

Программный комплекс предварительной обработки изображений для обнаружения и распознавания изображений

04, апрель 2014

DOI: 10.7463/0414.0707888

Буланов В. А., Волосатова Т. М.

УДК 621.0+681.5

Россия, МГТУ им. Баумана

vladimir_bulanov@mail.ru

tamaravol@gmail.com

Введение

Задача распознавания образов решается в самых разных областях: аэрокосмическая разведка; навигация летательных аппаратов; системы безопасности; управление с помощью систем технического зрения; медицина и биология.

Методы обнаружения, классификации и идентификации объектов находятся в непрерывной разработке. Единой теории решения этих задач пока не существует. Решение указанных задач почти невозможно, если изображение искажено, расфокусировано и смазано. Решение принципиально невозможно, если изображение зашумлено. На этом основании можно утверждать, что актуальной является разработка аппаратно-программных комплексов для предварительной обработки изображений.

Целью работы является разработка программного обеспечения автоматизированного рабочего места (АРМ), предназначенного для создания программ предобработки изображений с целью последующего обнаружения и распознавания объектов.

В публикации приведены результаты аналитического обзора алгоритмов цифровой обработки изображений [1, 2], а также задач деконволюции, сегментации и фильтрации изображений. Рассмотрены алгоритмы компенсации смаза и дефокусировки [5,6]. Регистрация изображений местности, зачастую, должна производиться оперативно, без возможности повторения. При этом условия фотосъемки не всегда позволяют получить качественные снимки. Изображения могут быть дефокусированы и смазаны, особенно если снимки сделаны с помощью аппаратуры, устанавливаемой на БПЛА без должной стабилизации АФА (см. ниже). В статье приведен интерфейс программной реализации большинства указанных выше алгоритмов, представлен пример использования разработанного АРМ.

В статье использованы следующие сокращения:

АРМ – автоматизированное рабочее место;

АФА – аэрофотоаппарат;

БПЛА – беспилотный летательный аппарат;

ГИС – геоинформационная система;

ДЗЗ – дистанционное зондирование Земли;

ЛА – летательный аппарат;
 ОС – операционная система;
 ПО – программное обеспечение;
 САПР – система автоматизированного проектирования;
 СТЗ – система технического зрения.
 ОЭС – оптико-электронная система.

1. Структура программного обеспечения АРМ

Функциональная схема программного обеспечения (ПО) представлена на рис. 1.

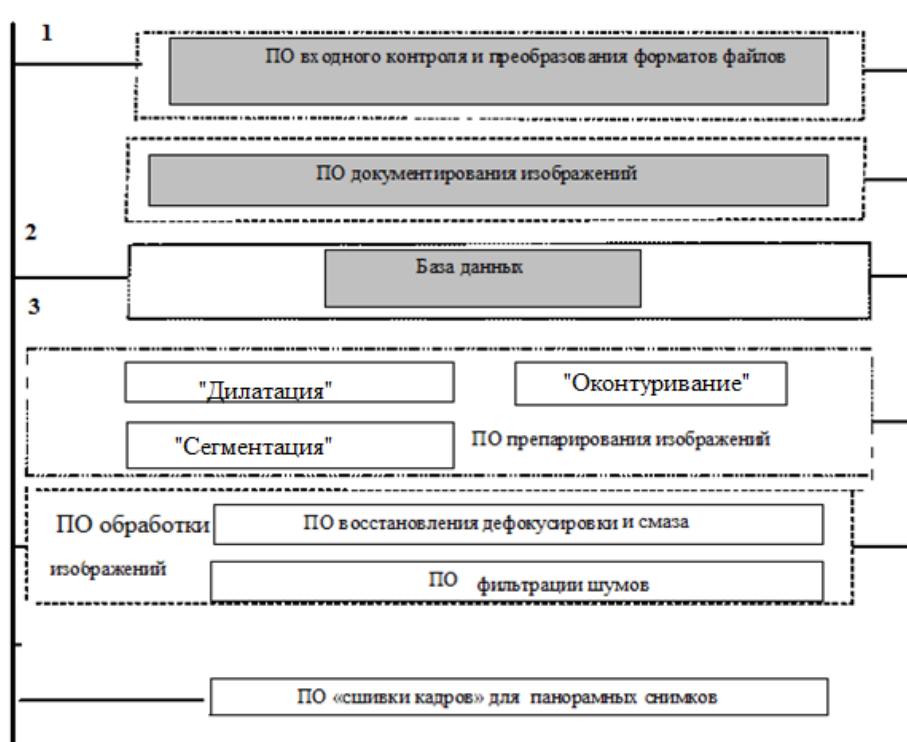


Рис.1. Функциональная схема ПО АРМ

Группа ПО, объединенная в блок 1, предназначена для сопряжения с другими программными комплексами (например, ГИС). К числу задач, решаемых данным ПО, относится преобразование форматов данных.

Группа ПО в блоке 2 предназначена для сопровождения процесса записи регистрируемых изображений, их описаний (в военных приложениях – разведдонесений) и навигационных данных.

ПО «База данных» обеспечивает обслуживание архива изображений. ПО предназначено для хранения изображений, записанных из АФА и СТЗ, а также для нахождения прототипов (эталонов) с помощью нечеткого запроса.

Группа ПО в блоке 3 обеспечивает подготовку (препарирование) изображений: сегментацию, оконтуривание, дилатацию.

Группа ПО обработки изображения включает в себя:

- ПО для восстановления дефокусировки и смаза (решение обратной задачи);
- ПО фильтрации;

- ПО «сшивки кадров» для синтеза кадра (получение панорамы) из фрагментов. Рассмотрим последнюю группу программного обеспечения.

2. Восстановление дефокусировки и смаза

Задача восстановления дефокусировки определяется интегральным уравнением Фредгольма первого рода [3,4]

$$\int_a^b k(\mathbf{p}, x, s_x, y, s_y) g(s_x, s_y) ds_x ds_y = f(x, y), \quad (1)$$

где s_x, s_y – пространственные координаты входной плоскости изображения, x, y – пространственные координаты плоскости регистрируемого изображения, \mathbf{p} – вектор конструктивных параметров ОЭС; a, b – пределы интегрирования; $k(\mathbf{p}, x, s_x, y, s_y)$ – ядро интегрального уравнения, описывающего ОЭС (функция рассеяния точки или импульсный отклик системы); $g(s_x, s_y)$ – входное распределение в координатном пространстве s_x, s_y , $f(x, y)$ – полученное изображение, которое является в некотором смысле искажением $g(s_x, s_y)$. Задача восстановления смаза отличается видом ядра уравнения. Заметим, что если хотя одна из подынтегральных функций в уравнении (1) является финитной, указанный интеграл записывается в бесконечных пределах.

Для дефокусировки в качестве импульсного отклика можно принять модель вида

$$k(r, x - s_x, y - s_y) = \begin{cases} \frac{1}{\pi r^2}, & \sqrt{(x - s_x)^2 + (y - s_y)^2} \leq r, \\ 0, & \sqrt{(x - s_x)^2 + (y - s_y)^2} > r. \end{cases}$$

Здесь вектор параметров модели определен лишь радиусом эквивалентного пятна рассеяния r .

Для линейной модели смаза функцию рассеяния точки можно принять в виде

$$k(d, x - s_x; y - s_y) = \begin{cases} \frac{1}{d}, & (x - s_x) \in [-d; 0], \\ 0, & (x - s_x) \notin [-d; 0]. \end{cases}$$

Здесь d – значение смаза.

Уравнение (1) в частотной области имеет вид

$$K(u, v) G(u, v) = F(u, v), \quad (2)$$

где u, v – координаты в частотной области; K, G, F – Фурье-образы соответствующих функций; $K(u, v)$ – оптическая передаточная функция.

Для решения задачи (2) используем регуляризацию по Тихонову [4,5], когда в качестве функции W – регуляризирующего множителя берется функция

$$W(u, v) = \frac{K^*(u, v)}{|K(u, v)|^2 + \alpha M(u, v)},$$

где $K^*(u, v)$ – функция комплексно сопряженная оптической передаточной функции; $M(u, v)$ – неотрицательная функция, например $M(u, v) = 1$; α – параметр регуляризации.

Спектральный метод основан на свойствах спектра смазанного изображения: в спектре присутствует характерная рябь, направленная по смазу и имеющая период, обратно пропорциональный модулю вектора (длине) смаза [6] (рис. 2).

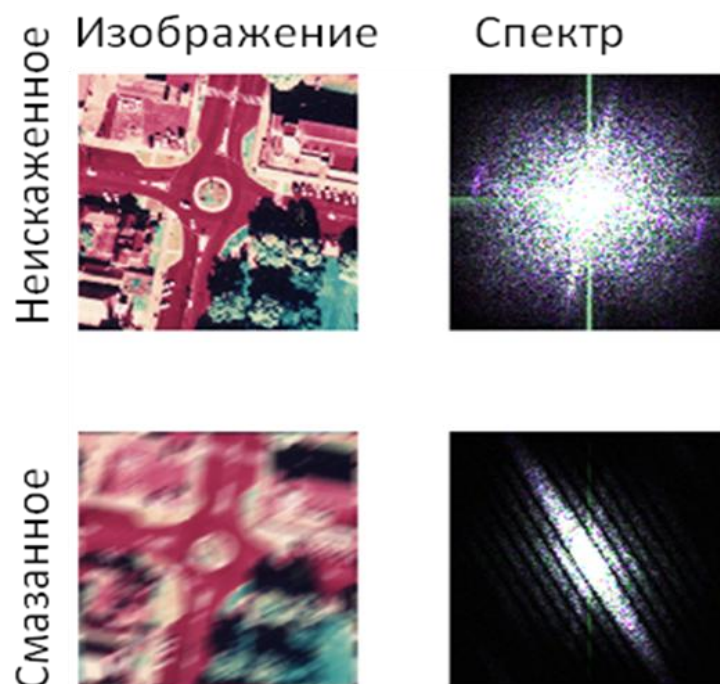


Рис. 2. Спектр смазанного изображения

Для удобного нахождения направления и длины смаза используем кепстр – вещественную часть преобразования Фурье от логарифма магнитуды спектра [7] (рис. 3). На кепстре характерно проявляются два симметричных относительно центра негативных пика, отстоящие от центра на величину, равную длине смаза. Линия, соединяющая эти пики, дает представление о направлении смаза.

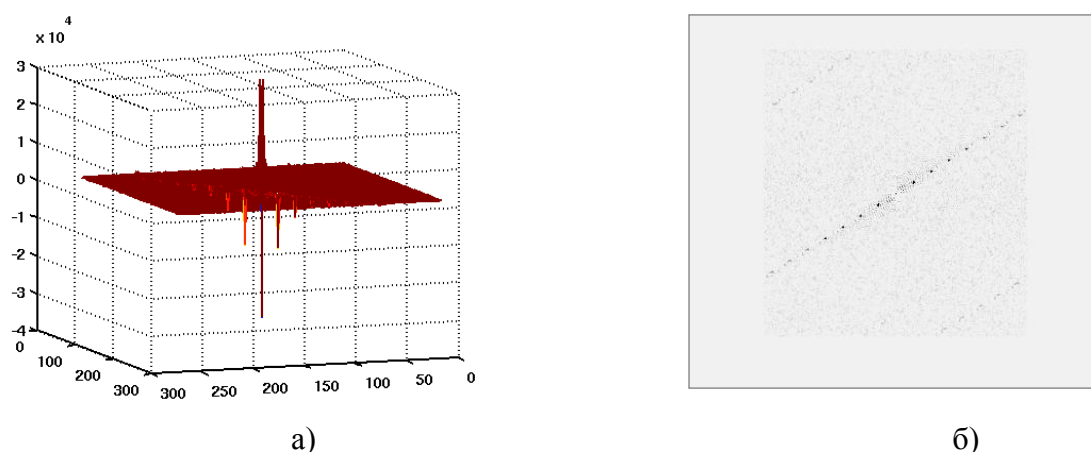


Рис. 3. Кепстр смазанного изображения: а – кепстр изображения; б – линия направления смаза.

Градиентный метод определения направления смаза [6] основан на том, что в направлении смаза градиент распределения амплитуды изображения ослаблен, в то время как в направлении, перпендикулярном смазу, величина градиента остается прежней. Таким образом, для определения вектора смаза необходимо найти градиент для каждой точки изображения. В обработке изображений для этого широко используются граничные градиентные фильтры. Определив градиенты для каждой точки, можно найти преимущественное направление градиента, тогда направление, перпендикулярное преимуществен-

ному, можно считать направлением смаза. Если же использовать не градиенты, а границы объектов смазанного изображения, то они будут вытянуты вдоль линии смаза. Найдя преимущественное направление линий границ объектов можно определить направление смаза.

Преимущественное направление линий можно определить с помощью преобразования Радона [6]

$$R(x, \theta)[f] = \int_{-\infty}^{\infty} f(x \cos \theta - y \sin \theta, x \sin \theta + y \cos \theta) dy.$$

Преобразование Радона имеет простой геометрический смысл – это интеграл от функции f вдоль прямой, перпендикулярной вектору $(\cos \theta; \sin \theta)$ и проходящей на расстоянии x от начала координат. Угол θ , для которого преобразование Радона максимально, определяет направление смаза.

3. Фильтрация изображений

Программное обеспечение поддерживает четыре режима фильтрации изображений:

- фильтрация медианным фильтром длиной 3, 5 или 7 единиц;
- нелинейная фильтрация;
- фильтрация за счет блендируания;
- фильтрация за счет применения тришолдинга.

На рис. 4 приведено время работы медианного фильтра в зависимости от размера изображения. Как видно, экономия при обработке изображений обратно пропорциональна длине фильтра.

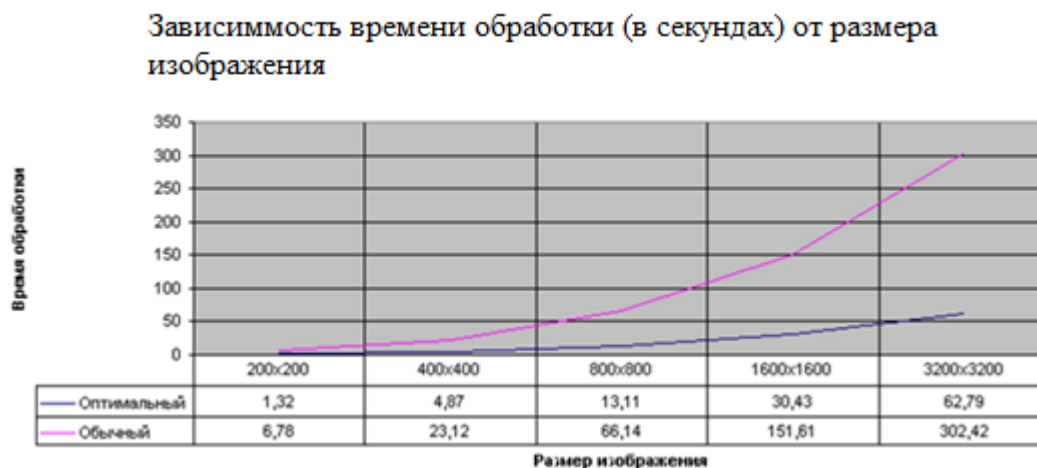


Рис. 4. График зависимости времени обработки медианным фильтром от размера изображения

Пользовательский интерфейс программы медианной фильтрации логически разделен на две зоны – зона просмотра изображения и зона управления программой (рис. 5). В верхней части окна расположены кнопки для загрузки изображения и выхода из программы. Зона управления состоит из четырех разделов, которые автоматически раскрываются при подводе к ним указателя мыши.



Рис.5. Зональное разделение пользовательского интерфейса

Раздел *Фильтрация* содержит настройки, применяемые для предварительной фильтрации изображения (рис. 6).

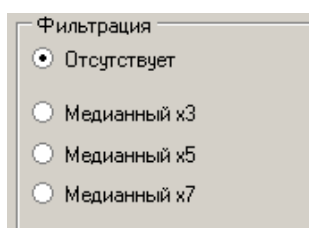


Рис. 6 Раздел *Фильтрация*

Раздел *Обработка сегментов* содержит настройки того, как алгоритм будет интерпретировать области при показе результатов обработки (рис. 7).

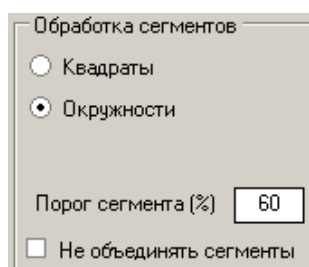


Рис. 7. Раздел *Обработка сегментов*

Раздел *Настройка сегментов*. Здесь указывается масштаб изображения – сколько пикселей содержит в себе один метр обрабатываемого снимка, минимальный и макси-

мальный размер искомых сегментов, а также порог расстояния между центрами сегментов, при котором происходит их объединение (рис. 8).

Настройка сегментов

Масштаб (pix/м)

Минимальный размер
искомых объектов (в
метрах)

- ширина

- длина

Максимальный размер
искомых объектов (в
метрах)

- ширина

- длина

Объединять, если
расстояние между центрами
не более (в метрах)

Рис. 8. Раздел Настройка сегментов

Раздел «Управление». В разделе собраны управляющие кнопки, позволяющие выполнить обработку изображения, обновить результаты в зависимости от измененных параметров, сохранить полученное изображение, а так же выгрузить полученные координаты сегментов (рис. 9).

Управление

Запустить обработку

Обновить

Сохранить изображение

Экспорт координат

Рис.9. Раздел Управление

Форма *Экспорт координат*. Позволяет выгружать полученные в ходе работы программы данные во внешние файлы – текстовый файл, либо в файл *Excel CSV* (рис. 10).

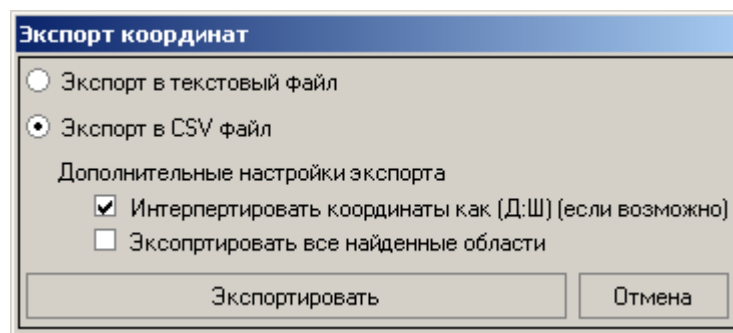


Рис. 10. Форма Экспорт координат

4. Сшивка изображений и координатная привязка

На рисунке 11 представлена структурная схема программного обеспечения сшивки полного изображения из его фрагментов и координатной привязки.

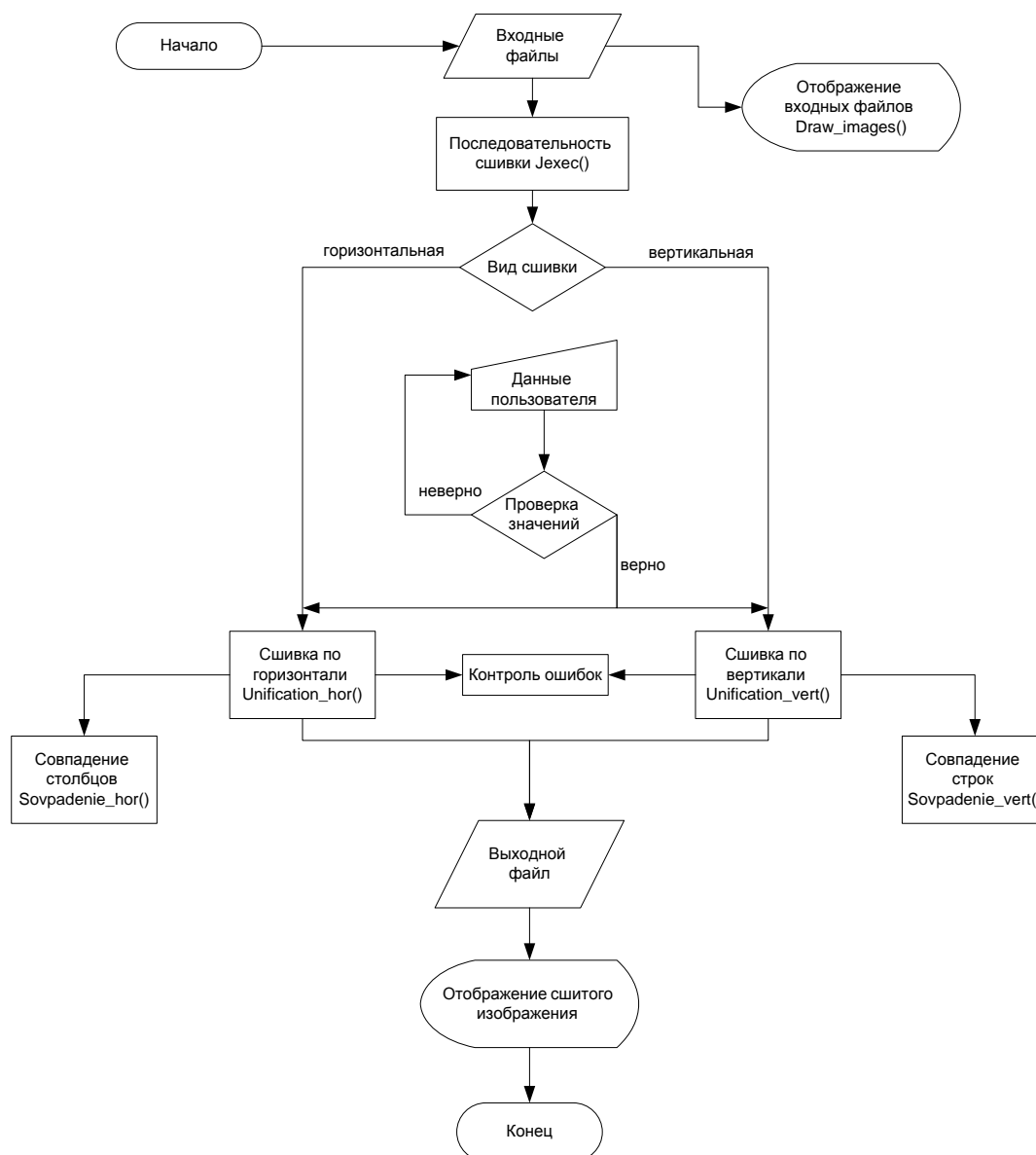


Рис. 11. Структурная схема ПО для сшивки полного изображения и координатной привязки

На рисунке 12 приведена структурная схема программного обеспечения для чтения координатной информации.

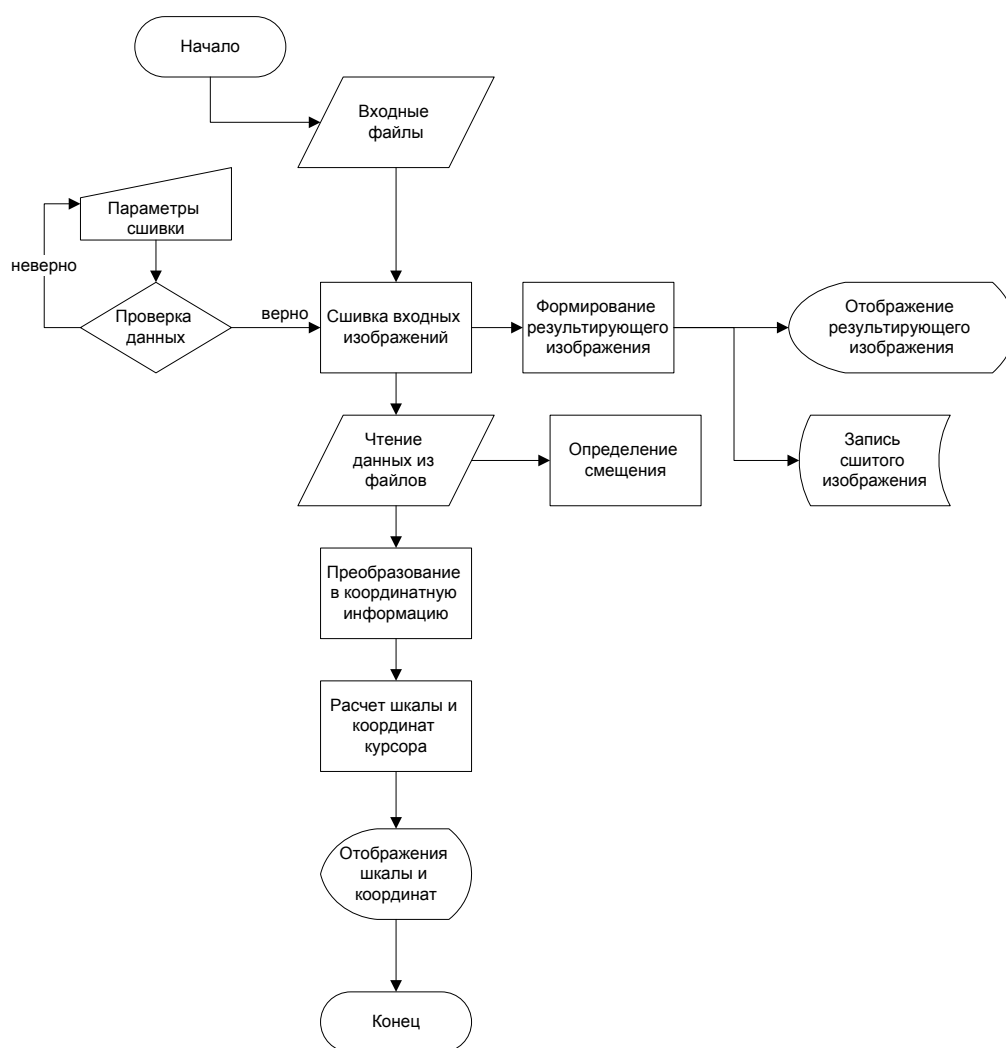


Рис. 12. Структура ПО для чтения координатной информации

При необходимости несложно осуществить конвертацию файла. На рисунке 13 приведен вариант хранения координатной информации в теле графического файла в двоичной кодировке. Перед записью данные кодируются условленным ранее способом. При использовании для хранения заголовка файла в формате **.bmp* и коррекции смещения основных данных можно добиться сохранения первоначального вида изображения и корректного распознавания файла всеми редакторами и служебными программами. После чтения координатной информации из файла она декодируется и может быть использована для осуществления привязки файла записи координат.

Структура *.BMP - файла:

BM	Смещение данных от начала файла
.....	Добавленная координатная информация	
Данные изображения			

☐ - Заголовок *.BMP - файла

Рис. 13. Структура *.bmp-файла для записи координат

На рисунке 14 показано содержимое модифицированного графического файла с добавленной координатной информацией (в данном случае координаты станции метро «Коньково» в Москве). Координатная информация в файле располагается сразу после основного заголовка файла в системе счисления по основанию 16, при этом корректируется смещение графических данных. Выбран формат записи координат в градусах и долях градуса, поскольку в условиях вычислительной техники данный способ удобнее способа хранения градусов, минут и секунд.

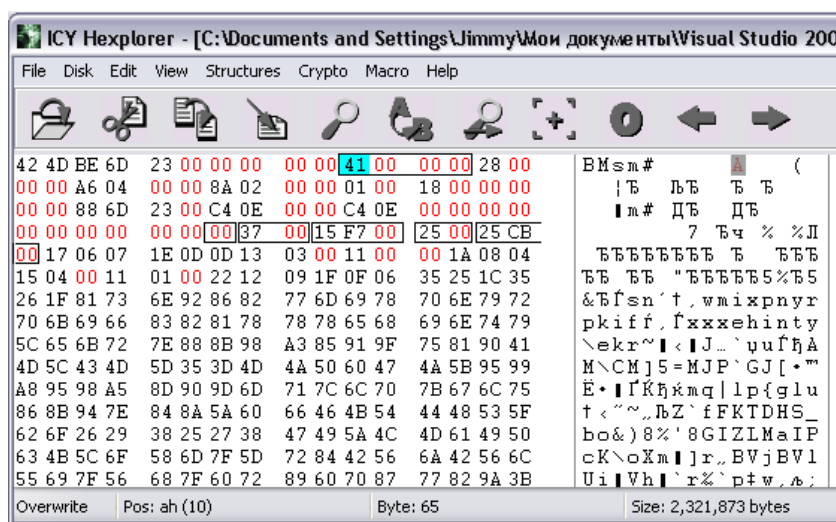


Рис. 14. Модифицированное представление bmp-файла после записи координат

В этот же файл записываются координаты геометрического центра снимка, так как предполагается, что съемка производится вертикально вниз с БПЛА или АФА, расположенного на ЛА (рис. 15). При расположении аэрофотоаппарата под углом, или в каких либо других случаях необходимо вводить соответствующую поправку.

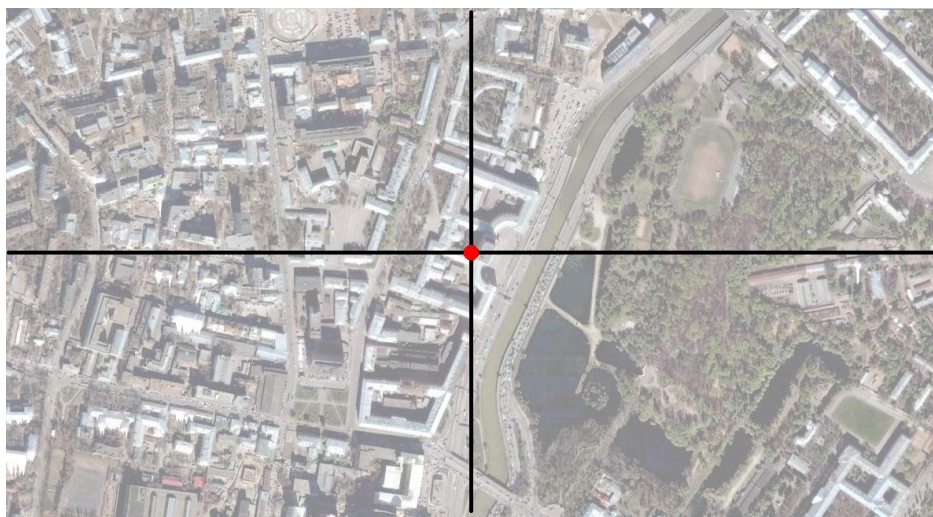


Рис. 15 Геометрический центр снимка

На рисунке 16 представлен интерфейс программы сшивки изображения, например из 9 фрагментов. Каждый из фрагментов располагается в отдельном подокне. Результирующее изображение выводится в отдельное подокно, предназначенное для вывода результата сшивки (рис. 16).

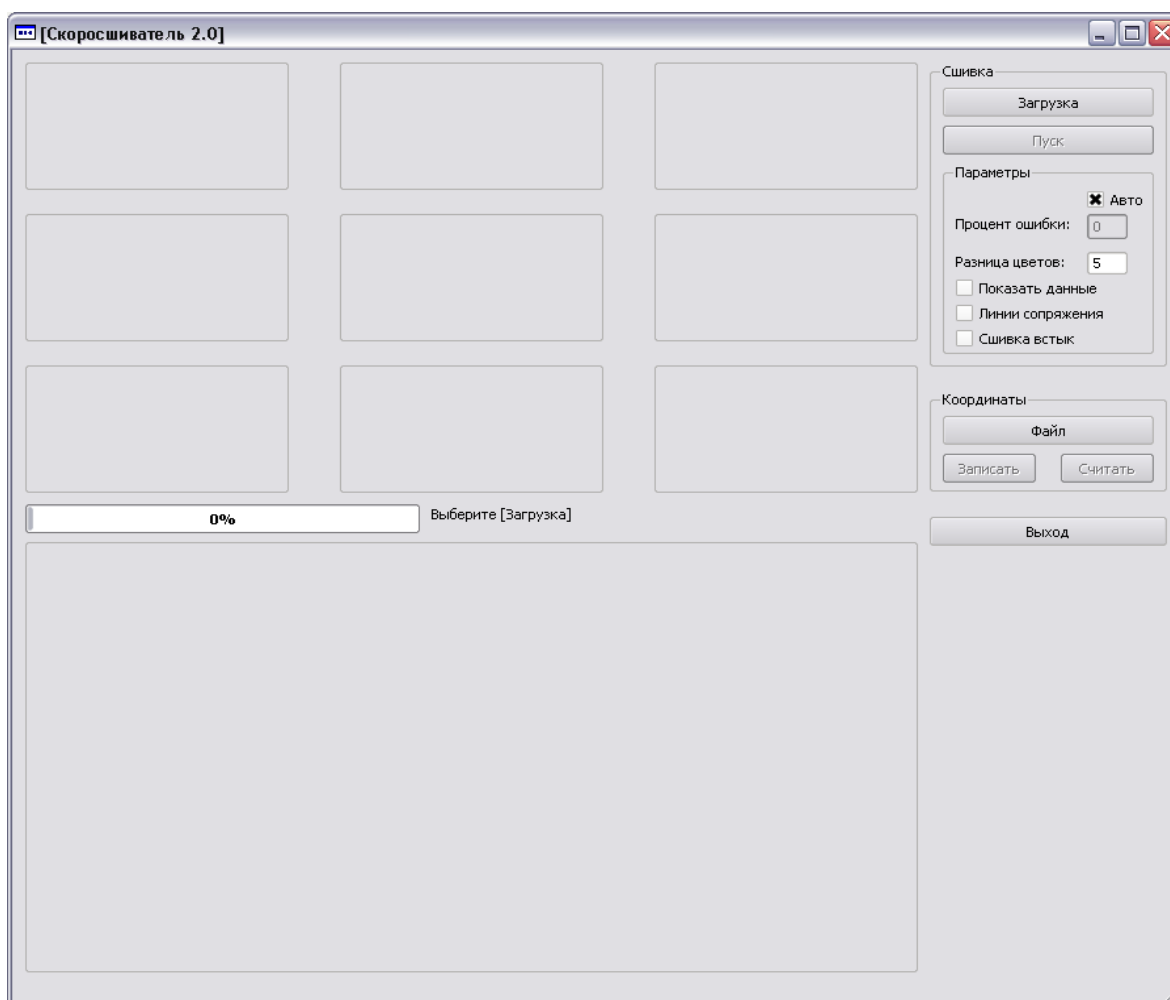


Рис. 16. Входной интерфейс программы Скоросшиватель

После запуска (см. рис. 16) пользователю предлагается выбрать сшиваемые изображения посредством кнопки «Загрузка». Результат выводится как на экран, так и записывается на жесткий диск под именем *result.bmp* (рис.17).

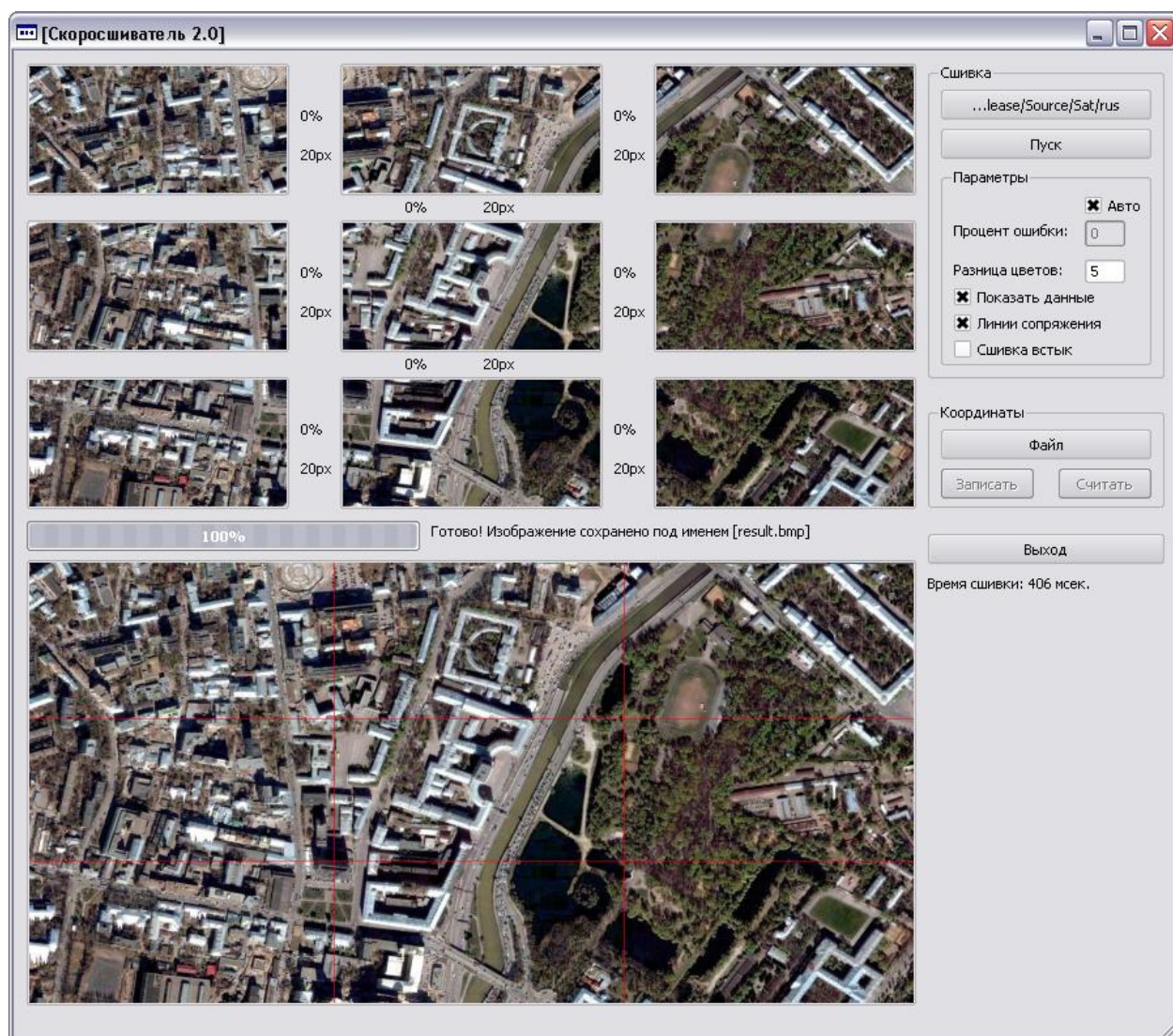


Рис.17. Окно программы после выполнения сшивки

Чтобы отобразить на результирующем изображении границы сшиваемых фрагментов, необходимо повторно нажать кнопку «Пуск» (рис.17). При этом в нижнем окне появятся красные линии сопряжения, показывающие границы составляющих фрагментов.



Рис. 18. Результат сшивки с линиями сопряжения

На рисунке 19 приведен интерфейс программы оцифровки результирующего изображения.

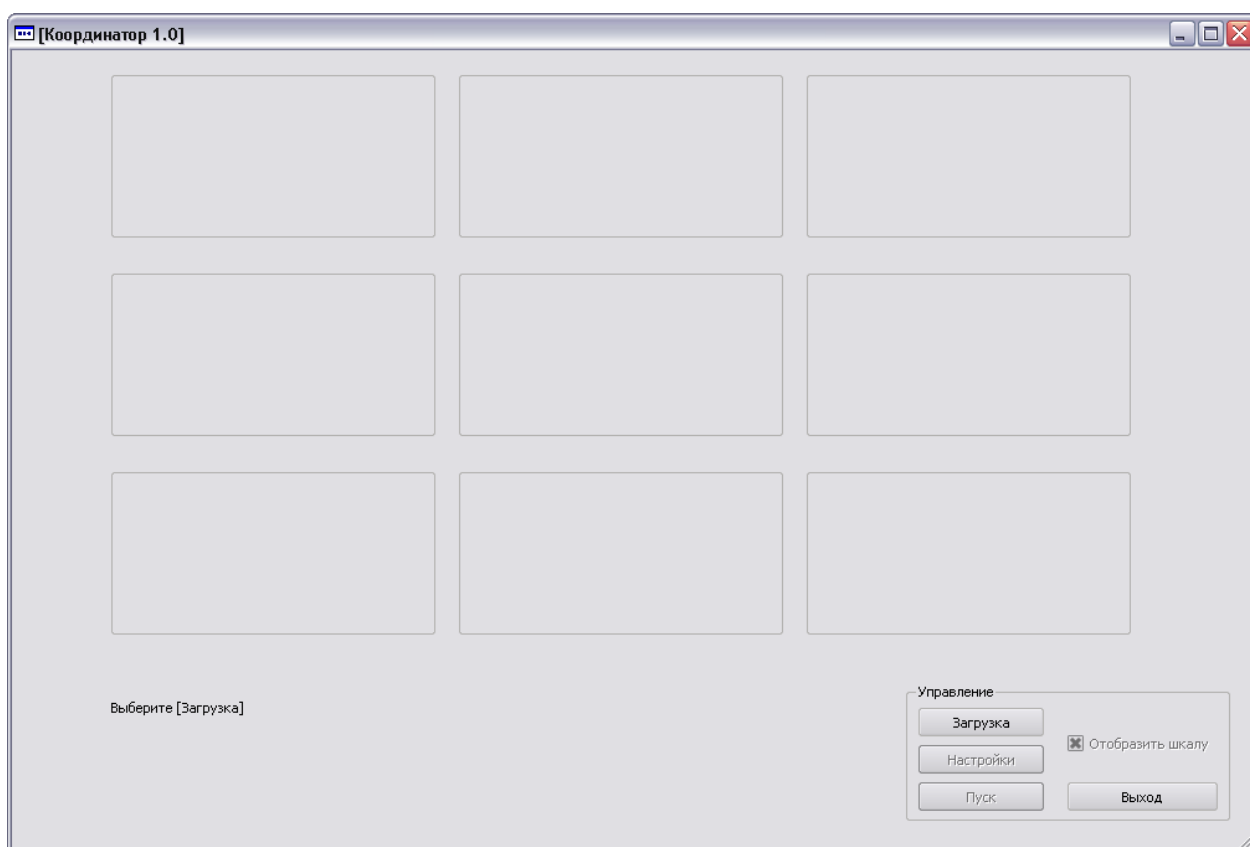


Рис.19. Интерфейс программы оцифровки результирующего изображения

Сшиваемые фрагменты изображений располагаются в соответствующих подокнах (рис. 19 и рис.20). После нажатии кнопки «Пуск» результат будет выведен на экран, а также записан на жесткий диск под именем *map.bmp*. Также будет сформирована координатная сетка и отображены шкалы широты и долготы (рис. 21).

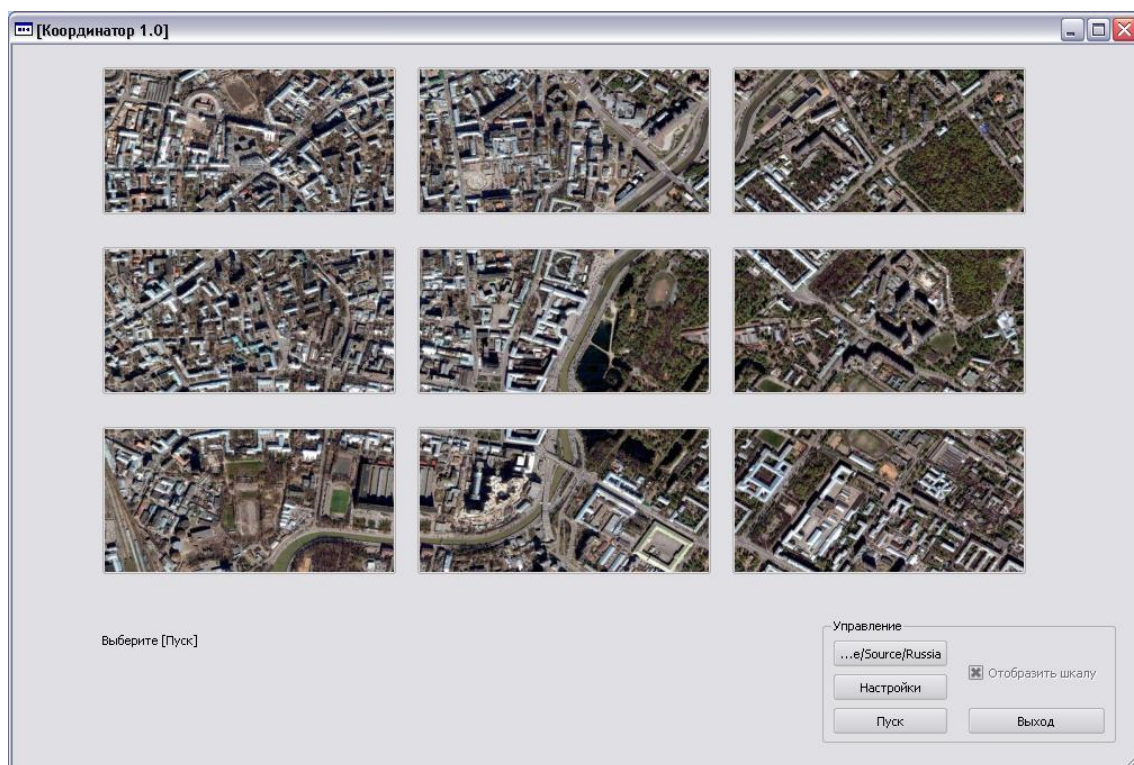


Рис.20. Исходные фрагменты изображения



Рис.21. Результат работы программы

Для анализа производительности программы сшивки изображений были выбраны четыре набора изображений, имеющих различные разрешения в девяти фрагментах (файлах).

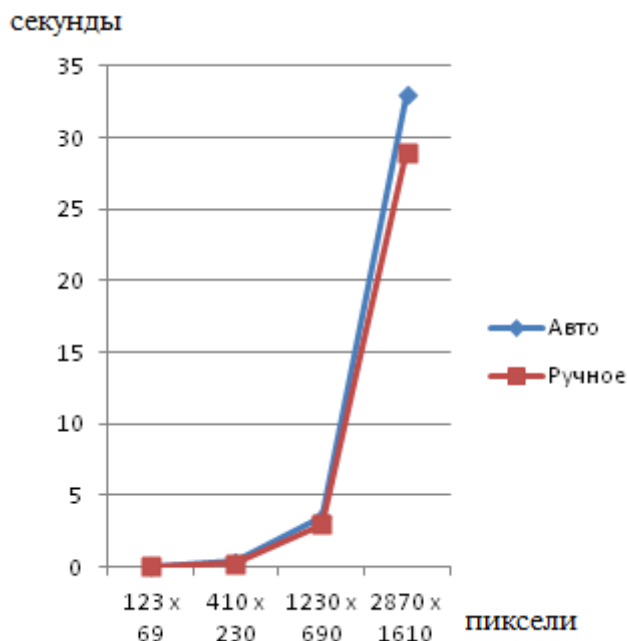


Рис. 22. Время выполнения сшивки фрагментов изображений различных размеров в зависимости от размера файла

Измерения производились на персональном компьютере с процессором *AMD Athlon* с тактовой частотой 2 ГГц и объемом оперативной памяти 1,5 Гб (см. рис.22). При измерении времени не учитывались временные затраты на запись результирующего файла на диск.

Анализ показал, что алгоритм, реализованный в программе, обладает удовлетворительной производительностью для решения поставленных задач.

Заключение

На основе аналитического обзора методов и алгоритмов обработки и распознавания изображений показана необходимость разработки соответствующего АРМ.

Предложена структура программного обеспечения для автоматизированного проектирования и синтеза алгоритмов обработки и распознавания изображений.

Разработано алгоритмическое и программное обеспечение АРМ.

Показана эффективность разработанных алгоритмов и программного обеспечения.

Список литературы

1. Афиногенов Е.А., Волосатова Т.М., Сельвесюк Н.И., Чичварин Н.В. Анализ методов и средств контроля систем дистанционного зондирования Земли // Наука и образование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2012. № 2. Режим доступа: <http://technomag.bmstu.ru/doc/322927.html> (дата обращения 01.04.2014).

2. Волосатова Т.М., Марченков А.М., Чичварин Н.В. Исследования и разработка комбинированного метода обнаружения и распознавания движущихся объектов // Информационные технологии. Приложение. 2013. № 12. С. 24-32.
3. Верлань А.Ф., Сизиков В.С. Методы решения интегральных уравнений с программой для ЭВМ. Киев: Наукова думка, 1978. 292 с.
4. Петров Ю.П., Сизиков В.С. Корректные, некорректные и промежуточные задачи с приложениями. СПб.: Политехника, 2003. 261 с.
5. Градов В.М., Филиппов М.В. Решение обратных задач методом регуляризации / под ред. К.А. Майкова. М.: Изд-во МГТУ им. Н.Э. Баумана, 1998. 20 с.
6. Волосатова Т.М., Воронов А.В., Чичварин Н.В. Математическое и программное обеспечение решения задачи синтеза для САПР оптико-электронных систем // Информационные технологии. Приложение. 2002. № 4. С. 3-34.
7. Childers D.G., Skinner D.P., Kemerait R.C. The cepstrum: A guide to processing // Proceedings of the IEEE. 1977. Vol. 65, iss. 10. P. 1428-1438. DOI: [10.1109/PROC.1977.10747](https://doi.org/10.1109/PROC.1977.10747)
8. Thilaka S., Donald I. Chapter 48. Medical Image Reconstruction with the FFT // In book: GPU Gems 2 / M. Pharr, ed. Addison-Wesley, 2005. P. 785-797.
9. Krueger N. An Algorithm for the Learning of Weights in Discrimination Functions Using a Priori Constraints // IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997. Vol. 19, iss. 7. P. 764-768. DOI: [10.1109/34.598233](https://doi.org/10.1109/34.598233)
10. Chamgoulov R., Lane P., MacAulay C. Three-dimensional optical computed-tomography microscope // Proc. SPIE. 5701. Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XII, 2005. DOI: [10.1117/12.591859](https://doi.org/10.1117/12.591859)

Program complex of pretreatment imaging for detection and image recognition

04, April 2014

DOI: 10.7463/0414.0707888

V.A. Bulanov, Volosatova T.M.

Bauman Moscow State Technical University, 105005, Moscow, Russian Federation

vladimir_bulanov@mail.ru

tamaravol@gmail.com

The article below describes an automatized workstation designed for developing programs, which preprocess images for further image detection and recognition. A software structure for an automatized workstation has been developed, and its functional block diagram is given. Methods and algorithms to solve the inverse problem of processing images obtained from unmanned aerial vehicles with no gyro stabilization system of aerial photography. Among them there are problems of defocusing and blurring recovery in resulting images. The article analyzes in-detail the software for image processing, that includes the following: software for defocusing and blurring recovery (an inverse problem solution); filtering software; software “joining frames” for frame synthesis (merging into panorama) from fragments. The interface of software support for most of algorithms described above is given. Computer-based experiment research results were also presented.

Publications with keywords: [filtering](#), [pattern recognition](#), [algorithms](#), [recognition](#), [segmentation](#), [blur](#), [defocusing](#), [inverse problem of image processing](#), [image composition](#)

Publications with words: [filtering](#), [pattern recognition](#), [algorithms](#), [recognition](#), [segmentation](#), [blur](#), [defocusing](#), [inverse problem of image processing](#), [image composition](#)

References

1. Afinogenov E.A., Volosatova T.M., Sel'vesyuk N.I., Chichvarin N.V. [Analysis of methods and means of controlling Earth remote sensing systems]. Nauka i obrazovanie MGTU im. N.E. Baumana - Science and Education of the Bauman MSTU, 2012, no. 2. Available at: <http://technomag.bmstu.ru/en/doc/322927.html> , accessed 01.04.2014. (in Russian).

2. Volosatova T.M., Marchenkov A.M., Chichvarin N.V. [Research and Development of the Combined Method of Detection and Recognition of Moving Objects]. *Informatsionnye tekhnologii. Suppl.*, 2013, no. 12, pp. 24-32. (in Russian).
3. Verlan' A.F., Sizikov V.S. *Metody resheniya integral'nykh uravneniy s programmami dlya EVM* [Methods for solving integral equations with computer programs]. Kiev, Naukova dumka Publ., 1978. 292 p. (in Russian).
4. Petrov Yu.P., Sizikov V.S. *Korrektnye, nekorrektnye i promezhutochnye zadachi s prilozheniyami* [Correct, incorrect, and intermediate problems with applications]. St. Petersburg, Politehnika Publ., 2003. 261 p. (in Russian).
5. Gradov V.M., Filippov M.V. *Reshenie obratnykh zadach metodom regulyarizatsii* [Solution of inverse problems by method of regularization]. Moscow, Bauman MSTU Publ., 1998. 20 p. (in Russian).
6. Volosatova T.M., Voronov A.V., Chichvarin N.V. [Mathematical and software of solutions of synthesis problem for CAD for optoelectronic systems]. *Informatsionnye tekhnologii. Suppl.*, 2002, no. 4, pp. 3-34.
7. Childers D.G., Skinner D.P., Kemerait R.C. The cepstrum: A guide to processing. *Proceedings of the IEEE*, 1977, vol. 65, iss. 10, pp. 1428-1438. DOI: [10.1109/PROC.1977.10747](https://doi.org/10.1109/PROC.1977.10747)
8. Sumanaweera T., Liu D. Ch. 48. Medical Image Reconstruction with the FFT. In book: Pharr M., ed. *GPU Gems 2*. Addison-Wesley, 2005, pp. 785-797.
9. Krueger N. An Algorithm for the Learning of Weights in Discrimination Functions Using a Priori Constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, vol. 19, iss. 7, pp. 764-768. DOI: [10.1109/34.598233](https://doi.org/10.1109/34.598233)
10. Chamgoulov R., Lane P., MacAulay C. Three-dimensional optical computed-tomography microscope. *Proc. SPIE*. 5701. Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XII, 2005. DOI: [10.1117/12.591859](https://doi.org/10.1117/12.591859)