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](#_Toc71474367)

[1 Описание разработанного метода 8](#_Toc71474368)

[1.1 Общее описание метода 8](#_Toc71474369)

[1.2 Учащение сетки дискретизации кадров 9](#_Toc71474370)

[1.3 Формирование дополнительного канала обработки 10](#_Toc71474371)

[1.4 Геометрическое согласование 10](#_Toc71474372)

[1.4.1 Методы, основанные на детектировании особых точек 12](#_Toc71474373)

[1.4.2 Методы, основанные на пирамидальном подходе 13](#_Toc71474374)

[1.4.3 Методы, основанные на вычислении пиков взаимокорреляционной функции 14](#_Toc71474375)

[1.4.4 Метод, основанный на оптическом потоке 14](#_Toc71474376)

[1.5 Взвешенное суммирование 15](#_Toc71474377)

[2 Описание программной реализации 16](#_Toc71474378)

[2.1 Модуль графического интерфейса 17](#_Toc71474379)

[2.2 Модуль предобработки 23](#_Toc71474380)

[2.3 Модуль интерполяции 24](#_Toc71474381)

[2.4 Модуль вычисления ошибки интерполяции 25](#_Toc71474382)

[2.5 Модуль согласования 25](#_Toc71474383)

[2.6 Модуль комплексирования 26](#_Toc71474384)

[2.7 Модуль автоматизации экспериментальных исследований 26](#_Toc71474385)

[3 Экспериментальное исследование 29](#_Toc71474386)

[3.1 Типичные для криминалистической экспертизы видео данные 29](#_Toc71474387)

[3.2 Сравнительное исследование методов геометрического согласования 31](#_Toc71474388)

[3.3 Исследование эффективности разработанного сверхразрешения 32](#_Toc71474389)

[3.3.1 Первый эксперимент 33](#_Toc71474390)

[3.3.2 Второй эксперимент 35](#_Toc71474391)

[3.4 Сравнительное исследование методов сверхразрешения 36](#_Toc71474392)

[Заключение 37](#_Toc71474393)

[Определения, обозначения и сокращения 38](#_Toc71474394)

[Список использованных источников 39](#_Toc71474395)

[Приложение А 40](#_Toc71474396)

[Приложение Б 41](#_Toc71474397)

[Приложение В 42](#_Toc71474398)

[Приложение Г 43](#_Toc71474399)

[Приложение Д 44](#_Toc71474400)

[Приложение Е 45](#_Toc71474401)

[Приложение Ж 46](#_Toc71474402)

[Приложение И 47](#_Toc71474403)

[Приложение К 48](#_Toc71474404)

[Приложение Л 49](#_Toc71474405)

[Приложение М 50](#_Toc71474406)

[Приложение Н 51](#_Toc71474407)

[Приложение О 52](#_Toc71474408)

[Приложение П 53](#_Toc71474409)

[Приложение Р 54](#_Toc71474410)

[Приложение С 55](#_Toc71474411)

[Приложение Т 56](#_Toc71474412)

[Приложение У 57](#_Toc71474413)

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 непреднамеренным искажениям в результате аппаратных ограничений используемой техники. Существуют различные ситуации, когда потерянное качество требуется восстановить. Менее искаженные изображения большего разрешения позволят более точно проводить экспертизу доказательственной ориентирующей информации в целях досудебного производства и предварительного расследования. В задачи экспертизы помимо установления параметров съёмки входит ещё и установление параметров объектов на видеозаписях.

На предприятии в случае нарушения коммерческой тайны на видеокамере могут быть зафиксированы действующие лица, документы, экраны мониторов, которые могут выступать объектами экспертизы. В ситуациях дорожно-транспортных происшествий интерес представляют марки/модели машин, номерные знаки, дорожные знаки или сигналы светофора, зафиксированные на видеорегистратор. В некоторых случаях проводить экспертизу видеосъёмки представляется невозможным из-за низкого качества видеокадров, что в свою очередь приводит к неразличимости характеристик интересуемых объектов.

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

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.

Формула накладываемых искажений на непрерывный сигнал, поступающий на матрицу видеокамеры:

 (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.

* 1. Общее описание метода

В основе метода лежит оптимальное комплексирование серии изображений. Основной принцип работы предлагаемого метода состоит в том, что дискретизированные отсчёты изображения, называемые пикселями, дискретизируются с ещё большим шагом. Каждый новый получившийся отсчёт имеет свой уникальный весовой коэффициент и вносит тот или ной вклад в результирующее изображение.

В качестве входного параметра в алгоритм подаётся последовательность кадров. Затем для этих кадров увеличивается сетка пикселей с интерполяцией значений, что обеспечивает сверхразрешающую способность метода. После этого создаётся дополнительный канал обработки информации, в который записываются поля дисперсий ошибок интерполяции кадров. Созданный канал обработки необходим для вычисления коэффициентов взвешенного суммирования на последнем шаге. Следом происходит геометрическое согласование кадров с соответствующим преобразованием созданного канала. После чего производится взвешенное суммирование пикселей кадров. Схема алгоритма изображена на рисунке 666:

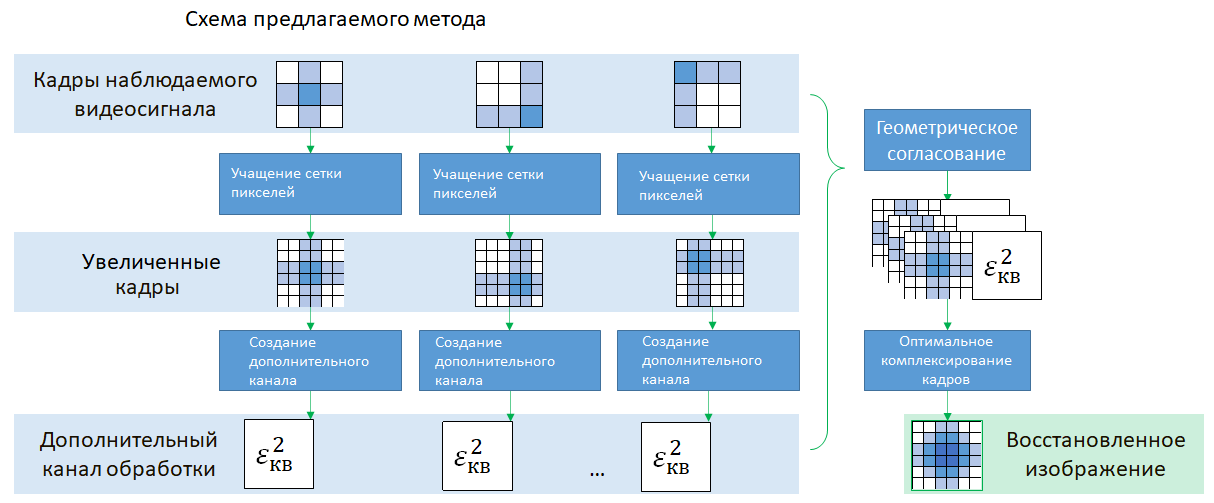


Рисунок 666 - Схема предлагаемого метода

Алгоритм оптимального комплексирования можно разбить на 4 шага:

1. учащение сетки дискретизации кадров;
2. вычисляются поля дисперсии ошибок интерполяции для каждого кадра;
3. геометрическое согласование серии кадров;
4. формирование восстановленного изображения.
   1. Учащение сетки дискретизации кадров

Первым шагом алгоритма является повышение частоты дискретизации кадров видеопоследовательности, что обеспечивает сверхразрешающую способность метода. Увеличить частоту дискретизации возможно разными способами, однако оптимальным считается интерполяция отсчётов. Интерполяцией называют вычисление промежуточных значений сигнала по дискретному набору данных. В рассматриваемой задаче используется линейная интерполяция. Со схемой линейной интерполяции можно ознакомиться на рисунке 666:

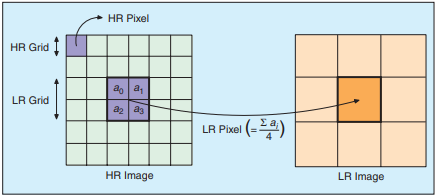


Рисунок 666 - Схема линейной интерполяции

Линейная интерполяция двумерного сигнала вычисляется по формуле:

(1)

* 1. Формирование дополнительного канала обработки

После того, как сетки пикселей всех кадров 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.

При учащении сетки дискретизации возможно получить только некоторую оценку исходной непрерывной сцены, причём оценка определяется с некоторой погрешностью. Эту погрешность для линейной интерполяции можно оценить теоретически. На этапе комплексирования больший вес будет даваться тем отсчетам, ошибка интерполяции которых ниже.

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. При согласовании одно изображение из последовательности принимается за эталонное. Такое изображение статично на протяжении всего процесса согласования. С эталоном сравниваются последующие кадры последовательности и затем изменяются, причём таким образом, чтобы они максимально точно соответствовали эталонному.

На рисунке 666 можно видеть пример согласования двух кадров:

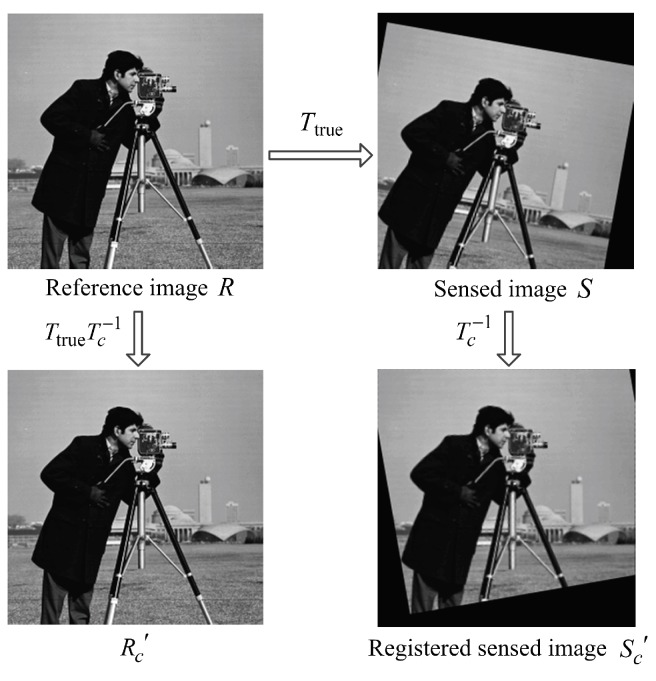


Рисунок 666 – Пример согласования двух кадров видеопоследовательности

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).

На изображении 666 можно видеть пример детектирования особых точек на изображении:



Рисунок 666 – Пример визуализации детектирования особых точек

* + 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.

На изображении 666 можно видеть пример пирамидальной модели:

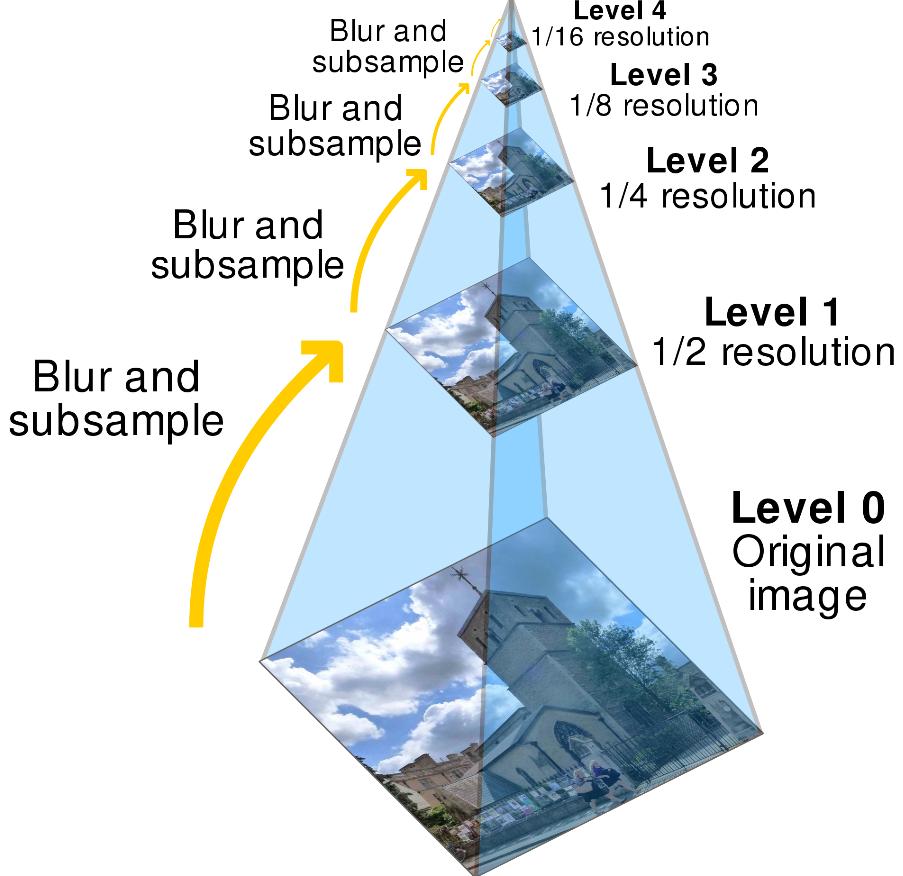


Рисунок 666 – Пример визуализации пирамидальной модели

* + 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».

На изображении 666 можно видеть визуальное изображение корреляционных пиков:

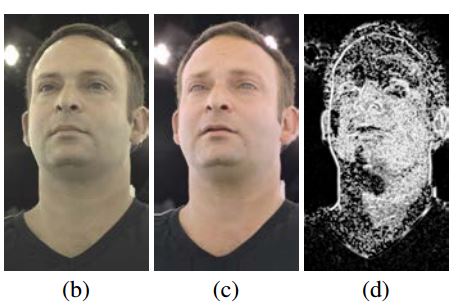


Рисунок 666 – Пример визуализации корреляционных пиков

* + 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.

На изображении 666 можно видеть визуальное изображение оптического потока:



Рисунок 666 – Пример визуализации оптического потока

* 1. Взвешенное суммирование

На последнем шаге алгоритма происходит оптимальное суммирование неравноточных измерений согласованного набора изображений.

Так как для наблюдения доступны только искаженные кадры, то необходимо построить линейную оценку, обеспечивающую наибольшую точность в терминах среднеквадратической ошибки. Представим наблюдаемый сигнал в виде формулы 666:

, (2)

где – доступная для наблюдений величина;

– исходная величина, оценку которой необходимо построить;

– шумовая компонента;

– целочисленный аргумент;

– количество доступных наблюдений.

Так как оптимальность обеспечивается минимизацией среднеквадратической ошибки, то в таком случае линейная оценка будет выглядеть следующим образом:

, (666)

, (666)

где – оценка исходной величины по наблюдаемой;

– минимизируемая среднеквадратическая ошибка;

– весовая функция суммирования.

Необходимо рассчитать весовую функцию . Необходимым условием оптимального суммирования является то, что в сумме для каждого отсчёта коэффициенты дают единицу:

, (666)

Другим условием является минимизация дисперсии ошибки:

, (666)

где – дисперсия ошибки;

Рассмотрим :

, (666)

Продифференцируем выражение (666) для среднеквадратической ошибки:

, (666)

Решив данное уравнение относительно , получаем выражение для весовой функции суммирования:

, (666)

Выражение для среднеквадратической ошибки (666) будет следующим:

, (666)

С учетом полученных результатов, выражение для оценки значения (666) примет следующий вид:

(666)

Полученные оценки позволяют восстановить наблюдаемое изображение оптимальным образом. В качестве меры оптимизации используется среднеквадратичная ошибка.

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.

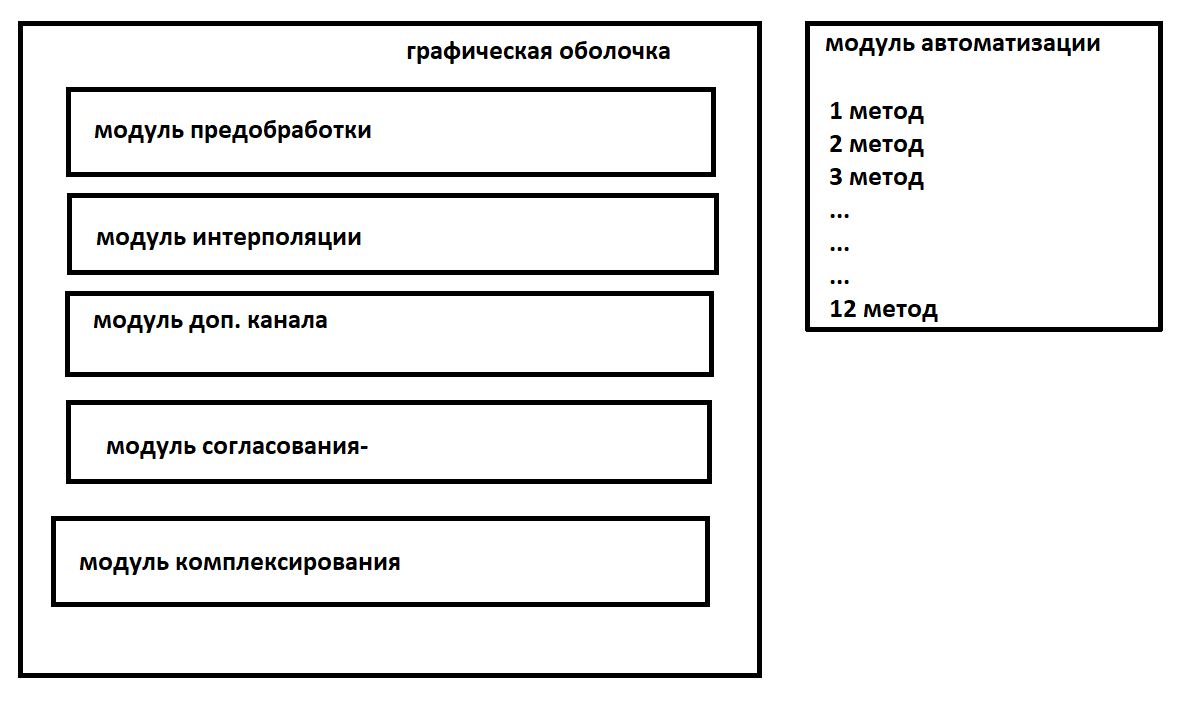


Рисунок 666 – Общая схема ПО

* 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.

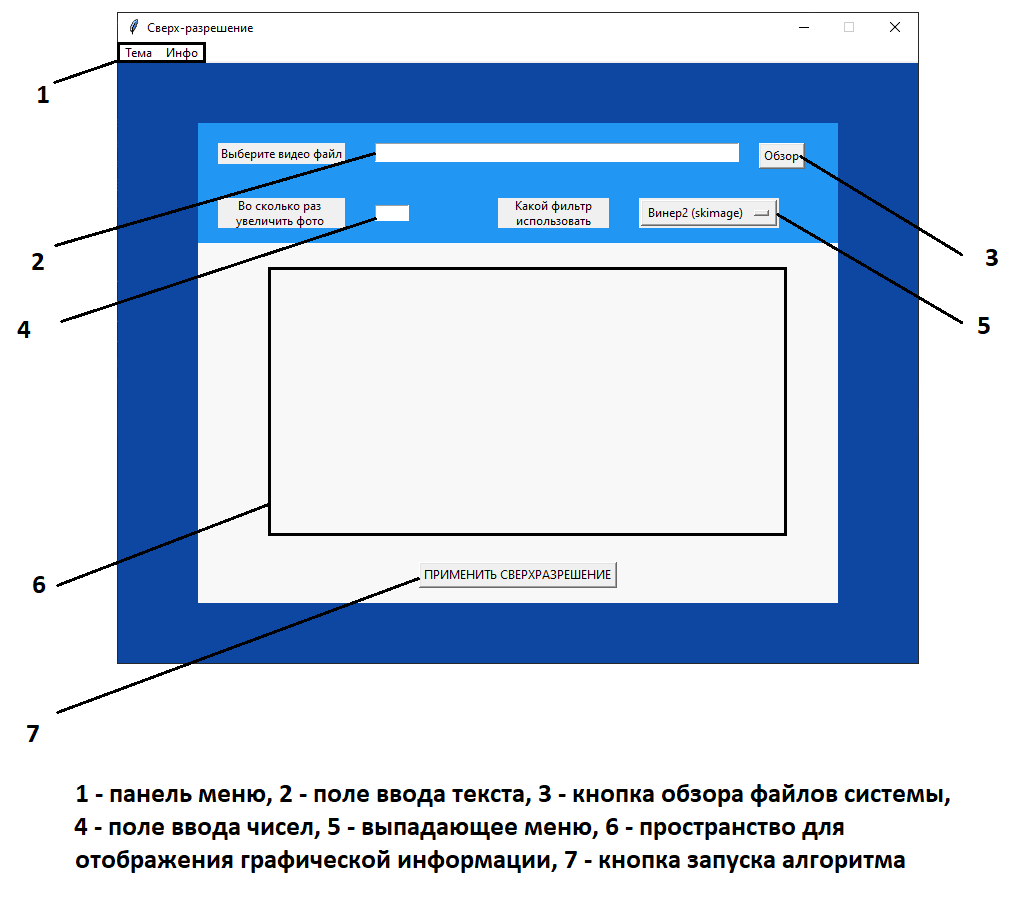


Рисунок 666 – Интерфейс разработанной программы

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.

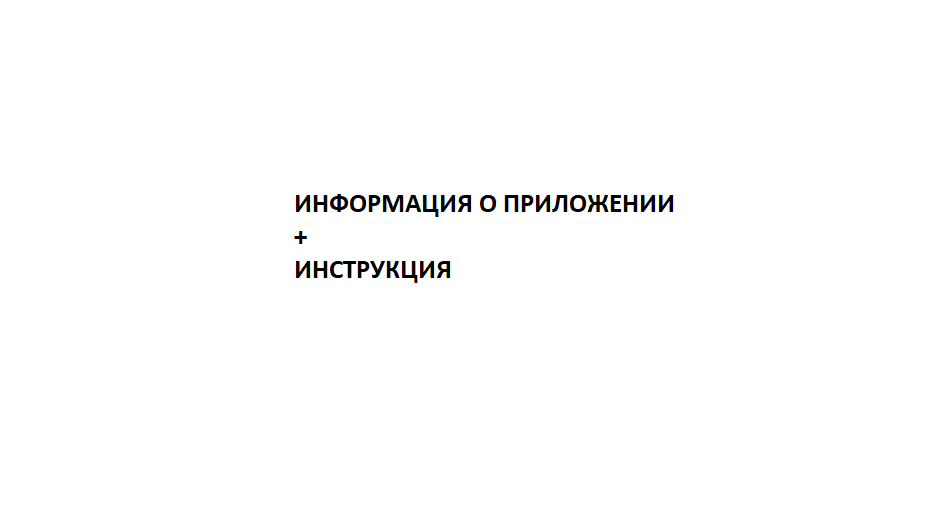


Рисунок 666 – Дополнительное окно с информацией о приложении

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:

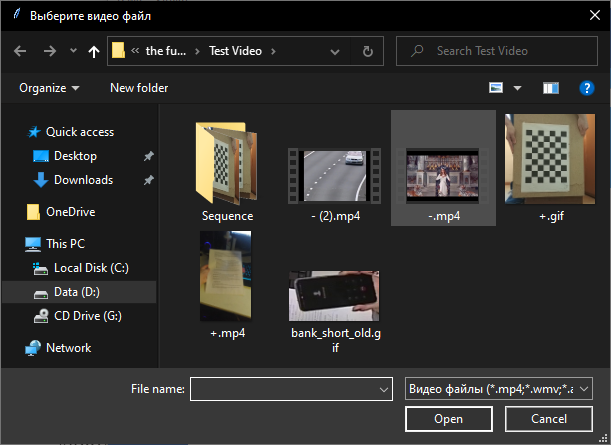


Рисунок 666 – Пример диалогового окна с видео файлами

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:

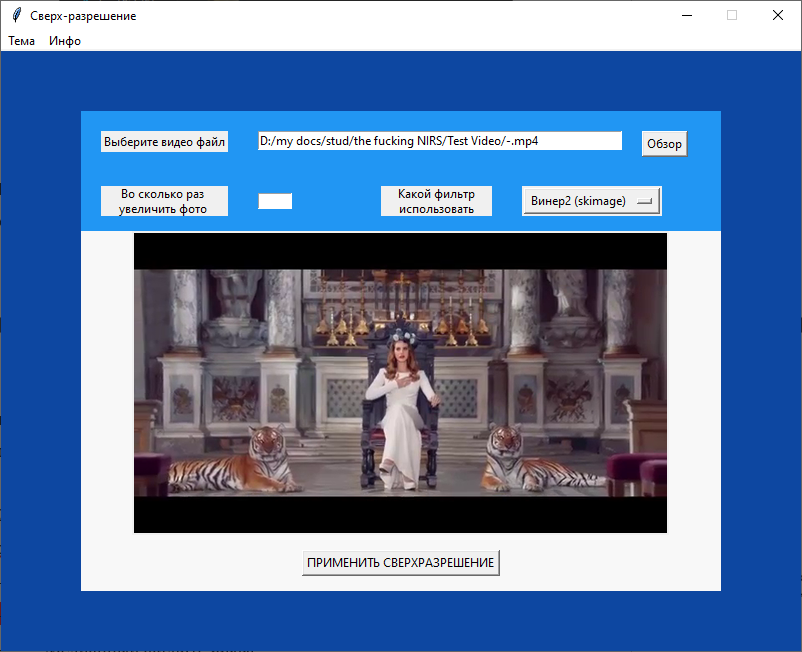


Рисунок 666 – Пример анимации

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:

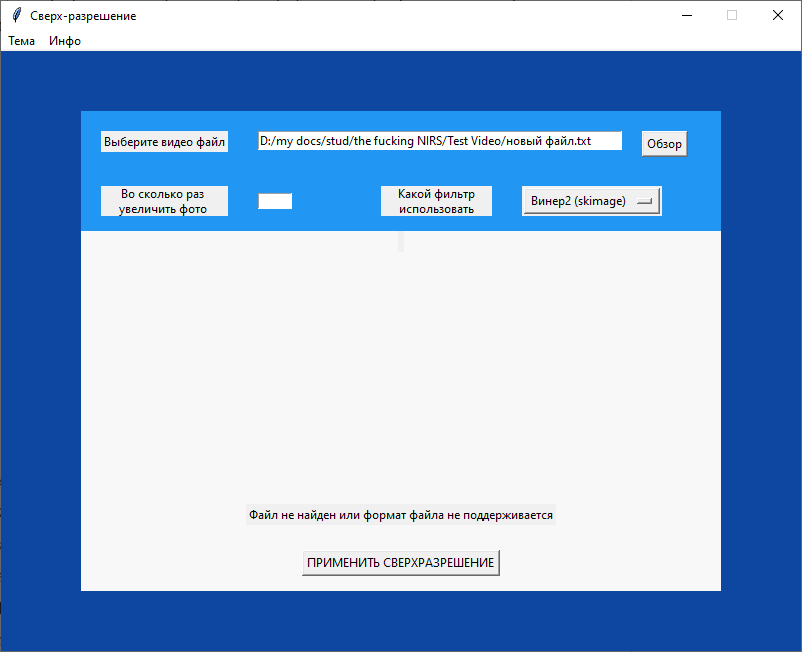


Рисунок 666 – Пример вывода ошибки

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:

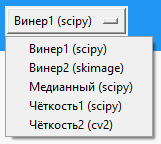


Рисунок 666 – Элементы выпадающего меню

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. Модуль автоматизации экспериментальных исследований

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

Для 12 различных методов согласования реализованы при помощи следующих программных библиотек: OpenCV, pystackreg, image\_registration, skimage.

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:

mse = 1/mn ∑\_(i=0)^(m-1)▒∑\_(j=0)^(n-1)▒〖|I(i,j)-K(i,j)|〗^2 (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.

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. Экспериментальное исследование

Было произведено три экспериментальных исследования.

Целью первого исследования являлось определение наиболее эффективного в терминах СКО алгоритма согласования серии изображений. Результат исследования позволил оптимизировать этап согласования в алгоритме комплексирования путём реализации метода, дающего наименьшую ошибку согласования.

Целью второго исследования являлось определение эффективности восстановления искусственно искаженного видео до исходного.

Целью третьего исследования являлось сравнение эффективности разработанного метода с другими методами сверхразрешения. Разработанный метод сравнивался с методом, основанным на усреднении по общей выборке и с методом, основанным на использовании искусственного интеллекта.

* 1. Типичные для криминалистической экспертизы видео данные

Для проведения экспериментов были использованы следующие последовательности кадров:

1. 3D модели геометрических фигур. Кадры сделаны в программе для 3D моделирования. На кадрах воспроизведено перемещение в пространстве одной фигуры на неподвижном фоне. Кадры не имеют пространственных искажений. Теоретический замысел такой последовательности – проверка работоспособности программного обеспечения без больших вычислительных затрат. Ожидаемый результат тестирования последовательности – получение за короткое время единственного изображения без видимых искажений;
2. шахматная доска, записанная на смартфон. Видеозапись производилась на неподвижную статически закреплённую камеру. Шахматной доске придавалось умеренное движение, фон оставался неизменным. Доска не выходила за пределы видимости, не трансформировалась и не искажалась. На камере присутствуют аппаратные искажения. Теоретический замысел такой последовательности – тривиальная проверка работоспособности разрабатываемого метода. Ожидаемый результат – получение кадра большей размерности без шумов и динамических искажений;
3. автомобиль, записанный на камеру наружного наблюдения. На кадрах неразборчив номер транспортного средства. Сам автомобиль не трансформируется, не искажается и сдвигается на несколько десятков пикселей за кадр, что можно считать интенсивным движением. Теоретический смысл – воспроизведение реальной ситуации, возникающей в случаях распознавания номеров автомобилей нарушителей правил дорожного движения. Ожидаемый результат – получение кадра с чётким читаемым номерным знаком транспортного средства;
4. экран телефона, зафиксированный камерой видеонаблюдения. Полная версия видеозаписи демонстрирует телефонный разговор сотрудника банка с другим субъектом, чья личность не установлена. Теоретический смысл – воспроизведение реальной ситуации нарушения банковской тайны. Видеозапись являлась доказательственной базой во время проведения расследования о нарушении банковской тайны. Экран телефона содержит имя второго субъекта, однако аппаратные искажения камеры делают текст не читаемым. Ожидаемый результат – получение кадра с читаемым текстом на экране телефона;
5. человеческое лицо, записанное на веб-камеру. Кадры представлены в низком разрешении, также кадры претерпевают умеренные искажения. Лицо умеренно передвигается в пространстве. Видео взято из общедоступного источника тестовых видеопоследовательностей. Теоретический смысл – воспроизведение реальной задачи распознавания лиц. Ожидаемый результат – получение кадра с более высоким разрешением с более выраженными чертами лица;
6. текст на бумажном носителе. На камеру смартфона были сделаны несколько кадров неподвижной сцены, однако сама камера перемещалась в пространстве, отчего все снимки получились разные. На кадрах изображен бумажный документ с неразборчивым текстом. Теоретический смысл – воспроизведение реальной задачи распознавания текста. Ожидаемый результат – получение кадра с более высоким разрешением с более чётким и читаемым текстом.

Все последующие исследования проводились по всем перечисленным последовательностям. В результатах исследований представлены усреднённые значения шести экспериментов.

* 1. Сравнительное исследование методов геометрического согласования

Было произведено экспериментальное исследование среднеквадратической ошибки (СКО) согласования в зависимости от выбора метода геометрического согласования.

Был исследован метод, основанный на использовании особых точек изображений. При этом были исследованы различные дескрипторы особых точек – SIFT, SURF, BRIEF, ORB.

Был исследован метод, основанный на пирамидальном подходе, при этом были исследованы различные модели искажения кадра – сдвиг, сдвиг и поворот, сдвиг-масштаб-поворот, аффинное преобразование, билинейное преобразование.

Был исследован метод, использующий пики взаимной корреляционной функции (ВКФ) двух изображений, предложенный в работе – как его классический вариант, так и его модификация, использующая «хи-квадрат» меру.

Также был исследован метод геометрического согласования изображений с использованием оптического потока.

Экспериментальное исследование было выполнено следующим образом – при помощи выбранного метода кадры последовательности геометрически согласовывались с первым изображением последовательности, после чего вычислялась среднеквадратическая ошибка согласования. Ошибка усреднялась по всему тестовому набору. Полученные результаты представлены на рисунке 666 и в таблице 666.

Таблица 2 – СКО согласования исследуемых методов

|  |  |
| --- | --- |
| Наименование метода | Среднее СКО согласования |
| 1. С использованием особых точек, дескриптор SIFT | 29,017 |
| 2. С использованием особых точек, дескриптор SURF | 29,264 |
| 3. С использованием особых точек, дескриптор BRIEF | 29,205 |
| 4. С использованием особых точек, дескриптор ORB | 28,841 |
| 5. Пирамидальный подход, сдвиг | 27,551 |
| 6. Пирамидальный подход, сдвиг и поворот | 23,506 |
| 7. Пирамидальный подход, сдвиг-масштаб-поворот | 23,465 |
| 8. Пирамидальный подход, аффинное преобразование | 23,155 |
| 9. Пирамидальный подход, билинейное преобразование | 23,272 |
| 10. С использованием пиков ВКФ | 27,879 |
| 11. С использованием пиков ВКФ, с χ2-мерой | 27,842 |
| 12. С использованием оптического потока | 29,419 |

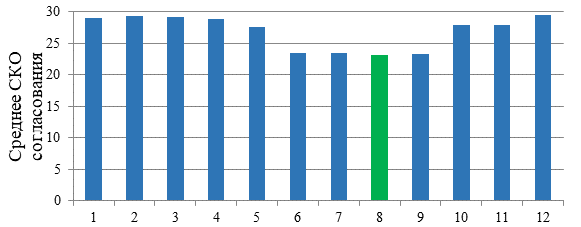


Рисунок 666 – Диаграмма усредненных по тестовому набору СКО для исследуемых методов. Номера столбцов диаграммы соответствуют номерам методов в таблице 666

Наилучшие результаты были получены при исследовании методов, основанных на пирамидальном подходе. Среди них наименьшее СКО было получено при использовании аффинного преобразования в качестве модели искажения кадра.

* 1. Исследование эффективности разработанного сверхразрешения

Для исследования эффективности было проведено два варианта экспериментов.

* + 1. Первый эксперимент

Эксперимент заключался в том, что исходная выборка принималась за эталон, то есть допускалось, что изображения, снятые на камеру, являются идеальными и не имеют каких-либо аппаратных искажений.

Для анализа работы программы было необходимо искусственно применить различные виды искажений и сравнить результат работы программы с эталонной выборкой.

В качестве искажений было принято решение использовать децимацию, динамически искажения и добавление аддитивных шумов. Применяемые программные библиотеки позволяют работать только с изображениями в оттенках серого.

Дальнейшие действия выполнялись согласно алгоритму комплексирования: учащена сетка пикселов обрабатываемых кадров, вычислена дисперсия ошибки интерполяции, кадры согласованы наилучшим алгоритмом – пирамидальным подходом с аффинным преобразованием.

На рисунке 666 можно видеть пример работы программы.

|  |  |  |
| --- | --- | --- |
| (а) | (б) | (в) |
| (г) | | |

Рисунок 666 – Пример полного цикла алгоритма. Один из исходных (а), один из искаженных (б), один из отфильтрованных и увеличенных (в) и несколько согласованных (г) кадров

После того, как вся серия кадров была согласована, происходило вычисление оценки исходного кадра по формуле (666).

В результате работы программы сгенерировалось полностью восстановленное изображение с разрешением, соответствующим исходным файлам. С результатом эксперимента можно ознакомиться на рисунке 666:

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

Рисунок 666 – Пример результата работы программы. Слева – модель искаженного кадра. Справа – результат работы алгоритма

Вычисление СКО двух изображений относительно друг друга дало следующий результат: ~0. Ошибка комплексирования близка к нулю.

* + 1. Второй эксперимент

Эксперимент заключался в том, что исходная выборка считалась искаженной. Задача эксперимента состояла в том, чтобы повысить визуальное качество видеокадров.

До применения алгоритма комплексирования необходимо попытаться восстановить качество изображений до исходного. Для этого использовался классический фильтр Винера, который даёт не идеальное, но оптимальное восстановление.

Дальнейшие действия выполнялись согласно алгоритму комплексирования: учащена сетка пикселов обрабатываемых кадров, вычислена дисперсия ошибки интерполяции, кадры согласованы наилучшим алгоритмом – пирамидальным подходом с аффинным преобразованием.

На рисунке 666 можно видеть пример работы программы.

|  |  |  |
| --- | --- | --- |
| (а) | (б) | (в) |
| (г) | | |

Рисунок 666 – Пример полного цикла алгоритма. Один из исходных (а), один из искаженных (б), один из отфильтрованных и увеличенных (в) и несколько согласованных (г) кадров

После того, как вся серия кадров была согласована, происходило вычисление оценки исходного кадра по формуле (666).

В результате работы программы сгенерировалось частично восстановленное изображение с увеличенным разрешением. С результатом эксперимента можно ознакомиться на рисунке 666:

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

Рисунок 666 – Пример результата работы программы. Слева – модель искаженного кадра. Справа – результат работы алгоритма

* 1. Сравнительное исследование методов сверхразрешения

Было произведено экспериментальное исследование среднеквадратической ошибки (СКО) сверхразрешения в зависимости от применяемого метода сверхразрешения.

Целью эксперимента было сравнение двух существующих методов получения сверхразрешения с разработанным новым методом.

Первый метод заключается в том, что согласованная серия кадров усредняется по всей выборке.

Второй метод основан на использовании искусственного интеллекта, принимающего в качестве входного параметра единственное изображение.

Эксперименты проводились на всех шести видеокадрах. Ниже приведены результаты экспериментов.

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