МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П.КОРОЛЕВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

М.В. Гашников, Н.И. Глумов, В.В. Мясников, Е.В. Мясников,
В.В. Сергеев, А.В. Чернов

Компьютерная обработка картографических и спутниковых изображений

Электронное учебное пособие

САМАРА
2010

Авторы: ГАШНИКОВ Михаил Валерьевич
ГЛУМОВ Николай Иванович,
МЯСНИКОВ Владислав Валерьевич,
МЯСНИКОВ Евгений Валерьевич,
СЕРГЕЕВ Владислав Викторович,
ЧЕРНОВ Андрей Владимирович

Пособие представляет собой описание методов и алгоритмов обработки аэрокосмических и картографических изображений. Рассмотрена специфика методов обработки изображений применительно к задачам, возникающим в геоинформационных системах. Представлены некоторые задачи бортовой обработки видеоинформации, а именно, задачи стабилизации скорости потока и повышения помехоустойчивости сжатых данных. Даны описания и пути решения некоторых задач наземной обработки: задачи обнаружения транспортных средств на изображениях, полученных с низколетящих летательных аппаратов; задачи автоматической геопривязки и калибровки картографических изображений; задачи распознавание контуров зданий на картографических изображениях. Описывается структура, задачи, программные средства, алгоритмические решения необходимые при создании и ведении регионального банка космических снимков.

Пособие предназначено для магистров по направлению 010400.68 “Прикладная математика и информатика”, обучающихся по программе «Математические и компьютерные методы обработки изображений и геоинформатики».

ОГЛАВЛЕНИЕ

1. Обработка изображений в геоинформационных системах: задачи, особенности и возможности 4
2. Некоторые задачи бортовой обработки видеоинформации в системах ДЗЗ 8
2.1. Общее описание метода бортовой обработки видеоинформации 8
2.2. Стабилизация скорости потока сжатых данных 9
2.3. Повышение помехоустойчивости сжатых данных 16
3. Некоторые задачи наземной обработки данных дистанционного зондирования и картографических изображений 25
3.1. Обнаружение транспортных средств на изображениях, полученных с низколетящего летательного аппарата 25
3.2. Нормализация положения движущихся объектов при последовательной регистрации RGB-каналов 28
3.3. Метод автоматической геопривязки и калибровки картографических изображений 32
3.4. Автоматическое распознавание контуров зданий на картографических изображениях 36
4. Региональный центр приема и обработки данных дистанционного зондирования . 42
4.1. Общая характеристика Центра 42
4.2. Региональный банк космических снимков 43
4.2.1. Автоматическое наполнение регионального банка данных космических снимков в режиме постоянного приема 45
4.4. Некоторые программные и программно-аппаратные решения в задачах обработки цифровых изображений и данных ДЗЗ 47
4.4.1. Инструментальная программная система для тематической обработки и анализа цифровых изображений 48
4.4.2. Программная система распределенной обработки изображений 52
4.5. Геоинформационный портал работы с пространственными данными и метаданными 56
5. Библиографический список 61

1. ОБРАБОТКА ИЗОБРАЖЕНИЙ В ГЕОИНФОРМАЦИОННЫХ СИСТЕМАХ: ЗАДАЧИ, ОСОБЕННОСТИ И ВОЗМОЖНОСТИ

В последние несколько десятилетий сформировался и продолжает бурно развиваться особый класс автоматизированных комплексов - компьютерные геоинформационные системы (ГИС), которые обрабатывают, накапливают, обновляют и предоставляют пользователям пространственно-координированные сведения об объектах, расположенных на земной поверхности.

Основная форма представления информации в ГИС – цифровые карты. Они создаются на основе различных информационных источников: традиционных картографических материалов, геодезических измерений и т.д. Одним из важнейших источников данных для ГИС являются цифровые изображения, получаемые в результате аэрофотосъемок и космических съемок (дистанционного зондирования) Земли.

Аэрокосмические снимки Земли имеют существенные отличия от многих других типов цифровых изображений:

- они формируются в сложных условиях наблюдения, включают в себя специфичные искажения, обусловленные (а) самой технологией их получения, основанной на применении движущегося летательного аппарата, (б) специальным видом используемых оптикоэлектронных видеосенсоров, (в) многочисленными внешними искажающими факторами;
- им соответствуют чрезвычайно большие объемы цифровых данных, связанные (а) с размерами получаемых изображений (в несколько десятков тысяч пикселов по каждой координате), (б) с их многокомпонентностью (число спектральных каналов имеет типичные значения 6-8, а иногда достигает и 30-40);
- при этом формирование данных дистанционного зондирования, как правило, происходит с очень высокой скоростью, что диктует жесткие требования (а) к быстродействию регистрирующей аппаратуры, (б) к пропускной способности каналов передачи данных, (в) к скорости бортовой и наземной обработки данных, к вычислительной сложности применяемых алгоритмов обработки изображений;
- их применение в ГИС приводит к необходимости решения ряда новых задач обработки, а для известных задач – выдвигает новые требования к вычислительным алгоритмам, задает специфичные критерии качества обработки;

- с другой стороны, использование информации, уже имеющейся в ГИС (цифровых карт, планов, семантических данных и т.д.), открывает новые возможности повышения эффективности обработки поступающих космических снимков.

Итак, зададимся вопросами, какие задачи обработки изображений - данных аэрокосмического дистанционного зондирования - характерны для геоинформатики, какие специфические требования предъявляют ГИС к обработке, что дают ГИС в смысле дополнительных возможностей и дополнительных ограничений, которые необходимо учитывать разработчику методов обработки изображений?

Рассуждая с общих позиций, обработку аэрокосмических изображений в ГИС мы можем интерпретировать как задачу компрессии данных. Действительно, в главной целью обработки здесь является преобразование огромного массива данных, соответствующего исходному изображению, в его кодированное представление в виде векторной цифровой карты, которая:

- имеет существенно меньший объем,
- сохраняет всю информацию о зондированной территории, необходимую для дальнейшего использования в различных прикладных задачах.

Разумеется, что в действительности это общее (компрессирующее, кодирующее) преобразование данных разбивается на большое число шагов и частных преобразований, преследующих различные внутренние цели обработки. Рисунок 1.1 иллюстрирует процесс обработки аэрокосмических изображений для ГИС, условно разделенный на стадии: бортовой обработки, наземной предварительной обработки, тематической обработки и векторизации (т.е. превращения в цифровую карту, размещенную в базе данных ГИС).

Указанные стадии обработки аэрокосмических изображений являются достаточно очевидными и часто используются в научной и технической литературе. Тем не менее следует дополнительно пояснить смысл термина "тематическая обработка". К таковой мы будем относить задачи обработки и анализа изображений, решаемые в интересах отдельных групп потребителей информации – специалистов в области природопользования, геологии, градостроительства и т.д. (в отличие от предварительной обработки, выполняемой "для всех"). Особенностью тематической обработки является извлечение из изображений лишь той части информации, которая интересует данную группу специалистов, и использование специфических критериев

качества обработки. Очень часто тематическая обработка осуществляется не в автоматическом, а в интерактивном режиме, т.е. с привлечением опыта и профессиональных знаний человека-оператора. Именно на этой стадии в обработке может быть использована дополнительная информация из ГИС. Например, это могут быть векторные карты, описывающие территорию в период времени, предшествующий зондированию, и позволяющие выявить на изображениях свежие изменения. Собственно, основной целью тематической обработки как раз и является формирование или обновление "слоев" картографической информации, интересующих конкретную группу пользователей, в базе данных ГИС.

Цифровые изображения, получаемые в результате дистанционного зондирования Земли, являются важным, но разумеется, не единственным источником данных для ГИС. Другим, не менее важным классом изображений, используемых для создания векторных цифровых карт, являются результаты сканирования "традиционных" картографических материалов: "бумажных" карт, планов, схем и т.д. Процесс обработки сканированных картографических изображений имеет свои особенности, но в целом тоже соответствует рисунку 1.1. Конечно при этом надо исключить из рассмотрения стадию "бортовой обработки" и некоторые процедуры "предварительной обработки", связанные с собственно формированием двумерных изображений: анализа стереопар, синтеза цифровой модели местности (ЦМР), обработки радиоголограмм, и т.п.

Из рисунка 1.1 следует, что на разных стадиях показанного процесса преобразований данных реализуются различные и весьма многочисленные частные процедуры обработки изображений. Если говорить об основных общих принципах построения этих процедур, обеспечивающих скорость и высокую эффективность обработки изображений в ГИС, то, на наш взгляд, их пять:

- иерархическое (многоуровневое) представление данных и алгоритмов обработки,

- рекурсивная и параллельная реализация обрабатывающих алгоритмов,

- компактное описание изображений системой признаков,

- настройка параметров обработки путем обучения, основанного на прецедентах.

- использование при обработке априорной информации из ГИС.

Ниже мы более подробно остановимся на некоторых задачах и алгоритмах обработки как аэрокосмических, так и картографических изображений, иллюстрирующих эти принципы.

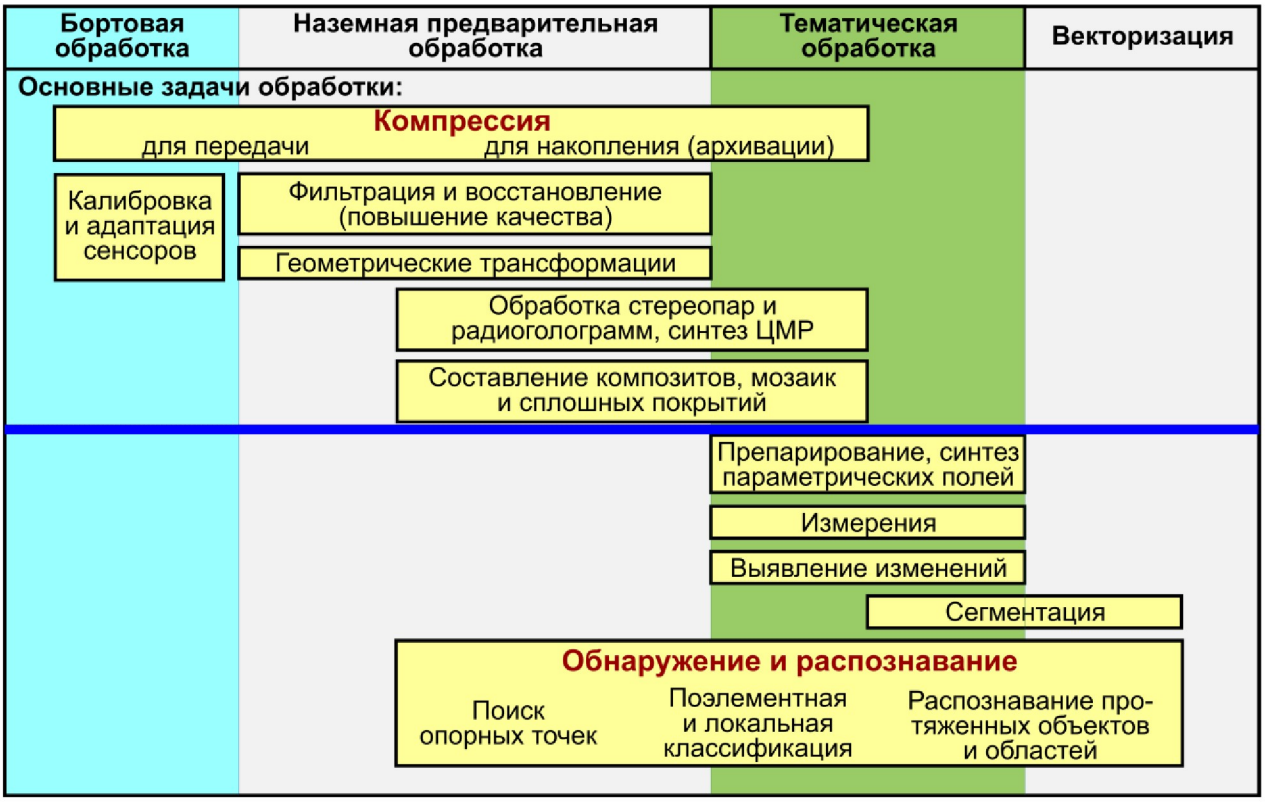


Рисунок 1.1 - Обработка аэрокосмических изображений для ГИС

2. НЕКОТОРЫЕ ЗАДАЧИ БОРТОВОЙ ОБРАБОТКИ ВИДЕОИНФОРМАЦИИ В СИСТЕМАХ ДЗЗ

2.1. Общее описание метода бортовой обработки видеоинформации

Специфика задачи обработки изображений в системах реального времени (в частности, в системах оперативного дистанционного зондирования поверхности Земли) определяется в первую очередь такими особенностями этих систем как:

- высокая скорость формирования входного изображения;
- наличие цифрового канала связи с постоянной пропускной способностью;
- возможность появления сбоев в цифровом канале связи;
- строгие ограничения на технические характеристики устройств (вес, энергопотребление и др.);
- широкий и неопределенный заранее круг задач, для решения которых может быть использованная получаемая видеоинформация.

Эти особенности определяют следующие требования к методам бортовой обработки видеоинформации:

- высокая степень сжатия;
- строгий контроль качества;
- низкая вычислительная и структурная сложность;
- постоянная скорость формирования выходного потока (сжатых данных);
- высокая помехозащищенность выходных данных.

Наиболее эффективными по большинству критериев и для основных классов изображений в настоящее время считаются методы компрессии на основе дискретного вейвлет-преобразования [17]. Однако они не удовлетворяют вышеперечисленным требованиям, в первую очередь по сложности и возможности контроля качества для бортовой реализации. Более простые методы, основанные на дифференциальном кодировании не обеспечивают достаточную эффективность, тем не менее благодаря их простоте используются в системах реального времени.

Разработанный авторами метод сжатия изображений на основе иерархической сеточной интерполяции (ИСИ-метод) удовлетворяет требованиям по степени сжатия, контролю качества и сложности алгоритма. Однако потребовалось дальнейшее развитие метода с целью решения не только задачи сжатия данных, но и решения задач обеспечения постоянной скорости формирования выходного потока данных и их помехоустойчивости [8, 9, 10**Ошибка! Источник ссылки не найден.**].

На рисунке 2.1 показана общая схема предлагаемого метода бортовой

обработки видеоинформации [5], на которой выделены три отдельных блока, обеспечивающие решение задач компрессии, стабилизации скорости выходного потока данных и защиты кодированной информации от сбоев в канале связи.

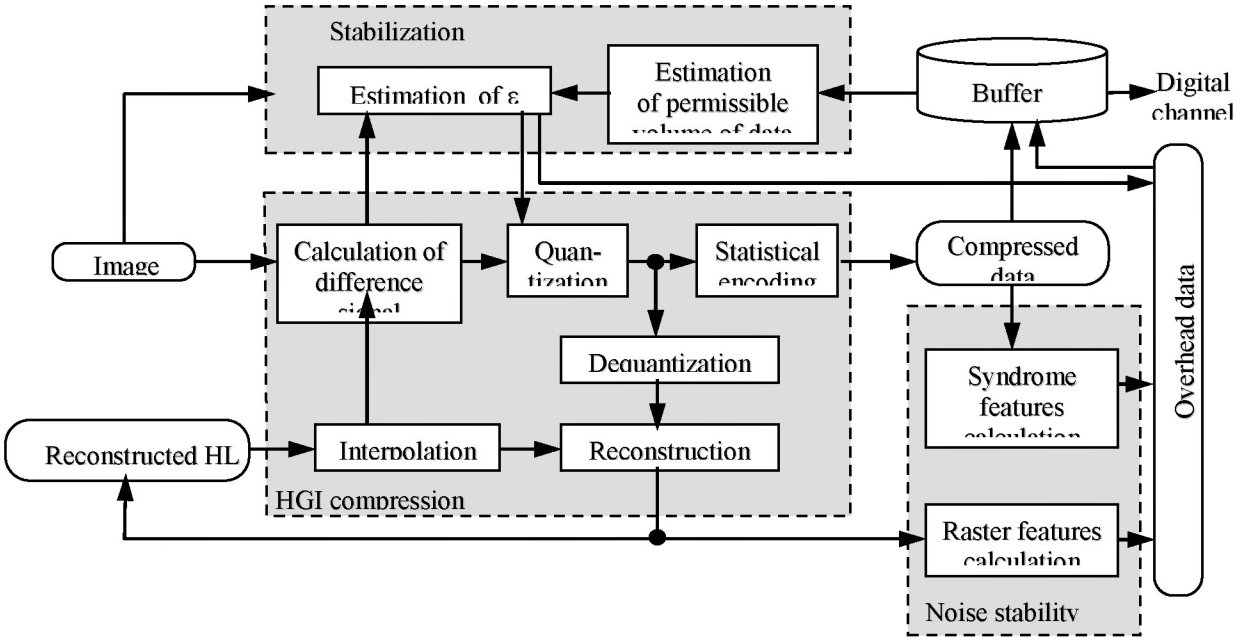


Рисунок 2.1 - Общая схема бортовой обработки видеоинформации

Метод компрессии обеспечивает последовательное сжатие иерархических уровней (ИУ) изображения и включает в себя алгоритмы интерполяции, предсказания, квантования и статистического кодирования. Отметим только, что за последние годы эти алгоритмы получили дальнейшее развитие [5, 7], были разработаны новые схемы интерполяции, шкалы квантования, улучшена эффективность статистического кодирования.

2.2. Стабилизация скорости потока сжатых данных

В системах дистанционного зондирования изображение формируется в виде длинной полосы фиксированной ширины. Это изображение делится на подполосы (substrips) (или блоки), которые компрессируются независимо. Затем каждый сжатый блок помещается в буферную память, из которой данные передаются в канал связи с постоянной пропускной способностью. Чтобы обеспечить скорость формирования выходного потока близкую к постоянной, необходимо определить управляющий параметр компрессии каждого блока. Для ИСИ-метода в качестве управляющего параметра используется максимальная погрешность сжатия.

Пусть кодируемое изображение представляет собой бесконечную вертикальную полосу шириной в M отсчетов. Такой формат изображения делает

целесообразным поблочную реализацию сжатия. При этом кодирование разных блоков размером $L \times M$ отсчетов (L - число строк в блоке) может осуществляться с различными управляющими параметрами, что приводит к отклонениям выходных показателей качества (степени сжатия и погрешности восстановления) от требуемых. Предлагаемая буферизация кодированных блоков должна по возможности гасить колебания погрешности и одновременно обеспечивать стабилизацию скорости выдачи закодированных данных в канал связи.

Построим математическую модель, дискретно описывающую во времени процесс буферизации видеоинформации, обрабатываемой поблочно. Пусть буферное запоминающее устройство имеет объем V_0 бит, к началу обработки K-го блока заполненный на $V(K\text{-}1)$ бит. За квант времени формируется с постоянной скоростью блок изображения объемом $\Delta V = LMb^0$ бит, который кодируется с задаваемой максимальной ошибкой $\varepsilon(K)$, обеспечивающей реальное сжатие блока $B(K)$ (бит/отсчет). За это же время в канал связи передается информация объемом $\frac{B_0 \Delta V}{b^0}$ бит, где B_0 - степень сжатия, соответствующая пропускной способности канала, b^0 - разрядность отсчетов несжатого изображения (бит/отсчет). К началу следующего кванта времени относительная заполненность буфера составит:

$$\bar{V}(K) = \frac{V(K)}{V_0} = \frac{1}{V_0}\left(V(K-1) + \frac{B(K)\Delta V}{b^0} - \frac{B_0 \Delta V}{b^0}\right) = \bar{V}(K-1) + \frac{B(K) - B_0}{k_v b^0}, \qquad (2.1)$$

где $k_v = \frac{V_0}{\Delta V}$ - параметр буфера (отношение объема буфера к объему блока до сжатия).

Основная проблема заключается в определении при сжатии каждого блока управляющего параметра - ошибки $\varepsilon(K)$. Задаваемая ошибка должна обеспечивать такую степень сжатия, при которой гарантируется условие отсутствия переполнения буфера $\bar{V}(K) \le 1$. Рассмотрим несколько подходов к определению $\varepsilon(K)$.

Алгоритм I. Для всех кодируемых блоков изображения задается фиксированная ошибка

$$\varepsilon(K) = \varepsilon_{max}.$$

Поскольку при постоянной ошибке различные блоки изображения сжимаются с различной степенью сжатия, то при ее выборе можно ориентироваться на наихудшую

(в смысле сжатия) ситуацию на изображении. Другими словами, ошибка должна гарантировать для произвольного фрагмента степень сжатия не менее требуемой. При этом отпадает потребность в буферной памяти для хранения сжатых блоков, т.к. любой блок сжимается сильнее, чем это необходимо для передачи в канал связи.

Алгоритм II. Ошибка на каждом шаге корректируется непосредственно в зависимости от заполненности буфера к моменту начала обработки K-го блока, например,

$$\varepsilon(K)=\varepsilon(K-1)+\left[\bar{V}(K-1)-\bar{V}_p\right]\bar{q}, \qquad (2.2)$$

где $0<\bar{V}_p<1$ - пороговое значение, при превышении которого ошибка увеличивается, а степень сжатия уменьшается; a - экспериментально подбираемый параметр.

Алгоритм III. Задаваемая максимальная ошибка сжатия K-го блока определяется с учетом вычисляемых статистических характеристик блока изображения (например, дисперсии D_x и коэффициента корреляции ρ) и требуемой степени сжатия $\hat{B}(K)$:

$$\varepsilon(K)=f(D_x,\rho,\hat{B}(K)),$$

где $\hat{B}(K)$ определяется исходя из наличия свободной памяти в буфере:

$$\hat{B}(K)=B_0+\left[\bar{V}_p-\bar{V}(K-1)\right]b^0\frac{k_v}{k_s}, \qquad (2.3)$$

$\bar{V}_p\approx 1$ - пороговое значение, обеспечивающее "запас прочности" по заполненности буфера,

$k_s\geq 1$ - коэффициент стабилизации, обеспечивающий более плавное изменение всех параметров при поблочном кодировании изображения.

Ниже предлагается конкретный метод расчета максимальной ошибки ε_{max} по требуемой степени сжатия $\hat{B}$, построенный в предположении изотропной экспоненциальной модели АКФ входного сигнала.

Предполагая достаточную эффективность метода статистического кодирования, в качестве оценки степени сжатия $\hat{B}$ будем использовать величину $\hat{H}_q$ энтропии квантованного разностного сигнала, предсказанную на основании коэффициента корреляции изображения ρ, дисперсии изображения D_x и ошибки ε_{max}. Можно записать:

$$\hat{H}_q=\left(\hat{H}^{(R)}+3\sum_{r=1}^{R}4^{r-1}\hat{H}^{(R-r)}\right)\Big/4^R$$

где $\hat{H}^{(r)}$ - предсказанное значение энтропии квантованного разностного сигнала на r-м ИУ. По определению энтропии:

$$\hat{H}^{(r)} = -\sum_{i=0}^{N_q-1} p_q^{(r)}(i)\log\left(p_q^{(r)}(i)\right),$$

где $p_q^{(r)}(i)$ - вероятность i-го уровня квантования разностного сигнала, а N_q-количество уровней квантования. В соответствии с известными рекомендациями [25] будем полагать, что распределение вероятностей неквантованного разностного сигнала на каждом ИУ является экспоненциальным. Тогда при использовании квантователя, вероятность i-го уровня квантования разностного сигнала может быть записана в виде:

$$p_q^{(r)}(i) = \sum_{k=i\cdot(2\varepsilon_{max}+1)-\varepsilon_{max}}^{i\cdot(2\varepsilon_{max}+1)+\varepsilon_{max}} \left(1-p_r\right)\left(1+p_r\right)p_r^{|i|},$$

где p_r- параметр упомянутого экспоненциального распределения, который связан с дисперсией $D^{(r)}$ разностного сигнала на r-м ИУ следующим образом:

$$p_r = \left(2\cdot D^{(r)} + 1 - \sqrt{2\cdot D^{(r)} + 1}\right) \Big/ D^{(r)}.$$

В свою очередь можно доказать, что:

$$D^{(r)} = a_1 \cdot 2^r \ln\left(1/\rho\right) D_x + a_2 \varepsilon_{max}^2,$$

где a_1, a_2 - коэффициенты, которые определяются схемой предсказания.

Таким образом, получена зависимость $\hat{H}_q = \hat{H}_q\left(\rho, D_x, \varepsilon_{max}\right)$, которая может быть заранее табулирована. В процессе кодирования в качестве искомого значения ошибки используется наименьшее значение ε_{max}, которое удовлетворяет неравенству:

$$\hat{H}_q\left(\rho, D_x, \varepsilon_{max}\right) < \hat{B}. \qquad (2.4)$$

Итак, предлагаемый алгоритм стабилизации с определением значения управляющего параметра (максимальной ошибки) алгоритма сжатия $\varepsilon(K)$ в K-ый квант времени требует:

1) вычисления статистических характеристик $D_x(K)$, $\rho(K)$ для K-го блока изображения;
2) определения требуемой степени сжатия $\hat{B}(K)$ согласно (2.3);

3) подбор (по таблице) значения ε(K), удовлетворяющего условию (2.4).

Для исследования рассмотренных алгоритмов было проведено имитационное моделирование процесса сжатия изображения со стабилизацией скорости выдачи кодированной видеоинформации с использование крупноформатного аэрокосмического изображения земной поверхности. На рисунке 2.2а показан пример такого изображения (размером 10000×512 отсчетов), хорошо отражающего динамику изменения локальной информативности изображения на маршруте съемки. Измеренные по блокам размером 33×512 статистические характеристики ρ и $\sigma=\sqrt{D_x}$ приведены на рисунке 2.2б.

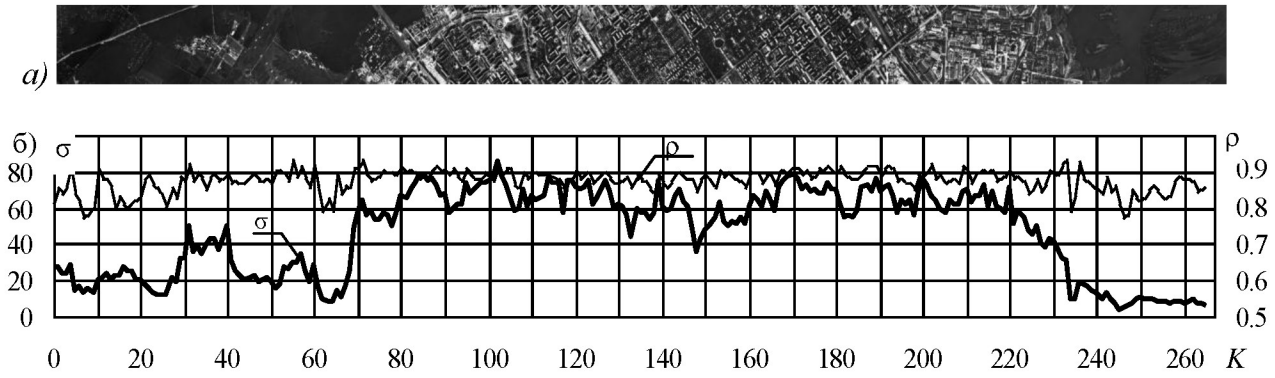


Рисунок 2.2 - Тестовое изображение и его статистические характеристики

С целью выбора оптимального алгоритма стабилизации скорости сжатия изображения были проведены экспериментальные исследования рассмотренных выше методов формирования управляющего параметра (максимальной ошибки восстановления) алгоритмом сжатия. При исследовании контролировалось заполнение буферной памяти согласно математической модели (2.1). В качестве критериев качества использовалась усредненная (по блокам) максимальная ошибка $E\{\varepsilon(K)\}$ восстановления. Все эксперименты выполнены для наиболее типичных требуемых степеней сжатия B_0=1 бит/отсчет, 2 бит/отсчет.

Алгоритм I. Вычислительный эксперимент показал, что при данном подходе нерационально эксплуатируются технические ресурсы (буферная память, канал связи), велика опасность переполнения буфера при неверном прогнозе допустимой максимальной ошибки, чрезмерна чувствительность к малым изменениям задаваемой ошибки (см. рисунок 2.3). При этом качество восстановления (по критерию $E\{\varepsilon(K)\}$) является плохим в сравнении с другими алгоритмами стабилизации. Реальное

применение данного способа задания управляющего параметра возможно только при большом объеме буферной памяти, а также при точном прогнозе управляющего параметра, что делает алгоритм практически непригодным для сжатия непрерывно поступающих видеоданных в системе реального времени.

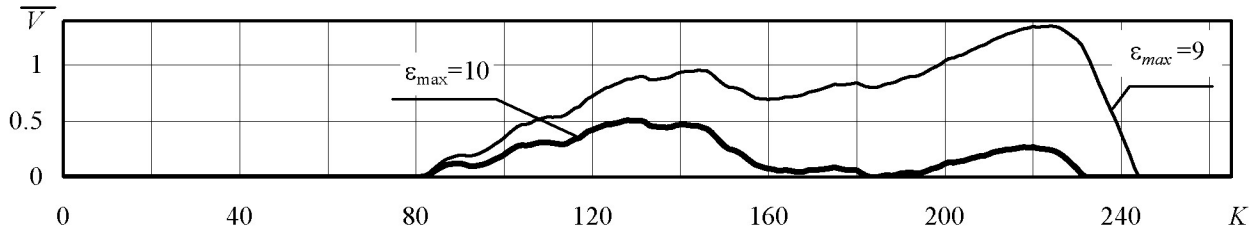


Рисунок 2.3 - Результаты моделирования алгоритма I стабилизации (B_0=2)

Алгоритм II. Больший интерес представляет формирование управляющего параметра в зависимости от относительной заполненности буфера согласно (2.2). Проведенные экспериментальные исследования (при варьировании коэффициента в пределах $2{,}5 \le a \le 20$, параметре буфера k_v=2, $\overline{V}_p = 0{,}5$) показали, что хотя параметр сжатия адаптивно настраивается в зависимости от состояния буфера, метод в целом можно охарактеризовать как недостаточно стабильный. Главной причиной, по которой метод не может быть рекомендован для реальной эксплуатации, является пульсирующий характер зависимостей $\overline{V}(K), \varepsilon(K)$ (см. рисунок 2.4), приводящий к неприемлемому качеству выходной видеоинформации, резко и многократно изменяющемуся в пределах одного маршрута съемки.

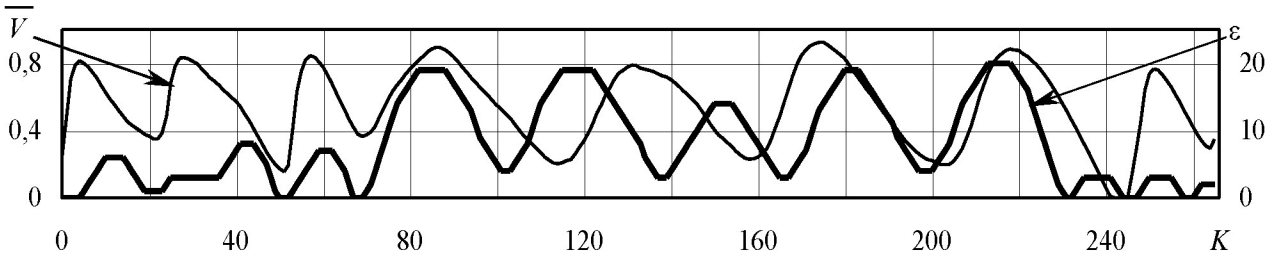


Рисунок 2.4 - Результаты моделирования алгоритма II стабилизации (B_0=2, a=10)

Алгоритм III. Наиболее перспективным представляется метод формирования заданной ошибки на основе анализа статистических характеристик кодируемого блока. Результаты моделирования метода показывают достаточно высокую эффективность и устойчивость. Большую роль при моделировании по формулам (**Ошибка! Источник ссылки не найден.**, **Ошибка! Источник ссылки не найден.**)

играет коэффициент стабилизации k_s, позволяющий смягчить последствия прохождения высокоинформативных участков изображения и получить лучшие показатели качества (см. рисунки 2.5 и 2.6а). Влияние параметра буфера k_v (при L=33 и оптимальном k_s) и числа строк в блоке L (при k_v=1) на качество восстановления отражено на рисунках 2.6б и 2.6в. Рост значения k_v, соответствующий увеличению размеров памяти, приводит к некоторому улучшению качества (см. рисунок 2.6б). Однако, наращивание памяти, если оно возможно, оказывается более целесообразным использовать на увеличение размера кодируемых блоков (см. рисунок 2.6в).

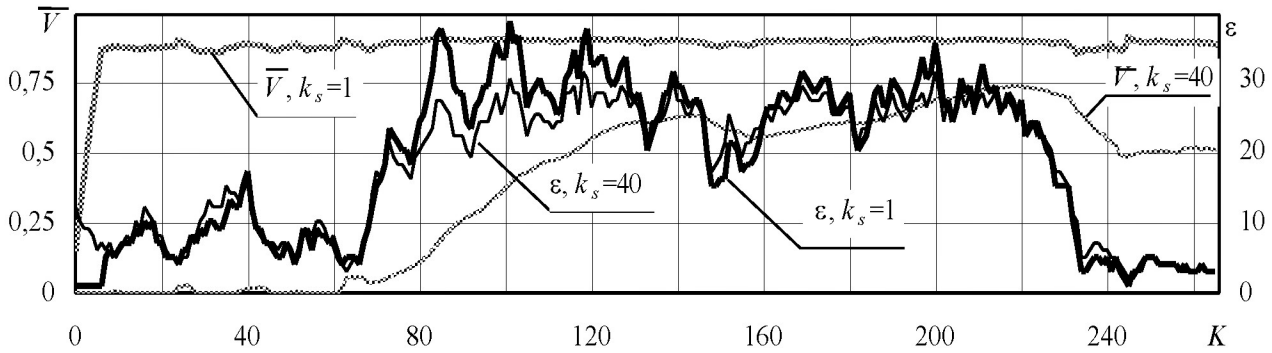
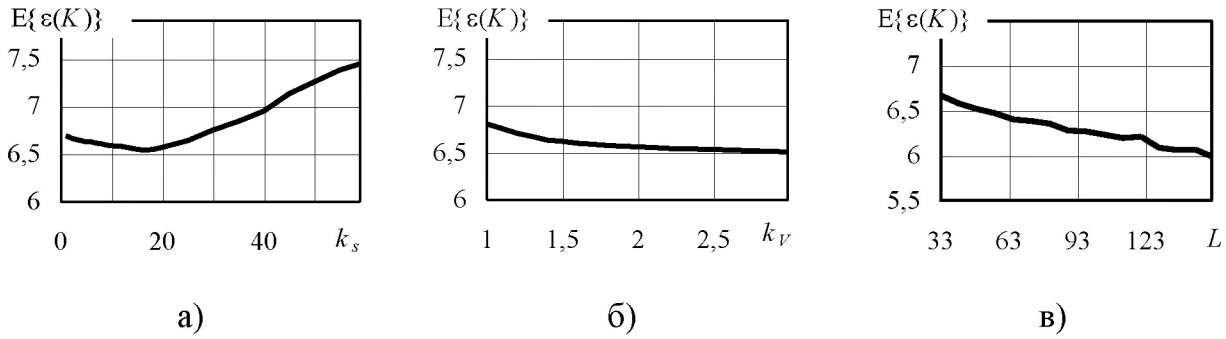


Рисунок 2.5 - Результаты моделирования алгоритма IV стабилизации (B_0=1 бит/отсчет)



а) б) в)

Рисунок 2.6 - Результаты исследования влияния параметров стабилизации

Поставленные вычислительные эксперименты не дают окончательных рекомендаций по выбору конкретных значений параметров метода стабилизации. Они возможны лишь при выполнении следующих условий:

1) точного задания ограничений, обусловленных техническими ресурсами;

2) решении весьма трудоемкой задачи оптимизации в многомерном пространстве параметров (k_s, k_v, L, $\overline{V}_p$) с использованием большого набора реальных сюжетов.

Однако изложенные выше результаты проведенных экспериментов позволяют провести сравнительный анализ различных алгоритмов стабилизации и наметить пути достижения наилучших показателей их качества.

Итак, результаты проведенного моделирования показывают, что для достаточно точного прогноза ошибки $\varepsilon(k)$, обеспечивающей требуемую степень сжатия на текущем блоке и стабильную скорость формирования сжатой видеоинформации в целом, нельзя опираться лишь на текущее состояние буфера и значения параметров сжатия на предыдущих шагах. Необходимо учитывать статистические свойства самого кодируемого в данный момент блока изображения. Только в этом случае алгоритм сжатия данных будет легко адаптироваться к любой, самой неблагоприятной (в смысле возможной степени сжатия при приемлемом качестве) ситуации на маршруте съемки, не выходя при этом из ограничений, обусловленных техническими ресурсами.

2.3. Повышение помехоустойчивости сжатых данных

В системах формирования и передачи цифровых изображений (СФПЦИ) по каналам связи наиболее важными являются следующие алгоритмы преобразования информации. Во-первых, для сокращения объема передаваемой информации по каналу связи с ограниченной пропускной способностью используется алгоритм сжатия. Во-вторых, с целью защиты передаваемых данных от помех канала связи используется алгоритм помехоустойчивого кодирования.

Поскольку данные алгоритмы реализуются в различных подсистемах СФПЦИ, разработка алгоритмов обычно осуществляется независимо друг от друга, без учета взаимной специфики. С одной стороны, при выборе алгоритма сжатия не учитывается устойчивость сжатых данных к сбоям. С другой стороны, все известные алгоритмы помехоустойчивого кодирования (АПК) являются универсальными по отношению к исходным данным и не используют ограниченность набора возможных комбинаций массива сжатых данных известной длины. Кроме того, не учитывается вероятностный характер последствий неисправленных сбоев, которые могут привести как к незначительным искажениям декомпрессированной информации (по сравнению с искажениями, внесенными при сжатии), так и к существенным искажениям вплоть до полной потери изображения.

В данной работе рассматривается комплексный подход к выбору алгоритмов сжатия и помехоустойчивого кодирования для реализации в СФПЦИ. При этом предполагаем следующую модель формирования и преобразования передаваемой в канал связи информации. Пусть изображение формируется путем построчного сканирования. Строки изображения объединяются в кадры размером $N \times M$ пикселов. Кадры подвергаются компрессии с заданной степенью сжатия B_c независимо друг от друга. Далее сжатая информация делится на блоки длиной k бит, каждый из которых преобразуется АПК (при этом размер блока увеличивается до размера $n > k$) и передается в канал связи. Полагаем, что сбои (инвертирование) бит при передаче данных происходят независимо друг от друга и известна вероятность p_0 сбоя одного бита.

Пусть S_i событие, заключающееся в появлении i сбоев в массиве сжатой информации, $P(S_i) = C_n^i p_0^i (1 - p_0)^{n-i}$ – вероятность данного события. Несложно показать, что вероятность наличия сбоев $P(\bar{S}_0) = 1 - P(S_0) = 1 - (1 - p_0)^n$ в передаваемом блоке недопустима велика при значениях $p_0 > 10^{-7}$ (см. рисунок 2.7), что может привести к непредсказуемым последствиям, если сбои останутся неисправленными. Например, при передаче блока с размерами $N = M = 128$ пикселов в режиме $B_c = 2$ бит/пиксел (при $p_0 = 10^{-6}$) приблизительно каждый тридцатый блок окажется со сбоями.

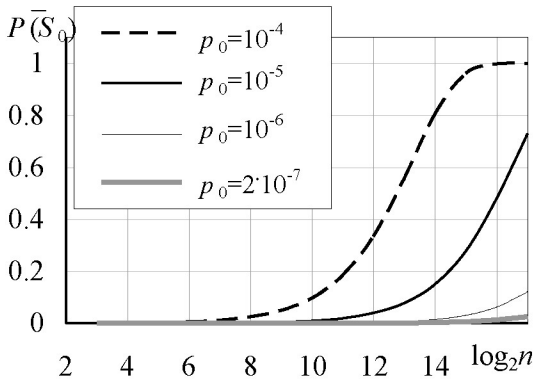


Рисунок 2.7 - Вероятности появления сбоев при передаче по каналу связи блока длиной n бит

Критерии выбора алгоритма сжатия и АПК

Пусть A_c, A_{ecc} – соответственно алгоритмы сжатия и помехоустойчивого

кодирования. Выбор АПК сводится к решению задачи оптимизации

$$\begin{cases} P(Q(A_{ecc}) = 0) \geq P_{bl} \\ \eta(A_{ecc}) \to \min\limits_{A_{ecc}} \end{cases},$$

где $P(Q(A_{ecc}) = 0)$ - достоверность переданной информации, определяемая как вероятность безошибочного декодирования блока (потери информации отсутствуют), P_{bl} - требуемая достоверность передачи блока информации, $\eta = n/k$ - избыточность помехоустойчивого кода.

При выборе алгоритма сжатия и настройке его параметров эффективность оценивается в координатах "степень сжатия – качество восстановленной информации". При заданной степени сжатия B_c (бит/пиксел), ограниченной характеристиками канала связи и последующим внесением избыточности при помехоустойчивом кодировании, выбирается алгоритм, минимизирующий потери качества.

$$\begin{cases} B(A_c) \leq B_c \\ Q(A_c) \to \min\limits_{A_c} \end{cases}$$

В качестве меры качества может использоваться, например, среднеквадратичное отклонение ε_{mse} восстановленного после декомпрессии изображения от исходного.

Выше описанный подход с раздельным выбором алгоритмов требует применения АПК с высокой избыточностью, чтобы исключить потерю кадров вследствие неисправленных сбоев. Это приводит к ограничению допустимой степени сжатия и, соответственно, к ухудшению качества выходной информации. Более оправданной представляется следующая комплексная постановка задачи

$$\begin{cases} B(A_c)\,\eta(A_{ecc}) \leq B_{fr} \\ P(Q(A_c) + Q(A_{ecc}) \leq Q_{fr}) \leq P_{fr} \\ Q_{fr} \to \min\limits_{A_c, A_{ecc}} \end{cases}, \qquad (2.5)$$

где B_{fr}, P_{fr} - требуемые степень сжатия и достоверность передачи всего кадра, Q_{fr} - суммарные потери качества восстановленного кадра вследствие применения алгоритма сжатия и АПК (полагаем соответствующие искажения независимыми). Следует отметить, что согласно данному критерию допускаются небольшие потери

качества и при помехоустойчивом кодировании.

Оценка влияния сбоев на качество выходной информации

При сбоях различных бит в массиве сжатой информации наблюдается большой разброс значений показателя качества выходного изображения. Следовательно, оценка эффективности АПК только числом гарантированно исправляемых бит не совсем правомерна применительно к сжатой информации. Необходимо использовать вероятностные оценки качества выходных изображений, которые можно получить путем моделирования всей цепочки преобразования изображения – сжатия, помехоустойчивого кодирования, внесения сбоев в информацию, декодирования и декомпрессии.

Однако непосредственное моделирование сбоев с малой вероятностью (при $p_0 B_c NM << 1$) требует большого количества вычислительных экспериментов. Вместо этого предлагается проводить моделирование с априорно заданным количеством сбоев в массиве кодированной информации. Ограничиваясь при моделировании событиями S_i, $i = \overline{1, I}$, можно построить гистограммы распределения выходного показателя качества и получить оценки условных вероятностей $P(Q \le Q_{fr} / S_i)$. При этом необходимое количество моделируемых событий I определяется из условия заданной точности

$$\sum_{i=I+1}^{n} P(S_i) < \varepsilon .$$

Так, например, для моделирования сбоев в блоке длиной $n = 2^{15}$ при $p_0 = 10^{-5}$ достаточно ограничиться количеством $I = 4$, чтобы обеспечить погрешность менее $2 \cdot 10^{-5}$.

Окончательно качество выходного изображения оценивается с помощью интегральной гистограммы

$$F(Q_{fr}) = \sum_{i=0}^{I} P(Q \le Q_{fr} / S_i) P(S_i). \qquad (2.6)$$

При сравнении различных алгоритмов удобнее пользоваться интегральными оценками вероятностей. Задавая некоторые пороговые значения Q_1, Q_2, введем следующую классификацию искажений:

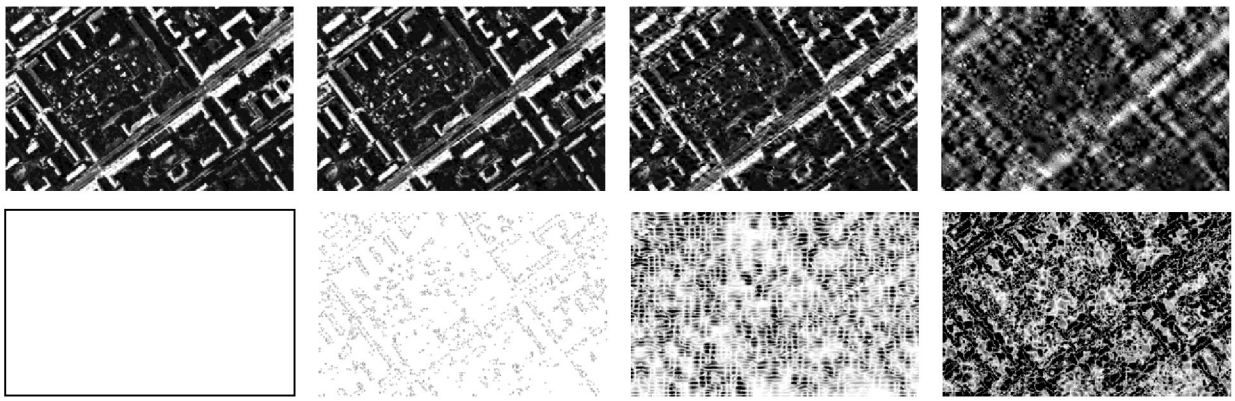
1) $Q = 0$ - изображение не содержит искажений,

2) $0 < Q \le Q_1$ - изображение содержит незначительные искажения (без потери визуального качества),

3) $Q_1 < Q \le Q_2$ - изображение содержит существенные искажения, тем не менее, изображение может быть пригодно для использования в каких-либо целях,

4) $Q > Q_2$ - изображение содержит искажения, которые делают информацию непригодной для дальнейшего использования.

Пусть $\Omega_0, \Omega_1, \Omega_2, \Omega_3$ – соответственно события, заключающиеся в попадании изображения в указанные выше классы. Тогда для сравнительного анализа алгоритмов сжатия и АПК можно использовать оценки вероятностей данных событий, легко рассчитываемые с помощью (2.6). На рисунке 2.8 показаны примеры изображений, принадлежащие описанным классам (кроме того, иллюстрирующие влияние неисправленных сбоев на качество восстановленной после компрессии методом на основе вейвлет-преобразования).



а) класс $\Omega_0, \varepsilon_{mse} = 0$ *б) класс* $\Omega_1, \varepsilon_{mse} = 3.1$ *в) класс* $\Omega_2, \varepsilon_{mse} = 22.3$ *г) класс* $\Omega_3, \varepsilon_{mse} = 84.6$

Рисунок 2.8 - Примеры изображений, различных классов (сверху – восстановленные после декомпрессии со сбоями; снизу - модули разностей изображений, восстановленных со сбоями типа S_2 *и без сбоев)*

Повышение помехозащищенности сжатых изображений

В [10] предложен способ повышения помехозащищенности сжатых изображений, согласно которому информация разделяется на две части – растровую и служебную, включающую заголовок изображения, параметры алгоритмов и т.д. Доля служебной информации в кадре составляет не более 1%, но она обладает особой ценностью для восстановления изображения - ее повреждение приводит практически

всегда к потере изображения. Следовательно, для защиты служебной информации необходимо использовать эффективные АПК (например, коды Боуза-Чоудхури-Хоквингема (БЧХ), Рида-Соломона [14]), обеспечивающие требуемую надежность передачи данных.

При гарантированном восстановлении служебной информации могут быть снижены требования к защите растровых данных при условии, что искажения вследствие неисправленных сбоев незначительны по сравнению с искажениями, внесенными при сжатии. При этом согласно критерию (2.5) будет уменьшена избыточность помехоустойчивого кода, улучшена степень сжатия B_c, и в итоге повышено качество выходного изображения.

Дополнительные возможности по повышению помехозащищенности предоставляет иерархическая структура сжатых данных. Так, в предлагаемом ИСИ-методе сжатые данные формируются последовательно по иерархическим уровням изображения, начиная со сжатия прореженного в 2^{L-1} изображения (где L – число иерархических уровней). Поскольку при переходе на очередной уровень объем данных увеличивается приблизительно в 4 раза, то также растет и вероятность наличия сбоев на очередном уровне. Однако даже при невозможности исправления сбоев на младшем уровне изображение не теряется полностью, а может быть получено путем интерполяции восстановленных старших уровней. На рисунках 2.9а-в показаны примеры декомпрессированных изображений со сбоями (типа S_2 - на младших иерархических уровнях $l=0$ и $l=1$), восстановленного до различных иерархических уровней l, а также разностные изображения (модули разностей изображений, восстановленных со сбоями и без). В случае невозможности устранения сбоев использование интерполяции позволяет восстановить изображение (рисунок 2.9 г), хотя и с худшим разрешением.

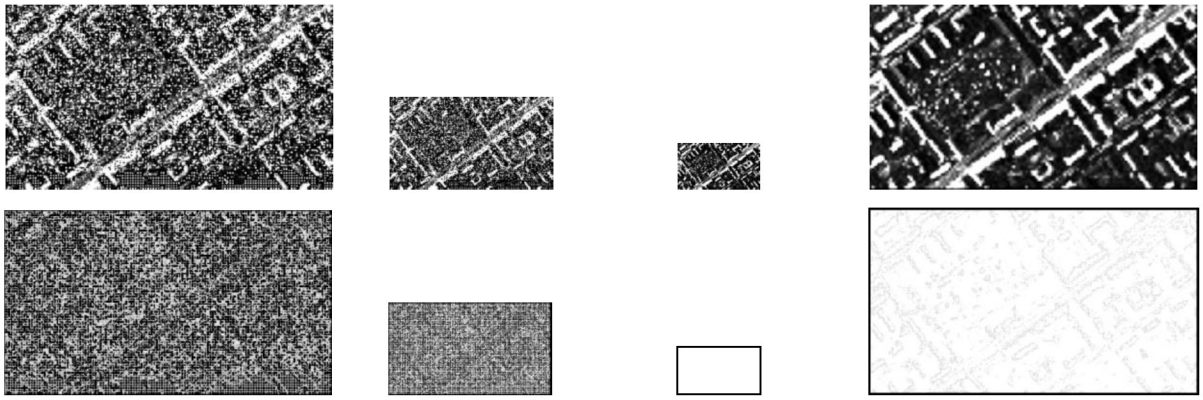
Результаты исследований

Проведено исследование трех методов сжатия цифровых изображений – JPEG, на основе вейвлет-преобразования и ИСИ-метода, доработанного с целью повышения помехозащищенности сжатых данных. Модификация ИСИ-метода заключалась:

1) во внесении в служебную информацию контрольных признаков для каждого иерархического уровня (вычисление растровых признаков на рисунке 2.1);

2) в выделении и кодировании служебной информации с помощью БЧХ-кодов

(вычисление синдромных признаков на рисунке 2.1).



а) $l=0,\ \varepsilon_{\text{mse}}=64.5$ *б)* $l=1,\ \varepsilon_{\text{mse}}=57.1$ *в)* $l=2,\ \varepsilon_{\text{mse}}=0$ *г)* $\varepsilon_{\text{mse}}=31{,}6$

Рисунок 2.9 - Использование интерполяционного восстановления при неустранимых сбоях (сверху – декомпрессированные восстановленные до l-го уровня изображения; снизу - разностные изображения)

Вычисление при декомпрессии изображения контрольных признаков и сравнение их с истинными значениями позволяет обнаружить и исправить сбои типов S_1, S_2. На рисунке 2.10 представлены зависимости вероятностей потери иерархических уровней изображения от вероятности сбоя одного бита (при $B_c NM = 2^{15}$ бит). При этом вероятность безошибочной передачи всего изображения составляет $1-P_{loss}$. Легко видеть, что вероятность потери старших уровней пренебрежимо мала.

На рисунке 2.11 представлены результаты исследования влияния сбоев на сжатые изображения для различных методов сжатия. При этом помехоустойчивое кодирование (за исключением кодирования служебной информации в ИСИ-методе) не применялось. Для сравнения помехоустойчивости сжатых данных приведены интегральные гистограммы показателя качества (среднеквадратичного отклонения ε_{mse}) и зависимости вероятностей принадлежности декомпрессированного изображения к классам Ω_0 и Ω_3 от вероятности сбоя p_0.

Результаты демонстрируют значительное превосходство ИСИ-метода по помехозащищенности данных. Так, с вероятностью 0.986 последствия сбоев в сжатой ИСИ-методом информации будут полностью устранены (при $p_0 = 10^{-6}$), а с вероятностью 0.997 погрешность восстановления не превысит 1, что соответствует классу Ω_1. Как показали исследования, влиянием сбоев на качество восстановленного

изображения можно пренебречь на всем диапазоне $p_0 < 10^{-5}$.

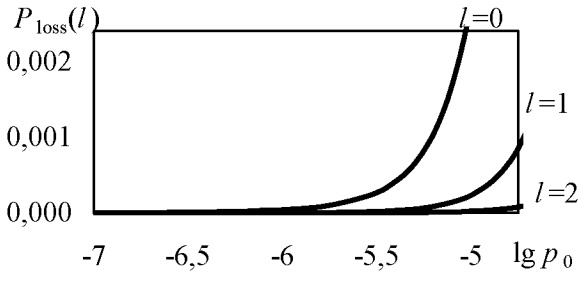


Рисунок 2.10 - Оценка вероятностей потери данных иерархических уровней изображения в ИСИ-методе

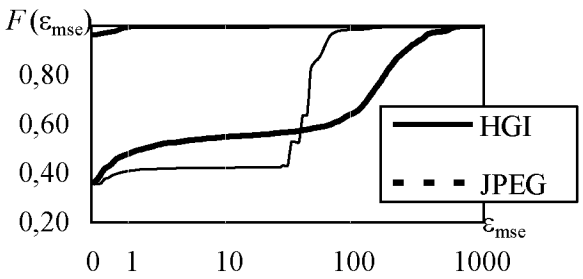


Рисунок 2.11 - Интегральные гистограммы показателя качества (среднеквадратичного отклонения)

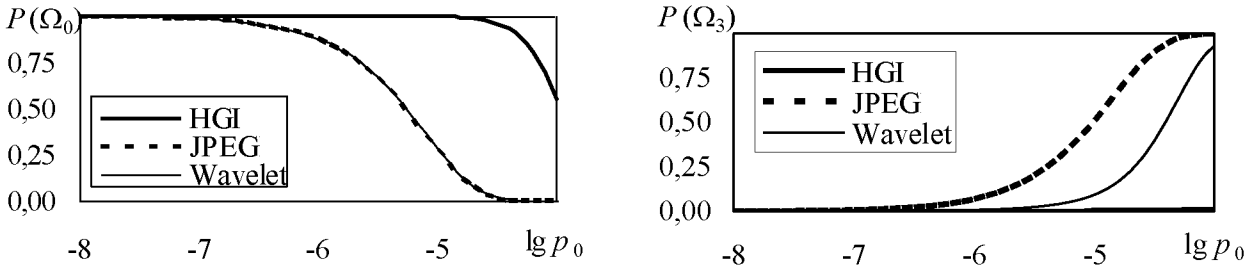


Рисунок 2.12 - Зависимости оценок качества выходной информации от вероятности сбоя одного бита

Для известных методов JPEG и на основе вейвлет-преобразования проблема помехозащищенности сжатых данных является весьма актуальной. С вероятностью не менее 0.5 восстановленные изображения будут неудовлетворительного качества, т.е. принадлежать классам Ω_2 и Ω_3. Для того чтобы защитить изображения сжатые этими методами до уровня ИСИ-метода, необходимо путем помехоустойчивого кодирования ввести избыточность не менее 10%. Однако при фиксированной пропускной способности канала связи в этом случае применение АПК приведет к необходимости большего сжатия, что неизбежно приведет к ухудшению качества выходной информации.

Таким образом, предлагается комплексный подход к выбору алгоритмов сжатия и помехоустойчивого кодирования для передачи сжатых изображений по каналу связи. Эффективность такого подхода особенно значительна при

использовании для сжатия ИСИ-метода, позволяющего оценить и значительно улучшить характеристики СФПЦИ на этапе проектирования.

3. НЕКОТОРЫЕ ЗАДАЧИ НАЗЕМНОЙ ОБРАБОТКИ ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ И КАРТОГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ

В этом разделе в первых двух параграфах рассматриваются задачи обнаружения объектов с аэрофото- и космических снимков, и нормализация их положения. В третьем и четвертом параграфах приводятся примеры решения задач обработки картографических изображений, полученных с отсканированных копий бумажных карт и планов.

3.1. Обнаружение транспортных средств на изображениях, полученных с низколетящего летательного аппарата

Задачей являлась разработка методов и алгоритмов обнаружения, локализации и определения основных параметров (ориентации и размеров) транспортных средств на фотоизображениях, полученных с низколетящего летательного аппарата. Ограничениями задачи являлись:

- разрешение получаемого изображения: 0.05 – 0.10 м,
- высота съемки: 100-200 м,
- скорость передвижения транспортных средств и летательного аппарата: до 100 км/час,
- ориентация оси камеры: надир.

Результатом поиска являлись координаты, размеры и (ненаправленная) ориентация каждого транспортного средства, обнаруженного на изображении. Пример изображения для обработки приведен на рисунке 3.1**Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**.

Для решения задачи была использована методика распознавания, получившая в литературе название «зрение на основе модели» [6, 15]. Для ее реализации использовались несколько групп признаков, описывающих основные характеристики объекта обнаружения и степень его «согласованности» с «идеальной» моделью обнаруживаемого объекта: геометрические, текстурные и структурные. Состав указанных групп признаков приведен ниже.

Группа геометрических признаков

- габаритные размеры объекта;
- дескрипторы границ и признаки границ объекта, рассчитываемые на их основе;
- признаки объекта обнаружения, рассчитываемые на основе дескрипторов границ;
- признаки объекта обнаружения, рассчитываемые на основе формы границ объекта;
- признаки, характеризующие степень выраженности геометрической структуры объекта.



Рисунок 3.1 - Задача обнаружения транспортного средства на аэрофотоснимках: пример изображения, полученного с низколетящего летательного аппарата

Группа текстурных признаков

- признаки, характеризующие локальный уровень выраженности границ;
- признаки, характеризующие степень гладкости функции яркости объекта;
- признаки, характеризующие распределение энергии в пространственно-частотном представлении изображения транспортного средства.

Группа структурных признаков

- признак, отражающий степень симметрии объекта обнаружения;
- признак, отражающий степень продольного подобия сечений изображения объекта;
- число поперечных границ на изображении объекта;
- признак, численно характеризующий отклонение объекта от «эталонной» модели объекта.

Результаты обнаружения транспортных средств с использованием разработанных методов и алгоритмов приведены на рисунке 3.2.



Рисунок 3.2 - Задача обнаружения транспортных средств на аэрофотоснимках: результат обнаружения и определения параметров транспортных средств на изображении, полученном с низколетящего летательного аппарата

Проведенная экспериментальная апробация и исследования подтвердили работоспособность разработанной и реализованной технологии обнаружения транспортных средств на изображениях и показали эффективность методов и алгоритмов, используемых в ее составе. При тестировании программных средств на реальных данных, были получены следующие оценки показателей качества обнаружения транспортных средств на отдельных снимках (среднее число

присутствующих на кадре объектов – 30 штук):

- вероятность верного обнаружения - 0.9467,
- среднее число ложно обнаруживаемых объектов на кадре – 2.8,
- вероятность ложного обнаружения - 0.01.

Указанные оценки вероятностей были получены по изображениям, полученным при усложненных (по отношению к указанным выше) условиях съемки. В частности, высота съемки составляла около 35 метров (при предполагаемой высоте в 100 м и выше) и угол наклона камеры по отношению к вертикали составлял 10-20 градусов (при предполагаемой съемке в надир, то есть строго вертикально).

Для повышения скорости обнаружения и разработки программных средств, работающих в режиме времени, близком к реальному, были использованы следующие методики:

- при построении дескрипторов границ объектов вычисления производились с использованием полиномиальных MBC-базисов;
- для ускорения процесса распознавания использовалась процедура двухэтапного обнаружения и распознавания.

3.2. Нормализация положения движущихся объектов при последовательной регистрации RGB-каналов

В этом параграфе разрабатывается алгоритм нормализации изображений движущихся объектов, полученных в результате последовательной регистрации RGB-каналов. Алгоритм позволяет объединять разноцветные копии движущихся объектов и вычислять их характеристики. Алгоритм основан на использовании взаимосвязи координат, размеров и ориентации разноцветных копий объектов.

В некоторых системах формирование цветных изображений осуществляется посредством последовательной регистрации цветовых каналов. К таким системам относятся некоторые системы дистанционного зондирования, в которых один и тот же участок земной поверхности последовательно регистрируется датчиком с различными светофильтрами. В результате получаются RGB-каналы [11], сформированные в различные моменты времени.

При таком способе съемки, главной проблемой синтеза цветного изображения из отдельных RGB-каналов является невозможность автоматическом совмещения движущихся объектов (например автомобилей). В результате, на

цветосинтезированном изображении присутствуют три RGB-копии каждого движущегося объекта (красная, синяя и зеленая), соответствующие различным моментам времени (рисунок 3.3).

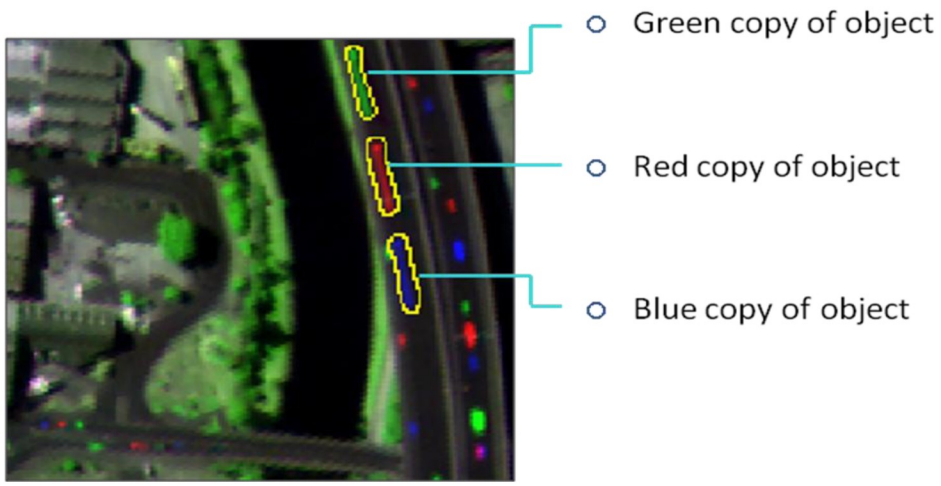


Рисунок 3.3 - Исходное изображение

Алгоритм разработан для нормализации изображений автомобилей на космических снимках, но может быть (с небольшими модификациями) настроен и для обработки объектов другого типа.

При обнаружении и цветокоррекции RGB-копий объектов возникает ряд трудностей:

- Линейные размеры некоторых автомобилей (например фуры и легкового автомобиля) могут различаться в несколько раз.
- Неоднородность яркости объектов даже внутри одной цветовой копии.
- Из-за невысокого разрешения космических снимков цвет и форма пятна яркости, описывающего автомобиль, сильно варьируется даже для автомобилей одинакового размера, из-за чего сложно определить их границы.
- Невысокое разрешение также является серьезным препятствием при определении скорости и направления движения объектов.
- Разноцветные копии объектов накладываются друг на друга. В результате кроме RGB-копий автомобиля, могут появиться его желтая и малиновая копии.
- На космическом снимке присутствуют цветовые пятна и объекты (например, участки растительности), похожие на искомые объекты.

В описанных условиях распространенные алгоритмы обнаружения (например, корреляционные) малоприменимы, стандартные алгоритмы анализа движения также

не работают, потому что последовательные кадры относятся к различным спектральным диапазонам и по ним не удается осуществить привязку по фоновой составляющей. Описанный далее алгоритм нормализации изображений движущихся объектов позволяет объединить три разноцветные копии объекта в одну. Алгоритм основан на использовании взаимосвязи координат, размеров и ориентации копий объектов различного цвета.

Описание алгоритма

При поиске все операции производятся в цветовом пространстве HSI [11] (цветность, насыщенность, яркость). Эксперименты показали, что надежнее всего обнаруживается красная копия объекта (на городском пейзаже нет ярко красных объектов) поэтому сначала ищутся все красные копии, а затем они используется для поиска синих и зеленых копий объектов.

Поиск красных копий объектов включает следующие этапы:

- построение стартовых областей
- расширение стартовых областей
- сегментация
- фильтрация областей по размерам

На этапе 1 производится выделение пикселов, для которых компоненты H, S, I попадают в заданные диапазоны, соответствующие насыщенному красному цвету:

$$M_1 = (h_1 < H < h_2) \wedge (s_1 < S < s_2) \wedge (i_1 < I < i_2) \tag{3.1}$$

где $\wedge$ - конъюнкция, а M_1 - булевская маска, описывающая найденные стартовые области. При настройке алгоритма диапазоны (h_1, h_2), (s_1, s_2), (i_1, i_2) выбираются как можно более узкими, но из условия, чтобы внутри всех (или почти всех) красных копий объектов присутствовала хотя бы одна отмеченная точка. В этом случае довольно мало стартовых областей попадает мимо объектов, но сами объекты не покрываются целиком стартовыми областями, поэтому области необходимо расширять.

На этапе 2 происходит итерационное расширение областей, обнаруженных на этапе 1:

$$O_2^0 = M_1,\ O_2^j = D(O_2^{j-1}) \cap M_2,\ j = 1,2,3,\ldots \tag{3.2}$$

где $D(\cdot)$ - оператор дилатации, а M_2 - маска, ограничивающая расширение областей.

Эта маска вычисляется аналогично маске M_1, но с более широкими диапазонами h_1, h_2 , s_1, s_2 , i_1, i_2 . Итерационный процесс останавливается на шаге *J*, на котором промежуточные результаты O_2^J и O_2^{J-1} совпадают. Использование такой процедуры позволяет осуществить “заливку” краев объектов.

На этапе 3 происходит сегментация бинарного изображения O_2^J по четырехсвязному критерию. В результате образуется список областей, каждая из которых может быть искомой красной копией объекта.

На этапе 4 оцениваются площади и размеры найденных областей. В качестве оценки размеров используются полуоси аппроксимирующих эллипсов, которые вычисляются на основе степенных моментов. Области, которые имеют площадь или размеры, не похожие на размеры автомобиля, отбрасываются. Остальные области считаются искомыми красными копиями объектов, которые затем используются для поиска синих и зеленых областей.

Поиск синих и зеленых копий объектов в целом аналогичен поиску красных, за исключением двух отличий.

Во-первых, стартовые области ищутся только в эллипсоидальной окрестности уже найденных красных копий объектов. Ориентация каждой такой эллипсоидальной области совпадает с ориентацией красной копии объекта, вокруг которой она строится (ориентация определяется на основе степенных моментов). Размер эллиптической области задается при настройке алгоритма и определяется максимальным расстоянием между копиями одного и того же объекта.
Использование такого ограничения позволяет сильно сузить зону поиска и существенно сократить количество ложных стартовых областей.

Во-вторых, после этапа 4 добавляется еще один дополнительный этап. На этом этапе отбрасываются синие и зеленые копии объектов, площади и размеры которых слишком сильно отличаются от площади и размеров красной копии, в эллипсоидальную окрестность которой они попадают.

Без использования описанного выше дополнительного этапа, многие изображения растительности порождают ложные обнаружения зеленых копий объектов.

После того, как получены списки областей, описывающих RGB-копии объектов, производится цветовая коррекция этих областей. Красные копии объектов

(они обнаруживаются наиболее уверенно) заменяются на серые соответствующей яркости. Синие копии стираются: в пределах соответствующих областей синяя компонента заменяется на полусумму красной и зеленой. Аналогично стираются зеленые копии объектов. Такой способ стирания не допускает сильных искажений ложно принятых за автомобиль небольших участков растительности.

В результате на изображении вместо трех разноцветных копий объекта остается один объект, имеющий более естественную окраску (рисунок 3.4):

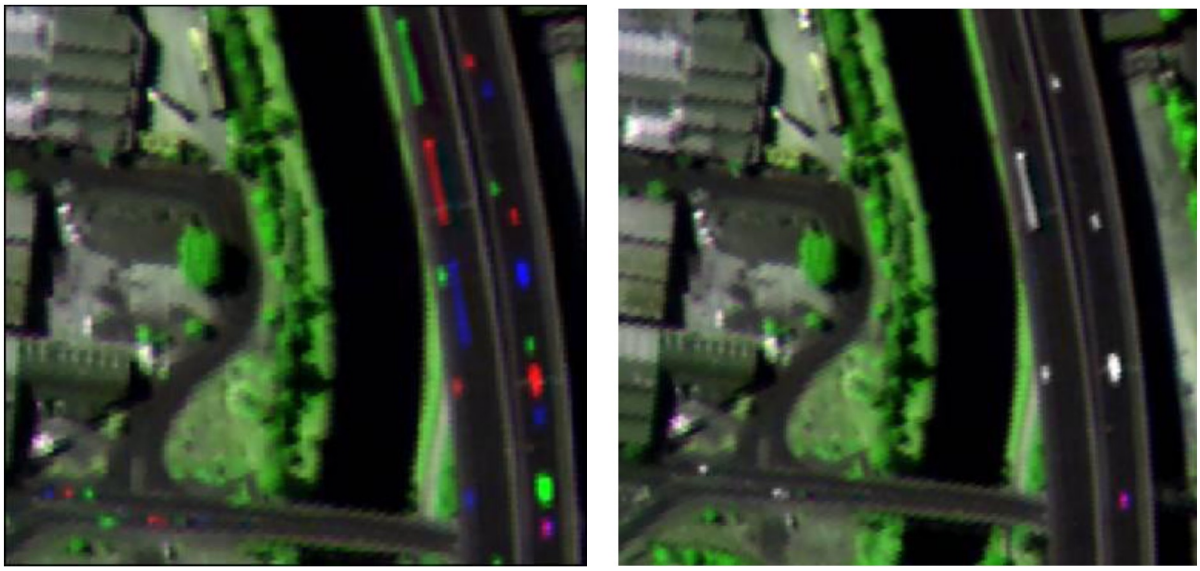


Рисунок 3.4 - Исходное изображение и результат работы алгоритма

Подведем итог рассматриваемым методам. Разработан алгоритм нормализации изображений движущихся объектов, полученных в результате последовательной регистрации RGB-каналов. Алгоритм показывает высокое качество работы при нормализации красных и синих копий объектов. Благодаря использованию взаимосвязи координат, размеров и ориентации копий объектов различного цвета, удается свести к минимуму искажения изображений растительности, а также измерить скорость передвижения транспортных средств.

3.3. Метод автоматической геопривязки и калибровки картографических изображений

В этом параграфе рассматривается один из этапов создания и обновления цифровых карт - геопривязка и калибровка картографических изображений (отсканированных топографических планов).

Источниками данных в геоинформационных системах (ГИС) являются бумажные топографические карты и планы, данные дистанционного зондирования, а также данные наземных наблюдений. Для создания и обновления крупномасштабных планов городов основным из них по-прежнему являются бумажные копии топографических планов, процесс перевода которых в цифровую форму включает три основных этапа:

1) сканирование бумажного оригинала топоплана (планшета), получение цифрового растрового изображения;
2) привязка по опорным точкам (по координатной сетке) и трансформация;
3) векторизация.

Второй этап традиционно выполняется с помощью интерактивных программных средств, позволяющих оператору указать на растровом изображении положение опорных точек, а затем выполнить геометрическую трансформацию (обычно используется ортогональный, аффинный, или кусочно-проективный вид преобразования).

Для расчета параметров аффинного преобразования по множеству вручную введенных опорных точек используется минимизация функционала,

$$\varepsilon^2 = \sum_{i=1}^{N}[(X_i^f - (a_1 x_i + a_2 y_i + a_3))^2 + \quad + (Y_i^f - (a_4 x_i + a_5 y_i + a_6))^2] \underset{a_j}{\rightarrow} \min,\ j = \overline{1,6}, \qquad (3.3)$$

где a_j - неизвестные параметры, которые находятся методом наименьших квадратов; (x_i, y_i) - координаты опорных точек на растре, полученные при указании оператором точки на изображении; (X_i^f, Y_i^f) - физические координаты опорных точек, значения которых вводятся оператором; ε^2 - среднеквадратическая ошибка преобразования; N – количество опорных точек (не менее 3).

Представленный в параграфе метод автоматического поиска опорных точек и геометрической коррекции изображений позволяет полностью исключить затраты времени оператора на выполнение второго этапа для источников информации в виде бумажных карт и планов (рисунок 3.5). Исходные изображения характеризуются наличием рамок листа – внешней (толстой) и внутренне (тонкой), обозначающей границы плана, а также координатной сеткой в виде крестов, нанесенных через каждые 10 см. Эти кресты и являются опорными точками привязки, положение которых указывает оператор. Задача сводится в автоматическом поиске положения

заданного количества крестов и указания их центра.

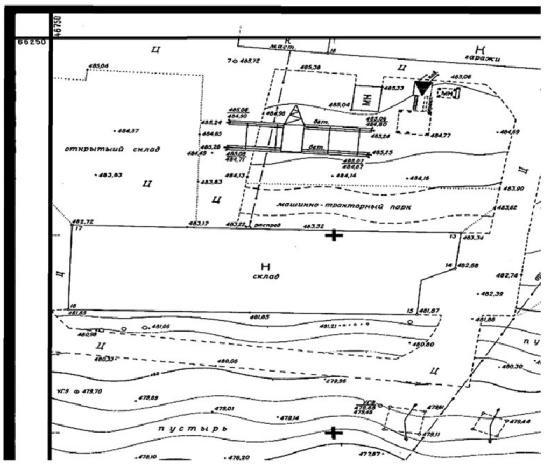


Рисунок 3.5 - Часть изображения топографического плана

Описание метода

Метод автоматической геопривязки состоит из следующих этапов:

1) нахождение положения «толстой» внешней и «тонкой» внутренней рамки планшета, расчет ориентировочного угла поворота при сканировании;
2) расчет начальных положений крестов, учитывая известную информацию об их количестве и расстоянии между ними;
3) Уточнение положения крестов и отброс ошибочно найденных;
4) аффинная или иная трансформация с записью выходного изображения и файла привязки.

Наиболее интересным с точки зрения алгоритмов распознавания представляется третий этап уточнения положения крестов по их предварительному положению в условиях сильных помех (рисунок 3.6)

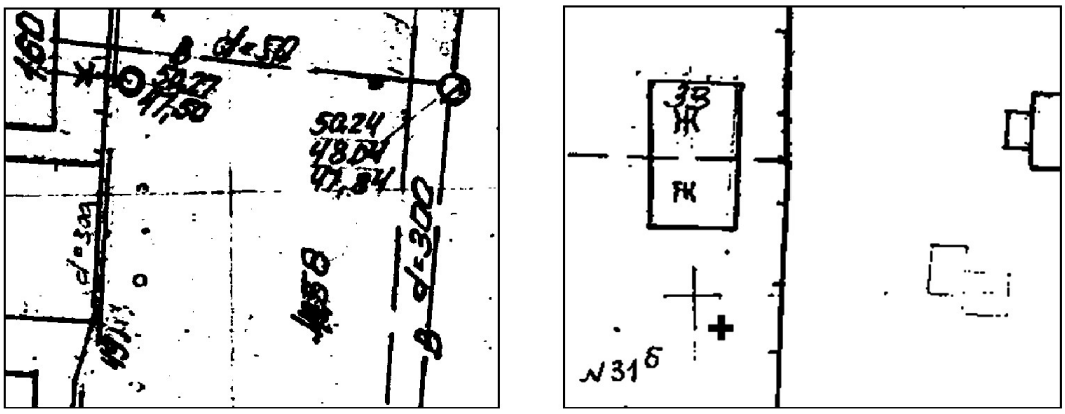


Рисунок 3.6 - Примеры плохо прорисованного крестов

Задача уточнения положения крестов сводится к задаче поиска в некоторой

окрестности изображения фигуры, инвариантной к повороту на углы, кратные 90 градусам Центр креста определяется следующим образом:

$$(n_1^*, n_2^*) = \arg\min_{(n_1,n_2)\in D} g^*(n_1, n_2), \tag{3.4}$$

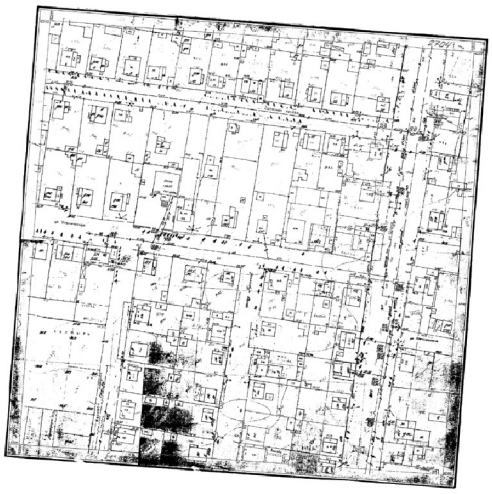
где

$$g^*(n_1, n_2) = \sum_{(m_1,m_2)\in M} g(m_1, m_2, n_1, n_2) = \sum_{(m_1,m_2)\in M} \max \begin{Bmatrix} v(n_1 + m_1, n_2 + m_2), \\ v(n_1 - m_2, n_2 + m_1), \\ v(n_1 - m_1, n_2 - m_2), \\ v(n_1 + m_2, n_2 - m_1), \end{Bmatrix}. \tag{3.5}$$

Здесь D - область поиска объекта, M - область крестообразной формы толщиной в 3-5 раз больше, чем толщина находимого объекта-креста.

После применения указанной процедуры будет найден набор точек, предположительно являющихся центрами крестов.

Из полученного набора с помощью модифицированного метода последовательного присоединения (Step-Up метода) на основе критерия (3.3) формируется окончательный набор опорных точек (reference point), по которому затем производится трансформация растра (обычно аффинная или кусочно-линейная), то есть компенсация геометрических искажений, внесенных сканированием (рисунок 3.7), и запись выходного файла привязки.



а) изображение планшета после сканирования;



б) изображение планшета после нахождения крестов и проведенной на их основе аффинной трансформации;

Рисунок 3.7 - Пример трансформации растра

3.4. Автоматическое распознавание контуров зданий на картографических изображениях

Задача массовой автоматической векторизации пространственных объектов с бумажных оригиналов встречается в процессе работ по переводу бумажных архивов топографических планов территории в цифровой вид. В предыдущем параграфе рассмотрена задача автоматизации процесса геопривязки. В этом параграфе рассмотрена задача автоматической векторизации объектов специального типа – зданий при наличии большого числа мешающих факторов в условиях Недостатком использования традиционных методов распознавания является их неустойчивость к шумовым и иным искажениям, плохая приспособленность для распознавания карт и планов со сложной нагрузкой, а также, зачастую, потеря топологических свойств конструкций (наличие «петель», «недоводов», несовпадающих контуров соседних объектов, лишних точек). В параграфе представлен метод, основанный на выделении площадных объектов по заданным геометрическим признакам с соблюдением их топологической связности.

На исходном изображении здание представляется в виде замкнутого контура с ортогональными сторонами, содержащего внутри надписи, полученные по значениям семантических атрибутов: номер дома, количество этажей и др. Пример объектов, подлежащих распознаванию, приведен на рисунке 3.8.

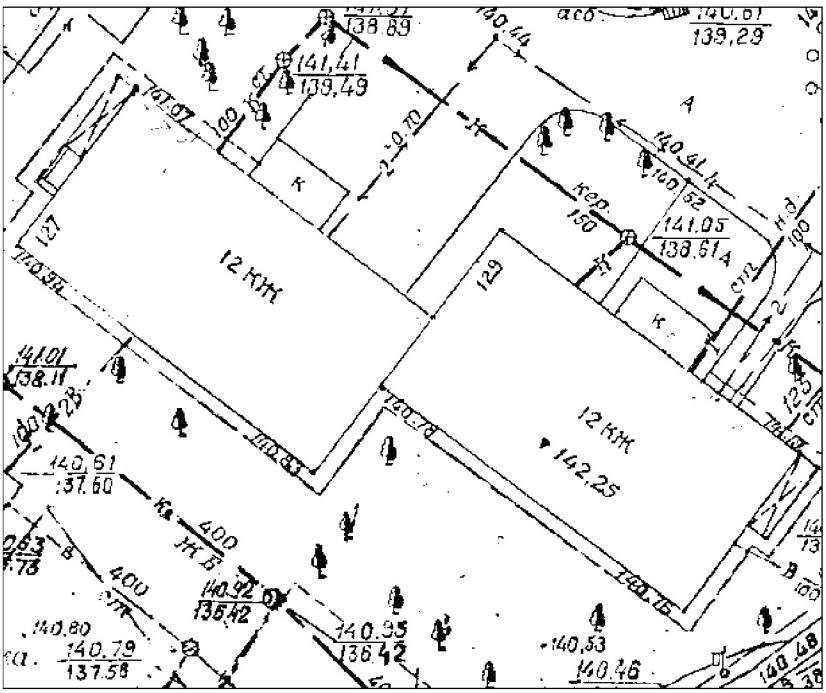


Рисунок 3.8 - Пример распознаваемых объектов

К процедуре распознавания предъявлялись следующие требования:

1) Строгие ограничения на допустимое положение результирующего контура относительно растрового оригинала с сохранением прямоугольности.
2) Соблюдение топологических отношений между граничащими объектами.
3) Устойчивость к шумам, разрывам линий на исходном изображении, наличию примыкающих к контуру посторонних объектов (надписей, элементов оформления и др.)
4) Высокая скорость векторизации и минимизация количества точек в результирующем представлении объектов.

Описание используемого метода

Исходными бинарными изображениями являлись результаты сканирования топографических планов. Данные изображения подвергались предварительной обработке в виде применения морфологических операций, призванных устранить шумы, мелкие объекты и разрывы.

Обнаружение объектов в данной работе базируется на первичной векторизации изображения и выделении объектов рассматриваемого класса из первичного векторного описания. Метод состоит из последовательного применения следующих процедур:

1) Скелетизация исходного растрового изображения.
2) Построение первичной векторной модели.
3) Устранение потенциальных разрывов.
4) Поиск всех базовых замкнутых контуров на векторной модели.
5) Выделение из множества замкнутых контуров и их объединений контуров зданий.
6) Постобработка данных с целью оптимизации результирующего представления.

Скелетизация

Формируемый на данном этапе скелетный образ должен обладать свойством строгой 8-связности и сохранять топологическую структуру исходного изображения. Для скелетизации (рисунок 3.9) был выбран метод попиксельного итеративного утончения, удовлетворяющий указанным требованиям. Следует отметить, что выбор алгоритма по точности представления растровых объектов не является в данном случае принципиальным, в силу того, что на этапе постобработки используется ряд

процедур, направленных на улучшение полученных векторных данных. [26]

Построение векторной модели

Первоначально на данном этапе осуществляется выделение особых точек на скелете (имеющих число связности, не равное двум). По найденному множеству создается полное описание скелетной структуры путем поиска для каждой особой точки всех исходящих из неё скелетных линий (поиск осуществляется до следующей особой точки). Полученная система скелетных линий подвергается уточнению, состоящему в генерализации каждой из линий и формировании на ней множества существенных (поворотных) точек. В результате получается векторная модель изображения в виде графа $G(V,E)$, где

V - множество вершин графа $V = V_0 \cup V_1$, V_0 - множество особых точек, каждая из которых соответствует изолированной, концевой или точке ветвления на скелете изображения,

V_1 - множество промежуточных существенных точек,

E - множество ребер графа, аппроксимирующих скелет изображения [13].

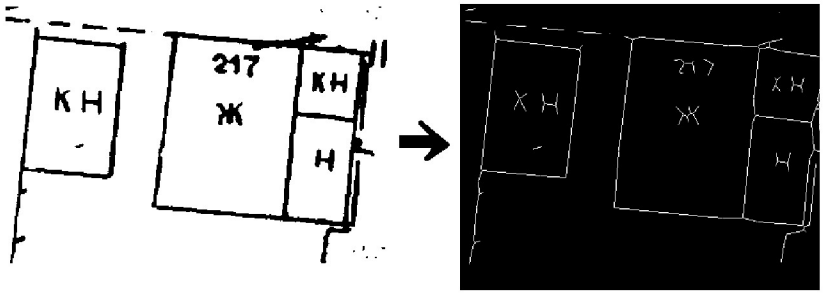


Рисунок 3.9 - Результат скелетизации

Устранение разрывов

Исходя из специфики исходных растровых данных, необходимо учитывать возможность появления разрывов на объектах изображения. В рамках рассматриваемой задачи можно ориентироваться на устранение разрывов между коллинеарными или ортогональными линиями. Таким образом, устранение разрывов может быть сведено к сравнению направлений ребер графа и восстановлению недостающих ребер в данных направлениях. В результате мы, возможно, получим лишние замкнутые контура, которые эффективно отсекаются на пятом этапе объединения и классификации.

Выделение замкнутых контуров

В данном случае использовался двухэтапный метод, позволяющий сократить вычислительные затраты и при этом дающий хорошие практические результаты.

На первом этапе осуществляется построение базиса пространства циклов Z_i для векторной модели изображения. Для этого производится растеризация полученной первичной векторной модели и составление растрового описания всех базисных циклов путем разметки результирующего бинарного изображения. Определение векторного описания множества базисных циклов Z_i сводится к поиску ребер, имеющих заданный идентификатор в качестве одной из разделяемых областей, и составлению из них целостного объекта.

Контуры зданий представляют собой объединения базовых контуров. На графе они представляются в виде в виде разложения по базисным циклам в следующем виде:

$$C = Z_{i_1} \oplus Z_{i_2} \oplus \ldots \oplus Z_{i_N} \tag{3.6}$$

В данной записи операция сложения по модулю 2 определяется как симметрическая разность множеств ребер заданных циклов. Таким образом, задача выделения контуров зданий сводится к поиску множества $Z_{i_t}{}_{t=0}^{N}$, обеспечивающего заданный критерий принадлежности к классу распознаваемых объектов. При этом может быть использована следующая процедура классификации контуров. Для ребер заданного контура C производится определение их ориентаций и группировка полученных значений по близости, что соответствует попаданию направлений ребер в некоторые сектора. Очевидно, что для контура с преимущественно прямыми углами большая часть ребер будет попадать в два сектора. Для классификации C производится сравнение отношения суммарной длины ребер L_0, попавших в данные сектора, к периметру контура L, с некоторым пороговым значением γ_0. Правило для принятия решения о том, что контур является зданием, при этом имеет следующий вид:

$$\gamma\ C\ = \frac{L_0}{L} > \gamma_0 \tag{3.7}$$

Для определения множества $Z_{i_t}{}_{t=0}^{N}$ применяется алгоритм эвристического поиска, в котором в качестве функции оценки текущего состояния выступает $\gamma\ C\ $.

После окончания поиска осуществляется проверка критерия (3.7) для контура C' с наилучшей оценкой $\gamma\ C'$ и принятие решения о том, является ли этот контур зданием [16].

Постобработка данных

В результате построения векторной модели и обнаружения контуров возникает задача улучшения контуров с целью приведения их к виду соответствующему требованиям, определенным в задании. Практически это сводится к упрощению описания результирующего подграфа зданий (см. **Ошибка! Источник ссылки не найден.**). При этом используется ограничение на расхождение между исходными растровыми линиями и векторным описанием (не больше заданного ε).

Целью данного этапа является улучшение полученных векторных данных, состоящее в удалении лишних вершин, спрямлении углов контуров и правильном позиционировании данных на исходном растре. Пусть $C = C_i \ {}^{N}_{j=1}$ - множество найденных контуров зданий, $R_i = r_j \ {}^{N_i}_{j=0}$ - множество ребер *i*-го контура, $\mathrm{A}_i = \alpha_k = \angle(r_k, r_{k+1}) \,|\, r_k, r_{k+1} \in R_i$ - множество углов *i*-го контура. Тогда оптимизация векторного представления может быть определена минимизацией целевой функции следующего вида:

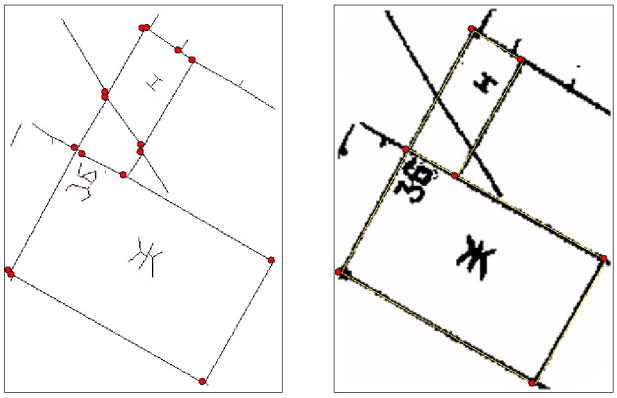
$$R(G) = \sum_{i=0}^{N-1} \sum_{k=0}^{N_i-1} \delta\left(\left|\alpha_k - \frac{\pi n}{2}\right| < \varepsilon\right) \cdot \left|\alpha_k - \frac{\pi n}{2}\right| \qquad (3.8)$$

Данная функция, по сути, отвечает за уменьшение отклонения углов контуров от значений $\frac{\pi n}{2}$. Оптимизация осуществляется поиском для каждой вершины положения, при котором обеспечивается минимум $R(G)$. При определении положения необходимо учитывать допустимость новых векторных данных, возможность удаления и слияния вершин.

На последнем этапе требуется повысить соответствие между полученной векторной моделью и исходным растровым изображением, используя только параллельный сдвиг ребер с сохранением всех ортогональных углов. Для этого производится дистантное преобразование исходного изображения, вычисление интегральных характеристик положения ребра $\lambda(G)$ по данному изображению и выполнение параллельного сдвига ребра в направлении увеличения $\lambda(G)$. Применение

такой процедуры приводит к перемещению ребра по направлению к средним линиям исходных растровых объектов [1].

На основе предложенного подхода было разработано программное обеспечение автоматического распознавания зданий, которое успешно прошло опытную эксплуатацию в ряде организаций при создании векторных планов г. Самары. Основная трудность состоит в извлечении из результатов множества контуров, соответствующих зданиям. На сегодня это делается оператором, при этом вероятность правильного распознавания составляет более 95%, а ручного труда сокращается в 2-3 раза. На рисунке 3.11 приведены результаты векторизации четырех зданий на растровом изображении рисунка 3.8.



а) Исходная векторная модель *б) После постобработки*

Рис 3.10. Результат постобработки

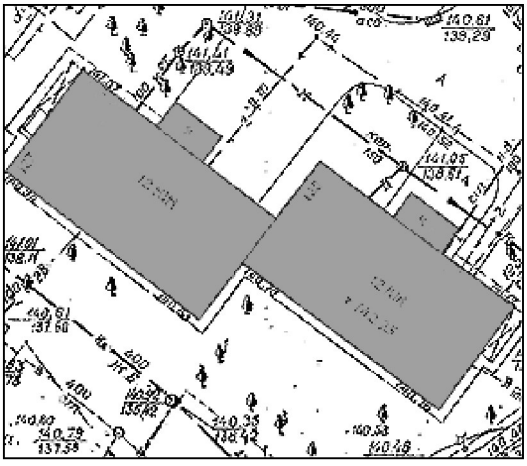


Рисунок 3.11 - Результаты распознавания

4. РЕГИОНАЛЬНЫЙ ЦЕНТР ПРИЕМА И ОБРАБОТКИ ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ

В этом параграфе рассматривается организация регионального центра приема и обработки данных дистанционного зондирования. Описывается организационная структура на базе Поволжского центра космической геоинформатики, организация базы данных дистанционного зондирования (региональный банк космических снимков), а также принципы построения программного обеспечения для работы с ним.

4.1. Общая характеристика Центра

Поволжский Центр космической геоинформатики (ПЦКГ) был создан в 2006 году по инициативе Правительства Самарской области Российской Федерации на базе Самарского государственного аэрокосмического университета. Главная цель создания Центра – решение следующих проблем:

- развитие космических информационных технологий в целях удовлетворения потребностей экономики Самарской области;
- геоинформационная поддержка управленческой деятельности и принятия решений органами власти Самарской области;
- развитие регионального, всероссийского и международного рынка услуг в области геоинформатики и космических информационных технологий;
- обучение студентов и повышение квалификации специалистов в области космических информационных технологий для предприятий Самарской области.

Центр позволяет найти новые эффективные решения указанных выше проблем за счет координации деятельности региональных органов власти, космической индустрии, высшей школы (Самарского государственного аэрокосмического университета) и наукоемкого бизнеса.

На момент написания этого текста Центр работает с набором из 10 спутников в режиме прямого сброса информации. Эти спутники обеспечивают Центр космическими изображениями региона, полученными в различных спектральных диапазонах, с различным детальностью и периодичностью (таблица 4.1).

Среди текущих задач, решаемых Центром с использованием спутниковых

изображений и современных геоинформационных технологий, можно отметить:

- организация и контроль использования земель, наблюдение за угодьями;
- инвентаризация посевов, мониторинг зерновых, оценка сельскохозяйственной деятельности;
- планирование городов и зонирование, мониторинг строящихся зданий;
- создание и обновление цифровых топографических и тематических карт;
- проектирование и мониторинг охраняемых зон и т.д.

Ниже описываются некоторые технические решения, относящиеся к структуре Центра, его программному и информационному обеспечению.

Таблица 4.1 - ДДЗ спутников, принимаемых в ПЦКГ

Спутник	Страна-разработчик	Детальность снимков (метров)	Число каналов, диапазон	Полоса обзора (км)	Средняя периодичность съемки (суток)
Terra (MODIS)	США	250-1000	36 (ИК, видимый)	2300	0.5 - 1
Aqua (MODIS)	США	250-1000	36 (ИК, видимый)	2300	0.5 - 1
SPOT-2	Франция	10 - 20	3 (ИК, видимый)	60	26
SPOT-4	Франция	10 - 20	3 (ИК, видимый)	60	26
RADARSAT-1	Канада	8 - 100	1 (радио 5.6 см)	50 -500	1 - 6
IPS-P6	Индия	5.8 - 55	4 (ИК, видимый)	23 - 740	5
IPS-P5	Индия	2.5	1 (видимый)	30	5
EROS-A	Израиль	2	1 (видимый)	13.5	3 - 4
EROS-B	Израиль	0.7	1 (видимый)	7	6 - 8

4.2. Региональный банк космических снимков

Для использования космических снимков, полученных со станции приема, необходимо произвести их радиометрическую, геометрическую коррекцию и преобразовать в проекцию, используемую в ГИС. Кроме того, большинство прикладных задач требует одновременной обработки нескольких снимков, создания композитов, единых покрытий рассматриваемой территории с возможностью быстрого доступа и управления набором информационных компонентов. Все это ведет к необходимости эффективной организации базы данных космических снимков.

На базе Поволжского центра космической геоинформатики создан региональный банк космических снимков (РБКС) Земли для их использования в различных отраслях экономики Самарской области.

Основными целями создания РБКС являются:

- автоматизация первичной обработки изображений, в первую очередь, геометрической коррекции и привязки, что позволяет автоматически вводить в банк данных все поступающие изображения;
- ведение базы данных хранения различных типов геопривязанных снимков со сжатием без потерь точности;
- организация бесшовных покрытий территории (композитов, мозаик из множества изображений) за различные периоды для обеспечения быстрого доступа к набору изображений как единому изображению;
- обеспечение доступа и возможности обработки и анализа изображений территории в реальном времени; благодаря уникальному формату сжатия данных и структуре хранения данных обеспечивается практически мгновенное извлечение и визуализация изображений из РБКС
- ведение базы метаданных об условиях съемки, привязке к системе координат, характеристиках спутников, а также векторная карта территории, используемая для поиска и отображения.
- сохранение качества изображений при вводе в банк данных (при первичной обработке и сжатии (компрессии)), что необходимо при многоцелевом использовании изображений.

Основные преимущества РБКС перед от аналогичными сервисами доступа к полным покрытиям территорий (Google Earth, Космоснимки.ру): мультимасшабность, мультивременность, оперативность и точность геопривязки ДДЗ.

При вводе в РБКС изображения подвергались геометрической трансформации с целью приведения к единой картографической проекции (Гаусса-Крюгера) и физической системы координат (СК-42) и к базовым разрешениям (не хуже исходного разрешения снимка), принятым в РБКС (в качестве базовых разрешений приняты следующие: 1 м, 2 м, 4 м, 8 м, 16 м, 128 м, 256 м). При помещении для растрового изображения создавалось и помещалось в РБКС индексное изображение, содержащее информацию о наличии облачности и контуре изображения. Все

изображения в РБКС хранятся в сжатом виде в ИСИ (HGI) -формате, который обеспечивает благодаря иерархической структуре хранения быстроту доступа к изображению в произвольном масштабе.

Кроме растровых изображений в РБКС хранится вся сопутствующая семантическая информация (метаданные) об условиях съемки, характеристиках спутников, а также векторная карта Самарской области (рисунок 4.1).

4.2.1. Автоматическое наполнение регионального банка данных космических снимков в режиме постоянного приема

Задача анализа территорий на основе последовательностей космических снимков находит широкое применение в задачах оценки состояния посевов, природопользования и лесоустройства, предупреждения чрезвычайных ситуаций и др. При этом, как правило, используются снимки среднего разрешения (5-50 м) типа Spot 2/4, IRS P6, которые относительно недороги и могут поступать в режиме постоянного сброса на региональные станции приема. Для совместного анализа снимков необходимо их привести к одному радиометрическому диапазону, к одной системе координат и, желательно, поместить в единое хранилище данных (региональный банк космических снимков - РБКС). Выделим особенности обработки таких снимков:

- орбитальная привязка снимков весьма неточна (ошибка составляет до 500 метров), поэтому невозможно проводить геометрическую трансформацию только на основе этих данных;
- необходимость использования цифровых моделей рельефа (ЦМР) при геометрической трансформации;
- необходимость выделения полезной части снимка и отсечения облаков;
- большое количество снимков (несколько десятков в неделю объемом около 1 Гб на территорию области).

Все это ведет к большому времени ручного труда по приему, радиометрической коррекции и геопривязке снимков.

Цель работы – создать сквозную автоматизированную технологию предварительной обработки космических снимков и их помещения в РБКС, состоящую из следующих основных этапов.

1. Прием, нарезка на фрагменты, радиометрическая коррекция и предобработка данных.
2. Создание маски облачности
3. Автоматическая геопривязка и геометрическая трансформация
4. Создание метаданных и помещение изображения в РБКС
5. Яркостная коррекция изображений (приведение к динамическому диапазону исходного покрытия) – при необходимости.

Основной идеей является использование базового начального покрытия из космических снимков и геометрическая привязка к нему на основе корреляционного сопоставления с использованием информации о ЦМР.

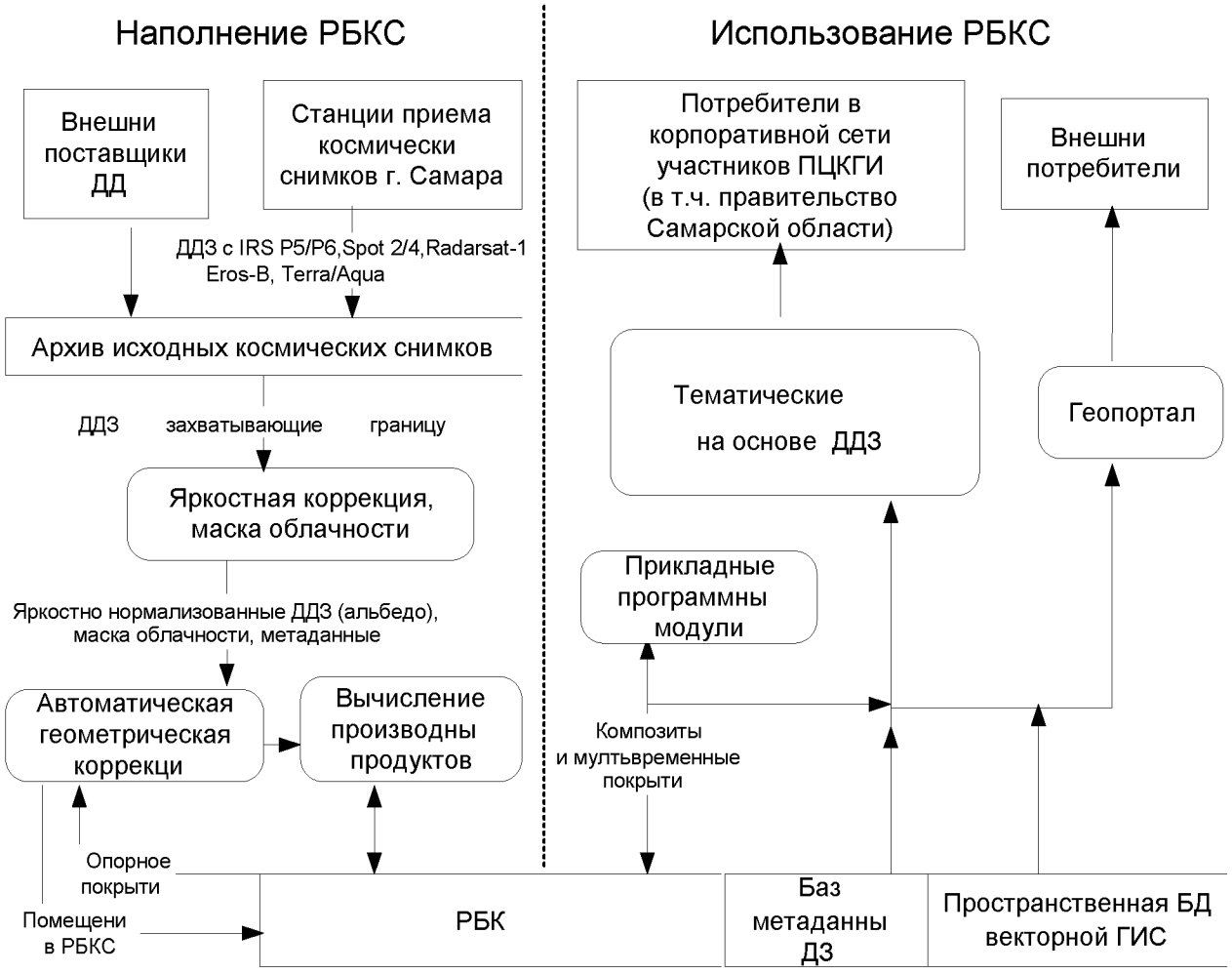


Рисунок 4.1 - Общая схема программного обеспечения ведения РБКС

4.4. Некоторые программные и программно-аппаратные решения в задачах обработки цифровых изображений и данных ДЗЗ

В настоящее время существует большое количество систем обработки и анализа цифровых изображений, специализированных для решения различных задач. Условно такие системы можно разделить на два класса. С одной стороны существуют программные системы или пакеты, подобные пакету MathLab, которые ориентированы на решение задач достаточно общего вида. Такие системы, в силу своей специфики, ориентированы на решение исследовательских задач общего назначения и не обязательно связаны с обработкой изображений. Однако они недостаточно хорошо приспособлены для построения узкоспециализированных производительных систем обработки изображений.

С другой стороны - это специализированные системы. Например, наиболее распространенные графические пакеты PhotoShop, CorelDraw и др. предназначены для художников или дизайнеров. Профессиональные системы подобные Erdas Imagine, ER Mapper, Idrisi, Envi ориентированы на еще более узкий круг специальных задач – обработку данных ДЗЗ. Все специализированные системы являются ресурсоемкими и достаточно дорогими. Даже если они обладают открытой архитектурой и возможностью встраивания в систему собственных алгоритмов обработки, для исследовательских целей эти системы малопригодны в силу ряда причин. Во-первых, использование известных систем обработки изображений подразумевает, как правило, визуальное восприятие изображений. Однако при разработке новых алгоритмов не менее важными являются другие формы представления изображений (в виде численных массивов, яркостных срезов, трехмерной визуализации и др.). Во-вторых, в конкретном случае использования известных систем основной задачей является обработка и анализ конкретного изображения (набора изображений). Однако для исследователя основной целью является определение параметров алгоритма для некоторого класса изображений, для чего необходимо проведение моделирования алгоритмов (или последовательности алгоритмов) на серии изображений с возможностью генерации случайных искажений, определения статистических характеристик изображений, и, в конечном итоге, оптимизации параметров обработки.

В настоящем подразделе рассматривается одно из возможных программно-

архитектурных решений [2, 3], разработанное и реализованное авторами. Оно позволяет удовлетворить противоречивым требования гибкости и наглядности, которые обычно предъявляются к исследовательским инструментальным системам и пакетам общего назначения (предназначенным для исследования методов, алгоритмов и информационных технологий обработки и анализа цифровых изображений, моделирования), и требованию производительности и массовости обработки, на которые ориентированы специализированные программные системы. Учитывая двойственный характер указанных выше требований, искомая программная система строится как объединение двух почти «независимых» систем, ядром которых являются общие программные модули, выполняющие различные функции тематической обработки, но реализованные программно в едином стандарте.

4.4.1. Инструментальная программная система для тематической обработки и анализа цифровых изображений

Прямым назначением системы ResLook [2, 3] является предоставление профессионального инструментария для выполнения научно-исследовательских работ в области обработки и анализа изображений. Однако помимо прямого назначения данная система может использоваться в качестве удобной среды для поддержки лабораторного практикума в учебном процессе по таким дисциплинам как математические методы обработки изображений, моделирование систем формирования изображений, методы распознавания образов и др.

Требования к системе

В плане функциональности программная система удовлетворяет следующим требованиям:

- наличие программного ядра для доступа и манипулирования данными с поддержкой различных форматов представления и хранения изображений;
- наличие ряда часто используемых функций обработки;
- наличие удобной среды для разработки сценариев с поддержкой отладки;
- отображение результатов обработки в различных представления, удобных исследователю;
- устойчивость системы к сбоям в функциях обработки;

- открытость архитектуры, что подразумевает:

  - возможность создания собственных функций обработки, встраиваемых в среду,

  - возможность создания собственных (новых) форматов представления и хранения изображений;

  - возможность расширения типов визуализируемых данных и способов визуализации.

Состав и назначение компонентов системы

Рассматриваемая система состоит из следующих подсистем:

- интерактивная программная среда,
- подсистема управления данными,
- библиотека прикладных модулей,
- подсистема запуска функций из ОС,
- визуализатор.

Схема информационного взаимодействия подсистем инструментальной системы представлена на рисунке 4.2. Далее рассматривается состав и назначение каждой из подсистем.

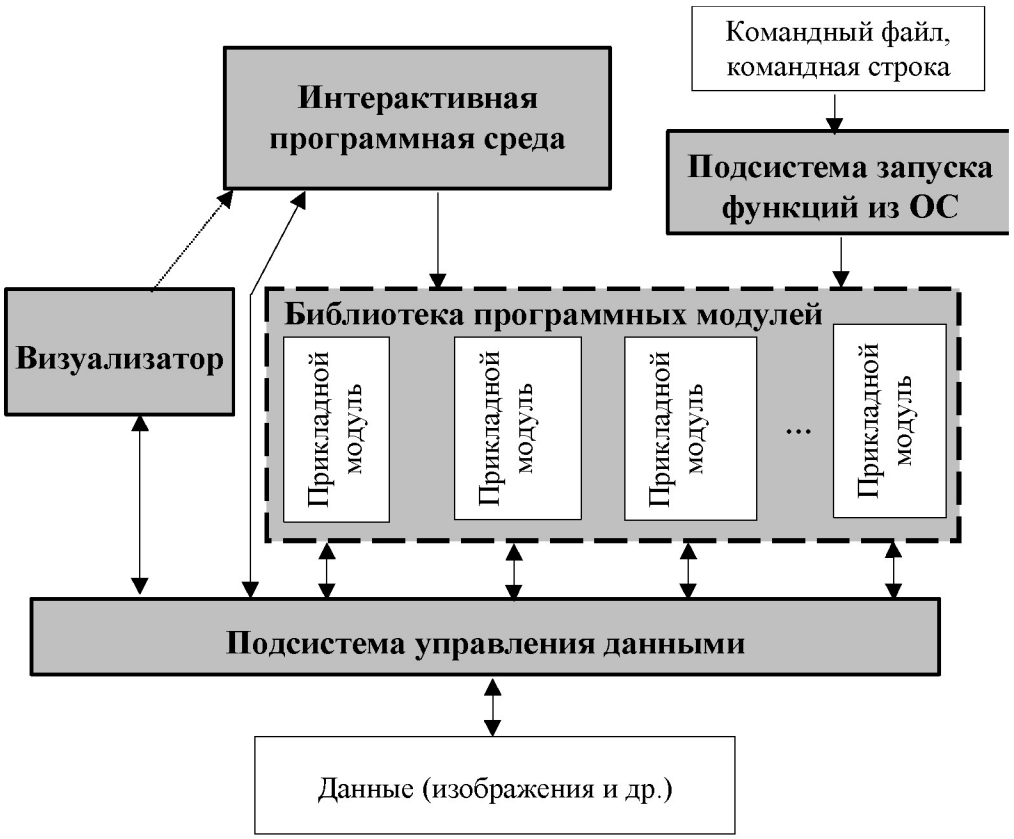


Рисунок 4.2 - Схема информационного взаимодействия подсистем

Подсистема управления данными обеспечивает взаимодействие прикладных модулей с входными/выходными данными для реализации алгоритмов обработки и анализа изображений. Данная подсистема поддерживает следующие типы изображений:

- полутоновые (серые) однокомпонентные изображения,
- цветные трехкомпонентные изображения,
- индексированные (палитровые) изображения,
- цветные четырехкомпонентные изображения в цветовых пространствах CMYK и YCbCr,
- многокомпонентные изображения (две и более компоненты, например, для представления комплексных значений).

Изображения могут быть представлены в форматах TIF, BMP, PIC (внутреннем формате системы ResLook [2, 3], в котором кроме 8-битных поддерживаются целочисленные 16-битные и вещественные 32-битные типы данных), а также формате HGI (см. раздел 5. настоящей книги), предназначенном для работы с изображениями значительного размера, например, данными ДЗЗ. Список поддерживаемых форматов расширяется путем написания соответствующих библиотек.

Базовым элементом системы является *прикладной программный модуль*, в котором сосредоточена логика, реализующая различного рода функции обработки. В настоящее время *библиотека программных модулей* включает в себя множество модулей, которые условно могут быть разделены по следующим категориям:

- синтез детерминированных и случайных полей;
- работа с отсчетами, фрагментами и атрибутами изображения;
- оценка статистических яркостных и пространственно-яркостных характеристик изображений;
- поэлементные преобразования изображений;
- выделение границ, фильтрация и локальная обработка изображений;
- геометрические и спектральные преобразования.

Интерактивная программная среда представляет собой набор исполняемых модулей и состоит из интегрированного редактора-отладчика, подсистемы выполнения прикладных модулей, а также подсистем настройки (администрирования). *Интегрированный редактор-отладчик* позволяет не только запускать отдельные прикладные модули, но и создавать, редактировать и выполнять сценарии обработки с возможностями отладки, близкими по составу к возможностям средств разработки прикладных программ общего назначения. *Подсистема выполнения прикладных модулей*, входящая в интерактивную программную среду, отвечает за выполнение функций обработки и осуществляет контроль над их выполнением. Отметим, что данная подсистема играет важную роль в обеспечении устойчивости системы к сбоям в прикладных модулях.

Подсистема визуализации инструментального программного комплекса (*визуализатор*) предназначена для отображения двумерных данных (изображений) для пользователя. Визуализатор позволяет просматривать данные в различном виде, используя обычное представление, табличное представление, вывод отдельно по цветовым каналам и т.п. Кроме того, визуализатор позволяет применять к изображениям различные преобразования, выполнять статистические измерения по областям изображения, а также реализует другие полезные исследовательские функции. Необходимо отметить, что визуализатор предоставляет разработчикам удобный интерфейс прикладного программирования, с использованием которого можно нарастить функциональность визуализатора, добавляя к нему новые

программные расширения для отображения данных, выполнения различных преобразований и т.п.

Замечание

Описанная архитектура обеспечивает базовую функциональность системы и отражает ее ориентацию на решение научных и исследовательских задач. В то же время при решении практических задач обработки данных ДЗЗ ключевая часть инструментальной программной системы, представленная в виде библиотеки исполняемых прикладных программных моделей и функций, дополняется программной системой распределенной обработки изображений. Краткое описание этой системы приводится ниже.

4.4.2. Программная система распределенной обработки изображений

Как уже было указано выше, отличие задач обработки данных ДЗЗ от задач исследовательского характера состоит в огромных объемах обрабатываемых данных. Рассмотрим следующий типовой пример, отражающий этапы получения изображения ДЗЗ.

После получения данных ДЗЗ на земле необходима их обработка для получения цифрового снимка заданного разрешения в заданной системе координат, содержащего слои с заданными спектральными диапазонами съемки. Однако, в силу значительных размеров и/или особенностей положения требуемого снимка, средства ДЗЗ зачастую не позволяют сразу сформировать все поле целиком, и изображения регистрируются полосами.

В таком случае процесс формирования искомого изображения может быть упрощенно представлен в рамках следующей схемы. Вначале для каждого полученного изображения полосы производится автономная обработка с целью компенсации разброса параметров элементов регистрации, нормализации динамического диапазона, фильтрации изображения для устранения линейных искажений (возможно, также и нелинейных искажений). Далее производится согласование полос и их геометрическая коррекция (для приведения в согласованную или во внешнюю систему координат). На завершающих этапах применяется совместная обработка всех изображений для формирования единого монохромного изображения. В случае, когда требуется получить мультиспектральное изображение,

приведенная выше схема обработки дополняется этапом совмещения изображений из различных спектральных диапазонов. Следует отметить, что похожие задачи возникают и при получении единого растрового поля по снимкам, полученным в разное время, с различных летательных аппаратов и т.д.

Учитывая значительные линейные размеры снимков и, возможно, наличие нескольких спектральных диапазонов, объемы обрабатываемых данных оказываются огромными и могут исчисляться десятками и сотнями гигабайт исходной информации. Использование стандартных средств и методов ускорения, основанных только на алгоритмических решениях (быстрые алгоритмы), в таких ситуациях оказывается недостаточным. Требуются программно-аппаратные решения, позволяющие увеличить пропускную способность системы обработки данных ДЗЗ.

Естественным способом увеличения производительности в рассматриваемом случае является распараллеливание процесса вычислений. Для этого в рамках общего решения необходимо выделить множество частных задач и определить зависимости по данным между задачами. Затем, используя полученные зависимости, определить подмножества задач, которые могут выполняться параллельно и независимо друг от друга. В большинстве случаев эти шаги интуитивно понятны специалисту в области обработки данных ДЗЗ и являются тривиальными. Так, например, «склейку» полос (формирование из двух полос единого изображения) можно производить только после устранения на исходном видеоматериале (полосе) геометрических искажений и искажений, вносимых средствами регистрации и оптикой. Вся схема обработки, позволяющая получить «базовый информационный продукт», может быть представлена в виде графа обработки.

Для решения задачи увеличения пропускной способности при обработке данных ДДЗ была разработана специальная распределенная система обработки данных ДДЗ [23], позволяющая осуществлять обработку данных ДЗЗ с использованием прикладных программных модулей инструментальной программной системы, описанной в п.4.4.1, в распределенном режиме на нескольких ПЭВМ. Следует отметить, что по построению рассматриваемая система близка так называемым GRID-системам, функционирующим в рамках вычислительных сетей предприятий.

Архитектура системы и функции основных программных модулей

Распределенная система обработки данных ДЗЗ состоит из программных компонентов:

- серверное приложение (диспетчер),
- программа работы с БД исходной информации (части изображений, данные о параметрах съемки, данные об искажениях и т.д.),
- программа, управляющая запуском и работой модулей тематической обработки, названная «виртуальной машиной»,
- прикладные модули, реализующие тематическую (содержательную) обработку ДДЗ. Эти модули одновременно являются и элементами инструментальной программной среды, описанной в п.4.4.1.

Основным назначением серверного приложения, представленного на рисунке 4.3, является обеспечение выполнения процесса обработки в соответствие с принятой схемой обработки путем распределения заданий между виртуальными машинами.

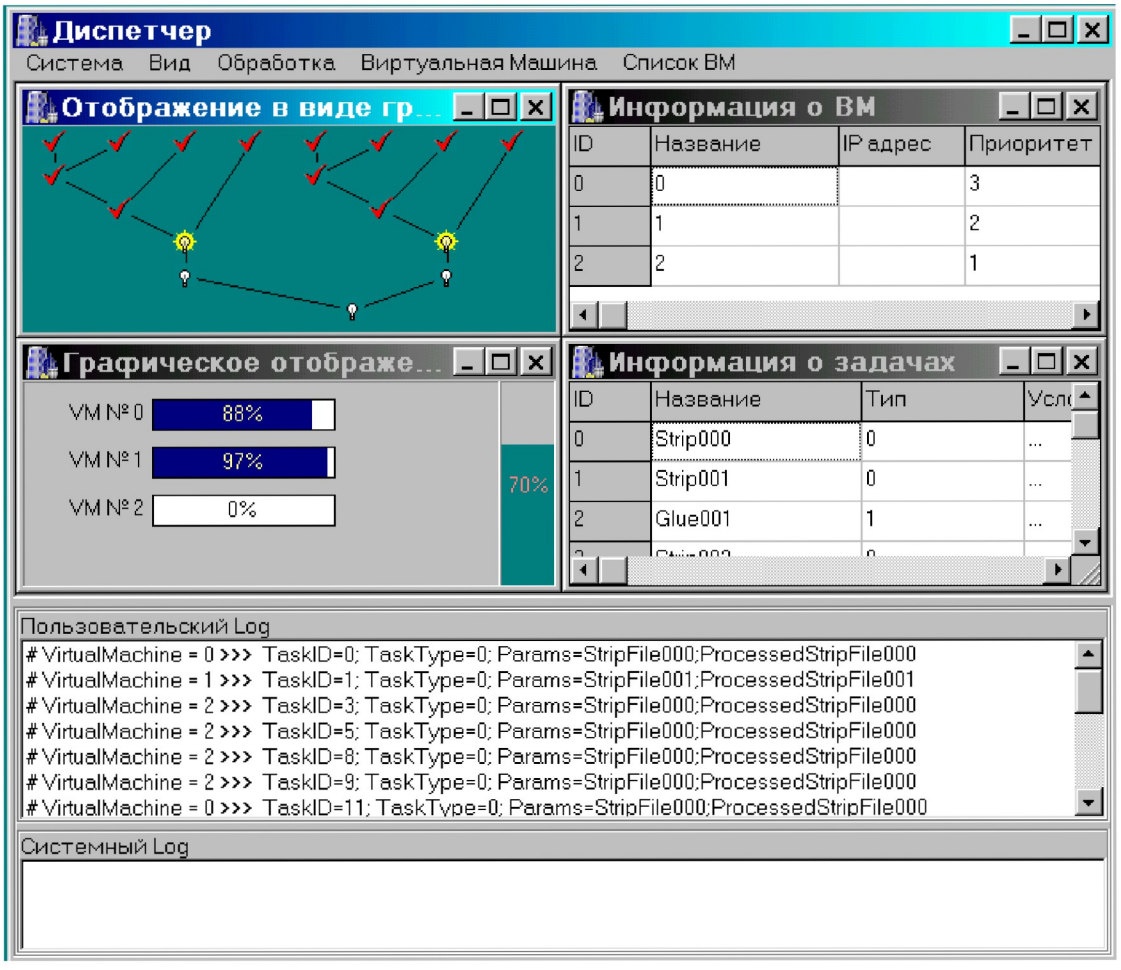


Рисунок 4.3 - Графический интерфейс серверного приложения

Наиболее важными функциями серверного приложения являются:

- администрирование комплекса обработки путем изменения состава виртуальных машин и привязки виртуальных машин к вычислительным ЭВМ, входящим в состав сети (в том числе во время выполнения обработки), а также предоставление функий интерактивного задания схемы обработки, представляемой в виде графа;
- управление производимой обработкой, осуществляя самостоятельный выбор виртуальных машин для выполнения отдельных задач, отслеживание работы виртуальных машин и степень их загрузки;
- управление процессом обработки, то есть осуществление запуска, приостановки и возобновления процесса обработки, полной остановки (отказ от обработки);
- позволяет осуществлять визуальный контроль за процессом обработки и состоянием виртуальных машин;
- автоматическая коррекция ошибок и сбоев, произошедших в процессе обработки за счет перераспределения вычислительных ресурсов (виртуальных машин и ЭВМ) в рабочем режиме.

Основным назначением *программы работы с БД* является управление исходными данными ДЗЗ и формирование схемы обработки, необходимой для получения результата. Наиболее важными функциями программы работы с БД являются:

- ведение базы исходных данных ДЗЗ, включающих как графическую, так и сопутствующую семантическую информацию с возможностью ввода, редактирования и просмотра данных;
- выбор набора исходных данных ДЗЗ (используемые кадры, полосы), используемых для получения информационного продукта, с последующим формированием схемы обработки.

Основным назначением *виртуальной машины* является выполнение указанных сервером прикладных программных модулей тематической обработки. Полный список функций виртуальной машины следующий:

- производит одну тематическую обработку за раз;
- отображает динамику процесса тематической обработки;
- реализует автоматическую связь с программой управления;

- отвечает на запросы сервера о состоянии (обработки и самой виртуальной машины);
- по запросу сервера запускает необходимую тематическую обработку;
- по окончании обработки передает серверу «код завершения тематической обработки».

Следует отметить, что на одной ЭВМ могут работать одновременно несколько виртуальных машин, что позволяет более эффективно использовать ЭВМ с многоядерной и многопроцессорной архитектурой. В настоящей версии системы связь между серверным приложением и виртуальными машинами осуществляется по стандарту DCOM (Distributed Component Object Model) фирмы Microsoft. В терминологии DCOM каждая виртуальная машина является удаленным COM сервером, который предоставляет свои вычислительные и алгоритмические ресурсы для DCOM - клиента. Клиентом COM-серверов в рассматриваемом случае является серверное приложение - диспетчер.

4.5. Геоинформационный портал работы с пространственными данными и метаданными

Цифровые карты, космические снимки и результаты их обработки представляют значительный интерес не только для специалистов, но и для широкого круга аудитории, региональных и муниципальных властей, предпринимателей и т.д. Популярным решением организации открытого доступа к ним является создание регионального геоинформационного портала работы с пространственными данными и метаданными. Центр приема, описываемый в настоящем приложении, является уникальным поставщиком космических снимков. В информационные ресурсы геопортала также включены базовые пространственные данные - обзорные цифровые карты региона, подробные адресные планы населенных пунктов, навигационные карты и пр.

В настоящем разделе рассматриваются основные технические аспекты создания регионального геопортала. Прежде чем перейти к описанию технической реализации, перечислим основные интеграционные требования, которые использовались при разработке регионального геопортала Самарской области.

1) Совместная обработка всех трех основных видов геоданных – векторных данных, растровых данных (аэрофото- и космические снимки) и метаданных.
2) Единая система разграничения доступа к ресурсам.
3) Полнота и открытость представленной информации.
4) Использование единой цифровой картографической основы (и единой базы данных).
5) Поддержка стандартов веб-картографии WMS,WFS [24], ISO-1915 [12].
6) Возможность представления пользователю интегрированной карты, собранной с более чем одного сервера предоставления пространственных данных.

Дополнительно к указанным интеграционным требованиям перечислим следующие требования к технической реализации.

1) Возможность работы в стандартном браузере без установки какого-либо клиентского программного обеспечения в режиме стандартной функциональности WMS (просмотр карт, поиск объектов, управление видимостью слоев).
2) Возможность использования расширенной функциональности по редактированию и анализу данных (ввод собственных объектов, аналитические сервисы) на клиентском месте.
3) Сокращение объема и защита передаваемых данных между сервером и клиентом.
4) Модульный принцип клиентского программного обеспечения, разграничение доступа к информационным ресурсам (на уровне картографических слоев, растровых покрытий и их групп) и к аналитическим сервисам.
5) Масштабируемость – возможность избежать значительного падения скорости работы при увеличении количества пользователей за счет привлечения дополнительных вычислительных ресурсов (критической по времени серверной операцией является растеризация изображения из векторных данных).

На рисунке 4.4 представлены основные компоненты программного обеспечения геопортала.

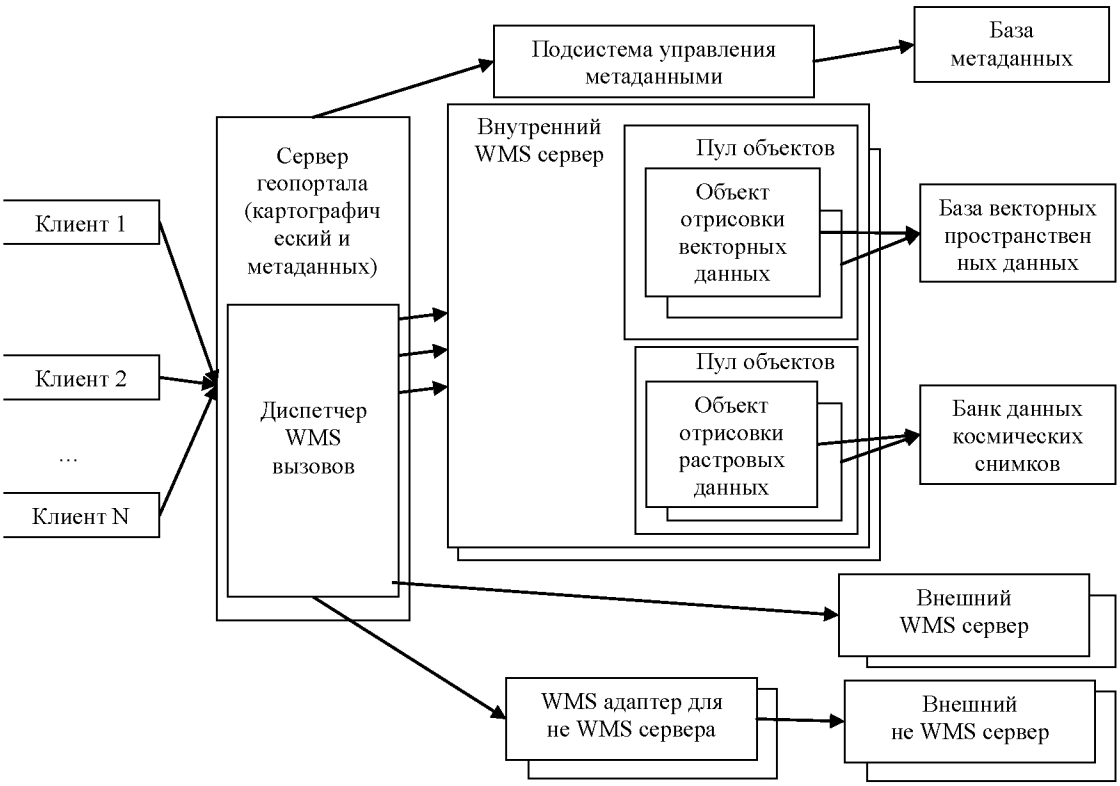


Рисунок 4.4 - Программная архитектура геопортала

В качестве тонкого клиента выступает удалённый компьютер с веб-браузером. Клиентские запросы, поступают по сети Интернет на сервер геопортала (диспетчер), отвечающий за авторизацию пользователей и перераспределение клиентских запросов между внутренними или внешними серверами. Внутренние (специализированные) WMS-серверы формируют ответ на поступающие от диспетчера типизированные запросы, обращаясь к поставщикам векторных картографических данных и поставщикам ДДЗ из регионального банка данных, объединенным в пулы соответствующих объектов. Такая архитектура обеспечивает эффективное использование вычислительных ресурсов за счет возможности перераспределения нагрузки между внутренними серверами портала и использования возможностей ЭВМ с многопроцессорной архитектурой в качестве внутренних серверов.

Интеграция геопортала с внешними геоинформационными ресурсами осуществляется в зависимости от типа ресурса. Наиболее просто вопрос интеграции решается в том случае, если ресурс поддерживает спецификацию WMS - достаточно перенаправлять запросы на внешний ресурс, при необходимости выполняя преобразование между используемыми проекциями. В случае если ресурс не поддерживает спецификацию WMS, задача интеграции неизбежно приводит к

необходимости создания промежуточного звена (адаптера), позволяющего полностью учесть все нюансы взаимодействия с ресурсом.

В сервер геопортала включена система кэширования запрашиваемых клиентом палеток (tile) для уменьшения нагрузки на сервер и ускорения работы системы. Передача изображений между сервером и клиентом может осуществляться в специальном сжатом виде.

При реализации программного обеспечения внутренних серверов были использованы технологии ASP.NET, COM, XML и др. Реализация клиента выполнена на основе JavaScript с использованием технологии AJAX (базовый вариант), а также на основе Flash-технологий (расширенный вариант, см. рисунок 4.5).

В рамках регионального портала также реализован пилотный вариант *подсистемы управления метаданными,* обеспечивающей решение следующих важнейших задач:

- добавление метаданных производителями и владельцами пространственных данных,
- поиск потребителями информации о пространственных данных на основе пространственных, временных и др. параметров,
- извлечение информации о пространственных данных, включающей информацию о назначении и пригодности данных, условия доступа к данным, необходимую информацию для обработки и использования пространственных данных и т.д.

Структура метаданных определяется в соответствии с международным стандартом ISO 19115 и национального стандарта ГОСТ Р 52573-2006 «Географическая информация. Метаданные» [12].

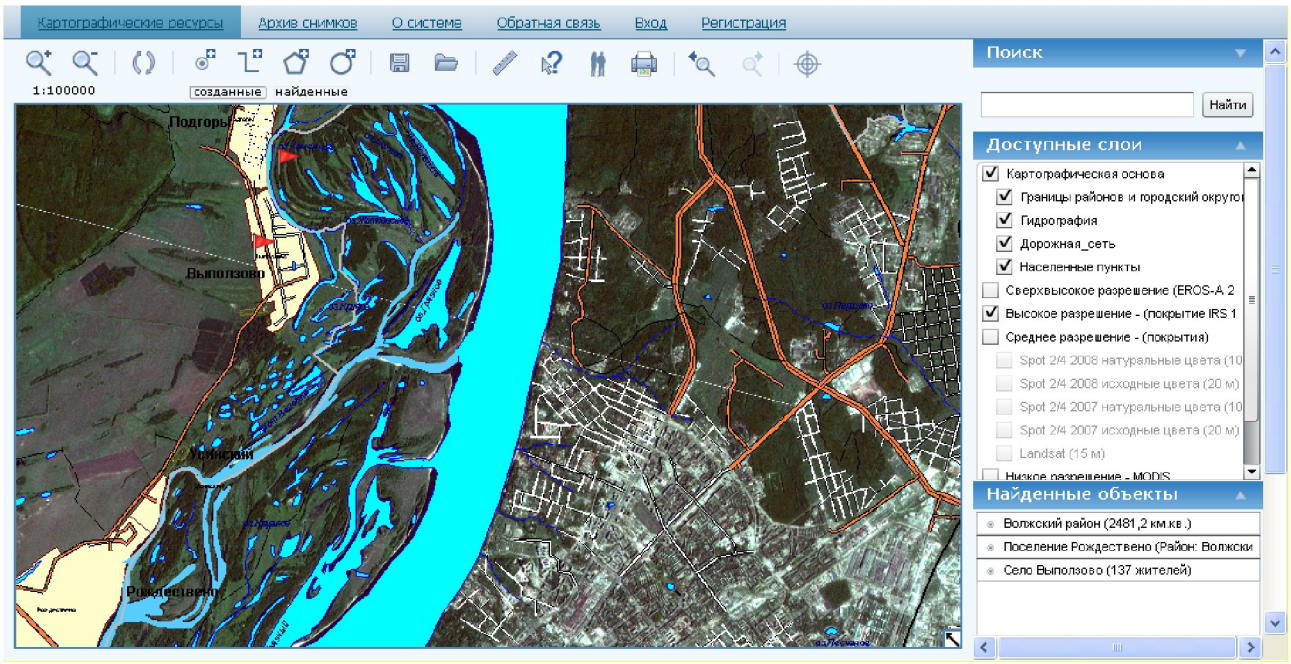


Рисунок 4.5 - Пользовательский интерфейс клиентского места

5. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Borgefors G. Distance Transforms in Digital Images [Текст] / G. Borgefors // Computer Vision, Graphics and Image Processing. – 1986. – P. 344–371.
2. Chernov A.V. Software System for Development Algorithms for Digital Image Processing and Analysis [Текст] / A.V. Chernov, M.A. Chicheva, M.V. Gashnikov, N.I. Glumov, E.V. Myasnikov, V.V. Sergeyev // Proceedings of the Second IASTED International Multi-Conference on ACIT - Software Engineering (ACIT-SE). – Novosibirsk, 2005. – P. 139-142.
3. Chernov A.V., System of Software Tools for Digital Image Processing and Analysis [Текст] / A.V. Chernov, M.A. Chicheva, M.V. Gashnikov, N.I. Glumov, E.V. Myasnikov, V.V. Sergeyev // Pattern Recognition and Image Analysis. – 2005. – Vol. 15, No. 2. – P. 368-370.
4. Chernov A.V. Fast Method for Local Image Processing and Analysis [Текст] / A.V. Chernov, V.V. Myasnikov, V.V. Sergeyev // Pattern Recognition and Image Analysis. –1999. – Vol. 9, No. 2. – P. 237-238.
5. Chicheva M.A. Image Compression Technology for Real-Time Systems of Remote Sensing [Текст] / M.A. Chicheva, M.V. Gashnikov, N.I. Glumov, V.V. Sergeyev //Proceedings of International Conference on Computing, Communications and Control Technologies. – Austin, Texas, USA, August 14-17, 2004. – Vol. 5. – P. 237-241.
6. Forsyth David A. Computer Vision - A modern approach [Текст] / David A. Forsyth, Jean Ponce. – Prentice Hall, Inc. Upper Saddle River, New Jersey, 2003. – 693 p.
7. Gashnikov M.V. Adaptive interpolation algorithm for hierarchical image compression [Текст] / M.V. Gashnikov, N.I. Glumov, V.V. Sergeyev // Proceedings of International Conference "Automation, Control, and Information Technology" (ACIT). – Novosibirsk, 2002. – P. 423-428.
8. Gashnikov M.V. Stabilization of the Compressed Data Formation Rate in Hierarchical Image Compression [Текст] / M.V. Gashnikov, N.I. Glumov, V.V. Sergeyev // Pattern Recognition and Image Analysis. – 2007. – Vol. 17, No. 1. – P. 79–81.
9. Glumov N.I. Comprehensive approach to choice of compression and error-control coding algorithms for digital image transmission through communication channels

[Текст] / N.I. Glumov // Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis. – Zagreb, Croatia, 2005. – P. 484-488.

10. Glumov N.I. Improving Noise Immunity of Transmission of Compressed Digital Images [Текст] / N.I. Glumov // Pattern Recognition and Image Analysis. – 2003. – Vol. 13, No. 2. – P. 273-276.
11. Gonzalez R.C. Digital Image Processing [Текст] / R.C. Gonzalez, R.E. Woods. – Prentice Hall, 2004. – 624 p.
12. INTERNATIONAL STANDARD ISO/FDIS 19115. – Geographic information. – Metadata.
13. Janssen R.D.T. Adaptive Vectorization of Line Drawing Images [Текст] / R.D.T. Janssen, A.M. Vossepoel // Computer Vision and Image Understanding. – 1997. – Vol. 65, No. 1. – P. 38-56.
14. Lin S. Error Control Coding: Fundamentals and Applications [Текст] / S. Lin, D.J. Costello. – second edition. – Prentice Hall: Englewood Cliffs, NJ, 2004.
15. Shapiro Linda G. Computer Vision [Текст] / Linda G. Shapiro, George C. Stockman. – Prentice Hall, Upper Saddle River, New Jersey, 2001. – 608 p.
16. Luger G. Artificial intelligence: structures and strategies for complex problem solving [Текст] / G. Luger. – Addison-Wesley Publishing Company, 2002.
17. Mallat S.G. A theory for multiresolution signal decomposition: the wavelet representation [Текст] / S.G. Mallat // IEEE Trans. on Pattern Anal. and Mach. Intell. – 1989. – № 11(7). – P. 674-693.
18. Myasnikov V.V. Algorithms for the Optimization of the Two-Stage Procedure of the Detection and the Recognition of Objects in an Image [Текст] / V.V. Myasnikov // Pattern Recognition and Image Analysis. – 1999. – Vol. 9, No. 4. – P. 702-707.
19. Myasnikov V.V. Construction of Integer-Valued Polynomials for Recursive Calculation of Convolution with an FIR Filter [Текст] / V.V. Myasnikov // Pattern Recognition and Image Analysis. – 2005. – Vol. 15, No. 1. – P. 264-267.
20. Myasnikov V.V. Fast Algorithm for Recursive Computation of the Convolution of an Image with a Two-Dimensional Inseparable Polynomial FIR Filter [Текст] / V.V. Myasnikov // Pattern Recognition and Image Analysis. – 2007. – Vol. 17, No. 3. P. 421-427.
21. Myasnikov V.V. On Recursive Computation of the Convolution of Image and 2-D

Inseparable FIR-Filter [Текст] / V.V. Myasnikov // The 9th World Multi-Conference on Systemics, Cybernetics and Informatics. – Orlando, Florida, USA, July 10-13, 2005. – P. 268-272.

22. Myasnikov V.V. Optimization Algorithms of the Two-Stage Pattern Detection and Recognition Procedure [Текст] / V.V. Myasnikov // Pattern Recognition and Image Analysis. – 1999. – Vol. 9, No. 1. – P. 81-83.
23. Myasnikov V.V. Program System for Distributed Image Processing [Текст] / V.V. Myasnikov, E.V. Myasnikov, V.V. Sergeev, A.V. Chernov // Pattern Recognition and Image Analysis. – 2003. – Vol. 13, No. 2. – P. 228-230.
24. OpenGIS ® Specification Standards. – http://www.opengeospatial.com/standarts.
25. Pratt W. Digital image processing [Текст] / W. Pratt. – 4ed. – Wiley, 2007.
26. Rosenfeld A. Sequential operations in digital picture processing [Текст] / A. Rosenfeld, J. L. Pfaltz // ACM., 1966. – Vol. 13. – P. 471-494.