

На правах рукописи

Дружинин Денис Вячеславович

**АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
СЖАТИЯ БЕЗ ПОТЕРЬ ВИДЕОДАННЫХ ГРАФИЧЕСКОГО
ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ**

05.13.11 – Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Автореферат
диссертации на соискание учёной степени
кандидата технических наук

Томск – 2018

Работа выполнена в федеральном государственном автономном учреждении высшего образования «Национальный исследовательский Томский государственный университет».

Научный руководитель: доктор технических наук, доцент
Замятин Александр Владимирович

Официальные оппоненты:

Спицын Владимир Григорьевич, доктор технических наук, профессор, федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Томский политехнический университет», Отделение информационных технологий, профессор

Турлапов Вадим Евгеньевич, доктор технических наук, доцент, федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского», кафедра математического обеспечения и суперкомпьютерных технологий, профессор

Ведущая организация: Федеральное государственное автономное образовательное учреждение высшего образования «Сибирский федеральный университет»

Защита состоится 04 апреля 2019 г. в 10 час. 30 мин. на заседании диссертационного совета Д 212.267.08, созданного на базе федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский Томский государственный университет», по адресу: г. Томск, пр. Ленина, 36 (учебный корпус № 2 ТГУ, аудитория 102).

С диссертацией можно ознакомиться в Научной библиотеке и на официальном сайте федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский Томский государственный университет» www.tsu.ru.

Материалы по защите диссертации размещены на официальном сайте ТГУ: <http://www.ams.tsu.ru/TSU/QualificationDep/co-searchers.nsf/newpublicationn/DruzhininDV04042019.html>

Автореферат разослан «___» февраля 2019 г.

Учёный секретарь
диссертационного совета
доктор технических наук,
профессор

Скворцов
Алексей Владимирович

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследования. В начале XXI века создание и применение видео различной природы стало доступно широкому кругу пользователей. Один из основных типов видео, распространенных среди пользователей персональных компьютеров (ПК), – видео графического интерфейса пользователя (Graphical User Interface видео, GUI-видео). Такое видео часто создают как необходимое дополнение к руководству пользователя информационно-программных комплексов или к описанию сценариев воспроизведения ошибок при взаимодействии команд разработчиков и инженеров по качеству программного обеспечения (ПО) для повышения эффективности их коммуникации.

Ввиду сравнительно значительного объема *видеоданных* (данных, получаемых в ходе формирования видео) возникают сложности, связанные с необходимостью отводить большой объем дискового пространства для их хранения. Как следствие, при решении задачи обработки GUI-видеоданных появляется условие минимизации их объема. Дополнительно эта задача осложняется тем, что GUI-видеоданные характеризуются высоким разрешением кадров.

Кроме того, приложения, осуществляющие фиксацию GUI-видео, выполняют лишь вспомогательную функцию, работают одновременно с другими приложениями и сервисами операционной системы, и по этой причине должны исполняться в низкоприоритетном режиме. Поэтому необходимо минимизировать уровень использования ими системных ресурсов компьютера, требуемых для эффективного исполнения основных задач пользователя. Как следствие, появляется условие повышения вычислительной эффективности алгоритмов фиксации и сжатия GUI-видеоданных при минимальном уровне использования центрального процессора (ЦП) и оперативной памяти (ОП) (максимальной *ресурсоэффективности* таких алгоритмов).

С середины XX века выполнен большой объем исследований теоретического и практического характера в области сжатия данных такими учеными как Котельников В.А., D.A. Huffman, A. Lempel, J. Ziv. В области сжатия графических данных, в том числе видеоданных, получены результаты, как предполагающие потери информации, так и обеспечивающие сжатие без потерь. В первой группе исследований следует отметить работы таких ученых как Бабкин В.Ф., Тропченко А.Ю., G. Sullivan, Y. Lu, M.R. El-Sakka, D. Taubman, A.A. Rodriguez. Полученные ими результаты получили широкое практическое применение и преимущественно предназначены для видео с видеокамеры (*традиционного* видео) и используют его особенности, заключающиеся в преобладании непрерывно-тоновых (плавных) цветовых переходов

дов в кадрах. Соответствующие алгоритмы сжатия видеоданных, как правило, предполагают устранение визуально малозаметных элементов изображения и потерю информации.

GUI-видеоданные существенно отличаются от традиционных видеоданных, что в значительной мере ограничивает возможности применения к ним традиционных алгоритмов сжатия. Так, в GUI-видеоданных преобладают дискретно-тоновые (резкие) цветовые переходы. Такие видеоданные, как правило, целесообразно сжимать без потерь информации специализированными алгоритмами, так как даже небольшая доля потерь может привести к неприемлемому визуальному ухудшению качества видеоданных. Во второй группе исследований, посвящённых сжатию графических данных без потерь информации, следует отметить работы таких учёных как B. Carpentieri, J.J. Ding, M. Hernández-Cabronero, M.W. Marcellin, I. Blanes, K. Hirakawa. Однако, лишь в отдельных исследованиях таких учёных как S. Mehrotra, P. Dondi, L. Lombardi, S. Wang, J. Xu, R. Joshi, R.A. Cohen учитываются особенности GUI-видео. Многие существующие алгоритмы, применяемые для сжатия GUI-видеоданных, не учитывают в полной мере отличия GUI-видео от традиционного видео, что приводит к низкой степени сжатия либо к низкому качеству декодированных видеоданных в случае сжатия с потерями информации. Часть алгоритмов сжатия не обладает достаточной вычислительной эффективностью и ресурсоэффективностью, что препятствует их использованию для сжатия GUI-видеоданных в фоновом режиме. Поэтому проблема создания эффективных алгоритмов сжатия GUI-видеоданных всё ещё остаётся малоизученной.

Учитывая всё вышеизложенное, следует сделать вывод об актуальности проблемы создания новых и модификации существующих алгоритмов сжатия, предназначенных для обработки GUI-видеоданных, позволяющих производить сжатие GUI-видеоданных без потерь информации не только с высокой степенью сжатия, но и с минимальным использованием вычислительных ресурсов компьютера.

Цель работы и задачи исследования. Целью диссертационной работы является разработка алгоритмического и программного обеспечения фиксации и локального сохранения GUI-видеоданных со сжатием без потерь с высокой степенью сжатия, обладающего высокими показателями ресурсо- и вычислительной эффективности.

Для достижения поставленной цели необходимо последовательное решение следующих задач:

1. Анализ существующего алгоритмического и программного обеспечения сжатия видеоданных, особенностей традиционного видео и GUI-видео, позволяющих определить основные направления иссле-

дования в области построения эффективных алгоритмов сжатия GUI-видеоданных.

2. Разработка ресурсо- и вычислительно эффективных алгоритмов сжатия без потерь информации, обеспечивающих высокую степень сжатия GUI-видеоданных. Решение данной задачи предполагает также исследование эффективности предлагаемых алгоритмов.

3. Создание высокопроизводительного ядра кодека (англ. codec, в данном случае – ПО преобразования видеоданных), предназначенного для сжатия и восстановления GUI-видеоданных. Результатом решения этой задачи должны явиться программные средства такого кодека, реализующие предложенные оригинальные алгоритмы.

4. Апробация разработанного кодека с использованием различных видеофрагментов при решении экспериментальных и реальных задач.

Научную новизну полученных в работе результатов определяют:

1. Алгоритм пространственного группового кодирования, отличающийся более полным учётом горизонтальной и вертикальной корреляции пикселей кадра за счёт выявления большего количества пространственных объектов в нём, и обеспечивающий большую степень сжатия GUI-видеоданных по сравнению с аналогами.

2. Алгоритм сжатия со сниженной пространственной избыточностью, отличающийся единовременным устранением основных типов пространственной избыточности GUI-видеоданных (частое чередование цветов, одноцветные области, градиентные переходы), и позволяющий существенно увеличить степень сжатия таких данных.

3. Алгоритм оценки движения с учётом классификационных признаков, отличающийся способом выбора векторов движения исключительно в наиболее вероятных направлениях перемещения объектов GUI-видеоданных, и превосходящий аналоги в вычислительной эффективности при сохранении высокой степени сжатия и ресурсоэффективности.

4. Архитектура кодека сжатия GUI-видеоданных, характеризующаяся наличием подсистемы обработки данных на видеокарте и динамическим подключением модулей сжатия, и позволяющая построить программное обеспечение кодека, обеспечивающего высокую степень сжатия данных при высоких показателях ресурсо- и вычислительной эффективности.

Положения и результаты, выносимые на защиту:

1. Алгоритм сжатия со сниженной пространственной избыточностью позволяет повысить по сравнению с аналогами степень сжатия дискретно-тоновых изображений при сопоставимых вычислительных затратах.

2. Алгоритм оценки движения с учётом классификационных признаков позволяет существенно повысить вычислительную эффективность по сравнению с аналогами при сохранении высокой степени сжатия.

3. Разработанное алгоритмическое и программное обеспечение кодека преобразования GUI-видеоданных превосходит аналогичные кодеки по степени сжатия и вычислительной эффективности, а также упрощает и ускоряет создание программных систем с его использованием.

Методы исследования и достоверность полученных результатов. В работе использованы методы теории сжатия информации, теории обработки изображений, теории алгоритмов и математической статистики. Экспериментальные исследования выполнены с использованием программной реализации алгоритмов с последующей оценкой полученных результатов и сравнением с экспериментальными данными из специальной литературы. Достоверность полученных результатов подтверждена применением методов математической статистики и успешными результатами практической апробации алгоритмического и программного обеспечения.

Теоретическая и практическая значимость работы. Теоретическая значимость диссертационной работы заключается в том, что разработанное алгоритмическое обеспечение обладает высокими показателями ресурсо- и вычислительной эффективности, позволяет сжимать GUI-видеоданные без потерь с высокой степенью сжатия, а также позволяет сформировать научно-методический задел для создания программного обеспечения фиксации и локального сохранения GUI-видеоданных.

Практически значимыми являются предложенное в работе семейство алгоритмов, а также ПО кодека сжатия GUI-видеоданных, созданное на их основе. Программные средства кодека функционируют на компьютерах типа IBM PC под управлением Windows XP/Vista/7/8/10. Объём исходного кода разработанного ПО составляет более 8000 строк на языке C++. Кодек сжатия GUI-видеоданных внедрён в Югорском НИИ информационных технологий при создании ПО для фиксации GUI-видео, а также в компании-разработчике ПО ООО «Армадэйт» в качестве самостоятельного программного продукта. Результаты внедрения подтверждены соответствующими актами.

Соответствие паспорту специальности. Данное диссертационное исследование выполнено в соответствии с паспортом специальности 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей», а именно соответствует следующим областям (номера соответствуют пункту в паспорте специальности): п. 7 – Человеко-машинные интерфейсы; моде-

ли, методы, алгоритмы и программные средства машинной графики, визуализации, обработки изображений, систем виртуальной реальности, мультимедийного общения; п. 8 – Модели и методы создания программ и программных систем для параллельной и распределенной обработки данных, языки и инструментальные средства параллельного программирования.

Апробация результатов исследования. Основные положения и отдельные результаты исследования докладывались и обсуждались на следующих конференциях: VI Международная научно-практическая конференция «Информационные технологии и математическое моделирование» (Анжеро-Судженск, 2007 г.), XLVIII международная научная студенческая конференция «Студент и научно-технический прогресс» (Новосибирск, 2010 г.), IX Всероссийская научно-практическая конференция с международным участием «Информационные технологии и математическое моделирование» (Анжеро-Судженск, 2010 г.), XV Всероссийская научно-практическая конференция «Научное творчество молодёжи» (Анжеро-Судженск, 2011 г.), X Всероссийская научно-практическая конференция с международным участием «Информационные технологии и математическое моделирование» (Анжеро-Судженск, 2011 г.), XII Всероссийская научно-практическая конференция с международным участием им. А.Ф. Терпугова (Анжеро-Судженск, 2013 г.), 52-я международная научная студенческая конференция МНСК-2014 (Новосибирск, 2014 г.), XVIII Всероссийская научно-практическая конференция «Научное творчество молодежи. Математика. Информатика» (Анжеро-Судженск, 2014 г.), II Всероссийская молодежная научная конференция с международным участием «Математическое и программное обеспечение информационных, технических и экономических систем» (Томск, 2014 г.), Всероссийская конференция «XII Сибирская научная школа-семинар с международным участием Компьютерная безопасность и криптография – SIBECRYPT'14» (Екатеринбург, 2014 г.), Всероссийская научно-практическая конференция «Информационно-телекоммуникационные системы и технологии» (Кемерово, 2014 г.), XV Международная конференция имени А.Ф. Терпугова «Информационные технологии и математическое моделирование» (пос. Катунь, 2016 г.), XVI Международная конференция имени А.Ф. Терпугова «Информационные технологии и математическое моделирование» (Казань, 2017 г.).

Публикации. По теме диссертации опубликовано 16 работ, в том числе 3 статьи в ведущих рецензируемых научных журналах, входящих в перечень ВАК.

Личный вклад. Все основные результаты исследования получены автором. Постановка задачи и результаты исследования обсуждались с А.В. Замятиным, В.А. Лавровым.

Структура и объем диссертации. Диссертация состоит из введения, четырех глав, заключения, списка условных обозначений и сокращений, списка использованных источников и литературы из 146 наименований и четырех приложений. Общий объем работы – 161 страница.

ОСНОВНОЕ СОДЕРЖАНИЕ ДИССЕРТАЦИИ

Во **введении** обосновывается актуальность темы диссертационной работы, формулируется цель и задачи исследования и приводится краткое содержание работы по главам.

В **первой главе** проводится анализ результатов исследований в области сжатия GUI-видеоданных. Приводится классификация ПО, осуществляющего фиксацию GUI-видеоданных на ПО для мультимедийного общения и для локального сохранения GUI-видеоданных. В диссертационной работе решается задача разработки алгоритмического и программного обеспечения для локального сохранения GUI-видео с оперативным сжатием.

Ключевые (опорные) кадры сжимаются независимо от других кадров, поэтому задача сжатия таких кадров сводится к задаче сжатия изображения. Приводится классификация изображений на монохроматическое, полутоновое, а также цветное с непрерывным тоном и цветное дискретно-тоновое. В соответствии с приведённой классификацией кадры GUI-видео в большинстве своём относятся к цветным дискретно-тоновым изображениям.

Рассмотрены алгоритмы сжатия дискретно-тоновых изображений. В частности, рассмотрены словарные, статистические алгоритмы, алгоритмы группового кодирования и классификации блоков изображений. Часть алгоритмов не подходит для сжатия GUI-видеоданных, так как осуществляют сжатие с потерями информации. Другая часть алгоритмов сжатия обладает недостаточной вычислительной эффективностью для сжатия ключевых кадров GUI-видео в низкоприоритетном режиме, характерном для приложений, осуществляющих фиксацию GUI-видео. И, наконец, алгоритмы, способные оперативно сжимать ключевые кадры без потерь информации обеспечивают недостаточный уровень сжатия, учитывая высокий уровень избыточности, характерный для GUI-видеоданных.

Помимо ключевых выделяют промежуточные кадры, которые сжимаются с использованием информации о смежных кадрах. Выявлены существенные отличия традиционного и GUI-видео, проявляющиеся при анализе последовательности кадров. Рассмотрены существующие алгоритмы сжатия промежуточных кадров, в частности, алгоритм отсечения неизменившихся блоков кадра и алгоритм оценки движения. Ни один из этих алгоритмов не обеспечивает одновременно

высоких показателей вычислительной эффективности и степени сжатия. Рассматриваются перспективные методы повышения эффективности сжатия GUI-видеоданных, включающие использование вычислительных ресурсов видеокарты и технологии определения изменившихся частей кадра в GUI-видео.

На основании результатов анализа проблемы сжатия GUI-видеоданных сформулированы цель и задачи диссертационного исследования.

Цель исследования формально представлена как задача условной оптимизации с неявными ограничениями в следующем виде:

$$\begin{aligned} \max_x E_c(x) & \quad \text{где } x - \text{алгоритм сжатия без потерь} \\ \text{при ограничениях (условиях):} & \quad \text{информации, } E_c - \text{вычислительная} \\ & \quad \text{эффективность, } C_{comp} - \text{степень сжа-} \\ \begin{cases} C_{comp}(x) \geq C_{min} \\ M_{avr}(x) \leq M_{max} \end{cases}, & \quad (1) \quad \text{тия данных, } M_{avr} - \text{средний уровень} \\ & \quad \text{использования ОП, } C_{min} - \text{минималь-} \\ & \quad \text{ное практически допустимое значение критерия } C_{comp} \text{ и } M_{max} - \text{макси-} \\ & \quad \text{мальное практически допустимое значение критерия } M_{avr}. \text{ Максими-} \\ & \quad \text{зация вычислительной эффективности алгоритма, его реализации и} \\ & \quad \text{ПО сжатия GUI-видеоданных выражается в минимизации трудоёмко-} \\ & \quad \text{сти, времени выполнения } t \text{ и степени загрузки ЦП } C_{cpu}, \text{ соответ-} \\ & \quad \text{ственно.} \end{aligned}$$

Во **второй главе** представлены оригинальные алгоритмы сжатия GUI-видеоданных, сформулированы требования к ПО кодека для сжатия GUI-видеоданных и предложены принципы построения подобных систем.

Для достижения высокой степени сжатия ключевых кадров GUI-видео необходимо устранить типы избыточности, связанные: с частым чередованием цветов (1) и с наличием групп одноцветных пикселей (2). Для устранения 1-го типа избыточности разработан *сдвиговый алгоритм* словарного сжатия, а для устранения 2-го – оригинальный *алгоритм пространственного группового кодирования (ПГК)*, учитывающий горизонтальную и вертикальную корреляцию пикселей кадра. Для устранения обоих перечисленных типов избыточности предложен новый *гибридный сдвигово-групповой алгоритм*, объединяющий преимущества сдвигового алгоритма и алгоритма ПГК. Для устранения остаточной избыточности результирующих наборов данных гибридного сдвигово-группового алгоритма предложен *алгоритм со сниженной пространственной избыточностью*, предполагающий обработку данных гибридным сдвигово-групповым алгоритмом и финальное сжатие его результирующих наборов данных известными алгоритмами. Аналитически показано, что все эти алгоритмы имеют трудоёмкость $O(n)$. Для изложения сути разработанных алгоритмов введём некоторые определения.

Остаточное изображение – это массив пикселей исходного изображения, которые не были заменены в ходе сжатия некоторыми служебными данными.

Сдвиговый алгоритм обрабатывает изображение, представленное в виде вектора. Вектор **В** формируется путём последовательной записи в него строк исходной матрицы пикселей **Р_{исх}**. Идея сдвигового алгоритма заключается в замене пикселя P_{cur} в векторе **В** адресной ссылкой на эквивалентный пиксель P_{prev} , встречавшийся ранее. Преимущество алгоритма заключается в способности устранять избыточность, связанную с частым чередованием цветов, свойственную именно GUI-видеоданным. Вместе с тем алгоритм не способен эффективно кодировать одноцветные группы пикселей. Таким образом, для обеспечения эффективности сдвигового алгоритма и раскрытия его потенциала он должен использоваться в комбинации с алгоритмом группового кодирования.

На каждом шаге алгоритма *пространственного группового кодирования* (рис. 1) происходит выявление пространственных объектов, начинающихся с текущего пикселя $P_{исх}[i+1; j+1]$ и состоящих из эквивалентных пикселей.



Рисунок 1 – Принцип построения алгоритма пространственного группового кодирования.

Для кодирования выбирается объект, содержащий максимальное количество таких пикселей. В результирующий массив **Р_{код}** записывается тип этого объекта, количество пикселей в нём и байты, кодирующие цвет пикселя. Предлагается включить в набор выявляемых про-

странственных объектов горизонтальную, вертикальную линии и прямоугольник ($P_{\text{верт}}$, $P_{\text{гор}}$ или $P_{\text{прям}}$). Преимущество алгоритма заключается в повышенной способности выявлять пространственную избыточность по сравнению с каноническим алгоритмом. Однако при этом сохраняется главный его недостаток, заключающийся в том, что эффективность сжатия резко падает при частом чередовании цветов.

Гибридный сдвигово-групповой алгоритм (рис. 2) позволяет устранить этот недостаток за счёт применения идей сдвигового алгоритма. На каждом шаге гибридного сдвигово-группового алгоритма выполняется поиск возможностей замены байтов пикселя на адресную ссылку, а также группового кодирования группы эквивалентных пикселей, включая текущий пиксель. Такой гибридный алгоритм сохраняет все преимущества алгоритма ПГК, обеспечивая дополнительное сжатие за счёт замены байтов чередующихся пикселей адресными ссылками.

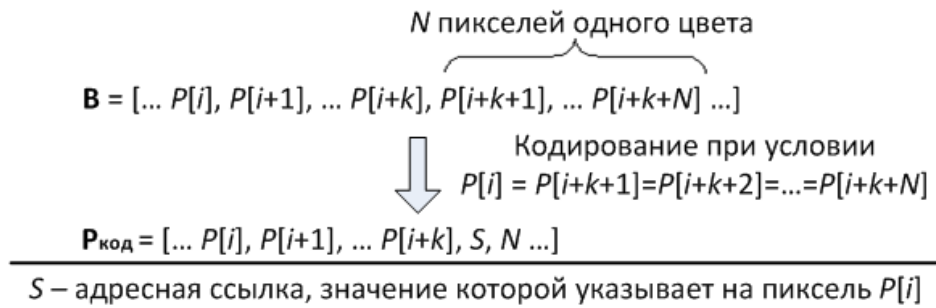


Рисунок 2 – Принцип построения гибридного сдвигово-группового алгоритма.

Чтобы повысить эффективность финального сжатия, необходимо учитывать тип избыточности, характерный для каждого из результирующих наборов данных гибридного сдвигово-группового алгоритма. Такие наборы данных включают массив флагов, массив сдвигов и количеств, а также массив пикселей (остаточное изображение). Для этого предложен новый *алгоритм сжатия со сниженной пространственной избыточностью* (АСПИ), использующий гибридный сдвигово-

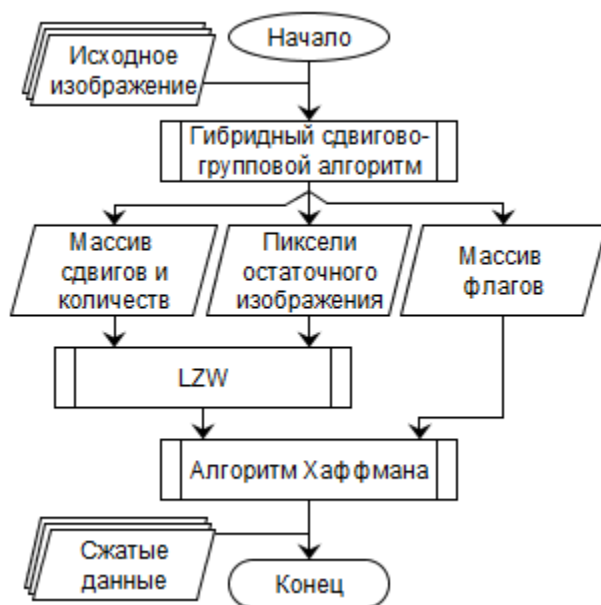


Рисунок 3 – Схема алгоритма сжатия со сниженной пространственной избыточностью.

групповой алгоритм на первом этапе, алгоритмы LZW и Хаффмана на втором этапе (рис. 3). Алгоритм LZW и Хаффмана в комбинации спо-

собны устранить как пространственную, так и статистическую избыточность, присутствующую в соответствующих результирующих наборах данных гибридного сдвигово-группового алгоритма.

Для достижения высокой степени сжатия промежуточных кадров GUI-видео необходимо устранить временную избыточность, учитывая особенности GUI-видеоданных. Для этого разработано несколько оригинальных алгоритмов сжатия:

1. *Алгоритм отсеечения неизменившихся строк и столбцов кадра*, повышающий степень сжатия по сравнению с аналогами в таких часто встречающихся случаях, как наличие точечных или близких по форме к прямоугольной изменений текущего кадра относительно предыдущего.
2. *Адаптивный алгоритм*, сочетающий преимущества существующих алгоритмов отсеечения неизменившихся областей кадра.
3. *Алгоритм оценки движения с учётом классификационных признаков*, обладающий более высокой вычислительной эффективностью по сравнению с аналогами.

Предлагается использовать адаптивный алгоритм отсеечения неизменившихся областей на первой стадии сжатия промежуточных кадров GUI-видео. Он обладает высокой вычислительной эффективностью и позволяет уменьшить объём входных данных для алгоритма оценки движения, который обеспечивает высокую степень сжатия, но имеет меньшую вычислительную эффективность.

Существующий алгоритм оценки движения, адаптированный для обработки GUI-видеоданных, учитывает многие их особенности, но не является вычислительно эффективным, что препятствует его использованию для оперативного сжатия. В такой ситуации логичной выглядит идея провести классификацию движений в GUI-видео и разработать алгоритмы, выявляющие некоторые из этих классов движений, и работающие при этом значительно быстрее алгоритма, выявляющего все классы движений. Предложена следующая классификация движений в GUI-видео:

1. Движения по вертикали и горизонтали (например, возникающие при скроллинге).
2. Движения в ближайшей окрестности (например, возникающие при перетаскивании пользователем окна).
3. Прочие движения.

Как правило, подавляющее большинство движений объектов на экране, возникающих в ходе работы пользователя, относятся к 1-му либо ко 2-му классу. Используя приведённую классификацию, удалось разработать оригинальный алгоритм оценки движения (рис. 4). Для выявления движений 1-го и 2-го классов последовательно выполняются две модификации алгоритма оценки движения. Это моди-

фикации, осуществляющие поиск блока, соответствующего текущему: только по вертикали и горизонтали; во всех направлениях в ближайшей окрестности текущего блока.

Для повышения вычислительной эффективности алгоритма оценки движения при обработке GUI-видеоданных предлагается использовать оригинальную *технику предварительного сравнения диагональных элементов*, учитывающую горизонтальную и вертикальную корреляцию пикселей в кадре. При сравнении двух блоков сначала происходит сравнение их диагональных элементов. И только в случае, когда все диагональные элементы попарно равны между собой, происходит сравнение всех элементов этих блоков.

Аналитически показано, что трудоёмкость разработанного алгоритма оценки движения с учётом классификационных признаков в наихудшем случае составляет $O(n \times \sqrt{n})$, в то время как трудоёмкость в наихудшем случае существующего алгоритма с полным перебором – $O(n^2)$, где n – количество пикселей. Это позволяет выполнить условие обработки GUI-видеоданных $t(x) \leq T_{max}$ в (1) и обеспечить высокую вычислительную эффективность сжатия.

Основываясь на результатах анализа особенностей GUI-видео, сформулированы требования к ПО кодека для сжатия GUI-видеоданных и предложены концептуальные основы его создания, упрощающие и ускоряющие создание аналогичных систем. Такой кодек должен осуществлять высокопроизводительное сжатие данных без потерь, обладать высокой ресурсоэффективностью. Предложена обобщённая архитектура кодека сжатия GUI-видеоданных, включающая подсистему обработки данных на видеокарте и динамически подключаемые модули, используемые только при достаточном уровне производительности платформы (рис. 5). Такие архитектурные решения способны уменьшить нагрузку на ЦП, что позволяет выполнить условие $\max_x E_c(x)$ в (1). На рис. 5 блоки с серым и белым фоном соответствуют подсистемам, входящим и не входящим в состав ПО кодека, соответственно.

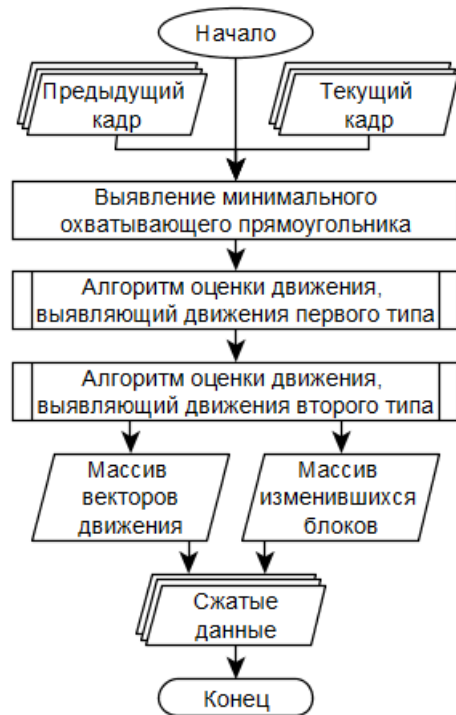


Рисунок 4 – Схема алгоритма оценки движения с учётом классификационных признаков.

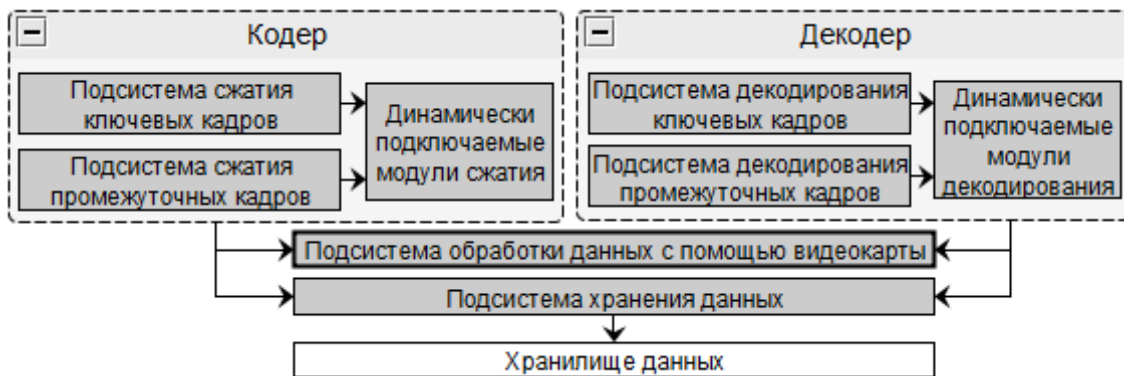


Рисунок 5 – Обобщённая структура ПО кодека сжатия GUI-видеоданных.

В **третьей главе** проводятся комплексные экспериментальные исследования разработанных алгоритмов с варьированием параметров в обширных пределах на широком спектре GUI-видеоданных.

Описаны способ выбора флагов для увеличения суммарной степени сжатия, а также детали формирования дополнительных структур данных для повышения вычислительной эффективности при выполнении гибридного сдвигово-группового алгоритма.

При тестировании алгоритмов сжатия применялась следующая методика. Каждый элемент набора данных сжимался 100 раз с усреднением полученных результатов. Проверка статистической значимости отличий в скорости и степени сжатия различных реализаций алгоритмов проводилась с помощью критерия Вилкоксона, а доверительная вероятность при этом принята равной 95%. Тестирование проводилось на платформе: процессор Intel Core 2 Duo E6750 2,66 ГГц, ОП DDR2 2Гб, ОС Windows 7.

Наборы тестовых данных включают спектр GUI-видеоданных с различными характеристиками. Так, для тестирования алгоритмов сжатия ключевых кадров использовались снимки экрана следующих типов: изображения с преобладанием текста (1), изображения с преобладанием деловой графики (2), Windows XP (3), Windows 7 (4), Ubuntu Gnome 14.04 (5).

На рис. 6 представлены данные экспериментальных исследований на 1-м наборе тестовых данных для тех реализаций, которые продемонстрировали показатели, близкие к лучшим по выборке: АСПИ, использующий zlib уровня 6 (АСПИ), zlib уровней 1 (zlib_1), 4 (zlib_4), 6 (zlib_6), 9 (zlib_9); LZO_X_999 уровня 6 (LZO_6), LZO_X_1 (LZO_X_1). Следует отметить, что zlib является реализацией алгоритма стандарта Deflate и включает как словарное, так и статистическое сжатие.

На подавляющем большинстве тестовых наборов данных АСПИ обеспечил наивысшую степень сжатия при приемлемой вычислительной эффективности. Так степень сжатия АСПИ до 23,5% выше, чем у zlib_9. Удалось увеличить скорость сжатия по сравнению с zlib_9 до 4

раз. Поэтому АСПИ позволяет выполнить условия $C_{comp}(x) \geq C_{min}$, $t(x) \leq T_{max}$ в (1).

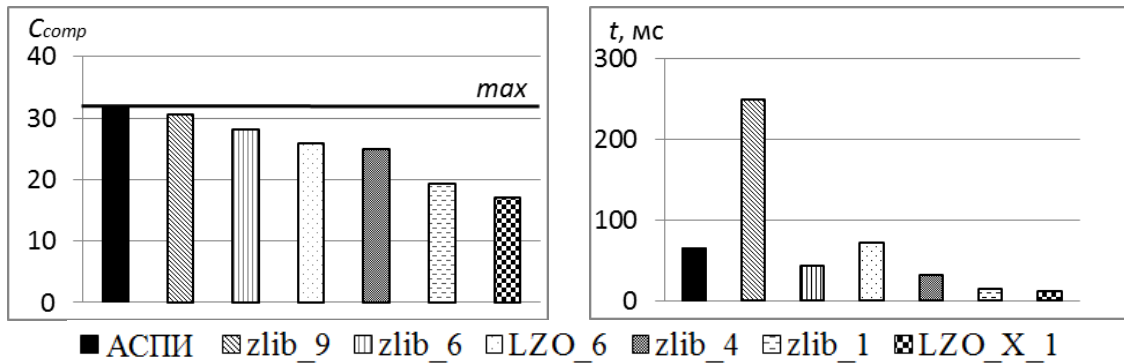


Рисунок 6 – Эффективность основных алгоритмов сжатия ключевых кадров (степень и время сжатия) на наборе тестовых данных № 1.

Для оценки эффективности алгоритмов оценки движения обратимся к результатам экспериментальных исследований этих алгоритмов (табл. 1, рис. 7). 2-ая и 3-я реализации, представленные в табл. 1, являются этапами разработанного алгоритма оценки движения с учётом классификационных признаков (1-ая реализация).

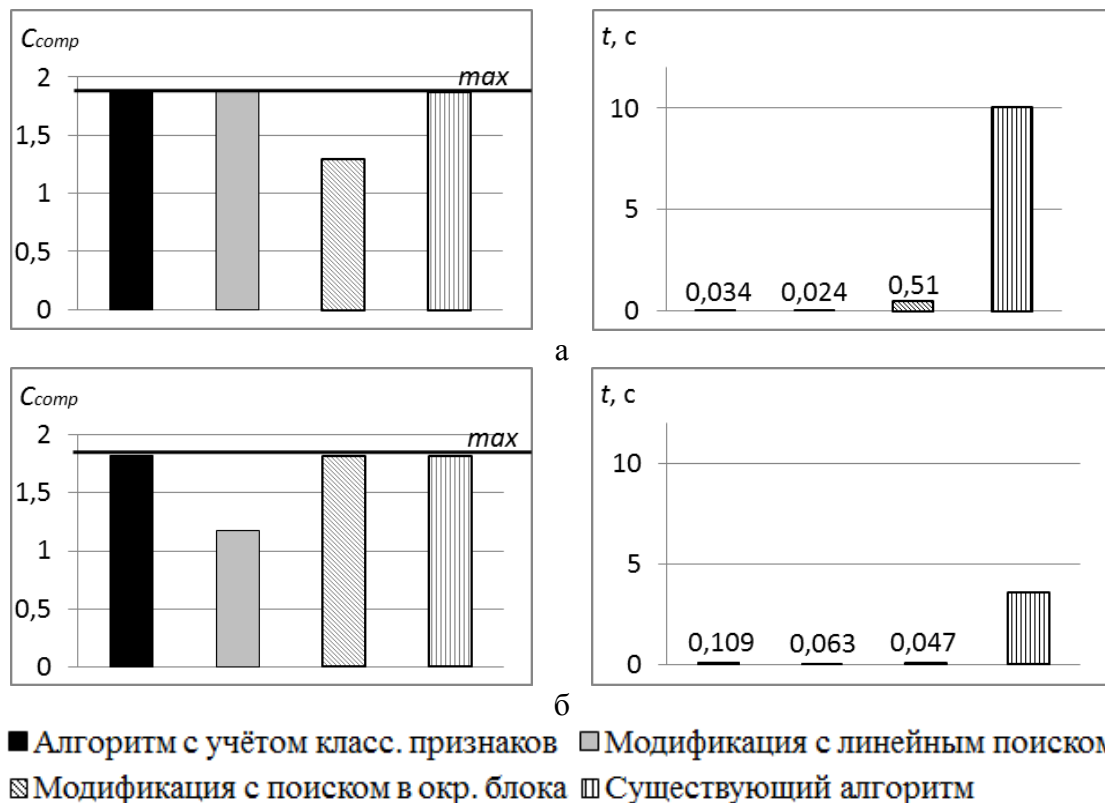


Рисунок 7 – Эффективность алгоритмов оценки движения (степень и время сжатия) на наборах тестовых данных: а) № 1; б) № 2.

Для тестирования использовались наборы данных, соответствующие типам движения в приведённой выше классификации:

1. Движения по вертикали и горизонтали потенциально на большие расстояния (использовался скроллинг).

2. Движения в произвольном направлении на небольшие расстояния (использовалось перетаскивание окна приложения).

Таблица 1 – Варианты реализации алгоритмов оценки движения.

№	Реализация	Краткое обозначение
1	Алгоритм оценки движения с учётом классификационных признаков	Алгоритм с учётом класс. признаков
2	Модификация алгоритма оценки движения, осуществляющая поиск блока по вертикали и по горизонтали	Модификация с линейным поиском
3	Модификация алгоритма оценки движения, осуществляющая поиск блока в ближайшей окрестности текущего блока	Модификация с поиском в окр. блока
4	Существующий алгоритм оценки движения, адаптированный для обработки GUI-видеоданных	Существующий алгоритм

Разработанный алгоритм с учётом классификационных признаков способен ускорить выполнение оценки движения до десяти раз по сравнению с существующим алгоритмом, адаптированным для обработки GUI-видеоданных, при эквивалентном количестве распознанных движений первого и второго типов, что позволяет выполнить условия $C_{comp}(x) \geq C_{min}$, $t(x) \leq T_{max}$ в (1).

Представлены результаты практического сравнения алгоритмов отсечения неизменившихся областей кадра. Показано, что на всех тестовых наборах данных разработанный адаптивный алгоритм способен выбрать конкретный алгоритм отсечения неизменившихся областей кадра, обеспечивающий более высокую степень сжатия.

В **четвёртой главе** приведены детали создания ПО кодека сжатия GUI-видеоданных, основанного на разработанных алгоритмах и результатах их исследований. Предложена архитектура кодека обработки GUI-видеоданных, отвечающая требованиям и соответствующая концептуальным основам создания таких систем, сформулированным выше.

Технология кодирования данных, реализованная в этом кодеке, предполагает использование подсистемы высокопроизводительной обработки данных с использованием видеокарты, которая позволяет при схожем уровне вычислительной эффективности высвободить ЦП для выполнения других операций (рис. 8). Последовательность реализаций алгоритмов, применяемых для сжатия кадра, варьируется в зависимости от наличия в аппаратной платформе видеокарты и в зависимости от того, является кадр ключевым или промежуточным. Второй этап АСПИ применяется только при наличии достаточного количества вычислительных ресурсов.

Для оценки эффективности разработанного кодека использовались видеоданные, полученные при взаимодействии пользователя с различными приложениями:

1. Прикладные приложения (Eclipse, Microsoft Word и др.).
2. Системные приложения.

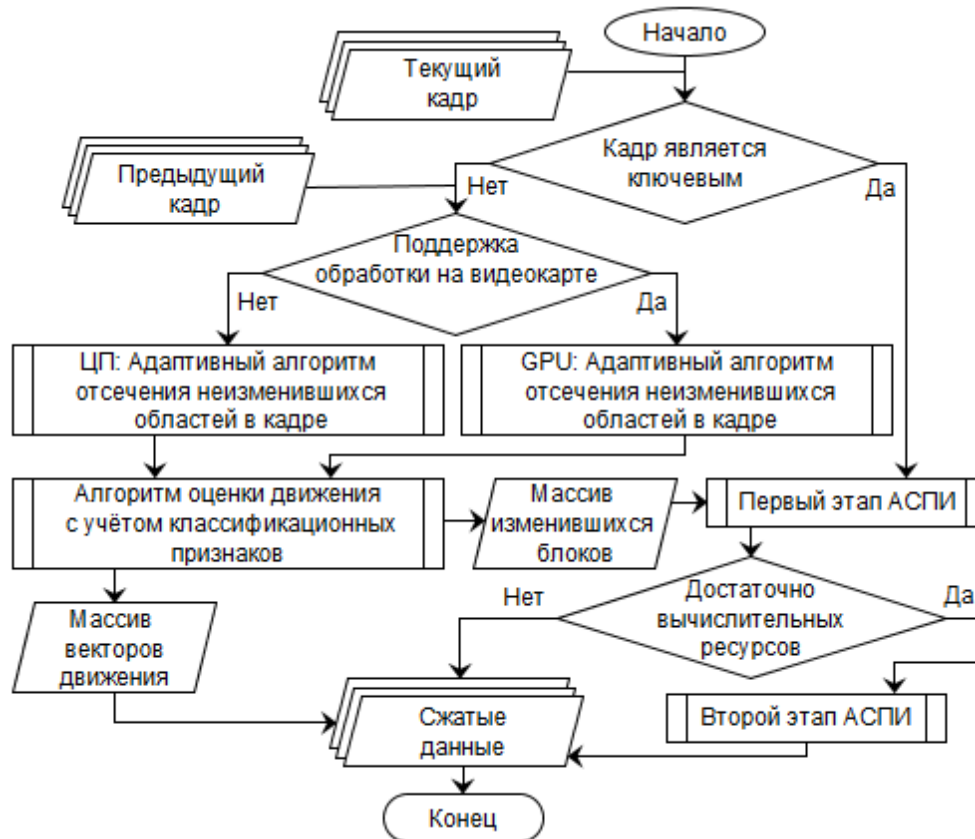


Рисунок 8 – Технология обработки данных кодека Butterfly Screen Video Codec.

В ходе тестирования оценены такие характеристики кодеков, как $C_{сри}$, C_{comp} и M_{avr} . Для всех кодеков выбран режим сжатия без потерь информации. В табл. 2 перечислены кодеки, продемонстрировавшие показатели, близкие к лучшим по выборке для обоих наборов данных.

Таблица 2 – Кодеки сжатия GUI-видеоданных.

№	Кодек	Краткое обозначение
1	Butterfly Screen Video Codec	BSVC
2	Windows Media Encoder 9	Windows Encoder
3	Google VP8 Video Codec (VBR, 5000 kbps)	Google Codec
4	inno Screen Capture Codec 1.9	inno Codec
5	H.264 (без потерь информации)	H.264

BSVC и Windows Encoder обеспечили наивысшую степень сжатия первого тестового набора видеоданных. Уровень использования ресурсов ЦП у BSVC ниже, чем у Windows Encoder в 6 раз на этом наборе видеоданных (рис. 9а). Параметр C_{comp} второго тестового набора видеоданных, продемонстрированный BSVC чуть ниже (на 11 %),

чем у Google Codec. При этом параметр C_{cpu} у представленного кодека ниже, чем у Google Codec в 4,3 раза (рис. 9б). Значение параметра M_{avr} кодека BSVC близко к минимальному по выборке на всех тестовых наборах данных. Стоит отметить, что на рис. 9 значения параметра C_{cpu} приведены в расчёте на одно ядро процессора.

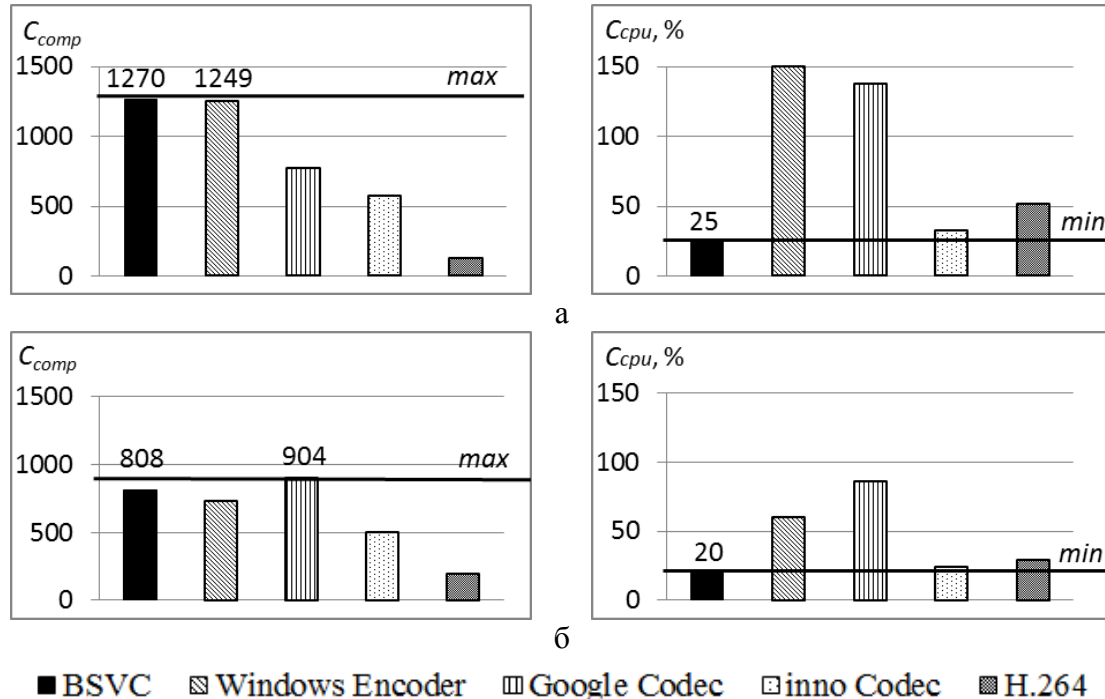


Рисунок 9 – Эффективность основных кодеков (степень сжатия и степень загрузки ЦП) на наборах тестовых данных: а) № 1; б) № 2.

На основании вышеизложенного следует сделать вывод о том, что разработанный кодек, занимая лидирующие позиции по параметрам C_{comp} и M_{avr} , демонстрирует значительно более низкие значения целевого параметра C_{cpu} , по совокупности характеристик превосходит аналоги на всех тестовых наборах видеоданных и позволяет выполнять все условия обработки GUI-видеоданных (1).

В **заключении** приведены основные выводы и результаты диссертационной работы.

В **приложениях** приведены акты о внедрении полученных результатов, свидетельство о государственной регистрации программы для ЭВМ, подробные результаты тестирования различных реализаций алгоритмов сжатия и описание функционального интерфейса разработанного кодека для сжатия GUI-видеоданных.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ ПО ТЕМЕ ДИССЕРТАЦИИ

Статьи в журналах, включенных в Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук:

1. **Дружинин Д. В.** Комбинированный алгоритм сжатия ключевых кадров экранного видео / Д. В. Дружинин // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2011. – № 3 (16). – С. 67–77. – 0,83 а.л.

2. **Дружинин Д. В.** Комбинированные алгоритмы сжатия ключевых кадров экранного видео / Д. В. Дружинин // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 4 (25). – С. 129–136. – 0,65 а.л.

3. **Дружинин Д. В.** Алгоритмы сжатия экранного видео, использующие корреляцию соседних кадров / Д. В. Дружинин // Известия Алтайского государственного университета. – 2014. – № 1/2 (81). – С. 91–95. – DOI: 10.14258/izvasu(2014)1.2-14. – 0,6 а.л.

Свидетельство о государственной регистрации программы для ЭВМ:

4. Свидетельство о государственной регистрации программы для ЭВМ № 2009615754. Butterfly Screen Video Codec / **Дружинин Д. В.**; правообладатель Дружинин Д. В. (RU). Заявка № 2009614530; дата поступления – 19.08.2009; дата государственной регистрации в Реестре программ для ЭВМ – 16.10.2009.

Публикации в прочих научных изданиях:

5. **Дружинин Д. В.** Сжатие экранного видео с помощью видеокарты. Сравнение технологий / Д. В. Дружинин // Вычислительные методы и программирование: новые вычислительные технологии. – 2008. – Т. 9, № 2. – С. 72–80. – 0,99 а.л.

6. **Дружинин Д. В.** Алгоритм оценки движения, адаптированный для обработки экранного видео / Д. В. Дружинин // Вычислительные методы и программирование: новые вычислительные технологии. – 2009. – Т. 10, № 2. – С. 117–122. – 0,68 а.л.

7. **Дружинин Д. В.** Гибридный алгоритм сжатия изображения. Сравнение алгоритмов сжатия изображений / Д. В. Дружинин // Информационные технологии и математическое моделирование (ИТММ-2007) : материалы VI Международной научно-практической конференции. Анжеро-Судженск, 09–10 ноября 2007 г. – Томск, 2007. – Ч. 2. – С. 70–73. – 0,21 а.л.

8. **Дружинин Д. В.** Модификации гибридного алгоритма сжатия изображений / Д. В. Дружинин // Обратные задачи и информационные технологии рационального природопользования : материалы IV научно-практической конференции. Ханты-Мансийск, 22–23 апреля 2008 г. – Ханты-Мансийск, 2008. – С. 218–222. – 0,5 а.л.

9. **Дружинин Д. В.** Алгоритм классификации блоков изображения, оптимизированный для выполнения на видеокарте / Д. В. Дружинин // Информационные технологии и математическое моделирование (ИТММ-2009) : материалы VIII Всероссийской научно-практической конференции с международным участием. Анжеро-Судженск, 12–13 ноября 2009 г. – Томск, 2009. – Ч. 2. – С. 206–211. – 0,3 а.л.

10. **Дружинин Д. В.** Практическое сравнение кодека butterfly screen video codec с его аналогами / Д. В. Дружинин // Информационные технологии и математическое моделирование (ИТММ-2010) : материалы IX Всероссийской научно-практической конференции с международным участием. Анжеро-Судженск, 19–20 ноября 2010 г. – Томск, 2010. – Ч. 2. – С. 36–41. – 0,28 а.л.

11. **Дружинин Д. В.** Применение гибридного алгоритма для сжатия экранного видео / Д. В. Дружинин // Информационные технологии и математическое моделирование (ИТММ-2013) : материалы XII Всероссийской научно-практической конференции с международным участием им. А. Ф. Терпугова. Анжеро-Судженск, 29–30 ноября 2013 г. – Томск, 2013. – Ч. 1. – С. 16–20. – 0,31 а.л.

12. **Дружинин Д. В.** Эволюция гибридного алгоритма сжатия дискретно-тоновой графики / Д. В. Дружинин // Интеллектуальный потенциал XXI века: ступени познания : сборник материалов XIX Молодежной международной научно-практической конференции. Новосибирск, 18 декабря 2013 г. – Новосибирск, 2013. – С. 117–121. – 0,28 а.л.

13. **Дружинин Д. В.** Развитие гибридного алгоритма сжатия дискретно-тоновой графики / Д. В. Дружинин // Научное творчество молодежи. Математика. Информатика : материалы XVIII Всероссийской научно-практической конференции. Анжеро-Судженск, 24–25 апреля 2014 г. – Томск, 2014. – Ч. 1. – С. 124–128. – 0,26 а.л.

14. **Дружинин Д. В.** Гибридный алгоритм сжатия дискретно-тоновой графики / Д. В. Дружинин // Прикладная дискретная математика. – 2014. – № 7 : Компьютерная безопасность и криптография : труды Всероссийской конференции «XII Сибирская школа-семинар с международным участием». – С. 116–118. – 0,28 а.л.

15. **Дружинин Д. В.** Адаптивный алгоритм сжатия видеоданных графического интерфейса пользователя / Д. В. Дружинин, А. В. Замятин // Информационные технологии и математическое моделирование (ИТММ-2016) : материалы XV Международной конференции имени А. Ф. Терпугова. Катунь, 12–16 сентября 2016 г. – Томск, 2016. – Ч. 2. – С. 96–102. – DOI: 10.17223/9785751124335/19. – 0,35 / 0,18 а.л.

16. **Дружинин Д. В.** Концептуальные основы создания программного обеспечения кодека для сжатия видеоданных графического интерфейса пользователя / Д. В. Дружинин, А. В. Замятин // Информационные технологии и математическое моделирование (ИТММ-2017) : материалы XVI Международной конференции имени А. Ф. Терпугова. Казань, 29 сентября – 03 октября 2017 г. – Томск, 2017. – Ч. 2. – С. 34–40. – 0,35 / 0,18 а.л.

Издание подготовлено в авторской редакции.
Отпечатано на участке цифровой печати Издательского Дома
Томского государственного университета.
Заказ № 4712-18 от «24» декабря 2018 г.
Тираж 100 экз.
г. Томск Московский тр.8 тел. 53-15-28