МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»

(Самарский университет)

Институт информатики, математики и электроники

Факультет информатики

Кафедра геоинформатики и информационной безопасности

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

«ПОВЫШЕНИЕ КАЧЕСТВА ВИДЕО ДЛЯ ЗАДАЧ КРИМИНАЛИСТИЧЕСКОЙ ЭКСПЕРТИЗЫ»

по направлению подготовки 10.05.03 Информационная безопасность автоматизированных систем

(уровень академического специалитета)

направленность (профиль) «Обеспечение информационной безопасности распределенных информационных систем»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Обучающийся |  |  |  | Цой Г.В. |
|  | (подпись) | | |  |
| Руководитель ВКР,  профессор, д.т.н |  |  |  | Сергеев В.В. |
|  | (подпись) | | |  |
| Рецензент |  |  |  |  |
|  | (подпись) | | |  |
| Нормоконтролёр |  |  |  | Клевцова Е.В. |
|  | (подпись) | | |  |

Самара, 2021

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»

(Самарский университет)

Институт информатики, математики и электроники

Факультет информатики

Кафедра геоинформатики и информационной безопасности

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / В.В. Сергеев /

*(подпись) И.О.Фамилия*

«25» января 2021 г.

**Задание на выпускную квалификационную работу (ВКР)**

|  |  |  |
| --- | --- | --- |
| Студенту | | Цою Глебу Владимировичу |
|  | | *(ФИО, полностью)* |
| группы | 6511-100503D | |

1. Тема ВКР: «Повышение качества видео для задач криминалистической экспертизы»   
утверждена приказом по университету № 47 от 25.01.2021 г.

2. Перечень вопросов, подлежащих разработке в ВКР:

3. Консультанты по разделам ВКР (при наличии):

раздел ВКР: −

разрабатываемые вопросы: −

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

*должность, степень подпись И.О.Фамилия*

4. Дата выдачи задания: «15» февраля 2021 г.

5. Срок представления на кафедру законченной ВКР: «10» июня 2021 г.

|  |  |  |
| --- | --- | --- |
| Руководитель ВКР  профессор, д.т.н |  | /В.В. Сергеев/ |
|  | *подпись* | *И.О.Фамилия* |
| Задание принял к исполнению |  | /Г.В. Цой/ |
|  | *подпись студента* | *И.О.Фамилия* |

Реферат

Пояснительная записка: 70 c., 20 рисунка(ов), 4 таблиц(ы), 20 источника(ов), 10 приложений(ия).

Графическая часть: 15 слайдов презентации Power Point.

ЛИНЕЙНОЕ СВЕРХРАЗРЕШЕНИЕ, ОШИБКА ВОССТАНОВЛЕНИЯ, ОПТИМАЛЬНОЕ КОМПЛЕКСИРОВАНИЕ, КРИМИНАЛИСТИКА.

Объектом исследования является

Целью работы является

В результате работы разработан

Содержание

[Введение 6](#_Toc70862998)

[1 Описание разработанного метода 8](#_Toc70862999)

[1.1 Общее описание метода 8](#_Toc70863000)

[1.2 Увеличение кадров 10](#_Toc70863001)

[1.3 Формирование дополнительного канала обработки 10](#_Toc70863002)

[1.4 Геометрическое согласование 10](#_Toc70863003)

[1.4.1 Методы, основанные на детектировании особых точек 11](#_Toc70863004)

[1.4.2 Методы, основанные на пирамидальном подходе 11](#_Toc70863005)

[1.4.3 Методы, основанные на вычислении пиков взаимнокорреляционной функции 12](#_Toc70863006)

[1.4.4 Метод, основанный на оптическом потоке 12](#_Toc70863007)

[1.5 Взвешенное суммирование 12](#_Toc70863008)

[2 Описание программной реализации 13](#_Toc70863009)

[2.1 Первый этап разработки 13](#_Toc70863010)

[2.1.1 Входной параметр 14](#_Toc70863011)

[2.1.2 Согласование кадров выбранным методом 15](#_Toc70863012)

[2.1.3 Вычисление среднеквадратической ошибки 16](#_Toc70863013)

[2.1.4 Выходной параметр 16](#_Toc70863014)

[2.2 Второй этап разработки 17](#_Toc70863015)

[2.2.1 Модуль графического интерфейса 18](#_Toc70863016)

[2.2.2 Модуль предобработки 24](#_Toc70863017)

[2.2.3 Модуль интерполяции 25](#_Toc70863018)

[2.2.4 Модуль вычисления ошибки интерполяции 26](#_Toc70863019)

[2.2.5 Модуль согласования 26](#_Toc70863020)

[2.2.6 Модуль комплексирования 27](#_Toc70863021)

[3 Экспериментальное исследование 28](#_Toc70863022)

[3.1 Сравнительное исследование методов геометрического согласования 28](#_Toc70863023)

[3.2 Исследование разработанного метода 28](#_Toc70863024)

[Заключение 29](#_Toc70863025)

[Определения, обозначения и сокращения 30](#_Toc70863026)

[Список использованных источников 31](#_Toc70863027)

[Приложение А 32](#_Toc70863028)

[Приложение Б 33](#_Toc70863029)

[Приложение В 34](#_Toc70863030)

[Приложение Г 35](#_Toc70863031)

[Приложение Д 36](#_Toc70863032)

[Приложение Е 37](#_Toc70863033)

[Приложение Ж 38](#_Toc70863034)

[Приложение И 39](#_Toc70863035)

[Приложение К 40](#_Toc70863036)

[Приложение Л 41](#_Toc70863037)

[Приложение М 42](#_Toc70863038)

[Приложение Н 43](#_Toc70863039)

[Приложение О 44](#_Toc70863040)

[Приложение П 45](#_Toc70863041)

[Приложение Р 46](#_Toc70863042)

[Приложение С 47](#_Toc70863043)

[Приложение Т 48](#_Toc70863044)

[Приложение У 49](#_Toc70863045)

Введение

Обеспечить информационную безопасность защищаемого объекта можно многими способами. Некоторые из них напрямую связаны с использованием изображений: изображения используются в стеганографии в качестве контейнера, на защищаемых объектах аудит обеспечивают камеры видеонаблюдения, совсем недавно стали применяться биометрические системы аутентификации. Изображения могут применяться не только как средство защиты, но и как защищаемые объекты.

Изображения могут быть подвержены искажениям в результате атаки либо непреднамеренно из-за аппаратных ограничений используемой техники. В любом случае всегда может потребоваться восстановить потерянное качество. Менее искаженные изображения большего разрешения позволят более точно фиксировать лица, производить снятие биометрии или обеспечивать восстановление доказательственной, ориентирующей информации в целях досудебного производства и предварительного расследования.

Существует класс методов восстановления изображений, называемый сверхразрешением. Такие методы способны повышать пространственное разрешение кадра с использованием различных технологий.

Подавляющее большинство методов способны визуально улучшать качество, принимая за меру качества коэффициент различимости объектов. Для задач криминалистики такой подход недопустим, так как в результате обработки изображений могут возникать несуществующие в реальности элементы, называемые артефактами.

Методы сверхразрешения можно проклассифицировать по количеству используемых кадров, необходимых для восстановления. Существуют методы, использующие как единственный кадр, так и серию кадров видеопоследовательности.

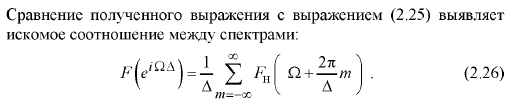
Также методы классифицируются по применяемым технологиям. Выделяют методы на основе искусственного интеллекта, вейвлет-преобразования, проекции на выпуклые множества, адаптивной фильтрации, а также методов преобразований Фурье.

Целью данной дипломной работы является разработка оптимального метода получения сверхразрешения, а также его программная реализация. Оптимизация будет обеспечиваться минимизацией среднеквадратичной ошибки (СКО) для серии последовательных кадров видеопоследовательности с использованием подхода, основанного на преобразованиях Фурье.

Разрабатываемый метод сможет применяться не только в задачах криминалистики, а также в таких областях, как медицина, компьютерное зрение, обработка данных дистанционного зондирования Земли, астрономия и микроскопия и других.

1. Описание разработанного метода
   1. Общее описание метода

Современные средства видеозаписи фиксируют кадры с некоторыми искажениями, которые препятствуют дальнейшей обработке полученных кадров. Изображение, фиксируемое на матрице видеокамеры, подвергается динамическим искажениям в непрерывной временной области, затем дискретизируется по времени, после чего на сигнал накладываются шумы. Таким образом для наблюдения доступен только дискретный выходной сигнал.



ТУТ БУДЕТ НОРМАЛЬНАЯ КАРТИНКА С НЕПРЕРЫВНО-ДИСКРЕТНОЙ МОДЕЛЬЮ НАБЛЮДЕНИЯ.

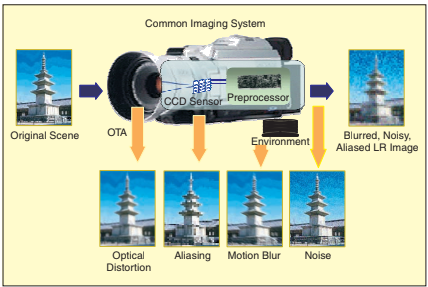


Рисунок 1 – Непрерывно-дискретная модель наблюдения

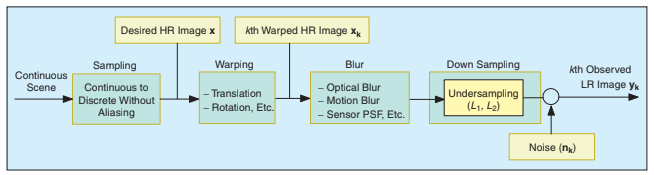


Рисунок 2 – Непрерывно-дискретная модель наблюдения

Итоговая формула непрерывно-дискретной модели наблюдения:

 (1)

Задача сверхразрешения – избавиться от искажений, вызванных аппаратными несовершенствами техники, и приблизиться к восстановлению дискретного изображения до непрерывного.

Необходимо учесть, что при компьютерной обработке сигналов невозможно обрабатывать сигналы в непрерывном виде, но можно преобразовывать сигнал в квазинепрерывный.

Идея разрабатываемого метода заключается в том, что изображение преобразуется из дискретного в квазинепрерывное путём учащения сетки дискретизации. В основе метода лежит оптимальное комплексирование серии изображений.

Алгоритм оптимального комплексирования изображений следующий:

* Увеличение кадров;
* Формирование дополнительного канала обработки;
* Геометрическое согласование кадров;
* Формирование восстановленного изображения.

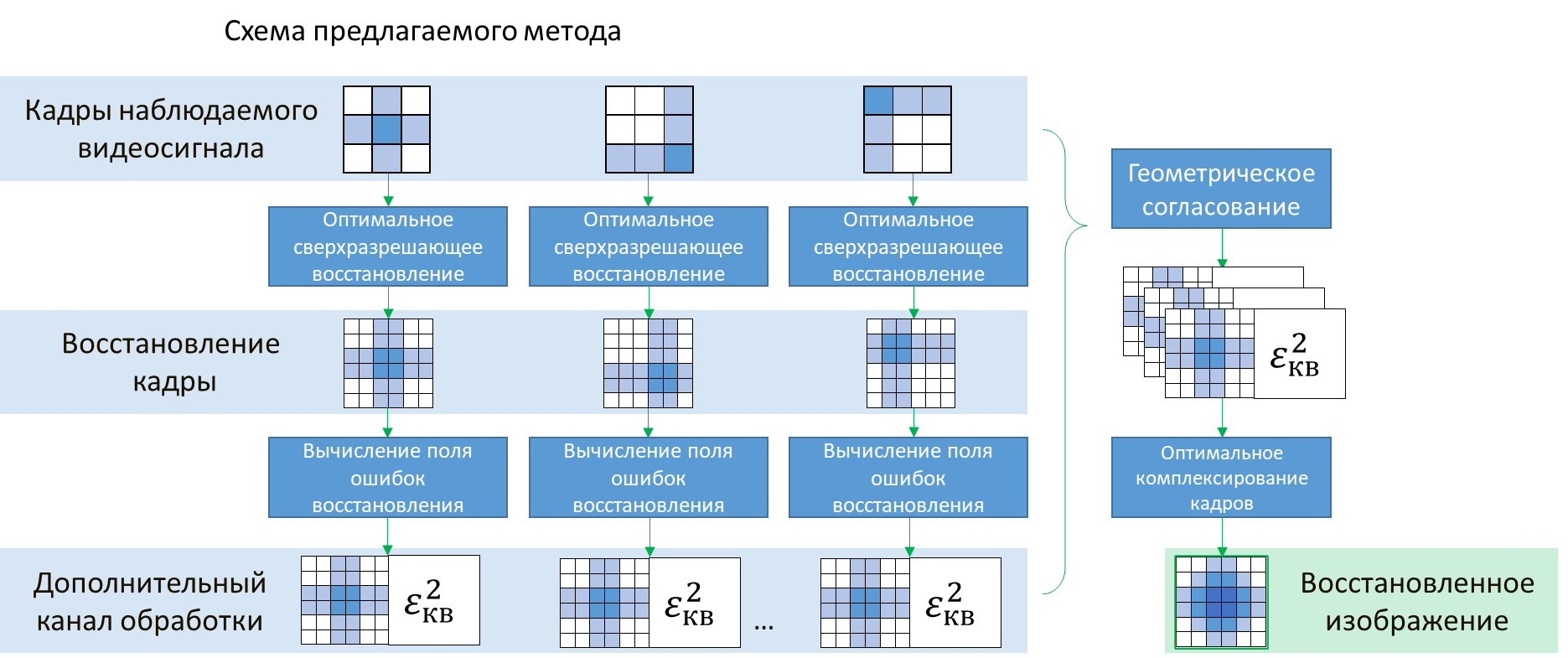


Рисунок 2 - Схема предлагаемого метода

* 1. Увеличение кадров

Первым шагом алгоритма является учащение сетки пикселей с интерполяцией значений. Интерполяцией называют вычисление промежуточных значений сигнала по дискретному набору данных. В рассматриваемой задаче используется линейная интерполяция, которая вычисляется по формуле:

(1)

* 1. Формирование дополнительного канала обработки

После того, как исходные кадры увеличены, необходимо дополнить каждый кадр видеопоследовательности вспомогательным каналом, заполненным дисперсиями ошибок интерполяции. В дальнейшем значения дисперсий будут преобразованы и использованы в итоговой формуле восстановления отсчётов.

ВЫВОД ФОРМУЛЫ

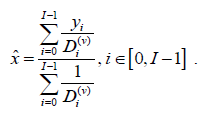
ВЫВОД ФОРМУЛЫ

* 1. Геометрическое согласование

Геометрическое согласование – это процесс преобразования последовательности кадров таким образом, чтобы интересуемый объект находился на каждом кадре в единой системе координат, соответствующей эталонному изображению. На этом шаге производится получение матриц перехода при помощи выбранных методов геометрического согласования. Исходные кадры и поля дисперсии ошибок, созданные на предыдущем шаге, преобразуются в соответствии с матрицами перехода. Ниже рассмотрены наиболее значимые группы методов согласования.

* + 1. Методы, основанные на детектировании особых точек. С помощью детекторов на изображении идентифицируются наиболее значимые точки и с помощью дескрипторов обеспечивается инвариантность нахождения одних и тех же особых точек относительно преобразований изображений. После того, как определены ключевые точки происходит их сопоставление на эталонном и согласуемом изображениях. Сопоставление точек сопровождается трансформацией изображения. Существует несколько способов поиска ключевых точек. Наиболее популярными являются:
* Алгоритм scale-invariant feature transform (SIFT);
* Алгоритм speeded up robust features (SURF);
* Алгоритм Binary Robust Independent Elementary Features + Star detector (BRIEF + Star)
* Алгоритм Oriented Features from Accelerated Segment Test and rotated BRIEF (ORB).
  + 1. Методы, основанные на пирамидальном подходе. Представление одного и того же изображения в нескольких масштабах называется пирамидой. Принцип действия метода пирамид заключается в том, что исходные эталонное и согласуемое изображения представляются в виде пирамиды. На некотором уровне пирамиды между эталоном и согласуемым изображениями находится соответствие в точке. Далее изображение трансформируется согласно найденным точкам. Существует пять возможных модификаций пирамидального подхода:
* Сдвиг;
* Сдвиг и поворот;
* Сдвиг, масштаб и поворот;
* Аффинное преобразование;
* Билинейное преобразование.
  + 1. Методы, основанные на вычислении пиков взаимокорреляционной функции. Принцип действия методов заключается в применении прямого и обратного преобразований Фурье. Вычислив спектры сигнала, можно получить пиковые значения взаимокорреляционной функции. Полученные функции трансформируются согласно эталонной, что соответственно приводит к трансформации изображений. Существуют разные модификации этого метода, но в основном выделяют следующие:
* Основанные на «классических» преобразованиях Фурье;
* Основанные на вычислении меры «хи-квадрат».
  + 1. Метод, основанный на оптическом потоке. Оптический поток – изображение видимого движения, представляющее собой сдвиг каждой точки между двумя изображениями. Суть оптического потока в том, что он не ищет какие-то особенные точки, а по параметрам изображений пытается определить, куда сместилась произвольная точка.
  1. Взвешенное суммирование

Необходимо построить линейную оценку, обеспечивающую наибольшую точность в терминах среднеквадратической ошибки. На этом шаге происходит вычисление значений отсчетов результирующего изображения в каждой точке с использованием выражения 3

 (3)

1. Описание программной реализации

Для программной реализации метода необходимо выбрать такой язык программирования, который обеспечит наибольшую эффективность при работе с цифровыми изображениями. Также язык должен поддерживать разработку графического интерфейса.

Было принято решение использовать язык Python 3.9, так как для него уже существуют библиотеки, оптимизированные для работы с изображениями и двумерными массивами. С помощью существующих библиотек для работы с пользовательским интерфейсом становится возможно создать минимально жизнеспособный продукт, пригодный для конечных пользователей.

Программная реализация проходила в два этапа:

1. Сравнительное исследование наиболее эффективного метода согласования серии кадров видеопоследовательности;
2. Разработка пользовательского интерфейса, позволяющего получить сверхразрешенное изображение по серии кадров видеопоследовательности.
   1. Первый этап разработки

Кадры необходимо согласовать с высокой точностью, так как ошибка согласования оказывает существенное влияние на результирующее изображение. Для выбора наиболее эффективного метода согласования было произведено сравнительное исследование двенадцати методов согласования кадров. В качестве меры сравнения применялась среднеквадратическая ошибка согласования кадров относительно единственного эталонного изображения. Каждый метод реализован по отдельности в виде консольного приложения.

Представленная ниже схема является общей для каждого метода согласования:

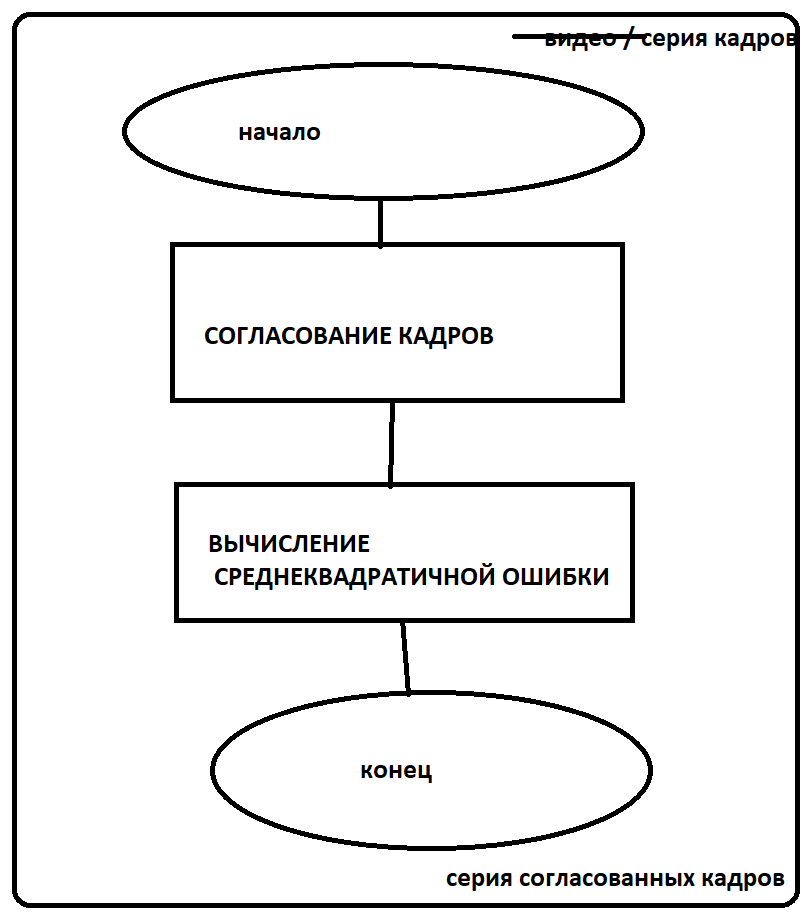


Рисунок 666 – Блок-схема первого этапа разработки

Программный код каждого рассмотренного метода представлен в приложениях А-Н.

Рассмотрим отдельно каждый блок схемы.

* + 1. Входной параметр

В качестве входного параметра используется последовательная серия кадров одного видео. Входные изображения конвертируются и представляются в форме двумерных массивов беззнакового байтового формата. Первый кадр последовательности принимается за эталонный. С эталонным изображением согласуются остальные кадры. Пример кадров можно видеть на рисунке 666:

|  |  |
| --- | --- |
|  |  |
|  |  |

Рисунок 666 – Пример кадров видеопоследовательности

* + 1. Согласование кадров выбранным методом

Для 12 различных методов согласования применяются 12 соответствующих функций из разных библиотек. В таблице 1 приведено соответствие каждого метода согласования с используемой библиотекой.

Таблица 1 – Программные библиотеки, соответствующие рассмотренным методам согласования

|  |  |
| --- | --- |
| Методы, основанные на детектировании особых точек | OpenCV |
| Методы, основанные на пирамидальном подходе | pystackreg |
| Методы, основанные на вычислении пиков взаимнокорреляционной функции | image\_registration |
| Метод, основанный на оптическом потоке | skimage |

* + 1. Вычисление среднеквадратической ошибки

Полученная согласованная последовательность кадров сравнивается с эталонным. В качестве меры сходства принята среднеквадратическая ошибка между эталоном и обработанным кадром. Расчёт СКО производится по формуле 666:

(666)

где mse – СКО;

m – ширина изображения в пикселях;

n – длина изображения в пикселях;

I(i, j) – значение i,j-ого пикселя результирующего изображения;

K(i, j) – значение i,j-ого пикселя эталонного изображения.

* + 1. Выходной параметр

В результате работы кода генерируются согласованные кадры, а также текстовый файл, содержащий вычисленные значения СКО. С примерами согласования можно ознакомиться на рисунке 666.

|  |  |
| --- | --- |
|  |  |
|  |  |

Рисунок 666 – Пример кадров видеопоследовательности

На основе полученных значений СКО было произведено сравнительное исследование наиболее эффективного метода согласования серии кадров.

* 1. Второй этап разработки

Целью второго этапа была разработка пользовательского приложения для получения сверхразрешенного изображения. В качестве входного параметра принимается видео либо набор кадров. Если на вход подаётся видео – программа преобразует его в последовательность кадров самостоятельно. Кадры хранятся в памяти компьютера как двумерные массивы беззнакового байтового формата. Входная последовательность кадров подвергается предобработке, затем она проходит через алгоритм оптимального комплексирования, реализованный в виде отдельных модулей. Для удобства работы предусмотрен модуль, отвечающий за графическую оболочку программы.

Принципиальная схема работы программы представлена на рисунке 666:

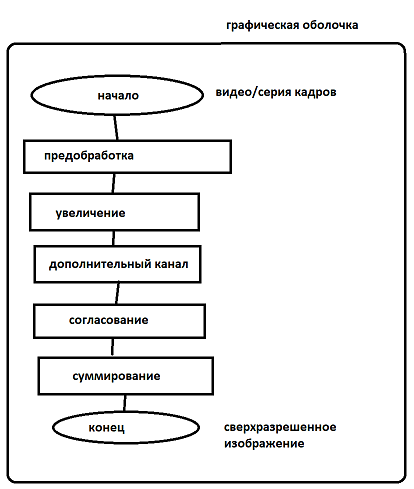


Рисунок 666 – Блок-схема второго этапа разработки

Таким образом в работе программы задействованы шесть модулей: графического интерфейса, предобработки, интерполяции, вычисления ошибки интерполяции, согласования и комплексирования. В результате работы программы генерируется единое сверхразрешенное изображение.

С полным кодом каждого разработанного модуля можно ознакомиться в приложениях О-У.

Ниже подробно описана работа каждого модуля.

* + 1. Модуль графического интерфейса

Графический интерфейс разрабатывался с использованием библиотеки «Tkinter». Визуально программа представляет собой окно 800 на 600 пикселей. При запуске исполняемого файла окно размещается по центру монитора.

Ниже представлено изображение окна сразу после запуска исполняемого файла.

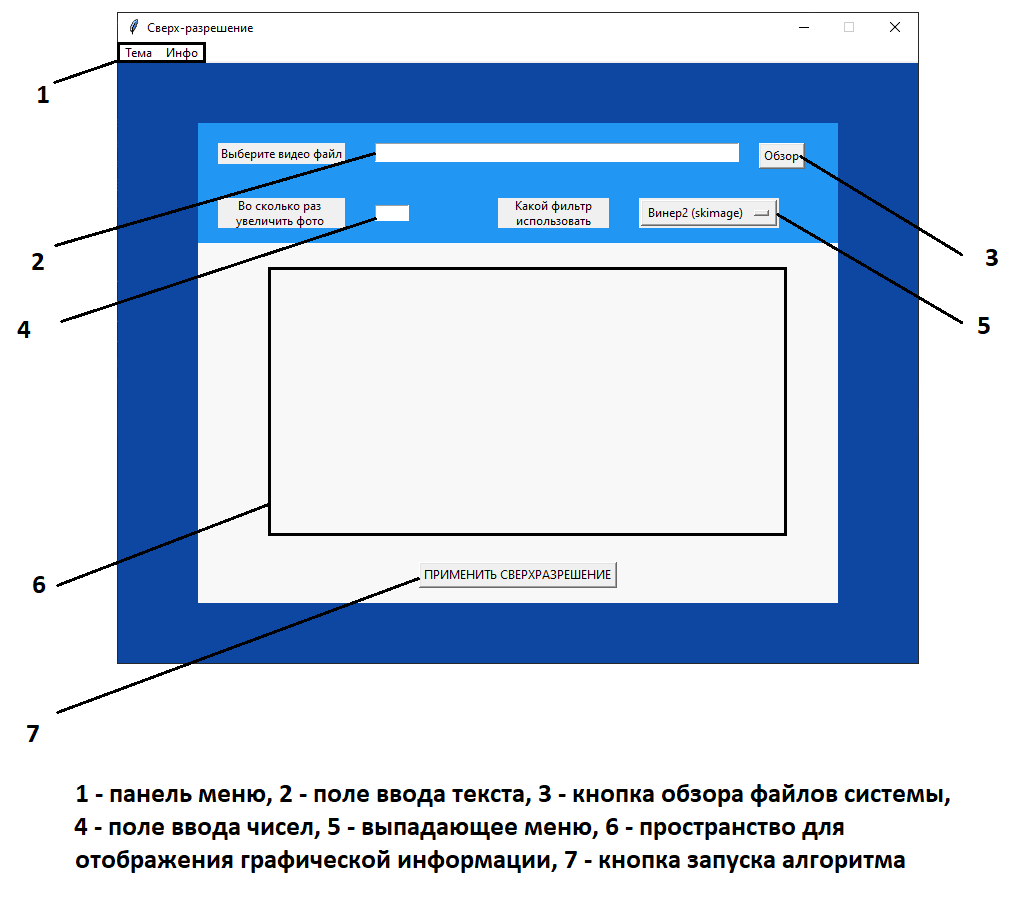


Рисунок 666 – Интерфейс разработанной программы

Ниже подробно описан каждый элемент графической оболочки.

В панель меню 1 ходит кнопка «Инфо», при нажатии на которую выводится новое окно с информацией о приложении, а также подробная инструкция по работе с приложением.

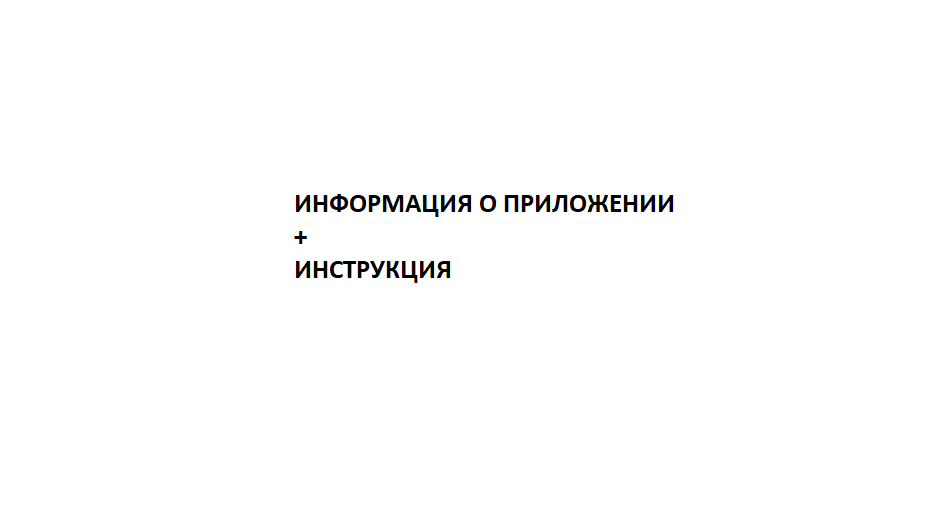


Рисунок 666 – Дополнительное окно с информацией о приложении

Поле для ввода текста 2 предназначено для указания расположения одного видеофайла или нескольких графических файлов.

Кнопка 3 «Обзор» осуществляет открытие нового диалогового окна для выбора файла из каталогов файловой системы. Для удобства работы реализована функция фильтрации файлов по доступным форматам: «mp4», «wmv», «avi», «gif», «jpeg» и «png». Пример открытого диалогового окна представлен на рисунке 666:

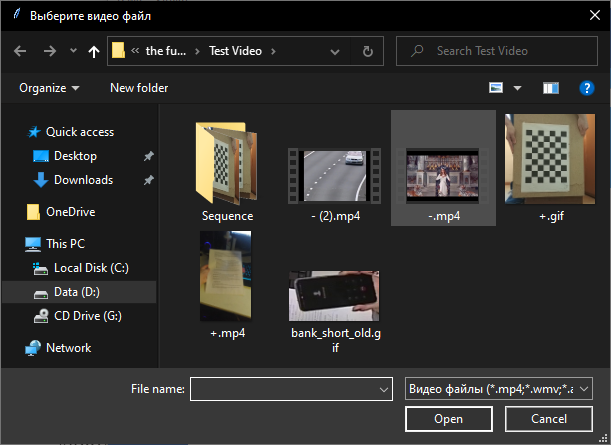


Рисунок 666 – Пример диалогового окна с видео файлами

После выбора файла диалоговое окно закрывается. Поле 2 заполняется абсолютным путём к выбранному файлу, а пространство 6 воспроизводит анимацию графического файла. Пример работы программы с выбранным файлом показан на рисунке 666:

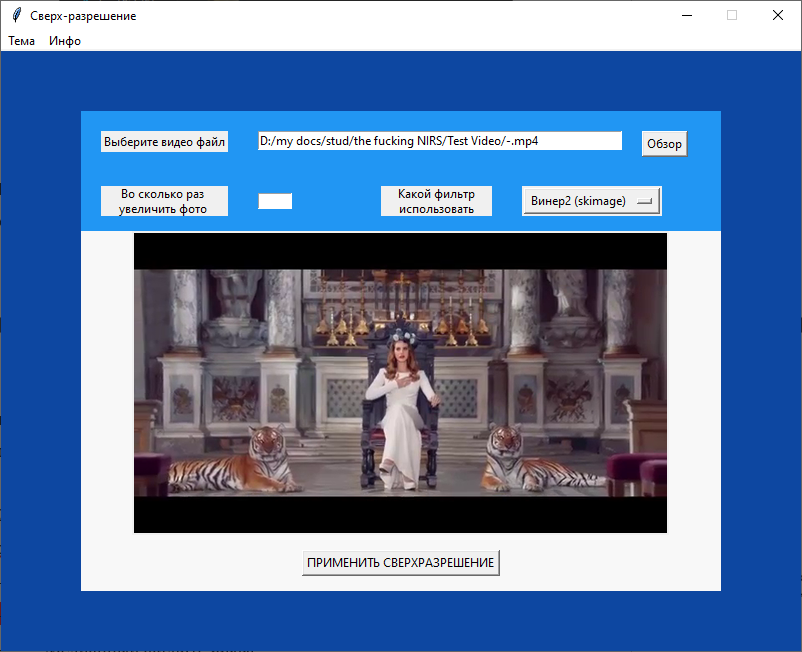


Рисунок 666 – Пример анимации

Анимация осуществляется при помощи параллельно выполняющегося процесса, который обрабатывает файл по каждому следующему кадру. После того как процесс обработает последний кадр последовательности, анимация прекращается.

В том случае, если пользователем выбран файл, неподдерживаемого формата, то в пространстве 6 отображается предупреждающее сообщение. Пример вывода сообщения представлен на рисунке 666:

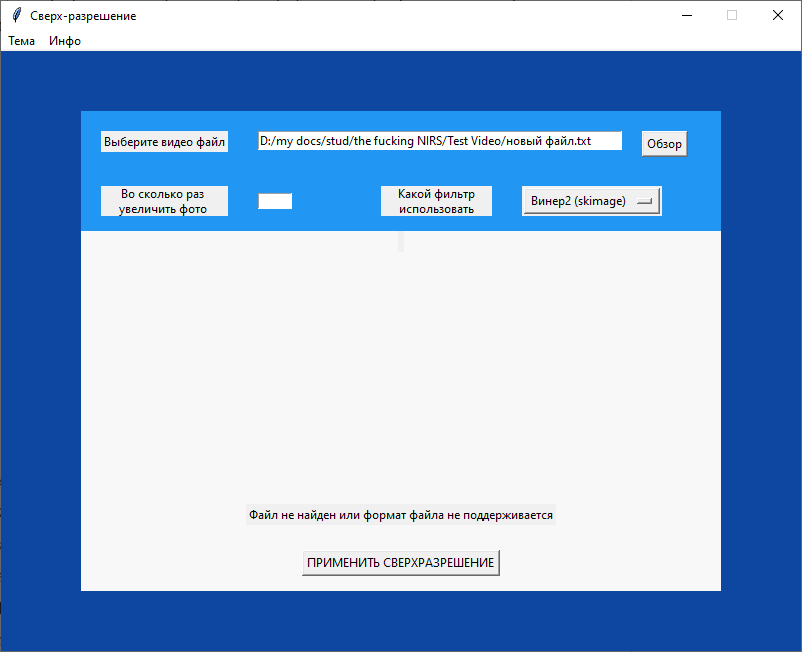


Рисунок 666 – Пример вывода ошибки

Поле ввода 4 предназначено для ввода чисел. Поле работает совместно с функцией форматно-логического контроля. Функция осуществляет проверку введённых значений. Согласно этой функции, поле ограничено по вводу не более чем двумя цифрами. При попытке ввода символов, отличающихся от арабских цифр, символы не будут заноситься в поле.

Выпадающее меню 5 содержит шесть активных элементов. Выбранный элемент характеризует дальнейший способ предобработки изображений.

* элемент «Без предобработки» соответствует пропуску этапа предобработки;
* элемент «Винер» соответствует обработке изображения фильтром Винера;
* элемент «Гаусс» соответствует обработке изображения фильтром Гаусса;
* элемент «Медианный» соответствует обработке изображения медианным фильтром;
* элемент «Контраст» соответствует обработке изображения методом линейного контрастирования;
* элемент «Резкость» соответствует обработке изображения при помощи фильтра маски нерезкости.

Изначально параметр проинициализирован значением «Винер».

Пример выпадающего меню изображен на рисунке 666:

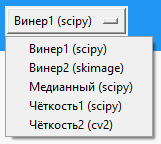


Рисунок 666 – Элементы выпадающего меню

Пространство 6 предназначено для вывода пользователю графической и текстовой информации. В процессе работы программы пользователю выводятся:

* анимация серии кадров;
* предупреждающие сообщения;
* прогресс выполнения алгоритма;
* результат работы программы.

Анимация кадров воспроизводится во время выбора файла из каталога файловой системы.

Предупреждающие сообщения возникают в результате ошибок, вызванных как действиями пользователя, так и нештатным поведением системы. Нештатные поведения могут быть вызваны нехваткой оперативной памяти или неверным представлением графической информации (поврежденный файл или неподдерживаемая кодировка).

Прогресс выполнения алгоритма представляется в виде индикатора, заполняющегося зелёной линией, соответствующей процентному соотношению от выполнения. Индикатор сопровождается текстовым полем с уточняющей информацией – на каком шаге находится выполнение алгоритма. Успешно завершенный шаг алгоритма соответствует полностью заполненному индикатору. Каждому шагу алгоритма соответствует свой индикатор. Как только алгоритм переходит на следующий шаг – старый индикатор очищается, и в окне появляется новый.

Ниже представлен пример индикации прогресса выполнения работы алгоритма:

|  |  |
| --- | --- |
| а) | б) |
| в) | г) |
| д) | е) |
| Рисунок 666 – Пример индикации прогресса работы алгоритма. Этап предобработки (а), увеличение сетки кадров (б), вычисление ошибки интерполяции (в), согласование серии кадров (г), взвешенное суммирование отсчётов (д), результат работы алгоритма (е) | |

Кнопка 7 осуществляет запуск алгоритма оптимального комплексирования изображений с дополнительной предварительный обработкой. По нажатию кнопки считываются значения во всех полях интерфейса, при этом в соответствующей функции осуществляется проверка данных на валидность:

1. если хотя бы одно из полей 2, 4, 5 пустое, то в поле 6 выводится ошибка;
2. если поле 2 заполнено несуществующим путём к файлу, то в поле 6 выводится ошибка;
3. если все поля заполнены валидными данными, то по нажатию кнопки считается строка из поля 2 и найденный видеофайл (набор изображений) запишется в памяти программы в виде массива, элементами которого являются двумерные массивы беззнакового байтового формата. Дальнейшая работа с массивом проходит в следующих модулях.
   * 1. Модуль предобработки

Модуль необходим для предварительной фильтрации кадров от шумов или для большего выделения границ объектов. Такие манипуляции не вносят ошибки в результат комплексирования и даже наоборот позволяют избавиться от некоторых искажений, следовательно применение модуля предобработки допустимо при работе с изображениями в том числе в криминалистических целях.

Данный модуль соответствует первому шагу алгоритма программы. В качестве входного параметра принимается массив изображений, индикатор прогресса и способ предобработки.

Для использования того или иного метода предобработки пользователем выбирается соответствующий элемент из выпадающего меню, изображенного на рисунке 666 под цифрой 5.

Ниже описана реализация использующихся методов:

* предобработка не производится, если в параметры функции передан аргумент «Без предобработки»;
* фильтр винера применяется в алгоритме, если в параметры функции передан аргумент «Винер». Фильтрация обеспечивается готовой функцией из библиотеки «Skimage»;
* фильтр гаусса применяется в алгоритме, если в параметры функции передан аргумент «Гаусс». Фильтрация обеспечивается готовой функцией из библиотеки «???»;
* медианный фильтр применяется в алгоритме, если в параметры функции передан аргумент «Медианный». Фильтрация обеспечивается готовой функцией из библиотеки «Scipy»;
* линейное контрастирование применяется в алгоритме, если в параметры функции передан аргумент «Контраст». Фильтрация обеспечивается готовой функцией из библиотеки «???».
* выделение границ объектов при помощи маски нерезкости применяется в алгоритме, если в параметры функции передан аргумент «Резкость». Фильтрация обеспечивается готовой функцией из библиотеки «???».

Если пользователем выбран один из элементов (кроме «Без предобработки»), то фильтрация проводится в цикле по каждому кадру. После успешного преобразования каждого кадра вычисляется процент пройденных шагов цикла. Процент выполнения выводится в интерфейсе в виде индикатора. В результате работы модуля выдаётся массив преобразованных изображений. После завершения работы модуля индикатор очищается.

* + 1. Модуль интерполяции

Данный модуль соответствует второму шагу алгоритма программы. В качестве входного параметра принимается массив изображений, полученных на первом шаге, индикатор прогресса и коэффициент увеличения сетки пикселей. Коэффициент указывается в графическом интерфейсе в поле, изображенном на рисунке 666 под цифрой 4.

Интерполяция обеспечивается готовой функцией из существующей библиотеки «OpenCV». Интерполяция проводится в цикле по каждому кадру. После успешного преобразования каждого кадра вычисляется процент пройденных шагов цикла. Процент выполнения выводится в интерфейсе в виде индикатора. В результате работы модуля выдаётся массив преобразованных изображений. После завершения работы модуля индикатор очищается.

* + 1. Модуль вычисления ошибки интерполяции

Данный модуль соответствует третьему шагу алгоритма программы. В качестве входного параметра принимается массив изображений, полученных на втором шаге и индикатор прогресса.

Вычисление ошибки обеспечивается функцией \_\_\_???. Вычисление проводится в цикле по каждому кадру. После успешного вычисления ошибки интерполяции каждого кадра вычисляется процент пройденных шагов цикла. Процент выполнения выводится в интерфейсе в виде индикатора. В результате работы модуля выдаётся список двумерных массивов, состоящих из десятичных чисел. После завершения работы модуля индикатор очищается.

* + 1. Модуль согласования

Данный модуль соответствует четвёртому шагу алгоритма программы. В качестве входного параметра принимается массив изображений, полученных на втором шаге, индикатор прогресса и список двумерных массивов, полученный на третьем шаге.

Сравнительное исследование методов согласования показало, что наиболее эффективным в терминах среднеквадратической ошибки методом является пирамидальный подход с аффинным преобразованием. В связи с этим в модуле был реализован только данный метод. Согласование аффинным методом обеспечивается готовой функцией из существующей библиотеки «pystackreg».

Согласование проводится в цикле по каждому кадру и соответствующему элементу списка ошибок интерполяции. После успешного согласования каждого кадра и преобразования соответствующего элемента списка вычисляется процент пройденных шагов цикла. Процент выполнения выводится в интерфейсе в виде индикатора. В результате работы модуля выдаётся массив преобразованных изображений и преобразованный список массивов ошибок интерполяции. После завершения работы модуля индикатор очищается.

* + 1. Модуль комплексирования

Данный модуль соответствует пятому шагу алгоритма программы. В качестве входного параметра принимается массив изображений и список массивов ошибок интерполяции, полученные на четвёртом шаге, а также индикатор прогресса.

Комплексирование обеспечивается функцией, вычисляющей формулу (3). Комплексирование проводится в цикле по каждому кадру. После успешного преобразования каждого кадра вычисляется процент пройденных шагов цикла. Процент выполнения выводится в интерфейсе в виде индикатора. В результате работы модуля выдаётся единое изображения, являющееся результатом формулы (3).

Также данный модуль обеспечивает вывод результирующего изображения в окне приложения и последующее сохранение изображения в файл в формате PNG в ту же папку, в которой находится исполняемый файл. После завершения работы модуля индикатор очищается и выводится сообщение о результате сохранения файла.

1. Экспериментальное исследование

Фыва.

* 1. Сравнительное исследование методов геометрического согласования

Фыва.

* 1. Исследование разработанного метода

Фыва.

Заключение

Фыва фываывафываф ывафывафыв ывафыва фываф ывафыва фывафыафывафываыв фываыфв аыва ыфва фываываывафыва фывафыв афывафы ыфва ы.

Определения, обозначения и сокращения

Фыва фываывафываф ывафывафыв ывафыва фываф ывафыва фывафыафывафываыв фываыфв аыва ыфва фываываывафыва фывафыв афывафы ыфва ы.

Список использованных источников

Фыва фываывафываф ывафывафыв ывафыва фываф ывафыва фывафыафывафываыв фываыфв аыва ыфва фываываывафыва фывафыв афывафы ыфва ы.

Приложение А

for i in range(1, len(files)-1):

#-------Процесс согласования НАЧАЛО------#

start\_time = time.time()

offset\_image = img\_as\_ubyte(rgb2gray(io.imread(join(pathIn,files[i]))))

noise = 0.1

xoff, yoff, exoff, eyoff = chi2\_shift(ref\_image, offset\_image, noise,

return\_error = True, upsample\_factor = 'auto')

corrected\_image = shift(offset\_image, shift = (-yoff, -xoff), mode = 'constant')

corrected\_image = img\_as\_ubyte((corrected\_image - np.min(corrected\_image)) / (np.max(corrected\_image) - np.min(corrected\_image)))

io.imsave(pathOut+files[i], corrected\_image)

time\_arr[i-1] = time.time() - start\_time

#-------Процесс согласования КОНЕЦ-------#

print("--- %s seconds ---" % (time.time() - start\_time))

sko\_arr[i-1] = np.sum((corrected\_image.astype("float") - ref\_image.astype("float")) \*\* 2)

sko\_arr[i-1] /= float(ref\_image.shape[0] \* ref\_image.shape[1])

print("SKO[" + str(i) + "]: " + str(sko\_arr [i-1]) + "\n")

sko += sko\_arr[i-1]

#-------------Рассчёт СКО----------------#

sko = sko/(len(files)-1)

sko\_arr[len(files)-2] = "Average SKO: " + str(sko)

time\_arr[len(files)-2] = "Total Time: " + str(time.time() - total\_time)

with open(pathOut + 'SKO.txt', 'w') as f:

for i in range(1, len(sko\_arr)+1):

if (i != 50):

f.write("SKO№%s:\t%s\tExecution time %s\n" % (i, sko\_arr[i-1], time\_arr[i-1]))

else:

f.write("\t%s\t%s\n" % (sko\_arr[i-1], time\_arr[i-1]))

#------------Запись в файл---------------#

print("Average SKO: " + str(sko))

print(str(time\_arr[len(files)-2]))

Приложение Б

Приложение В

Приложение Г

Приложение Д

Приложение Е

Приложение Ж

Приложение И

Приложение К

Приложение Л

Приложение М

Приложение Н

Приложение О

Приложение П

Приложение Р

Приложение С

Приложение Т

Приложение У