Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное   
учреждение высшего профессионального образования  
«Самарский государственный аэрокосмический университет   
имени академика С.П. Королева

(национальный исследовательский университет)»

(СГАУ)

Факультет информатики  
Кафедра технической кибернетики

**Зверев Алексей Николаевич**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРА**

**(МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)**

Тема: Разработка и исследование алгоритмов анализа устойчивости рекурсивных фильтров обработки изображений

Направление подготовки магистров: «010900.68 – Прикладные математика и физика»

магистерская программа: «Прикладные информационные технологии в бизнесе и управлении»

Научный руководитель

   д. т. н., профессор Фурсов В.А.

(ученая степень, ученое звание, фамилия, инициалы)

Магистрант

Зверев А.Н.

(ученая степень, ученое звание, фамилия, инициалы)

Самара – 2013 год

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего профессионального образования  
«Самарский государственный аэрокосмический университет   
имени академика С.П. Королева

(национальный исследовательский университет)»

(СГАУ)

Факультет информатики  
Кафедра технической кибернетики

|  |  |
| --- | --- |
|  | «УТВЕРЖДАЮ»  Заведующий кафедрой  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г. |
|  |  |

**ЗАДАНИЕ  
по подготовке магистерской диссертации**

студента группы 6226 М 308 Зверева Алексея Николаевича

1. Тема работы: Разработка и исследование алгоритмов анализа устойчивости рекурсивных фильтров обработки изображений

утверждена приказом по университету от «19» марта 2013 г. № 114-ст

1. Исходные данные к работе: Литература по методам цифровой обработки сигналов и изображений, исследованию устойчивости.
2. Перечень вопросов, подлежащих разработке: 1. Разработка алгоритма поиска и формирования тестовых фрагментов. 2. Разработка алгоритма идентификации восстанавливающего БИХ-фильтра. 3. Разработка алгоритма анализа устойчивости полученного БИХ-фильтра. 4. Разработка алгоритма стабилизации неустойчивого БИХ-фильтра.

График выполнения работы:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Этапы работы | % | Сроки выполнения по этапам | Итоги проверки | | |
| Отметка о вып. | Подпись магис-та | Подпись рук-ля |
| Анализ существующих методов восстановления изображений | 5 | 20.02.2013 | вып. |  |  |
| Определение общей схемы сквозной технологии | 15 | 01.03.2013 | вып. |  |  |
| Разработка алгоритма поиска фрагментов | 20 | 07.03.2013 | вып. |  |  |
| Разработка алгоритма формирования фрагментов | 35 | 22.03.2013 | вып. |  |  |
| Разработка алгоритма идентификации параметров БИХ-фильтра | 50 | 05.04.2013 | вып. |  |  |
| Разработка алгоритма анализа устойчивости | 80 | 06.05.2013 | вып. |  |  |
| Разработка алгоритма стабилизации | 90 | 17.05.2013 | вып. |  |  |
| Подготовка документа-ции по магистерской диссертации | 100 | 30.05.2013 | вып. |  |  |

1. Перечень графического материала (с точным указанием обязательных чертежей, плакатов, слайдов): название работы (слайд 1), цели и задачи работы (слайд 2), общая схема сквозной технологии (слайд 3), поиск и формирование тестовых фрагментов (слайд 4), математическое описание двумерного БИХ-фильтра (слайды 5-6), идентификация параметров восстанавливающего БИХ-фильтра (слайд 7), проверка устойчивости табличным методом (слайды 8-9), проверка устойчивости с использованием матриц Шура-Кона (слайды 10-11), стабилизация двумерного рекурсивного фильтра табличным методом (слайд 12), результаты (слайды 13-16).

Срок представления законченной работы: «6» июня 2013 г.

Дата выдачи задания: «15» января 2013 г.

Руководитель работы Фурсов В.А.

(подпись) (фамилия, инициалы)

Задание принял к исполнению

(дата)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

**РЕФЕРАТ**

**Выпускная квалификационная работа магистра:** 58c., 10 рисунков, 7 таблиц, 20 источников, одно приложение.

**Презентация:** 16 слайдов Microsoft PowerPoint.

ДВУМЕРНЫЙ РЕКУРСИВНЫЙ ФИЛЬТР, МАТРИЦЫ ШУРА-КОНА, ТАБЛИЦЫ ДЖУРИ, СКВОЗНАЯ ТЕХНОЛОГИЯ, АЛГОРИТМ ОЦЕНКИ И ОБЕСПЕЧЕНИЯ УСТОЙЧИВОСТИ

Объектом исследования являются двумерные рекурсивные фильтры второго порядка с бесконечной импульсной характеристикой (БИХ-фильтры).

Цель работы - разработка и исследование алгоритмов анализа и обеспечения устойчивости восстанавливающих БИХ-фильтров в рамках сквозной технологии адаптивного восстановления изображений.

В работе представлена сквозная технология адаптивного восстановления изображений, подробно рассмотрены все этапы этой технологии, реализованы алгоритм поиска и формирования тестовых фрагментов, алгоритм идентификации параметров восстанавливающего БИХ-фильтра, созданы и реализованы алгоритмы анализа и обеспечения устойчивости двумерного рекурсивного фильтра. Показана работоспособность разработанной технологии на примере обработанных и восстановленных изображений.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 7](#_Toc358194265)

[1 АНАЛИЗ ПРОБЛЕМ И ФОРМУЛИРОВКА ЗАДАЧ ИССЛЕДОВАНИЯ 9](#_Toc358194266)

[1.1 ПОСТАНОВКА ЗАДАЧИ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ 9](#_Toc358194267)

[1.2 ОБЗОР МЕТОДОВ СИНТЕЗА ВОССТАНАВЛИВАЮЩИХ ФИЛЬТРОВ И ПРОБЛЕМЫ ИХ ПОСТРОЕНИЯ ПРИ АПРИОРНОЙ НЕОПРЕДЕЛЕННОСТИ МОДЕЛЕЙ ПОМЕХ. 10](#_Toc358194268)

[1.3 КОНКРЕТИЗАЦИЯ ЗАДАЧ ИССЛЕДОВАНИЯ (В РАМКАХ АДАПТИВНОГО ПОДХОДА) 11](#_Toc358194269)

[2 ТЕХНОЛОГИЯ АДАПТИВНОГО ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ 13](#_Toc358194270)

[2.1 ОБЩАЯ СХЕМА СКВОЗНОЙ ТЕХНОЛОГИИ 13](#_Toc358194271)

[2.2 АЛГОРИТМ ПОИСКА И ФОРМИРОВАНИЯ ТЕСТОВЫХ ФРАГМЕНТОВ 14](#_Toc358194272)

[2.3 МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ ДВУМЕРНОГО РЕКУРСИВНОГО ЦИФРОВОГО ФИЛЬТРА ВТОРОГО ПОРЯДКА 16](#_Toc358194273)

[2.4 ИДЕНТИФИКАЦИЯ ПАРАМЕТРОВ ВОССТАНАВЛИВАЮЩЕГО БИХ-ФИЛЬТРА 19](#_Toc358194274)

[3 АНАЛИЗ И ОБЕСПЕЧЕНИЕ УСТОЙЧИВОСТИ РЕКУРСИВНЫХ ФИЛЬТРОВ 21](#_Toc358194275)

[3.1 АНАЛИЗ МЕТОДОВ ОЦЕНКИ УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ 21](#_Toc358194276)

[3.2 ПРОВЕРКА УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ ТАБЛИЧНЫМ МЕТОДОМ 27](#_Toc358194277)

[3.3 ПРОВЕРКА УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ С ИСПОЛЬЗОВАНИЕМ МАТРИЦ ШУРА-КОНА 31](#_Toc358194278)

[3.4 СТАБИЛИЗАЦИЯ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ ТАБЛИЧНЫМ МЕТОДОМ 36](#_Toc358194279)

[3.5 СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ОЦЕНКИ И ОБЕСПЕЧЕНИЯ УСТОЙЧИВОСТИ 37](#_Toc358194280)

[4 ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ 40](#_Toc358194281)

[4.1 РЕЗУЛЬТАТЫ ПРОВЕРКИ РАБОТЫ АЛГОРИТМА АВТОМАТИЗИРОВАННОГО ФОРМИРОВАНИЯ ТЕСТОВЫХ ФРАГМЕНТОВ 40](#_Toc358194282)

[4.2 ПРИМЕР РЕАЛИЗАЦИИ АЛГОРИТМА ИДЕНТИФИКАЦИИ БИХ-ФИЛЬТРА 41](#_Toc358194283)

[4.3 РЕЗУЛЬТАТЫ РЕАЛИЗАЦИИ СКВОЗНОЙ ТЕХНОЛОГИИ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ 43](#_Toc358194284)

[ЗАКЛЮЧЕНИЕ 46](#_Toc358194285)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 47](#_Toc358194286)

[ПРИЛОЖЕНИЕ А Листинг программного кода приложения 49](#_Toc358194287)

# ВВЕДЕНИЕ

В различных областях науки и техники существуют задачи, которые предусматривают цифровую обработку многомерных массивов данных. Это, например, анализ и обработка аэрофото- и космических снимков, геофизические исследования земной коры, компьютерная томография, анализ оптических, тепловых, рентгеновских, радиографических изображений в медицине, промышленной дефектоскопии, системы технического зрения и множество других важных научно-практических задач. Цифровые фильтры имеют преимущество по сравнению с аналоговым исполнением: они не требуют периодического контроля и калибровки, так как их работоспособность не зависит от дестабилизирующих факторов внешней среды, например, температуры. К тому же характеристики цифрового фильтра могут быть легко изменены программно. Поэтому они широко используются в телекоммуникациях, в приложениях адаптивной фильтрации, в цифровой обработке изображений и видеопоследовательностей.

Обработка многомерных сигналов имеет некоторые особенности, иногда весьма существенные, которые отличают ее от одномерной обработки. Во-первых, многомерная обработка оперирует со значительно большими объемами данных, что увеличивает затраты на их обработку, хранение и анализ. Во-вторых, математические методы описания многомерных систем не отличаются той завершенностью и законченностью, которая характерна для одномерных систем. В-третьих, многомерные системы обладают большим числом степеней свободы, в результате чего проектирование с одной стороны приобретает гибкость, а с другой стороны большую сложность, несвойственную одномерным системам. В общем случае одномерные методы обработки не обобщаются до многомерных, что приводит к необходимости разработки новых и модификации существующих алгоритмов.

Для обработки двумерных сигналов можно использовать как фильтры с бесконечной импульсной характеристикой (БИХ-фильтры), так и фильтры с конечной импульсной характеристикой (КИХ-фильтры). КИХ-фильтры всегда физически реализуемы и устойчивы. Однако при значительных искажениях требуется опорная область больших размеров, что приводит к существенному возрастанию вычислительной сложности. Кроме того, возможности улучшения качества изображения с помощью КИХ-фильтров ограничены.

С использованием БИХ-фильтра можно обеспечить компенсацию сильных искажений с использованием опорной области небольших размеров. Однако эти фильтры для некоторых форм опорной области физически нереализуемы, кроме того, они могут оказаться неустойчивыми. В связи с этим становится крайне важна задача проверки и обеспечения устойчивости БИХ-фильтров.

Несмотря на некоторые трудности реализации БИХ-фильтров, с точки зрения вычислительной сложности сквозной технологии оценки и восстановления, их использование, как правило, предпочтительнее. Использование модели БИХ-фильтра для решения задачи идентификации (особенно при малых размерах используемых для этого фрагментов изображений) выгоднее, даже если на этапе улучшения качества изображений, по каким-либо причинам, должен использоваться КИХ-фильтр.

# 1 АНАЛИЗ ПРОБЛЕМ И ФОРМУЛИРОВКА ЗАДАЧ ИССЛЕДОВАНИЯ

## 1.1 ПОСТАНОВКА ЗАДАЧИ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ

Известно, что изображение объекта, полученное с помощью оптического прибора, как правило, искажено вследствие дефокусировки, из-за влияния среды между объектом и прибором и в силу ряда других причин.

Восстановление изображений включает оценку параметров искажения и использование ее для коррекции исходных данных. Оно заключается в построении некоторого приближения исходного изображения . При этом точность приближения изображения функцией тем выше, чем больше известно об искажающем операторе и шуме [12].

Восстановление смазанных (размытых, смещенных, сдвинутых), а также зашумленных изображений является некорректной задачей и обычно описывается одномерным интегральным уравнением Фредгольма I рода типа свертки:

|  |  |
| --- | --- |
|  | (1.1) |

где – распределение интенсивности вдоль неискаженного (искомого исходного) изображения;

распределение интенсивности вдоль измеренного искаженного изображения (без шума) в функции прямоугольных координат (ось направлена вдоль смаза);

математически ядро интегрального уравнения, а физически функция рассеяния точки (импульсная характеристика), равная в случае равномерного и прямолинейного смаза

|  |  |
| --- | --- |
|  | (1.2) |

где величина смаза (сдвига) в пикселях.

Задача решения уравнения (1.2) является некорректной по причине нарушения не только первого и второго условий, но, главным образом, третьего условия корректности по Адамару [13]: даже очень малые относительные ошибки правой части (а также ФРТ и метода решения) могут приводить к настолько большим ошибкам, что численное решение не будет иметь практически ничего общего с точным. Кроме того, неустойчивость обратной задачи (существенное снижение точности обратного решения) может быть обусловлена и наличием шума, который на практике неизбежен.

## 1.2 ОБЗОР МЕТОДОВ СИНТЕЗА ВОССТАНАВЛИВАЮЩИХ ФИЛЬТРОВ И ПРОБЛЕМЫ ИХ ПОСТРОЕНИЯ ПРИ АПРИОРНОЙ НЕОПРЕДЕЛЕННОСТИ МОДЕЛЕЙ ПОМЕХ

Существует несколько методов решения интегральных уравнений (1.1). К наиболее распространенным из них относятся: инверсная фильтрация (метод преобразования Фурье), метод регуляризации Тихонова, метод параметрической фильтрации Винера, метод максимального правдоподобия Люси-Ричардсона (один из методов статистической регуляризации), итеративные методы (Фридмана, Ландвебера) и др. Особенность перечисленных методов заключается в том, что при их использовании необходимо иметь некоторую информацию об искажающей функции или энергетических спектрах шума и неискаженного изображения. Применение параметрического фильтра Винера позволяет эффективно восстанавливать искаженные изображения, однако использование константы в качестве оценки для отношения энергетических спектров не всегда приводит к удовлетворительному решению обратной задачи [14].

Анализ существующих методов показал, что в основе большинства из них лежит ПФ (или ДПФ), которое не дает в чистом виде устойчивого решения. Для получения устойчивого решения необходимо прибегать к различным фильтрам: регуляризующим, оптимальным или параметрическим.

Все представленные выше методы, как показывает практика и работа [14], обладают одним существенным недостатком – наличием эффекта Гиббса, эффекта ложных волн. Появление искажений типа "звоны" связано с ограниченной областью представления изображения и использованием ДПФ в методах реконструкции.

## 1.3 КОНКРЕТИЗАЦИЯ ЗАДАЧ ИССЛЕДОВАНИЯ (В РАМКАХ АДАПТИВНОГО ПОДХОДА)

Основными пунктами исследования в настоящей работе являются:

* Идентификация моделей систем формирования изображений по малым фрагмента.
* Формирование тестовых фрагментов изображений.
* Идентификация параметров рекурсивного фильтра.
* Анализ и обеспечение устойчивости рекурсивных фильтров.

Задача идентификации рассмотрена в узком смысле и состоит в оценивании параметров модели системы формирования изображений из априори заданного класса. Наиболее широко используются линейные модели с постоянными (по пространственным координатам) параметрами. Очевидны вычислительные преимущества, связанные с применением таких моделей. Если в действительности искажения оказываются пространственно-зависимыми, линейные модели с постоянными параметрами целесообразно применять на малых фрагментах изображений.

Формирование тестовых фрагментов изображений будет рассмотрено в рамках технологии адаптивного восстановления изображений. В традиционной схеме оценки характеристик модели сквозного видеоинформационного тракта на поверхности Земли создают мишенный комплекс, состоящий из тестовых фрагментов – мир, который затем регистрируется исследуемой системой. При этом на этапе оценивания могут использоваться либо выполненный на некотором носителе рисунок, либо исчерпывающее описание миры. Но часто задача ставится более "жестко". Требуется оценить характеристики модели искажений без использования специальной миры по фрагментам известных объектов на исследуемом изображении. Подходящими для такой цели являются, например, элементы дорожной и аэродромной разметки, фрагменты строительных конструкций, крыши промышленных зданий и сооружений на границе с отбрасываемой ими тенью и др. Актуальной становится задача поиска и формирования подобных фрагментов изображений.

После подготовительного этапа формирования фрагментов проводится идентификация параметров рекурсивного фильтра. Идентификация возможна как в классе КИХ-фильтров, так и в классе БИХ-фильтров. Несмотря на некоторые трудности в реализации, БИХ-фильтр дает значительное преимущество, так как моделью БИХ-фильтра невысокого порядка можно аппроксимировать характеристики искажений, для оценивания которых в классе КИХ-фильтров потребуется маска больших размеров. Поэтому в итоге при одинаковом качестве восстановления применение БИХ-фильтра может дать даже экономию вычислений.

Так как полученный фильтр может оказаться неустойчивым, важно обеспечить устойчивость путем стабилизации его коэффициентов. При использовании неустойчивого фильтра результаты его работы непредсказуемы и не представляют практической ценности.

# 2 ТЕХНОЛОГИЯ АДАПТИВНОГО ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ

## 2.1 ОБЩАЯ СХЕМА СКВОЗНОЙ ТЕХНОЛОГИИ

Поиск фрагментов-претендентов

Компьютерное ретуширование отобранных фрагментов

Идентификация параметров восстанавливающего БИХ-фильтра

Анализ устойчивости полученного фильтра

Стабилизация фильтра (если требуется)

Рисунок 2.1 – Общая схема технологии

В рамках настоящей работы предлагается рассмотреть технологию адаптивного восстановления изображений, общая схема которой представлена на рисунке 2.1.

Эта технология представляет собой готовое решение задачи восстановления изображения в условиях априорной неопределённости модели помех, а также при отсутствии тестовых изображений по умолчанию (мир). Подробный разбор каждого звена технологии представлен ниже.

## 2.2 АЛГОРИТМ ПОИСКА И ФОРМИРОВАНИЯ ТЕСТОВЫХ ФРАГМЕНТОВ

Схематично фрагмент со ступенчатой функцией яркости показан на рисунке 2.2. Здесь темные и светлые точки соответствуют отсчетам с низким и высоким уровнем яркости. Для отыскания таких фрагмент ищутся участки, имеющие резкие перепады яркости в заданном направлении, причем такие, что изменения яркости в пределах каждого из двух соседних уровней незначительны.

Поиск осуществляется следующим образом. Вначале для каждой точки изображения определяется перепад яркости:

|  |  |
| --- | --- |
|  | (2.1) |

где

отсчет яркости на фрагменте;

средние значения яркости, вычисленные в левой и правой половине прямоугольного участка размером (см. рисунок 2.2).

Далее на заданном (в процентах к общему числу) множестве точек изображения, являющихся центрами фрагментов, определяется минимальное значение величины перепада яркости (2.1) и сравнивается с заданным допустимым значением (. Если отобранное множество точек может использоваться для оценки разрешения, а величина принимается в качестве порогового значения, определяющего это множество фрагментов (претендентов).

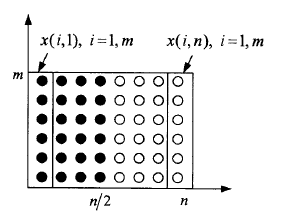


Рисунок 2.2 – Фрагмент изображения со ступенчатой функцией яркости

Для каждого отобранного таким образом фрагмента (претендента) вычисляются СКО яркости в каждой зоне (темной и светлой):

|  |  |
| --- | --- |
|  | (2.2) |

|  |  |
| --- | --- |
|  | (2.3) |

Затем из числа претендентов отбираются фрагменты, для которых полусумма СКО имеет допустимое значение. При этом участвуют одновременно фрагменты с перепадом яркости темный-светлый и светлый-темный либо только одного типа.

Заключительная операция первого этапа состоит в формировании тестовых неискаженных фрагментов. Она заключается в компьютерном ретушировании отобранных фрагментов. Предполагается, что наименьшим искажениям подвергаются участки фрагментов, наиболее удаленные от линии перепада яркостей. Поэтому процедура ретуширования сводится к замене значений яркости в каждой из половинок выбранного фрагмента их средними значениями, вычисленными по формулам:

где , параллельных линии, разделяющей светлую и темную зоны на уровне яркости (см. рисунок 2.2).

## 2.3 МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ ДВУМЕРНОГО РЕКУРСИВНОГО ЦИФРОВОГО ФИЛЬТРА ВТОРОГО ПОРЯДКА

В общем случае двумерный рекурсивный фильтр описывается разностным уравнением:

(2.4)

где – входной сигнал;

– выходной сигнал;

– коэффициенты нерекурсивной части фильтра;

—коэффициенты рекурсивной части фильтра.

В случае, если можно нормализовать коэффициенты *a* и *b,* разделив (1.1) на . Это позволяет принять без потери общности рассмотрения. Данный фильтр обладает свойством линейности – подчиняется принципу суперпозиции, а также он инвариантен к сдвигу – сдвиг входной последовательности приводит к соответствующему сдвигу выходной последовательности [1].

Двумерное z-преобразование от обеих частей уравнения (2.4) имеет вид:

|  |  |
| --- | --- |
|  | (2.5) |

Передаточная функция двумерного рекурсивного ЛИС-фильтра имеет вид:

|  |  |
| --- | --- |
|  | (2.6) |

Нетрудно заметить, что для вычисления значения конкретного выходного отсчета выходная маска должна накрывать только известные значения отсчетов (за исключением вычисляемого). На рисунке 2.3а приведен пример опорной маски 5×5, для которой требование рекурсивной вычислимости не выполняется (вычисляемый выходной отсчет обозначен кружком в центре опорной области).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| фильтр 5х5 | фильтр 5х5 I | фильтр 5х5 II | фильтр 5х5 III | фильтр 5х5 IV |
| а | б | в | г | д |
| Рисунок 2.3 – Исходная маска 5×5 – а и соответствующие ей маски и допустимые направления рекурсии для: 1-го квадранта – б; 2-го квадранта – в; 3-го квадранта – г; 4-го квадранта – д | | | | |

Известно [1], что импульсный отклик симметричного БИХ-фильтра с опорной областью на всей -плоскости можно разбить на четыре отдельных импульсных отклика, по одному на каждый квадрант. На рисунках 2.3, б – д приведены соответственно маски 1-го, 2-го, 3-го и 4-го квадрантов и направления рекурсии для них. С использованием указанного разбиения можно построить фильтры, соответствующие этим квадрантам и соединив их параллельно, получить общий импульсный отклик двумерного БИХ-фильтра. На рисунке 2.4 приведена общая схема параллельного соединения БИХ-фильтров одного квадранта, реализующая двумерный БИХ-фильтр на -плоскости.

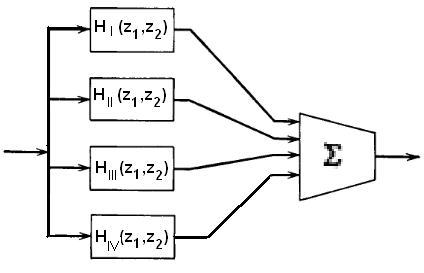


Рисунок 2.4 – Параллельное соединение 4-х фильтров одного квадранта, реализующее двумерный БИХ-фильтр на плоскости

## 2.4 ИДЕНТИФИКАЦИЯ ПАРАМЕТРОВ ВОССТАНАВЛИВАЮЩЕГО БИХ-ФИЛЬТРА

Задача идентификации модели, описываемой соотношением (2.4), заключается в оценке неизвестных коэффициентов по измерениям

Классическая постановка задачи такова: построить оценку вектора параметров по доступным для непосредственного наблюдения или сформированным каким-либо способом матрице и вектору уравнения типа линейной регрессии:

|  |  |
| --- | --- |
|  | (2.7) |

где – вектор ошибок.

Источниками ошибок являются погрешности измерений, ограничения на порядок модели и др.

Соотношение вида (2.7) формально легко выписывается для приведенной выше в виде соотношения (2.4) модели фильтра с бесконечной импульсной характеристикой. Если фрагмент (или изображение) содержит различных отсчетов , из них можно составить фигурирующий в (2.7) вектор размерности Соответствующая матрица , сформированная на том же фрагменте, имеет размерность Каждая строка этой матрицы состоит из отсчетов одной опорной области:

|  |  |
| --- | --- |
|  | (2.8) |

Соответствующий вектор искомых параметров имеет вид

|  |  |
| --- | --- |
|  | (2.9) |

Идентификация по малым фрагментам изображений порождает ряд принципиальных проблем. Имеют место следующие специфические особенности задачи идентификации по малым фрагментам изображений:

1. Всегда существует опасность того, что из-за попадания фрагмента изображения на участок фона задача идентификации окажется плохо обусловленной или даже вырожденной.
2. Условия предельных теорем теории вероятностей при малом числе наблюдений не выполняются и, как следствие, невозможно обосновать априорную вероятностную модель ошибок в исходных данных. Свойство устойчивости статистических характеристик шумов на фрагментах изображений может не проявляться даже в том случае, когда существует устойчивое распределение ошибок на множестве изображений данного класса.

Также основной проблемой является то, что при реализации указанной схемы идентификации по фрагментам, имеющим перепад функции яркости в одном направлении, может быть идентифицирована передаточная функция невысокого порядка. В то же время, формирование заданной функции яркости для фрагментов, содержащих объекты сложной геометрической формы, представляется не всегда возможным. Связано это обычно, как с недостатком априорной информации о деталях изображения, так и трудностями отыскания резких границ объектов сложной формы. В связи с этим матрицу  будем формировать из блочных матриц, каждая из которых формируется по простым фрагментам, соответствующим основным направлениям ориентации линии разрыва функции яркости в пространстве. При этом обеспечивается хорошая обусловленность задачи при идентификации параметров фильтра достаточно высокого порядка.

Полученная система линейных уравнений решается в данной работе с помощью метода сопряженных градиентов – одного из наиболее известных методов решения систем линейных уравнений. При этом используется встроенная функция системы Matlab bicgstab().

# 3 АНАЛИЗ И ОБЕСПЕЧЕНИЕ УСТОЙЧИВОСТИ РЕКУРСИВНЫХ ФИЛЬТРОВ

## 3.1 АНАЛИЗ МЕТОДОВ ОЦЕНКИ УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ

Любой переходный процесс в выходном сигнале, вызванный изменениями во входном множестве, должен быть ограниченным по своей протяженности, и установившееся поведение фильтра (т.е. отклик при больших значениях ) должно быть предсказуемым. Проблема устойчивости рекурсивных фильтров сводится к наложению ограничений на коэффициенты фильтра таким образом, чтобы при ограниченном входном сигнале выходной сигнал удовлетворял неравенству , то есть его импульсный отклик описывался неравенством:

|  |  |
| --- | --- |
|  | (3.1) |

где *P, Q –* положительные числа.  
 Данный критерий называется критерием ограниченного входа и ограниченного выхода (ОВОВ) [1].

Иными словами, необходимым и достаточным условием устойчивости указанного фильтра является абсолютная суммируемость его импульсного отклика.

Практическая применимость критерия (3.1) ограничена. При использовании его для проверки на устойчивость необходимо вычислять бесконечные суммы. Простое усечение суммы неприемлемо, поскольку усеченная сумма всегда будет ограниченной. Более того, требуется, чтобы импульсный отклик был известен. Однако алгоритмы синтеза фильтров обычно позволяют получить либо коэффициенты разностного уравнения, либо передаточную функцию фильтра. Поэтому желательно иметь возможность определить устойчивость фильтра непосредственно по коэффициентам или по передаточной функции.

Для двумерных цифровых фильтров первого порядка проблема устойчивости решается сравнительно легко, однако уже для фильтров второго порядка сложность задачи значительно возрастает. Рассмотрим основные положения и некоторые методы определения устойчивости двумерных БИХ-фильтров, делая акцент на фильтры второго порядка.

Рассмотрим теорему Хуанга [4]:

***Теорема 3.1*** Пусть является рекурсивным фильтром первого квадранта. Этот фильтр устойчив тогда и только тогда, когда удовлетворяет двум условиям:

1. (3.2)
2. (3.3)

То есть, если двумерная последовательность абсолютно суммируема, то ее z-преобразование аналитично на единичной поверхности Обратное утверждение так же справедливо: если z-преобразование последовательности аналитично на единичной поверхности, то последовательность является абсолютно суммируемой, а фильтр – устойчивым.

Для практически важного случая передаточных функций

эквивалентно требованию Это в свою очередь требует, чтобы единичная поверхность лежала в области сходимости передаточной функции.

Хотя устойчивость многомерного рекурсивного фильтра зависит от набора нулей полинома знаменателя иногда на устойчивость может влиять и полином числителя. Это происходит в том случае, когда на единичной биокружности имеются несущественные особенности второго рода, то есть на ней имеются такие точки, в которых одновременно числитель и знаменатель равны нулю. В этом случае нельзя будет сказать, устойчив фильтр или нет. Другими словами, можно подобрать такой фильтр с несущественными особенностями второго рода, который будет устойчив, однако не будет удовлетворять условиям теоремы 3.1. В настоящее время не имеется общих и прямых средств для определения того, является ли устойчивой передаточная функция, не имеющая полюсов вне единичной биокружности и обладающая несущественной особенностью второго рода на единичной биокружности. Таким образом, теорема 3.1 является лишь достаточным условием устойчивости для случаев нетривиального полинома в числителе. Она является необходимым условием устойчивости только в случае отсутствия на единичной биокружности несущественных особенностей второго рода.

Рассмотрим простой тест на устойчивость, который основывается на двух теоремах, представленных Бауэром в [5].

***Теорема 3.2*** Многомерный линейный инвариантный к сдвигу цифровой фильтр, имеющий вид:

|  |  |
| --- | --- |
|  | (3.4) |

где устойчив, если его коэффициенты удовлетворяют неравенству:

|  |  |
| --- | --- |
|  | (3.5) |

Считается, что передаточная функция фильтра не имеет несущественных особенностей второго рода.

Из этой теоремы следует, что для устойчивости уравнения (3.4) все коэффициенты рекурсивной части должны быть положительны, и их сумма не должна быть больше единицы. Достоинством этой теоремы является то, что она сформулирована для фильтров любой размерности и любого порядка, и такой метод может быть легко распространен на нелинейный случай. К недостаткам относится то, что все коэффициенты должны быть положительны. Так же теорема 3.2 не обладает достаточностью, и можно подобрать такой устойчивый фильтр, который не будет удовлетворять условию (3.5), но являться устойчивым.

***Теорема 3.3*** Многомерное линейное инвариантное к сдвигу рекурсивное уравнение вида:

|  |  |
| --- | --- |
|  | (3.6) |

устойчиво, если выполняется неравенство

|  |  |
| --- | --- |
|  | (3.7) |

В этом случае коэффициенты рекурсивной части могут принимать как положительные, так и отрицательные значения. При этом в уравнении отсутствуют члены, которые имеют задержку сразу в нескольких направлениях. Кроме того, в каждом из направлений присутствует только один рекурсивный член.

Хорошо видно, что тест на устойчивость фильтра, основанный на теоремах 3.2 и 3.3, можно выполнить, посчитав сумму рекурсивных коэффициентов. Хотя этот тест и налагает существенные ограничения на коэффициенты фильтра, но учитывая его простоту и то, что он позволяет тестировать фильтры любого порядка и любой размерности, он может с успехом использоваться в некоторых случаях.

Тест на устойчивость, использующий спектральное разложение на множители [6], заслуживает особого внимания. Этот алгоритм позволяет не только проверять устойчивость фильтров, но и стабилизировать их (сделать неустойчивый фильтр устойчивым). Ниже приведен краткий алгоритм получения коэффициентов устойчивого фильтра:

Шаг 1. Определить характеристики требуемого фильтра. Необходимо задать начальные условия: порядок фильтра, АЧХ фильтра, частоты среза, величину подавления сигнала, амплитуды пульсаций, ФЧХ фильтра, максимально допустимую величину отклонения от заданных параметров (критерий для шага 7).

Шаг 2. Получить начальный массив коэффициентов фильтра используя алгоритмы синтеза фильтров. Полученный массив коэффициентов может быть как устойчивым, так и неустойчивым.

Шаг 3. Определить функцию автокорреляции для коэффициентов

|  |  |
| --- | --- |
|  | (3.8) |

Шаг 4. Получить кепстр [1] из полученного массива, используя гомоморфное преобразование [7]:

Шаг 5. Проектировать кепстр на заданную опорную плоскость (первый квадрант):

Если массив коэффициентов не устойчив, то кепстр , полученный на шаге 4, будет не полностью находиться в той же опорной области, что и Ограничим кепстр только опорной областью, а отсчеты, лежащие вне опорной области, приравняем к нулю. Таким образом, получим новый кепстр

Шаг 6. Получить устойчивый массив коэффициентов фильтра, используя гомоморфное преобразование:

Шаг 7. Сравнить полученный устойчивый массив с исходным массивом .

Приведенный алгоритм является процедурой стабилизации начального фильтра с коэффициентами . Однако в процессе стабилизации коэффициенты фильтра изменяются, что сказывается на характеристиках. Поэтому процедуры оптимизации начального массива должны пошагово корректировать коэффициенты так, чтобы АЧХ фильтра менялась незначительно.

Итак, тест на устойчивость, использующий спектральное разложение на множители, обладает следующими преимуществами:

* не налагает никаких ограничений на коэффициенты и порядок фильтра;
* позволяет изменить коэффициенты первоначально неустойчивого фильтра для достижения устойчивости.

Недостатками же являются:

* излишняя громоздкость и выполнение за длительное время, так как используются дискретное преобразование Фурье и рекурсивная процедура оптимизации;
* большое количество априорной информации, требующейся для корректной работы алгоритма.

## 3.2 ПРОВЕРКА УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ ТАБЛИЧНЫМ МЕТОДОМ

Рассмотрим табличный метод определения устойчивости двумерного цифрового фильтра [8].

Перепишем передаточную функцию в виде:

|  |  |
| --- | --- |
|  | (3.9) |

Для определения устойчивости физически реализуемого фильтра (3.9) применим следующую теорему:

***Теорема 3.4*** Физически реализуемый рекурсивный фильтр с передаточной функцией (3.9), где – некоторый многочлен двух переменных , называется устойчивым в смысле ОВОВ, тогда и только тогда, когда многочлен не обращается в ноль ни при каких значениях из области

|  |  |
| --- | --- |
|  | (3.10) |

Устойчивость фильтра согласно теореме 3.1 достигается выполнением условий:

|  |  |
| --- | --- |
|  | (3.11)  (3.12) |

Условие б) теоремы 3.4 получается из условия б) теоремы 3.1, если принять . Таким образом, условие б) стало проще. Поскольку передаточная функция фильтра не должна иметь неопределенностей второго рода, то условие устойчивости выглядит так:

|  |  |
| --- | --- |
|  | (3.13) |

Представим полином

|  |  |
| --- | --- |
|  | (3.14) |

Полиномы , указанные в таблице 3.1, для строк, начиная со второй, определяются так:

|  |  |
| --- | --- |
|  | (3.15) |

Начальные полиномы определяются по формуле (3.14).

Таблица 3.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | … |  |  |
| 0 |  |  | … |  |  |
| 1 |  |  | … |  |  |
| … | … | … |  |  |  |
|  |  |  |  |  |  |

Для определения устойчивости рекурсивного фильтра воспользуемся следующей теоремой.

***Теорема 3.5*** Двумерный полином не имеет нулей внутри единичной биокружности, если комплексные полиномы из таблицы 3.1 удовлетворяют двум условиям:

|  |  |
| --- | --- |
|  | (3.16)  (3.17) |

Применим данную теорему для исследуемого в работе двумерного рекурсивного цифрового фильтра второго порядка. Результат приведен в таблице 3.2:

Таблица 3.2

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 |  |  |  |
| 1 |  |  |  |

где

Используя теорему 3.5 и таблицу 3.2, запишем условия устойчивости двумерного БИХ-фильтра второго порядка:

а.1) (3.18)

а.2) (3.19)

а.3) (3.20)

а.4) (3.21)

б.1) (3.22)

б.2) (3.23)

б.3) + (3.24) ;

б.4) (3.25)

Условия б.1-б.4 (формулы 3.22-3.25) представляют собой условия устойчивости для одномерных фильтров.

Таким образом, проверка на устойчивость двумерного цифрового фильтра второго порядка по сравнению с двумерным цифровым фильтром первого порядка значительно сложнее. Так как, во-первых, количество условий увеличилось в два раза, во-вторых, при проверке двумерного фильтра второго порядка возникает необходимость тестировать одномерные полиномы четвертого порядка, тогда как при проверке двумерного фильтра первого порядка – одномерные полиномы первого порядка.

## 3.3 ПРОВЕРКА УСТОЙЧИВОСТИ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ С ИСПОЛЬЗОВАНИЕМ МАТРИЦ ШУРА-КОНА

Представим передаточную функцию двумерного БИХ-фильтра, используя разложение по положительным степеням z:

|  |  |
| --- | --- |
|  | (3.26) |

Используя теорему 3.4, рассмотрим еще один метод определения устойчивости [9]. Выполнение условия (3.12) можно проверить непосредственно, так как многочлен является многочленом одной переменной. Существует несколько способов, позволяющих выяснить, лежат ли все нули интересующего нас многочлена вне единичного круга (приложение 1). Эти способы используют иннорные определители, табличную форму или симметрические матрицы Шура-Кона. Последняя матрица должна быть отрицательно определенной (т.е. все ее главные миноры нечетного порядка должны быть отрицательными, а четного порядка - положительными) тогда и только тогда, когда у многочлена все нули попадают в область как этого требует условие (3.12). Если заменить многочлен одной переменной на обратный ему, т.е. на то условие (3.12) сведется к необходимому и достаточному условию того, что все корни уравнения лежат внутри единичного круга. В результате получим условие устойчивости одномерного рекурсивного цифрового фильтра с соответствующим характеристическим многочленом.

Для проверки выполнения условий (3.11) можно применить критерий Шура-Кона. Рассмотрим в качестве параметра, тогда является многочленом переменной , коэффициентами которого служат функции от параметра В силу устойчивости многочлен не может обращаться в ноль внутри области это значит, что соответствующая матрица Шура-Кона для комплексных коэффициентов должна быть отрицательно-определенной. В этом случае элементами матрицы Шура-Кона служат многочлены переменной

Проверка отрицательной определенности матрицы Шура-Кона требует вычисления ее главных миноров, которые должны иметь соответствующие знаки. Эти миноры представляют собой многочлены переменных Кроме того, они вещественны, поскольку матрица Шура-Кона эрмитова.

Отсюда видно, что условие (3.11) эквивалентно требованию знакоопределенности в области некоторого множества функций, принимающих вещественные значения и являющихся многочленами Принимая во внимание, что в области имеет место равенство   получим, что эти функции для области имеют вид:

|  |  |
| --- | --- |
|  | (3.27) |

Для завершения проверки выполнения (3.11) по-прежнему необходим критерий положительной определенности многочлена (3.27) для всех из множества Можно сказать, что функция положительна на множестве тогда и только тогда, когда (или функция положительно определена в одной произвольной точке множества , а нулей функции лежат внутри области . Тогда нулей функции лежат вне единичного круга. Таким образом, на множестве у нее нет нулей [10]. Следовательно, функция положительна и значит, условие устойчивости выполнено.

Для упрощения проверки отрицательной определенности матрицы Шура-Кона, можно заменить многочлен рассматриваемый как многочлен переменной обратным к нему аналогично тому, как это делается при проверке выполнения условия (3.12). Так что условие отрицательной определенности превращается в условие положительной определенности. Эрмитова [11] матрица где

|  |  |
| --- | --- |
|  | (3.28) |

является симметрической матрицей Шура-Кона для многочлена

|  |  |
| --- | --- |
|  | (3.29) |

Для проверки условия (3.11) заменим многочлен на записанный как многочлен переменной :

|  |  |
| --- | --- |
|  | (3.30) |

Коэффициенты для формулы 3.11 теперь нужно заменить на

Условия (3.11) и (3.12) можно заменить условиями, в которых переменные поменялись ролями. Подобная замена может оказаться выгодной с вычислительной точки зрения.

Определим условия устойчивости для исследуемого фильтра второго порядка. Многочлен имеет вид:

|  |  |
| --- | --- |
|  | (3.31) |

Условие (3.12) в этом случае преобразуется к следующему виду:

|  |  |
| --- | --- |
|  | (3.32) |

Матрица размером полученная из выражения (3.28), имеет вид:

|  |  |
| --- | --- |
|  | (3.33) |

Из отрицательной определенности матрицы следуют первые два условия устойчивости рекурсивного фильтра второго порядка:

|  |  |
| --- | --- |
| 1. . | (3.34)  (3.35) |

Для проверки выполнения условия (3.11) запишем уравнение для

|  |  |
| --- | --- |
|  | (3.36) |

где

|  |  |
| --- | --- |
|  | (3.37) |

Матрица Шура-Кона имеет вид:

|  |  |
| --- | --- |
|  | (3.38) |

Матрица отрицательно определена если:

|  |  |
| --- | --- |
|  | (3.39)  (3.40) |

Выражения (3.39) и (3.40) являются условиями устойчивости рекурсивного фильтра второго порядка.

Расписав в явном виде выражение (3.39) и сделав соответствующие преобразования, получим неравенство, в левой части которого стоит симметричный многочлен второй степени:

|  |  |
| --- | --- |
|  | (3.41) |

где

|  |  |
| --- | --- |
|  | (3.42) |

Выражение (3.40) также сводится к неравенству, в левой части которого находится симметричный многочлен четвертой степени:

|  |  |
| --- | --- |
| *,* | (3.43) |

где

|  |  |
| --- | --- |
|  | (3.44) |

и

|  |  |
| --- | --- |
|  | (3.45) |

а коэффициенты в (3.44) подставляются из (3.42).

Условия (3.41) и (3.43) разбиваются на четыре следующих условия:

|  |  |
| --- | --- |
|  | (3.46)  (3.47)  (3.48)  (3.49) |

Эти условия в дополнении выражениями (3.34) и (3.35) есть полный набор условий устойчивости двумерного рекурсивного цифрового фильтра второго порядка в смысле ОВОВ.

Преимущества метода проверки на устойчивость с использованием матрицы Шура-Кона:

* применение теории матриц позволяет вычислять условия устойчивости быстрее, чем в методе, использующем спектральное разложение на множители, который требует вычисления как минимум четырех двумерных преобразований Фурье;
* не налагаются никакие ограничения на коэффициенты и порядок фильтра;
* задача тестирования двумерной передаточной функции сводится к тестированию одномерных функций.

В качестве недостатка стоит отметить, что метод не дает возможности стабилизации коэффициентов фильтра.

## 3.4 СТАБИЛИЗАЦИЯ ДВУМЕРНЫХ РЕКУРСИВНЫХ ЦИФРОВЫХ ФИЛЬТРОВ ТАБЛИЧНЫМ МЕТОДОМ

Тесты на устойчивость в результате выполнения выдают ответ: либо "фильтр устойчив", либо "фильтр неустойчив". Для разработчика фильтра такое положение дел отнюдь не всегда удобно, так как возникает задача изменения коэффициентов фильтра для достижения устойчивости при сохранении заданных в условиях синтеза параметров, то есть нужно стабилизировать полученный в результате синтеза фильтр. Кроме того, для обеспечения хороших частотных характеристик фильтра необходимо выбирать коэффициенты не на грани устойчивости, а в некотором отдалении, то есть иметь "запас" устойчивости.

Табличный метод можно представить в виде:

|  |  |
| --- | --- |
|  | (3.50) |

где

вещественная константа;

набор некоторых функций.

Например, для табличного метода условия "а" сразу имеют вид (3.50), где условия "б" так же можно свести к виду (3.50), используя условия устойчивости для одномерных цифровых фильтров. Рассмотрим -ю функцию из выражения (3.50). Она может принимать вещественные значения .

Рисунок 3.1

## 3.5 СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ОЦЕНКИ И ОБЕСПЕЧЕНИЯ УСТОЙЧИВОСТИ

Проанализируем условия устойчивости для двумерных фильтров -го порядка, полученные методом, использующим симметрические матрицы Шура-Кона, и табличным методом. За основу для сравнения возьмем три величины: количество условий, необходимых для тестирования фильтра (Т), количество условий, представляющих вид комплексных полиномов (П) и наибольшую степень комплексного полинома, который необходимо проверить на распределение нулей (C).

Общее количество условий, предоставляемых табличным методом, всегда в раз больше, чем количество условий устойчивости, полученных в результате применения метода Шура-Кона, но, несмотря на это, для фильтра второго порядка величины Т, П, и С будут принимать значения 6, 2 и 8 – для метода Шура-Кона; 8,4 и 4 – для табличного метода. Поэтому в случае фильтра второго порядка предпочтительнее использовать табличный метод, так как при его реализации таблица для проверки распределения корней комплексного полинома должна строиться для полинома четвертой степени, в отличие от метода Шура-Кона, при котором следует строить эту таблицу для полинома восьмой степени. В таблице 3.3 показаны изменение количества условий для проверки устойчивости и максимальная степень тестируемых полиномов в зависимости от порядка двумерного рекурсивного фильтра.

Таблица 3.3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Порядок фильтра (N) | Общее количество условий (Т) | | Количество полиномиальных условий (П) | | Наибольшая степень многочлена (С) | |
| Метод Шура-Кона | Табличный метод | Метод Шура-Кона | Табличный метод | Метод Шура-Кона | Табличный метод |
| *1* | *3* | *4* | *1* | *2* | *2* | *1* |
| *2* | *6* | *8* | *2* | *4* | *8* | *4* |
| *3* | *9* | *12* | *3* | *6* | *18* | *12* |
| *4* | *12* | *16* | *4* | *8* | *32* | *32* |
| *5* | *15* | *20* | *5* | *10* | *50* | *80* |
| *6* | *18* | *24* | *6* | *12* | *72* | *192* |
| *…* | *…* | *…* | *…* | *…* | *…* | *…* |
| *N* | *3N* | *4N* | *N* | *2N* | *2N2* | *2N-1N* |

Общее количество условий устойчивости при использовании табличного метода всегда больше, чем при применении метода с симметричными матрицами. То же самое можно сказать и об условиях, представляющих собой проверку распределения нулей комплексного многочлена относительно единичной окружности. Но решающим фактором при выборе метода реализации в данном случае является максимальный порядок многочлена, который необходимо тестировать. Из таблицы видно, что для фильтров до третьего порядка включительно, выгоднее реализовывать тест на основе табличного метода. Для фильтров, начиная с четвертого порядка, предпочтительнее использовать метод Шура-Кона. Для фильтров высоких порядков тесты, использующие алгебраические методы, становятся труднореализуемыми.

# 4 ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

## 4.1 РЕЗУЛЬТАТЫ ПРОВЕРКИ РАБОТЫ АЛГОРИТМА АВТОМАТИЗИРОВАННОГО ФОРМИРОВАНИЯ ТЕСТОВЫХ ФРАГМЕНТОВ

На рисунке 4.1, а, б приведены изображения участка автодороги с разметкой, которые использовались для оценки эффективности технологии. Исходное изображение (см. рисунок 4.1, а) подверглось смазыванию. Выделенные на расфокусированном изображении в соответствии с описанной в разделе 2 методикой тестовые фрагменты показаны на рисунке 4.1, б прямоугольниками.

a b

|  |  |
| --- | --- |
| C:\Users\Алексей\Pictures\Test2.tif | C:\Users\Алексей\Pictures\figure.tif |

Рисунок 4.1 – Тестовые изображения: исходное (a), смазанное с выделенными фрагментами (b)

Для идентификации параметров фильтров одного квадранта использовались тестовые фрагменты малых размеров, показанные на рисунках 4.2, а – h, с изменением функции яркости в одном из 8-ми направлений. Соответствующие им «неискажённые» фрагменты, сформированные путём компьютерного ретуширования, показаны на рисунках 4.2, i – p. В соответствии с количеством пар тестовых фрагментов было сформировано 8 матричных блоков размерности 24×17, из которых затем составлена матрица *X* размером 192×17.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| fr_hor_1 | fr_hor_2 | fr_ver_1 | fr_ver_2 | fr_L45_1 | fr_L45_2 | fr_R45_1 | fr_R45_2 |
| a | b | c | d | e | f | g | h |
| fr_hor_1_ret | fr_hor_2_ret | fr_ver_1_ret | fr_ver_2_ret | fr_L45_1_ret | fr_L45_2_ret | fr_R45_1_ret | fr_R45_2_ret |
| I | j | k | l | m | n | o | p |
| Рисунок 4.2 – Тестовые фрагменты: (a – h) – на исходном изображении,  (i – p) – те же фрагменты после компьютерного ретуширования | | | | | | | |

## 4.2 ПРИМЕР РЕАЛИЗАЦИИ АЛГОРИТМА ИДЕНТИФИКАЦИИ БИХ-ФИЛЬТРА

На рисунках 4.2, а – в, приведены соответственно исходное, искаженное и восстановленное изображения. Последнее получено суммированием результатов обработки фильтрами, построенными путем идентификации параметров четырех БИХ-фильтров одного квадранта. Для восстановления использовалось параллельное соединение простых БИХ-фильтров одного квадранта, опорные области которых показаны на рисунке 2.1.

|  |  |  |
| --- | --- | --- |
| C:\Users\Алексей\Pictures\Mirra0.tif | C:\Users\Алексей\Pictures\Mirra2.tif | C:\Users\Алексей\Pictures\Mirra1.tif |
| а | б | в |
| Рисунок 4.2 – Тестовые изображения: а – исходное изображение, б – искаженное изображение, в – восстановленное изображение | | |

При идентификации параметров восстанавливающего фильтра решается задача нахождения приближенного решения системы (формула 2.7). Это делается с помощью метода бисопряженных градиентов и реализующей его Matlab-функции bicgstab.

Построим график зависимости относительной нормы невязки от номера итераций (рисунок 4.3) с учетом особенности функции bicgstab: массив номеров итераций формируем с шагом, равным 0.5.

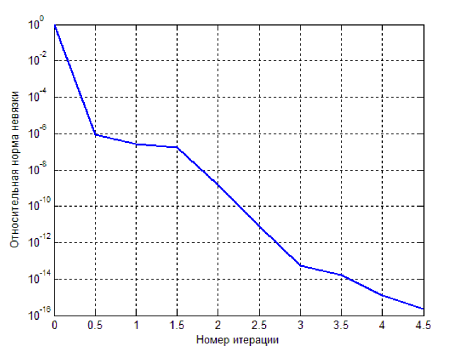


Рисунок 4.3 – Зависимость относительной нормы невязки от номера итерации для устойчивого итерационного метода бисопряженных градиентов

## 4.3 РЕЗУЛЬТАТЫ РЕАЛИЗАЦИИ СКВОЗНОЙ ТЕХНОЛОГИИ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ

На рисунках 4.4, а, б приведен другой пример применения описанной технологии к обработке черно-белых аэрокосмических снимков. Как и в предыдущем случае использовались четыре параллельно соединенных БИХ-фильтра одного квадранта с опорными областями, показанными на рисунках 2.1, б – д. Устойчивость всех четырех полученных фильтров одного квадранта гарантируется алгоритмом проверки устойчивости и стабилизации коэффициентов фильтра.

|  |  |
| --- | --- |
| s01small_sp5_t.bmp | s01small_r (восстановленное) |
| а | б |
| Рисунок 4.4 – Спутниковые черно-белые изображения: а – исходное, б – восстановленное | |

В таблице 4.1 представлены полученные коэффициенты фильтров четырех опорных квадрантов после прохождения алгоритма идентификации.

Таблица 4.1

|  |  |  |  |
| --- | --- | --- | --- |
| I квадрант | II квадрант | III квадрант | IV квадрант |
| 0,0587 | 0,0581 | 0,0590 | 0,0582 |
| 0,0586 | 0,0588 | 0,0588 | 0,0580 |
| 0,0586 | 0,0590 | 0,0581 | 0,0573 |
| 0,0588 | 0,0578 | 0,0586 | 0,0584 |
| 0,0588 | 0,0584 | 0,0584 | 0,0582 |
| 0,0587 | 0,0587 | 0,0578 | 0,0575 |
| 0,0589 | 0,0572 | 0,0581 | 0,0582 |
| 0,0588 | 0,0578 | 0,0578 | 0,0580 |
| 0,0587 | 0,0581 | 0,0572 | 0,0574 |
| 0,0589 | 0,0584 | 0,0598 | 0,0592 |
| 0,0587 | 0,0598 | 0,0584 | 0,0579 |
| 0,0585 | 0,0575 | 0,0602 | 0,0565 |
| 0,0593 | 0,0589 | 0,0589 | 0,0599 |
| 0,0591 | 0,0603 | 0,0575 | 0,0585 |
| 0,0590 | 0,0565 | 0,0593 | 0,0571 |
| 0,0596 | 0,0579 | 0,0579 | 0,0591 |
| 0,0594 | 0,0593 | 0,0565 | 0,0578 |

После анализа устойчивости полученных БИХ-фильтров оказалось, что фильтр III квадранта неустойчив. Итоговая таблица 4.2 коэффициентов устойчивых фильтров после процедуры стабилизации представлена ниже.

Таблица 4.2

|  |  |  |  |
| --- | --- | --- | --- |
| I квадрант | II квадрант | III квадрант | IV квадрант |
| 0,0587 | 0,0581 | 0,0590 | 0,0582 |
| 0,0586 | 0,0588 | 0,0588 | 0,0580 |
| 0,0586 | 0,0590 | 0,0584 | 0,0573 |
| 0,0588 | 0,0578 | 0,0586 | 0,0584 |
| 0,0588 | 0,0584 | 0,0584 | 0,0582 |
| 0,0587 | 0,0587 | 0,0578 | 0,0575 |
| 0,0589 | 0,0572 | 0,0581 | 0,0582 |
| 0,0588 | 0,0578 | 0,0578 | 0,0580 |
| 0,0587 | 0,0581 | 0,0572 | 0,0574 |
| 0,0589 | 0,0584 | 0,0598 | 0,0592 |
| 0,0587 | 0,0598 | 0,0584 | 0,0579 |
| 0,0585 | 0,0575 | 0,0602 | 0,0565 |
| 0,0593 | 0,0589 | 0,0582 | 0,0599 |
| 0,0591 | 0,0603 | 0,0575 | 0,0585 |
| 0,0590 | 0,0565 | 0,0593 | 0,0571 |
| 0,0596 | 0,0579 | 0,0579 | 0,0591 |
| 0,0594 | 0,0593 | 0,0565 | 0,0578 |

# ЗАКЛЮЧЕНИЕ

В данной работе получены следующие основные результаты.

1. Исследованы существующие методы восстановления изображений.
2. Исследованы существующие методы анализа и обеспечения устойчивости двумерных рекурсивных БИХ-фильтров.
3. Реализован алгоритм поиска и формирования тестовых фрагментов.
4. Реализован алгоритм идентификации параметров восстанавливающего БИХ-фильтра.
5. Разработан и реализован алгоритм анализа устойчивости двумерного рекурсивного фильтра второго порядка.
6. Разработан и реализован алгоритм обеспечения устойчивости двумерного рекурсивного фильтра второго порядка.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Даджион Д., Мерсеро Р. Цифровая обработка многомерных сигналов. М.: Мир, 1988. 488 с.
2. Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. М.: МИР, 1978. 848 с.
3. Сергеев В.В., Фурсов В.А., Парфенов С.И. Оценка разрешающей способности видеотракта по фрагментам регистрируемых изображений // Автометрия / СО РАН. 2001. №5. С. 25-36.
4. Huang T.S. Stability of 2-D recursive filters // IEEE Trans. Audio Electroacoust. 1972. No 6. P. 158-163.
5. Bauer P. A simple stability criterion for nonlinear multi-dimensional (M-D) direct form digital filters // ISCAS'89. IEEE.1989.V.2, P.1091-1094.
6. Ekstrom M., Woods J. Two dimensional spectral factorization with applications in recursive digital filtering // IEEE Transactions on Acoustics, Speech and Signal Processing. April 1976. No 2. P.115-128.
7. Оппенгейм А., Шафер Р. Цифровая обработка сигналов. М.: Связь, 1979. 416 с.
8. Yang X., Unbehauen R. New stability test algorithm for two-dimensional digital filters // IEEE Transaction on Circuits and Systems-I: Fundamental Theory and Applications. 1998. No.7. P.739-741.
9. O'Connor B.T., Huang Th.S. Stability of general two-dimensional recursive filters // IEEE Trans. Acoustics, Speech and Signal Processing. 1978. ASSP-26, No.6. P.550-560.
10. Джури Э. Инноры и устойчивость динамических систем. М.: Наука, 1978. 299 с.
11. Воеводин В.В. Линейная алгебра. М.: Наука, 1980. 400 с.
12. Гонсалес Р., Вудс Р., Эддинс С. Цифровая обработка изображений в среде MATLAB. M.: Техносфера, 2006. 616 с.
13. Петров Ю.П., Сизиков В.С. Корректные, некорректные и промежуточные задачи с приложениями: Учебное пособие для вузов. СПб.: Политехника, 2003. 261 с.
14. Гонсалес Р., Вудс Р. Цифровая обработка изображений. М.: Техносфера, 2005. 1072 с.
15. Дроздов М.А, Зимин Д.И, Скуратов С.А, Попов С.Б, Фурсов В.А. Технология определения восстанавливающих фильтров и обработки больших изображений // Компьютерная оптика. 2003. № 25. С. 17-20.
16. Фурсов В.А. Идентификация моделей систем формирования изображений в классе фильтров с бесконечной импульсной характеристикой при малом числе наблюдений // Компьютерная оптика. 1998. №18. С. 25-28.
17. Фурсов В.А. Идентификация моделей систем формирования изображений по малому числу наблюдений. Самара: Изд-во Самарского гос. аэрокосмического ун-та, 1998. 218 с.
18. Джури Э. Импульсные системы автоматического регулирования. М.: Наука, 1963. 455 с.
19. Прэтт У. Цифровая обработка изображений. М.: Мир, 1982. Т. 1. 312 с.
20. Хэмминг Р.В. Цифровые фильтры. М.: Мир, 1980. 224 с.

# ПРИЛОЖЕНИЕ А

Листинг программного кода приложения

В данном приложении приведен листинг основного кода реализованных алгоритмов поиска и формирования фрагментов, идентификации параметров восстанавливающего фильтра, анализа устойчивости и стабилизации.

%ПОИСК И ФОРМИРОВАНИЕ ФРАГМЕНТОВ

size(I)

blur\_size=20;

blur\_I=[I zeros(column\_size,blur\_size)];

size(blur\_I)

imshow(blur\_I)

for i=1:column\_size

for j=1:row\_size

pixel\_sum=0;

for k=j:j+blur\_size

pixel\_sum=pixel\_sum+blur\_I(i,k);

end

blur\_I(i,j)=pixel\_sum/blur\_size;

end

end

finalI=blur\_I;

imshow(finalI)

array\_size\_column=floor((column\_size-frag\_column\_size)/4)+1;

array\_size\_row=floor((row\_size-frag\_row\_size)/4)+1;

array=zeros(1,array\_size\_column\*array\_size\_row);

t=1;

s=int64(column\_size-frag\_column\_size)

k=int64(frag\_column\_size/2)

%HORISONTAL

i=1;

selectedHor=zeros(500,3);

while i<= column\_size-frag\_column\_size

j=1;

while j<=row\_size-frag\_row\_size

x1=brightnessJump(blur\_I,i,j,frag\_column\_size,frag\_row\_size);

x2=brightnessJump(blur\_I,i,j+frag\_row\_size/2,frag\_column\_size,frag\_row\_size);

array(t)=abs(x1-x2);

%сделать массив динамическим

selectedHor(t,1)=abs(x1-x2);

selectedHor(t,2)=i;

selectedHor(t,3)=j;

t=t+1;

j=j+frag\_row\_size/2;

end

i=i+frag\_column\_size/2;

end

k=1;

fragmentIndex=1;

fragmentArrayHor=zeros(200,2);

index=1;

while k<=length(selectedHor)

if (selectedHor(k,1)>0.05)

fragmentArrayHor(fragmentIndex,1)=selectedHor(k,2);

fragmentArrayHor(fragmentIndex,2)=selectedHor(k,3);

col=selectedHor(k,2);

row=selectedHor(k,3);

if (row==0)

row=row\_size-frag\_row\_size;

end;

fragmentIndex=fragmentIndex+1;

end

k=k+1;

end

figure

imshow(finalI)

sizefragArrayHor=size(find(fragmentArrayHor(:,1)));

fragBlurArrayHor=zeros(frag\_column\_size,frag\_row\_size,sizefragArrayHor(1));

fragArrayHor=zeros(frag\_column\_size,frag\_row\_size,sizefragArrayHor(1));

for i=1:sizefragArrayHor(1)

col=fragmentArrayHor(i,1);

row=fragmentArrayHor(i,2);

fragBlurArrayHor(:,:,i)=finalI(col:col+frag\_column\_size-1, row:row+frag\_row\_size-1);

end;

for t=1:sizefragArrayHor(1)

a1=0;

a2=0;

fragment=fragBlurArrayHor(:,:,t);

for i=1:frag\_column\_size

a1=a1+fragment(i,1,1);

a2=a2+fragment(i,frag\_row\_size,1);

end;

a1=a1/frag\_column\_size;

a2=a2/frag\_column\_size;

for i=1:frag\_column\_size

for j=1:frag\_row\_size/2

fragment(i,j,1)=a1;

end;

for k=frag\_row\_size/2+1:frag\_row\_size

fragment(i,k,1)=a2;

end;

end;

fragArrayHor(:,:,t)=fragment;

end;

%10,20

%VERTICAL

selectedVert=zeros(500,3);

frag\_column\_size2=frag\_row\_size; %8

frag\_row\_size2=frag\_column\_size; %6

i=1;

while i<= column\_size-frag\_column\_size2

j=1;

while j<=row\_size-frag\_row\_size2

x1=brightnessJumpVert(blur\_I,i,j,frag\_column\_size2,frag\_row\_size2);

x2=brightnessJumpVert(blur\_I,i+frag\_column\_size2/2,j,frag\_column\_size2,frag\_row\_size2);

array(t)=abs(x1-x2);

selectedVert(t,1)=abs(x1-x2);

selectedVert(t,2)=i;

selectedVert(t,3)=j;

t=t+1;

j=j+frag\_row\_size/2;

end

i=i+frag\_column\_size/2;

end

k=1;

fragmentIndex=1;

fragmentArrayVert=zeros(200,2);

while k<=length(selectedVert)

%col=floor(selected(k)/(row\_size-frag\_row\_size))+1;

%row=mod(selected(k),row\_size-frag\_row\_size);

if (selectedVert(k,1)>0.05)

fragmentArrayVert(fragmentIndex,1)=selectedVert(k,2);

fragmentArrayVert(fragmentIndex,2)=selectedVert(k,3);

fragmentIndex=fragmentIndex+1;

end

k=k+1;

end

sizefragArrayVert=size(find(fragmentArrayVert(:,1)))

fragBlurArrayVert=zeros(frag\_column\_size2,frag\_row\_size2,sizefragArrayVert(1));

fragArrayVert=zeros(frag\_column\_size2,frag\_row\_size2,sizefragArrayVert(1));

for i=1:sizefragArrayVert(1)

col=fragmentArrayVert(i,1);

row=fragmentArrayVert(i,2);

fragBlurArrayVert(:,:,i)=finalI(col:col+frag\_column\_size2-1, row:row+frag\_row\_size2-1);

end;

for t=1:sizefragArrayVert(1)

a1=0;

a2=0;

fragment=fragBlurArrayVert(:,:,t);

for i=1:frag\_row\_size2

a1=a1+fragment(1,i,1);

a2=a2+fragment(frag\_column\_size2,i,1);

end;

a1=a1/frag\_row\_size2;

a2=a2/frag\_row\_size2;

for i=1:frag\_row\_size2

for j=1:frag\_column\_size2/2

fragment(j,i,1)=a1;

end;

for k=frag\_column\_size2/2+1:frag\_column\_size2

fragment(k,i,1)=a2;

end;

end;

fragArrayVert(:,:,t)=fragment;

end;

%ИДЕНТИФИКАЦИЯ

%ПЕРВЫЙ КВАДРАНТ

y=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),1);

matr=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),17);

% y=zeros(24\*sizefragArrayVert(1),1);

% matr=zeros(24\*sizefragArrayVert(1),17);

k=1;

for t=1:sizefragArrayHor(1)fragmentArrayHor

fragmentBlur=reshape(fragBlurArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

fragment=reshape(fragArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

for i=1:frag\_column\_size-2

for j=1:frag\_row\_size-2

maskBlur=fragmentBlur(i:i+2,frag\_row\_size-j-1:frag\_row\_size-j+1);

mask=fragment(i:i+2,frag\_row\_size-j-1:frag\_row\_size-j+1);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,:) mask(2,:) mask(3,2) mask(3,3)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(3,1);

k=k+1;

end;

end;

end;

for t=1:sizefragArrayVert(1)

fragmentBlur=reshape(fragBlurArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

fragment=reshape(fragArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

for i=1:frag\_column\_size2-2

for j=1:frag\_row\_size2-2

maskBlur=fragmentBlur(i:i+2,frag\_row\_size2-j-1:frag\_row\_size2-j+1);

mask=fragment(i:i+2,frag\_row\_size2-j-1:frag\_row\_size2-j+1);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,:) mask(2,:) mask(3,2) mask(3,3)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(3,1);

k=k+1;

end;

end;

end;

%ВТОРОЙ КВАДРАНТ

y=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),1);

matr=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),17);

k=1;

for t=1:sizefragArrayHor(1)

fragmentBlur=reshape(fragBlurArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

fragment=reshape(fragArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

for i=1:frag\_column\_size-2

for j=1:frag\_row\_size-2

maskBlur=fragmentBlur(frag\_column\_size-i-1:frag\_column\_size-i+1,frag\_row\_size-j-1:frag\_row\_size-j+1);

mask=fragment(frag\_column\_size-i-1:frag\_column\_size-i+1,frag\_row\_size-j-1:frag\_row\_size-j+1);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,2) mask(1,3) mask(2,:) mask(3,:)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(1,1);

k=k+1;

end;

end;

end;

for t=1:sizefragArrayVert(1)

fragmentBlur=reshape(fragBlurArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

fragment=reshape(fragArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

for i=1:frag\_column\_size2-2

for j=1:frag\_row\_size2-2

maskBlur=fragmentBlur(frag\_column\_size2-i-1:frag\_column\_size2-i+1,frag\_row\_size2-j-1:frag\_row\_size2-j+1);

mask=fragment(frag\_column\_size2-i-1:frag\_column\_size2-i+1,frag\_row\_size2-j-1:frag\_row\_size2-j+1);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,2) mask(1,3) mask(2,:) mask(3,:)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(1,1);

k=k+1;

end;

end;

end;

tol=1e-1; maxit=100;

c=bicgstab(matr'\*matr,matr'\*y,tol,maxit)

%ТРЕТИЙ КВАДРАНТ

y=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),1);

matr=zeros(24\*sizefragArrayHor(1)+24\*sizefragArrayVert(1),17);

%

% y=zeros(24\*sizefragArrayVert(1),1);

% matr=zeros(24\*sizefragArrayVert(1),17);

k=1;

for t=1:sizefragArrayHor(1)

fragmentBlur=reshape(fragBlurArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

fragment=reshape(fragArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

for i=1:frag\_column\_size-2

for j=1:frag\_row\_size-2

maskBlur=fragmentBlur(frag\_column\_size-i-1:frag\_column\_size-i+1,j:j+2);

mask=fragment(frag\_column\_size-i-1:frag\_column\_size-i+1,j:j+2);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,1) mask(1,2) mask(2,:) mask(3,:)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(1,3);

k=k+1;

end;

end;

end;

for t=1:sizefragArrayVert(1)

fragmentBlur=reshape(fragBlurArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

fragment=reshape(fragArrayVert(:,:,t),frag\_column\_size2,frag\_row\_size2);

for i=1:frag\_column\_size2-2

for j=1:frag\_row\_size2-2

maskBlur=fragmentBlur(frag\_column\_size2-i-1:frag\_column\_size2-i+1,j:j+2);

mask=fragment(frag\_column\_size2-i-1:frag\_column\_size2-i+1,j:j+2);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,1) mask(1,2) mask(2,:) mask(3,:)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(1,3);

k=k+1;

end;

end;

end;

tol=1e-1; maxit=100;

c=bicgstab(matr'\*matr,matr'\*y,tol,maxit)

%ЧЕТВЕРТЫЙ КВАДРАНТ

for t=1:sizefragArrayHor(1)fragmentArrayHor

fragmentBlur=reshape(fragBlurArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

fragment=reshape(fragArrayHor(:,:,t),frag\_column\_size,frag\_row\_size);

for i=1:frag\_column\_size-2

for j=1:frag\_row\_size-2

maskBlur=fragmentBlur(i:i+2,frag\_row\_size-j-1:frag\_row\_size-j+1);

mask=fragment(i:i+2,frag\_row\_size-j-1:frag\_row\_size-j+1);

rowBlur=[maskBlur(1,:) maskBlur(2,:) maskBlur(3,:)];

row=[mask(1,:) mask(2,:) mask(3,2) mask(3,3)];

rowAll=[rowBlur row];

matr(k,:)=rowAll;

y(k)=mask(3,1);

k=k+1;

end;

end;

end;

%АНАЛИЗ УСТОЙЧИВОСТИ

STABILITY.m

d0=b00\*b00-b01\*b02;

d1=2\*b00\*b10-b01\*b12-b11\*b02;

d2=2\*b00\*b20+b10\*b10-b01\*b22-b11\*b12-b21\*b02;

d3=2\*b10\*b20-b11\*b22-b21\*b12;

d4=b20\*b20-b21\*b22;

g0=b00\*b01-b02\*b02;

g1=b00\*b11+b10\*b01-2\*b02\*b12;

g2=b00\*b21+b10\*b11+b20\*b01-b12\*b12-2\*b02\*b22;

g3=b10\*b21+b20\*b11-2\*b12\*b22;

g4=b20\*b21-b22\*b22;

Cond1=b00 + b10 + b20 + b02 + b12 + b22;

Cond2=b00-b10+b20-b02+b12-b22;

Cond3=d0+g0+d1+g1+d2+g2+d3+g3+d4+g4;

Cond4=d0-g0-d1+g1+d2-g2-d3+g3+d4-g4;

y1=(Cond1>0)&(Cond2>0)&(Cond3>0)&(Cond4>0);

A05=abs(b00+b02);

A15=(b10+b12)\*sign(b00+b02);

A25=(b20+b22)\*sign(b00+b02);

A06=abs(b00-b02);

A16=(b10-b12)\*sign(b00-b02);

A26=(b20-b22)\*sign(b00-b02);

C05=A05+A15+A25;

C15=2\*(A05-A25);

C25=A05-A15+A25;

C06=A06+A16+A26;

C16=2\*(A06-A26);

C26=A06-A16+A26;

y2=(C05>0)&(C15>0)&(C25>0);

y3=(C06>0)&(C16>0)&(C26>0);

%For cond7 and cond8

A07=abs(d0+g0);

A17=(d1+g1)\*sign(d0+g0);

A27=(d2+g2)\*sign(d0+g0);

A37=(d3+g3)\*sign(d0+g0);

A47=(d4+g4)\*sign(d0+g0);

A08=abs(d0-g0);

A18=(d1-g1)\*sign(d0-g0);

A28=(d2-g2)\*sign(d0-g0);

A38=(d3-g3)\*sign(d0-g0);

A48=(d4-g4)\*sign(d0-g0);

C07=A07+A17+A27+A37+A47;

C17=4\*(A07-A47)+2\*(A17-A37);

C27=6\*(A07+A47)-2\*A27;

C37=4\*(A07-A47)+2\*(A37-A17);

C47=A07-A17+A27-A37+A47;

C08=A08+A18+A28+A38+A48;

C18=4\*(A08-A48)+2\*(A18-A38);

C28=6\*(A08+A48)-2\*A28;

C38=4\*(A08-A48)+2\*(A38-A18);

C48=A08-A18+A28-A38+A48;

y4=(C07>0)&(C17>0)&(C27>0)&(C37>0)&(C47>0);

y5=(C08>0)&(C18>0)&(C28>0)&(C38>0)&(C48>0);

yfinal=y1&y2&y3&y4&y5

