

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В. Ломоносова**

Факультет Вычислительной Математики и Кибернетики

На правах рукописи

Насонов Андрей Владимирович

**РЕГУЛЯРИЗИРУЮЩИЕ МЕТОДЫ ПОВЫШЕНИЯ
РАЗРЕШЕНИЯ ИЗОБРАЖЕНИЙ**

Специальность 05.13.18 — математическое моделирование,
численные методы и комплексы программ

ДИССЕРТАЦИЯ

**на соискание ученой степени
кандидата физико-математических наук**

Научный руководитель: доктор физико-математических наук,
А.С. Крылов

Москва 2011

Оглавление

Введение	5
1 Регуляризирующие методы повышения разрешения изображений и суперразрешения	11
1.1 Обзор методов повышения разрешения изображений . . .	12
1.1.1 Линейные методы повышения разрешения изображений	12
1.1.2 Нелинейные методы повышения разрешения изображений	16
1.2 Постановки задач повышения разрешения изображений . .	20
1.2.1 Модель получения цифровых изображений	20
1.2.2 Постановка задачи ресамплинга изображений	22
1.2.3 Постановка задачи суперразрешения	24
1.3 Построение и анализ регуляризирующих методов решения задачи повышения разрешения изображений и суперразрешения	27
1.3.1 Построение регуляризирующих методов решения задачи повышения разрешения изображений и суперразрешения	29
1.3.2 Оптимизация параметров регуляризирующих методов	38
1.3.3 Регуляризирующий метод нахождения соответствий между изображениями в задаче суперразрешения .	49
1.3.4 Неитерационный метод решения задачи суперразрешения	50

2	Методы анализа и постобработки для задачи повышения разрешения изображений	55
2.1	Метрики для базовых контуров на изображениях	56
2.1.1	Метод нахождения базовых контуров на изображении и их окрестностей	60
2.1.2	Метрики точек базовых контуров и окрестностей базовых контуров	64
2.1.3	Применение метрик базовых контуров для задачи комбинирования результатов нескольких методов повышения разрешения изображений	66
2.2	Задача подавления эффекта Гиббса на изображениях . . .	68
2.2.1	Алгоритм обнаружения и оценки эффекта Гиббса .	68
2.2.2	Регуляризирующий метод подавления эффекта Гиббса	75
2.3	Задача повышения резкости изображений	78
2.3.1	Постановка задачи повышения резкости изображений	78
2.3.2	Регуляризирующий метод адаптивного повышения резкости низкочастотной части изображения	81
3	Программный комплекс	84
3.1	Используемые численные методы и их программная реализация	85
3.1.1	Субградиентный метод минимизации регуляризирующих функционалов	85
3.1.2	Метод минимизации квадратичных функционалов на множестве функций с ограниченной полной вариацией	91
3.1.3	Практический выбор параметра регуляризации для задачи повышения разрешения изображений	94

3.1.4	Алгоритм определения уровня эффекта Гиббса на изображениях	97
3.2	Структура программного комплекса	98
	Заключение	104

Введение

Задача повышения разрешения изображений является важной для широкого класса практических приложений, таких как обработка и анализ медицинских изображений, обработка аэрокосмических снимков, обработка данных видеонаблюдения, трансляция видеопотока низкого разрешения на современных широкоформатных дисплеях и ряда других задач.

Высокая производительность компьютеров в настоящее время даёт возможность в реальном времени использовать сложные итерационные методы повышения разрешения изображений, а также решать задачу *суперразрешения* [1,2]. Суперразрешение позволяет сразу по нескольким различным изображениям низкого разрешения одного и того же объекта построить одно изображение высокого разрешения. Это позволяет достичь большего качества по сравнению с повышением разрешения каждого из изображений низкого разрешения по отдельности.

Цифровое изображение

$$v = \{v_{i,j}\}, \quad i = 0, 1, \dots, N_x, \quad j = 0, 1, \dots, N_y,$$

представляет собой конечномерную двумерную таблицу размера $(N_x + 1) \times (N_y + 1)$, элементам которой, называемым *пикселями*, присвоено одно или несколько значений из некоторого конечного множества значений, в качестве которого обычно используется множество целых чисел на отрезке от 0 до 255. В случае изображения в градациях серого значением пикселя является одно число — интенсивность, в случае цветного изображения значением пикселя является вектор из трёх значений, со-

ответствующих цветовым компонентам.

Задача повышения разрешения изображений обладает особенностями, не позволяющими эффективно применять общие методы интерполяции функций для её решения, поэтому для повышения разрешения изображений требуется разработка специальных методов. Среди таких особенностей можно выделить следующие:

1. Наличие специфичной априорной информации о содержании и структуре изображения. Примером такой информации является информация о спектре непрерывного изображения: если изображение удовлетворяет условиям теоремы Котельникова [3], то возможно его однозначное восстановление по дискретному изображению низкого разрешения. Другими примерами являются предположения о сохранении полной вариации изображения при повышении разрешения, предположения о структуре контуров объектов.

Для различных классов изображений эта априорная информация различается, поэтому невозможно разработать универсальный метод повышения разрешения изображений для произвольных изображений.

2. Значимость субъективной оценки результата повышения разрешения. Например, в случае отображения видеопотока низкого разрешения на экранах высокой чёткости, визуальное качество изображений имеет решающее значение. При этом имеет значение как отсутствие артефактов, так и правдоподобность получаемого результата: отсутствие искажений, таких как исчезновение мелких деталей или появление новых деталей.

Наиболее распространёнными артефактами, возникающими при повышении разрешения изображений, являются артефакты, связанные с искажением высокочастотной информации: *эффект размытия*, *алиасинг* (ступенчатость контуров) и *эффект Гиббса*. В задачах обработки изображений эффект Гиббса проявляется как эффект *ложного оконтуривания*, возникающий при недостатке информации о высоких частотах

изображения и проявляющийся в виде ореолов возле резких контуров. В отличие от настоящего эффекта Гиббса, в случае ложного оконтуривания наблюдается обычно только одна или две осцилляции.

На рис.1 приведены примеры этих артефактов.



Рис. 1: Примеры артефактов, возникающих при повышении разрешения изображений.

Для подавления артефактов, внесённых методами повышения разрешения изображений, применяются методы постобработки.

3. В большинстве случаев требуется не построение непрерывного изображения, а переход с более грубой сетки на более мелкую. Такой процесс называют *ресамплингом изображений*, а коэффициент отношения шага крупной сетки к коэффициенту шага мелкой сетки — *коэффициентом увеличения изображений*.

4. Имеет значение вычислительная сложность алгоритмов повышения разрешения изображений при их применении в реальном времени.

Объективный анализ качества методов повышения разрешения изображений осуществляется с использованием специальных метрик, учитывающих как близость результата повышения разрешения к эталонному изображению, так и субъективную оценку качества изображения.

Большинство алгоритмов повышения разрешения изображения работают с изображениями, содержащими только одну компоненту. Применение таких алгоритмов к цветным изображениям заключается в представлении цветного изображения в виде трёх однокомпонентных изобра-

жений и повышении разрешения каждого из этих изображений по отдельности. При этом часто производится переход в другое цветовое пространство, например, YUV , где первая компонента представляет из себя значение интенсивности — яркостная компонента Y , а оставшиеся две компоненты определяют цвет. В связи с тем, что чувствительность человеческого восприятия к яркости выше, чем к цвету, такой подход позволяет понизить вычислительную сложность при повышении разрешения цветных изображений, применяя для повышения разрешения цветовой компонент быстрые алгоритмы.

Существуют алгоритмы, которые работают с компонентами пикселя цветного изображения как с единым целым. Например, в задачах выделения контуров используется цветовой градиент [4], позволяющий находить контуры, которые нельзя найти, используя только яркостную компоненту изображения. Однако в задачах повышения разрешения изображения такие алгоритмы не получили широкого распространения из-за высокой сложности и практически полного отсутствия разницы в субъективном качестве изображений по сравнению с методами, не учитывающими цветовой градиент.

В настоящее время алгоритмы интерполяции цветных изображений применяются, в основном, для демозаикинга — интерполяции байеровских шаблонов [5]. В большинстве устройств получения цветных цифровых изображений используются матрицы, состоящие из различных фотоэлементов, чувствительных к свету с определенной длиной волны. Используются три типа элементов: чувствительных к красному, зеленому и синему цветам. Эти три типа элементов расположены в виде мозаики, называемой обычно байеровским шаблоном. Задача демозаикинга состоит в получении полноцветного изображения по его байеровскому шаблону.

Целью диссертационной работы является построение регуляризирующих методов повышения разрешения изображений и суперразрешения, а также методов их анализа и постобработки, включающих в себя разра-

ботку метрик оценки качества обработки контуров и методов подавления артефактов, внесённых алгоритмами повышения разрешения изображения.

В первой главе рассматриваются задачи повышения разрешения изображений и суперразрешения. Глава начинается с краткого обзора существующих методов повышения разрешения изображений, которые можно разделить на два класса: класс линейных методов и класс нелинейных методов. Преимуществом линейных методов является их простота и вычислительная эффективность, но, в отличие от нелинейных методов, невозможно построить линейный метод, в котором не было бы ни одного из артефактов, приведённых на рис.1.

В диссертационной работе задача повышения разрешения изображений поставлена как обратная некорректная задача для задачи понижения разрешения изображений:

$$Az = u, \quad (1)$$

где u — известное изображение низкого разрешения, z — искомое изображение высокого разрешения, A — оператор понижения разрешения. Оператор A строится на основе математической модели получения цифровых изображений с помощью камеры.

Для нахождения решения поставленной обратной задачи (1) применяется метод регуляризации А.Н.Тихонова [6]. Задача регуляризации рассматривается в трёх постановках:

- I. Нахождение $\inf (\|Az - u\| + \lambda\Psi[z]), \lambda > 0$.
- II. Нахождение $\inf \Psi[z]$ для $\|Az - u\| \leq \delta, \delta > 0$.
- III. Нахождение $\inf \|Az - u\|$ для $\Psi[z] \leq C, C > 0$.

Определяются условия, при которых постановки I, II и III являются эквивалентными. Это позволяет выбирать наиболее удобную с вычислительной точки зрения постановку в зависимости от ситуации.

Также производится анализ выбора норм пространств и стабилизато-

ра среди наиболее часто используемых в задачах обработки изображений норм и стабилизаторов.

Вторая глава посвящена оценке качества изображений, обнаружению и подавлению артефактов на изображениях, возникающих после повышения разрешения. Оценка качества изображений основана на анализе характерных артефактов алгоритмов повышения разрешения изображений — размытия и эффекта Гиббса. Предлагается алгоритм нахождения областей, соответствующих этим артефактам, основанный на использовании математической морфологии. Алгоритм находит базовые контуры на изображении — контуры, которые сохраняются при определённом уровне деградации качества изображения, а затем выделяет окрестности, прилегающие к базовым контурам. Предлагаемые метрики оценки качества изображений в областях базовых границ проиллюстрированы на примере алгоритма комбинирования результатов двух алгоритмов повышения разрешения изображений.

Рассматриваются задачи подавления эффекта Гиббса и повышения резкости изображений. Для нахождения решений этих задач применяются регуляризирующие методы. Параметр регуляризации в данных задачах адаптивно определяется с использованием предложенного алгоритма оценки уровня эффекта Гиббса. Данная оценка основана на анализе зависимости значения полной вариации профилей базовых границ от масштаба.

Третья глава посвящена описанию разработанного программного комплекса и реализации используемых в работе методов минимизации регуляризирующих функционалов. Для минимизации используемых в работе недифференцируемых выпуклых функционалов используется субградиентный метод. В данной главе также рассматривается задача практического выбора значения параметра регуляризации для задачи повышения разрешения изображений.

Глава 1

Регуляризирующие методы повышения разрешения изображений и суперразрешения

В данной главе рассматривается задача повышения разрешения изображений в градациях серого. Адаптация методов повышения разрешения изображений для цветных изображений предлагается в главе 3.

Мы используем представление изображения в виде вещественной сеточной функции $\tilde{u}(ih_x, jh_y) = v_{i,j}$, заданной на двумерной равномерной сетке

$$\begin{aligned}\tilde{\Omega}_{h_x, h_y} &= \{(ih_x, jh_y) : 0 \leq i \leq N_x, 0 \leq j \leq N_y\}, \\ h_x &> 0, h_y > 0.\end{aligned}$$

Для того, чтобы избавиться от задания граничных условий в явном виде для каждого конкретного метода повышения разрешения изображений, удобно произвести продолжение сеточной функции \tilde{u} с конечномерной сетки $\tilde{\Omega}_{h_x, h_y}$ на бесконечномерную

$$\begin{aligned}\Omega_{h_x, h_y} &= \{(ih_x, jh_y) : i, j \in \mathbb{Z}\}, \\ h_x &> 0, h_y > 0.\end{aligned}\tag{1.1}$$

При этом значения функции u совпадают со значениями пикселей изображения \tilde{u} для $0 \leq i \leq N_x, 0 \leq j \leq N_y$, а значения за пределами этого прямоугольника вычисляются путём чётного (можно также использовать и другие способы, например нечётное продолжение) продолжения

относительно его сторон. Чётное продолжение для изображения размера $(N_x + 1) \times (N_y + 1)$ выглядит следующим образом:

$$u_{i+2N_x p, j+2N_y q} = \begin{cases} \tilde{u}_{i,j} & \text{при } 0 \leq i < N_x, 0 \leq j < N_y, \\ \tilde{u}_{2N_x-i,j} & \text{при } N_x \leq i < 2N_x, 0 \leq j < N_y, \\ \tilde{u}_{i,2N_y-j} & \text{при } 0 \leq i < N_x, N_y \leq j < 2N_y, \\ \tilde{u}_{2N_x-i,2N_y-j} & \text{при } N_x \leq i < 2N_x, N_y \leq j < 2N_y, \end{cases} \quad (1.2)$$

где $p, q \in \mathbb{Z}$.

Мы также будем использовать δ -представление дискретного изображения в виде суммы дельта-функций в тех случаях, когда необходимо произвести переход от дискретного представления к непрерывному без изменения самого изображения

$$\hat{u}(x, y) = \sum_{i,j=-\infty}^{+\infty} u_{i,j} \delta(x - ih_x) \delta(y - ih_y),$$

где $\delta(t)$ — дельта-функция:

$$\int_{-\infty}^{+\infty} f(t) \delta(t - t_0) dt = f(t_0).$$

1.1 Обзор методов повышения разрешения изображений

1.1.1 Линейные методы повышения разрешения изображений

Простейшим классом методов интерполяции изображений является класс линейных методов. В общем случае методы этого класса основаны на использовании непрерывной свёртки δ -представления изображения и некоторого ядра $K(x, y)$ [7]

$$\begin{aligned} f(x_0, y_0) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{u}(x_0 - x, y_0 - y) K\left(\frac{x}{h_x}, \frac{y}{h_y}\right) dx dy = \\ &= \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} u_{i,j} K\left(\frac{x_0}{h_x} - i, \frac{y_0}{h_y} - j\right), \end{aligned} \quad (1.3)$$

где $f(x, y)$ — интерполированное изображение. На данном этапе не накладывается никаких ограничений на значения пикселей и размер исходного изображения. Как правило, используется ядро вида $K(x, y) = K(x)K(y)$. Это позволяет существенно повысить скорость работы алгоритма, так как в этом случае интерполяция разбивается на последовательную одномерную интерполяцию сначала по одной оси, затем по другой:

$$\begin{aligned} f_i(y) &= \sum_{j=-\infty}^{+\infty} u_{i,j} K\left(\frac{y}{h_y} - j\right), \\ f(x, y) &= \sum_{i=-\infty}^{+\infty} f_i(y) K\left(\frac{x}{h_x} - i\right). \end{aligned} \tag{1.4}$$

При этом на ядро $K(t)$ накладываются следующие условия:

1. Условие интерполяции

$$\begin{aligned} K(0) &= 1, \\ K(n) &= 0, \quad n = \pm 1, \pm 2, \dots \end{aligned}$$

Это условие нужно для выполнения равенства значений исходного и интерполированного изображений

$$f(ih_x, jh_y) = u_{i,j}, \quad i, j \in \mathbb{Z}$$

в узлах сетки Ω_{h_x, h_y} .

2. Финитность ядра

$$K(t) = 0, \quad |t| > p > 0.$$

Так как вычислительная сложность метода интерполяции (1.3) пропорциональна количеству ненулевых слагаемых в сумме, то для достижения высокой скорости интерполяции выбираются ядра с небольшим p .

3. Условие нормировки коэффициентов

$$\sum_{i=-\infty}^{+\infty} K(t+i) = 1, \quad 0 \leq t < 1.$$

Данное условие следует из того, что интерполяция изображения, все пиксели которого имеют одинаковое значение $u_{i,j} = C$, должна давать константную функцию $f(x, y) = C$.

В работах [8–10] собраны воедино, обобщены и подробно рассмотрены практически все используемые в настоящее время методы линейной интерполяции.

К наиболее популярным линейным методам можно отнести следующие методы:

1. Метод ближайшего соседа:

$$K(t) = \begin{cases} 1 & \text{для } -1/2 \leq t < 1/2, \\ 0 & \text{иначе.} \end{cases}$$

Это самый быстрый метод интерполяции. В случае ресемплинга с целым коэффициентом увеличения он представляет собой простое повторение пикселей изображения. Данный метод обладает серьёзным недостатком: получаемая функция $f(x, y)$ является разрывной, а на изображении ярко выражен эффект ступенчатости.

2. Метод интерполяции первого порядка:

$$K(t) = \begin{cases} 1 - |t| & \text{для } -1 < t < 1, \\ 0 & \text{иначе.} \end{cases}$$

В двумерном случае этот метод называют *билинейной интерполяцией*. Получаемая с помощью этого метода функция $f(x, y)$ является непрерывной с кусочно-постоянной первой производной. Артефактами метода является эффект ступенчатости, но менее выраженный, чем в методе ближайшего соседа, и небольшой эффект размытия.

3. Метод сплайновой интерполяции третьего порядка (*бикубическая*

интерполяция):

$$K(t) = \begin{cases} (a+2)|t|^3 - (a+3)|t|^2 + 1 & \text{для } |t| \leq 1, \\ a|t|^3 - 5a|t|^2 + 8a|t| - 4t & \text{для } 1 < |t| < 2, \\ 0 & \text{иначе.} \end{cases} \quad (1.5)$$

Этот метод наиболее часто применяется для повышения разрешения изображений из-за малого значения p ($p = 2$) и хорошего баланса между артефактами размытия, алиасинга и оконтуривания краёв.

4. «Идеальная» интерполяция, основанная на использовании теоремы Котельникова [3]:

$$K(t) = \text{sinc } t = \frac{\sin \pi t}{\pi t}. \quad (1.6)$$

Теорема Котельникова гарантирует восстановление непрерывного сигнала по дискретному при условии, что в спектре непрерывного сигнала не было частот выше $\frac{1}{2 \max(h_x, h_y)}$. На практике применение этого метода затруднительно из-за того, что ядро (1.6) не является финитным ($p = \infty$), а спектр естественных изображений не является ограниченным. При интерполяции изображения с помощью данного метода, на изображении высокого разрешения наблюдается ярко выраженный эффект ложного оконтуривания, причиной которого является эффект Гиббса.

5. Метод Ланцоша, представляющий собой приближение идеальной интерполяции финитным ядром:

$$K_p(t) = \begin{cases} \text{sinc}(t) \cdot \text{sinc}\left(\frac{t}{p}\right) & \text{для } |t| < p, \\ 0 & \text{иначе.} \end{cases} \quad (1.7)$$

Параметр $p \in \mathbb{N}$ задаёт размер ядра. На практике обычно используют $p = 2$ и $p = 3$.

6. Метод интерполяции с использованием функции Гаусса (функции нормального распределения):

$$K(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}}.$$

Данный метод интерполяции не удовлетворяет первому условию, тем не менее, он обладает рядом полезных свойств и широко используется в обработке изображений. Например, он используется для вычисления производных любых порядков в произвольных точках на изображении. Ядро фильтра не является финитным, тем не менее, оно быстро убывает при росте t , поэтому на практике обычно принимают $K(t) = 0$ при $|t| > t_\sigma$, где значение t_σ выбирается таким образом, чтобы ошибка не превышала заданного значения. При использовании $t_\sigma = 3\sigma$ значение ошибки существенно ниже разницы между отображаемыми градациями интенсивности современных мониторов, поэтому на практике чаще всего используют именно это значение. Для эффективного вычисления свёртки с фильтром Гаусса используется его приближение с помощью рекуррентных фильтров [11].

Среди класса линейных методов невозможно выбрать наилучший метод. Любой линейный метод представляет собой баланс между тремя типами артефактов: размытия, алиасинга и эффекта Гиббса (см. введение, рис. 1). Подавление одного из артефактов приводит к усилению других артефактов.

1.1.2 Нелинейные методы повышения разрешения изображений

Добиться более качественных результатов можно с помощью нелинейных методов, в которых функция усреднения $K(x, y)$ задаётся отдельно для каждого интерполируемого пикселя и зависит от значений пикселей интерполируемого изображения.

Примером нелинейных методов является класс градиентных методов [12–14]. В основе градиентных алгоритмов лежит тот факт, что интерполяция вдоль краёв (контуров) деталей изображения даёт результаты лучше, чем обычная линейная интерполяция. В целом результат, получаемый градиентными методами, близок к результату бикубической интерполяции, но, при этом, эффект алиасинга практически полностью

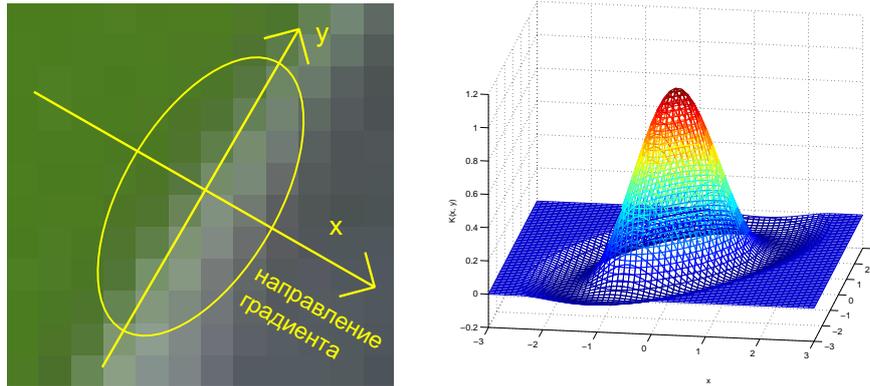


Рис. 1.1: Пример ядра $K(x, y)$ в точке контура при интерполяции с помощью градиентных методов.

отсутствует. Один из способов реализации такой интерполяции основан на использовании функции Гаусса с переменными радиусами по разным направлениям:

$$K(x', y', x_0, y_0) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{x'^2}{2\sigma_x^2} - \frac{y'^2}{2\sigma_y^2}}.$$

Здесь ядро $K(x', y', x_0, y_0)$ задано в декартовой системе координат $Ox'y'$ с центром в интерполируемом пикселе $O = (x_0, y_0)$ и осью Ox' , совпадающую с направлением градиента исходного изображения в (x_0, y_0) . Радиус σ_y фиксирован, а радиус σ_x выбирается на основе анализа модуля градиента изображения: чем больше модуль градиента, тем меньше значение σ_x . Пример ядра при интерполяции в точке границы приведён на рис. 1.1.

Ещё один алгоритм, основанный на использовании градиента — это метод WADI (Warped Distance) [15]. В нём значение интерполируемого пикселя представляет собой взвешенную сумму значений четырёх ближайших пикселей, причём веса выбираются в зависимости от расстояния до этих пикселей и модуля производной в этих пикселях. Чем больше производная, тем меньше весовые коэффициенты.

Алгоритм NEDI (New Edge-Directed Interpolation) увеличивает изображение в 2 раза, используя предположение о самоподобии фрагментов

изображения высокого разрешения и соответствующих им фрагментов изображения низкого разрешения [16, 17]. Значения интерполируемых пикселей представляют собой взвешенную сумму четырёх соседних пикселей, при этом веса вычисляются в предположении, что исходное изображение было получено с теми же весами путём увеличения уменьшенного в 2 раза исходного изображения. NEDI хорошо справляется с контурами объектов (визуально лучше, чем градиентные методы), но плохо обрабатывает участки изображения, для которых предположение о самоподобии неверно.

К методу NEDI близок класс фрактальных алгоритмов, в основе которых лежит идея о самоподобии целых блоков изображения. Принцип работы алгоритмов фрактального ресамплинга аналогичен алгоритмам фрактального сжатия и основан на методах фрактального кодирования. Различные методы фрактального кодирования, в том числе использующие итерации сжимающих отображений (Iterated Function System, IFS), рассматриваются в статьях [18–21].

Широкий класс методов представляет задачу повышения разрешения изображений как обратную задачу для уравнения

$$Az = u, \tag{1.8}$$

где u — исходное изображение низкого разрешения, z — искомое изображение высокого разрешения, A — оператор понижения разрешения. Таким образом, задача формулируется в виде: построить такое изображение высокого разрешения, которое после уменьшения даст известное изображение низкого разрешения.

Обратная задача, определяемая уравнением (1.8), является некорректно поставленной. Для её решения применяются следующие методы, основанные на использовании априорной информации об изображении:

1. Регуляризирующие методы [22]. В общем случае задаются два функционала: функционал соответствия изображения высокого разре-

шения изображению низкого разрешения $I_d(z, u)$, например, невязка

$$I_d(z, u) = \|Az - u\|_2^2$$

и функционал соответствия изображения высокого разрешения априорной информации $I_s(z)$ — стабилизатор [23]. Для нахождения изображения высокого разрешения производится минимизация регуляризирующего функционала

$$z_\lambda = \arg \min_z (I_d(z, u) + \lambda I_s(z)),$$

где параметр регуляризации λ контролирует баланс между соответствием изображению низкого разрешения и априорной информации.

Также используются следующие регуляризирующие методы [24–26]

$$z_C = \arg \min_{z \in M_C} I_d(z, u), \quad M_C = \{z | I_s(z) \leq C\}$$

и

$$z_\sigma = \arg \min_{z \in M_\sigma} I_s(z), \quad M_\sigma = \{z | I_d(z, u) \leq \sigma\}.$$

2. Метод обратной проекции ошибки [27, 28], заключающийся в итерационной минимизации невязки $\|Az - u\|$:

$$z_{k+1} = z_k + U(Az_k - u),$$

где U — оператор повышения разрешения изображений. Недостатком метода является, вообще говоря, отсутствие устойчивости и сходимости. Результат зависит от выбора начального приближения z_0 и оператора U . Тем не менее, данный метод используется в практических приложениях, так как часто позволяет за малое количество итераций получить результат с хорошим визуальным качеством.

3. Метод проекции на выпуклые множества, заключающийся в проектировании приближения z_0 на множество изображений M , где M — выпуклый компакт [29, 30]:

$$z_R = pr_M z_0 = \arg \min_{z \in M} \|z - z_0\|_2^2.$$

1.2 Постановки задач повышения разрешения изображений

1.2.1 Модель получения цифровых изображений

Рассмотрим модель получения изображений в градациях серого с помощью цифровой камеры.

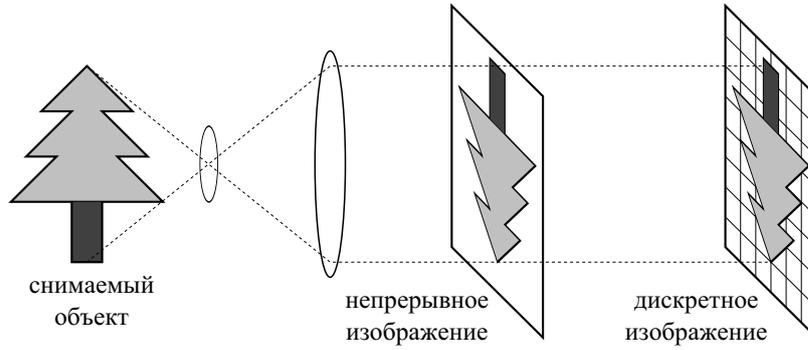


Рис. 1.2: Модель получения изображений с помощью цифровой камеры.

В цифровой камере изображение проецируется на матрицу, состоящую из светочувствительных элементов (сенсоров), каждый из которых соответствует пикселю на изображении. Значение пикселя $u_{i,j}$ равно суммарной интенсивности непрерывного светового потока

$$f(x, y) \in F = L(\mathbb{R}^2),$$

попавшего на сенсор:

$$u_{i,j} = \int_{x,y} f(x, y) K_{i,j}(x - x_i, y - y_i) dx dy, \quad (1.9)$$

где (x_i, y_i) — координаты центра пикселя, $K_{i,j}(x, y)$ — функция усреднения в координатах относительно центра пикселя, называемая также *point spread function* (PSF), или *функцией распределения точки*, пространство $L(\mathbb{R}^2)$ — банахово пространство вещественных функций, определённых на \mathbb{R}^2 с некоторой нормой $\|\cdot\|_F$.

Будем считать, что все сенсоры одинаково усредняют световой поток $K_{i,j}(x, y) = K(x, y)$ и расположены в узлах равномерной сетки Ω_{h_x, h_y} (1.1).

Дополнительно наложим условие центральной симметричности на $K(x, y)$. Этим условиям удовлетворяет большинство современных камер. При данных условиях модель (1.9) принимает вид:

$$u_{i,j} = [f * K](x_i, y_j), \quad (1.10)$$

где $f * K$ — операция свёртки

$$[f(x, y) * K(x, y)](x_0, y_0) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) K(x_0 - x, y_0 - y) dx dy.$$

Для прямоугольных сенсоров функция распределения точки принимает вид

$$K(x, y) = \begin{cases} 1 & \text{при } -\frac{h_x}{2} \leq x < \frac{h_x}{2}, -\frac{h_y}{2} \leq y < \frac{h_y}{2}, \\ 0 & \text{иначе.} \end{cases} \quad (1.11)$$

Будем использовать в качестве функции распределения точки двумерную функцию Гаусса

$$K(x, y) = G_{\sigma_x, \sigma_y}(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right),$$

где *радиусы* σ_x и σ_y (квадратные корни дисперсий σ_x^2 , σ_y^2) выбираются исходя из размеров сенсоров h_x и h_y .

Для сенсоров с одинаковыми размерами по вертикали и горизонтали $h = h_x = h_y$ мы берём одинаковые значения радиусов $\sigma = \sigma_x = \sigma_y$, и функция распределения точки записывается в виде

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \quad (1.12)$$

Значение радиуса σ берётся пропорционально размеру сенсоров $\sigma = \sigma_0 h$. Параметр σ_0 определяется конструкцией камеры, мы используем значения σ_0 в интервале $[0.4, 0.5]$.

В дальнейшем будем работать только с камерами с квадратными сенсорами $h_x = h_y = h$ с функцией распределения точки в виде (1.12).

1.2.2 Постановка задачи ресамплинга изображений

Обозначим пространство $\ell(\Omega_h)$ — банахово пространство вещественных сеточных функций, заданных на бесконечномерной равномерной сетке с шагом h

$$\Omega_h = \{(x_i, y_j) : x_i = ih, y_j = jh, \quad i, j \in \mathbb{Z}\}$$

с нормой $\|\cdot\|_{\ell(\Omega_h)}$.

Построение непрерывного изображения

Сформулируем задачу построения непрерывного изображения $f \in L(\mathbb{R}^2)$ по изображению $u \in \ell(\Omega_h)$, полученному с использованием введённой в предыдущем разделе математической модели. Согласно (1.10)

$$u_{i,j} = [f * G_{\sigma_0 h}](x_i, y_j) = [D_h H_{\sigma_0 h} f]_{i,j},$$

где $H_{\sigma_0 h}$ — оператор свёртки с функцией Гаусса $G_{\sigma_0 h}$,

$$D_h : L(\mathbb{R}^2) \rightarrow \ell(\Omega_h)$$

— оператор дискретизации:

$$[D_h f]_{i,j} = f(ih, jh).$$

Таким образом, в непрерывном случае задача повышения разрешения изображений ставится в виде обратной задачи для задачи получения дискретного изображения с камеры:

$$A_h f = D_h H_{\sigma_0 h} f = u, \quad f \in L(\mathbb{R}^2), u \in \ell(\Omega_h). \quad (1.13)$$

Ресамплинг изображений

В реальных задачах требуется не построение непрерывного изображения f , а переход на сетку с меньшим шагом — ресамплинг изображений.

Рассмотрим задачу понижения разрешения изображений в следующем виде: по имеющемуся изображению $z \in \ell(\Omega_{h/s})$, полученному с помощью камеры с размером сенсоров h/s , построить изображение

$u \in \ell(\Omega_h)$ для камеры с размером сенсоров h , где s — коэффициент масштабирования, $s > 1$.

Согласно математической модели (1.10):

$$\begin{aligned} z_{i,j} &= [f * G_{\sigma_0 h/s}] \left(\frac{ih}{s}, \frac{jh}{s} \right), \\ u_{i,j} &= [f * G_{\sigma_0 h}](ih, jh). \end{aligned}$$

Используя свойство $G_a * G_b = G_{\sqrt{a^2+b^2}}$, запишем вторую формулу в виде

$$u_{i,j} = [f * G_{\sigma_0 h/s} * G_{\sigma_0 h \sqrt{1-1/s^2}}](ih, jh). \quad (1.14)$$

Для вычисления $u_{i,j}$ необходимо иметь непрерывное изображение $f * G_{\sigma_0 h/s}$, но нам дано только дискретное изображение $z_{i,j}$. Используем δ -представление \hat{z} вместо $f * G_{\sigma_0 h/s}$ в (1.14) в качестве аппроксимации:

$$u_{i,j} = [\hat{z} * G_{\sigma_0 h \sqrt{1-1/s^2}}](ih, jh).$$

При этом остаётся лишь зависимость от s :

$$\begin{aligned} u_{i_0, j_0} &= [\hat{z} * G_{\sigma_0 h \sqrt{1-1/s^2}}](ih, jh) = \\ &= \frac{\sum_{i,j \in \mathbb{Z}} z_{i,j} G_{\sigma_0 h \sqrt{1-1/s^2}}(i_0 h - \frac{ih}{s}, j_0 h - \frac{jh}{s})}{\sum_{i,j \in \mathbb{Z}} G_{\sigma_0 h \sqrt{1-1/s^2}}(i_0 h - \frac{ih}{s}, j_0 h - \frac{jh}{s})} = \\ &= \frac{\sum_{i,j \in \mathbb{Z}} z_{i,j} G_{\sigma_0 \sqrt{1-1/s^2}}(i_0 - \frac{i}{s}, j_0 - \frac{j}{s})}{\sum_{i,j \in \mathbb{Z}} G_{\sigma_0 \sqrt{1-1/s^2}}(i_0 - \frac{i}{s}, j_0 - \frac{j}{s})} = \\ &= \frac{\sum_{i,j \in \mathbb{Z}} z_{i,j} G_{\sigma_0 \sqrt{s^2-1}}(i_0 s - i, j_0 s - j)}{\sum_{i,j \in \mathbb{Z}} G_{\sigma_0 \sqrt{s^2-1}}(i_0 s - i, j_0 s - j)}. \end{aligned}$$

Если коэффициент масштабирования s целый, то все узлы грубой сетки Ω_h являются узлами исходной сетки $\Omega_{h/s}$. В этом случае для нахождения u достаточно вычислить свёртку:

$$\begin{aligned} u_{i_0, j_0} &= \sum_{i,j \in \mathbb{Z}} z_{i,j} G_{\sigma_0 \sqrt{s^2-1}}(i_0 s - i, j_0 s - j) = \\ &= [z * G_{\sigma_0 \sqrt{s^2-1}}]_{i_0 s, j_0 s} = [H_{\sigma_0 \sqrt{s^2-1}} z]_{i_0 s, j_0 s}. \end{aligned}$$

В дальнейшем будем работать только с целыми значениями коэффициента увеличения s .

Таким образом, задача понижения разрешения изображений в s раз формулируется в виде:

$$u = A_s z = D_s H_{\sigma_0 \sqrt{s^2 - 1}} z, \quad (1.15)$$

где D_s — это оператор перехода на сетку с s раз большим шагом, действующий из пространства $\ell(\Omega_{h/s})$ в пространство $\ell(\Omega_h)$:

$$[D_s z]_{i,j} = z_{si,sj}.$$

Оператор A_s является линейным оператором.

Задача повышения разрешения изображений ставится в виде обратной задачи для задачи понижения разрешения:

$$A_s z = u, \quad u \in \ell(\Omega_h), \quad z \in \ell(\Omega_{h/s}). \quad (1.16)$$

1.2.3 Постановка задачи суперразрешения

Суперразрешение — это построение изображения высокого разрешения по нескольким слабо отличающимся изображениям объекта низкого разрешения. Основная идея заключается в использовании субпиксельных сдвигов снимаемого объекта для комбинации информации с нескольких изображений низкого разрешения. Поясним данную задачу на примере, представленном на Рис.1.3.

Пусть квадратные сенсоры камеры низкого разрешения с размером $2h$ имеют прямоугольную функцию распределения точки (1.11), а изображение было снято 4 раза: первый раз (u^1) без сдвига, второй (u^2) — со сдвигом на h по горизонтали, третий (u^3) — со сдвигом на h по вертикали, четвёртый (u^4) — со сдвигом на h по горизонтали и по вертикали. Требуется построить изображение z без сдвига с вдвое большим разрешением.

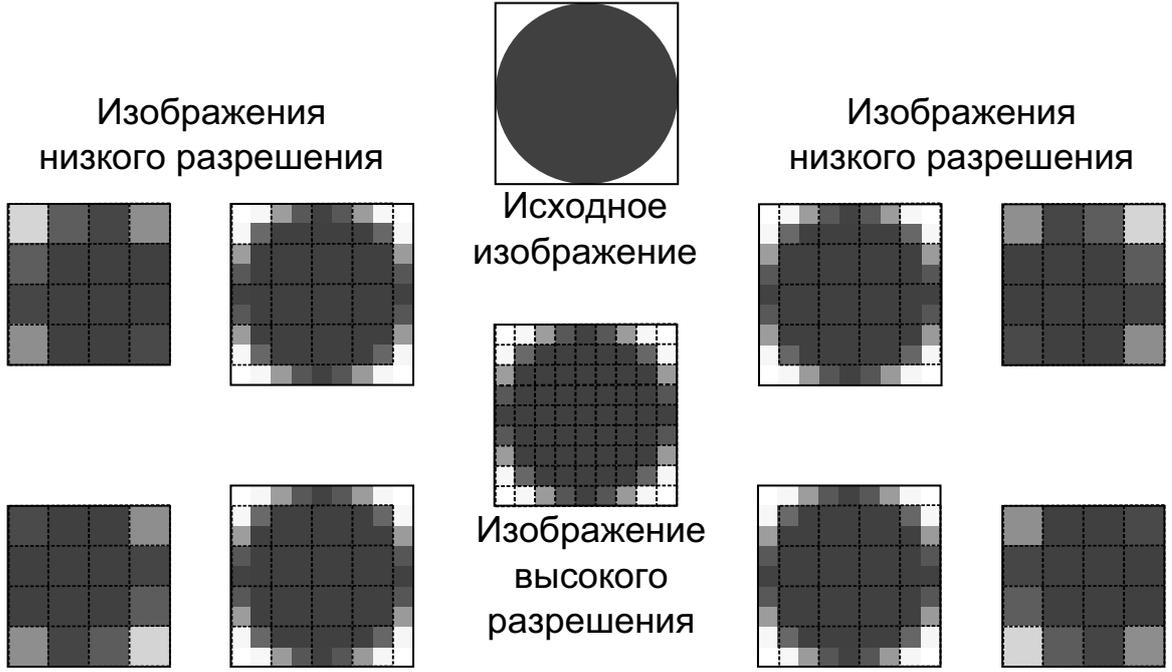


Рис. 1.3: Иллюстрация для задачи суперразрешения: восстановление изображения с вдвое большим разрешением по четырём изображениям низкого разрешения.

Значения пикселей первого изображения низкого разрешения можно выразить через пиксели z :

$$u_{i,j}^1 = \frac{1}{4}(z_{2i,2j} + z_{2i+1,2j} + z_{2i,2j+1} + z_{2i+1,2j+1}),$$

Для второго изображения справедлива формула

$$u_{i,j}^2 = \frac{1}{4}(z_{2i+1,2j} + z_{2i+2,2j} + z_{2i+1,2j+1} + z_{2i+2,2j+1}).$$

Аналогично выписываются соотношения для пикселей третьего и четвёртого изображений. Объединяя эти соотношения, получим:

$$\frac{1}{4}(z_{i,j} + z_{i+1,j} + z_{i,j+1} + z_{i+1,j+1}) = v_{i,j},$$

где

$$\begin{aligned} v_{2i,2j} &= u_{i,j}^1, \\ v_{2i+1,2j} &= u_{i,j}^2, \\ v_{2i,2j+1} &= u_{i,j}^3, \\ v_{2i+1,2j+1} &= u_{i,j}^4. \end{aligned}$$

Таким образом, рассмотренная задача свелась к задаче обращения свёртки изображения z с прямоугольным фильтром 2×2 .

Теперь перейдём к общей постановке задачи суперразрешения, используя предложенную модель понижения разрешения изображений.

Пусть $f \in L(\mathbb{R}^2)$ — исходное непрерывное изображение. С течением времени объекты на изображении смещаются, поэтому камера получает на вход изображения

$$f^k \in L(\mathbb{R}^2), \quad k = 1, 2, \dots, K,$$

полученные из f в результате применения операторов движения T_k . Оператор движения

$$T_k = (T_x^k, T_y^k)$$

определяется двумя функциями $T_x^k(x, y)$ и $T_y^k(x, y)$, задающими соответствие между координатами (x, y) на k -м изображении и координатами $(x + T_x^k(x, y), y + T_y^k(x, y))$ на исходном изображении:

$$[T_k f](x, y) = f(x + T_x^k(x, y), y + T_y^k(x, y)). \quad (1.17)$$

На выходе камеры получаются изображения низкого разрешения

$$u^k \in \ell(\Omega_h), \quad k = 1, 2, \dots, K,$$

причём

$$u^k = D_h H_{\sigma_0 h} T_k f. \quad (1.18)$$

Таким образом, в непрерывном случае задача суперразрешения изображений ставится в виде обратной задачи для задачи получения дискретных изображений с камеры, представленных системой уравнений:

$$\begin{aligned} A_h^k f &= D_h H_{\sigma_0 h} T_k f = u_k, \quad k = 1, 2, \dots, K, \\ f &\in L(\mathbb{R}^2), \quad u_k \in \ell(\Omega_h), \end{aligned} \quad (1.19)$$

где операторы D_h и $H_{\sigma_0 h}$ — это введённые ранее для задачи повышения разрешения изображений оператор дискретизации и оператор свёртки с фильтром Гаусса.

Аналогично формулируется задача для дискретного случая. Пусть

$$u^k \in \ell(\Omega_h), \quad k = 1, 2, \dots, K$$

— изображения низкого разрешения, полученные с помощью камеры с размером сенсоров h , $z \in \ell(\Omega_{h/s})$ — реконструируемое изображение высокого разрешения, полученное с помощью камеры с размером сенсоров h/s , s — коэффициент масштабирования, $s > 1$.

Согласно математической модели (1.10):

$$\begin{aligned} z_{i,j} &= [f * G_{\sigma_0 h/s}](ih/s, jh/s) \\ u_{i,j}^k &= [(T_k f) * G_{\sigma_0 h}](ih, jh), \quad k = 1, 2, \dots, K. \end{aligned}$$

Заменим вторую формулу на следующее приближение:

$$u_{i,j}^k = [(T_k(f * G_{\sigma_0 h/s})) * G_{\sigma_0 h \sqrt{1-1/s^2}}](ih, jh), \quad k = 1, 2, \dots, K.$$

Используя рассуждения, аналогичные рассуждениям для задачи ресамплинга изображений, получаем постановку задачи суперразрешения в дискретном случае в виде нахождения решения системы уравнений относительно z :

$$A_s^k z = D_s H_{\sigma_0 \sqrt{s^2-1}} T_k z = u^k, \quad k = 1, 2, \dots, K, \quad (1.20)$$

где операторы D_s и $H_{\sigma_0 \sqrt{s^2-1}}$ — это введённые ранее для задачи ресамплинга оператор перехода на сетку с s раз большим шагом и оператор дискретной свёртки с фильтром Гаусса. При применении оператора T_k к изображению z , заданному на дискретной сетке, мы используем метод билинейной интерполяции для вычисления значений в произвольных точках.

1.3 Построение и анализ регуляризирующих методов решения задачи повышения разрешения изображений и суперразрешения

Для нахождения решения задачи повышения разрешения изображений в непрерывной постановке (1.13) требуется построение обратного опе-

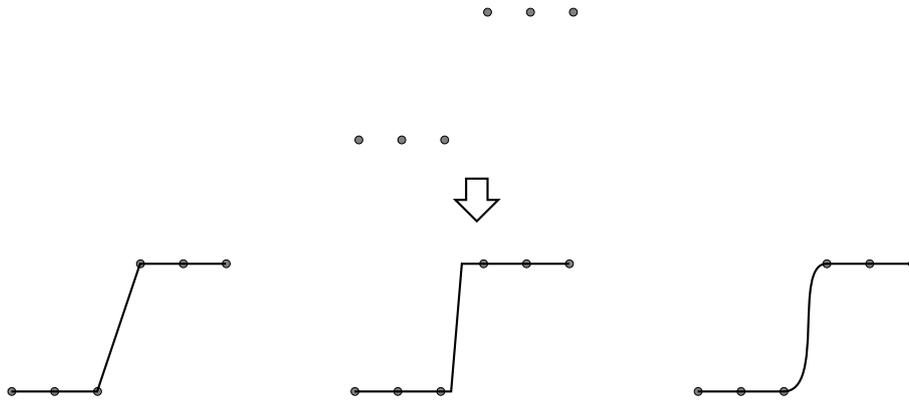


Рис. 1.4: Пример неоднозначной интерполяции ступенчатого контура на изображении. Все три способа являются допустимыми.

ратора для оператора $A_h = D_h H_{\sigma_0 h}$. Обратный оператор для оператора свёртки с фильтром Гаусса $H_{\sigma_0 h}$ является неограниченным, поэтому решение обратной задачи (1.13) является неустойчивым. В дискретной постановке (1.16) задача обращения свёртки является плохо обусловленной.

Помимо неустойчивости, отсутствует единственность решения, так как операторы D_h , D_s являются линейными операторами с

$$\det D_h = 0, \quad \det D_s = 0.$$

По изображению низкого разрешения можно построить несколько изображений высокого разрешения, причём все эти варианты будут допустимыми (см. рис.1.4). В связи с этим мы не будем требовать единственности решения задачи повышения разрешения изображений, однако потребуем компактность множества решений.

Вышеописанное верно и для задачи суперразрешения (1.19, 1.20). Но в отличие от задачи повышения разрешения изображений, всегда имеющей решение, решение задачи суперразрешения может отсутствовать в случае, если системы уравнений (1.19), (1.20) являются несовместными.

Таким образом, задачи повышения разрешения изображений и суперразрешения являются некорректно поставленными. Для нахождения решения задач повышения разрешения изображений и суперразрешения будем использовать регуляризирующие алгоритмы, базирующиеся на методе регуляризации А.Н. Тихонова [6].

1.3.1 Построение регуляризирующих методов решения задачи повышения разрешения изображений и суперразрешения

Рассмотрим решение операторного уравнения

$$Af = u \quad (1.21)$$

относительно f , где $f \in F$, $u \in U$. F и U — банаховы пространства с нормами $\|\cdot\|_F$ и $\|\cdot\|_U$ соответственно и метриками

$$\rho_F(f_1, f_2) = \|f_1 - f_2\|_F,$$

$$\rho_U(u_1, u_2) = \|u_1 - u_2\|_U,$$

оператор A — линейный непрерывный оператор, отображающий F на U .

Введём обозначения:

u_0 — точное значение правой части уравнения (1.21);

u_δ — приближённое значение правой части уравнения (1.21), причём $\rho_U(u_0, u_\delta) \leq \delta$;

$A^{-1}u = \{f \in F : Af = u\}$ — множество решений уравнения для правой части u ;

$A^{-1}u_0$ — множество точных решений уравнения при $u = u_0$;

$A^{-1}u_\delta$ — множество приближённых решений уравнения при $u = u_\delta$.

В случае единственности решения задача (1.21) называется устойчивой [6], если для любого $\varepsilon > 0$ существует $\delta > 0$ такое, что для любых допустимых значений правой части u_0 и u_δ таких, что $\rho_U(u_\delta, u_0) \leq \delta$ выполняется

$$\rho_F(A^{-1}u_\delta, A^{-1}u_0) < \varepsilon.$$

Для случая неединственности решения воспользуемся понятиями *бета-расстояния* и *бета-сходимости* для оценки отклонения множества решений $A^{-1}u_\delta$ для приближённой правой части от множества точных решений $A^{-1}u_0$ [31].

Определение 1. *Бета-расстоянием* между множествами M_1 и M_2 в метрическом пространстве называется величина

$$\beta(M_1, M_2) = \sup_{x \in M_1} \rho(x, M_2),$$

где

$$\rho(x, M) = \inf_{y \in M} \rho(x, y).$$

Определение 2. Последовательность множеств $\{M_n\}$ в метрическом пространстве называется *бета-сходящейся* к M при $n \rightarrow \infty$, если

$$\lim_{n \rightarrow \infty} \beta(M_n, M) = 0,$$

где

$$\beta(M_n, M) = \sup_{x_n \in M_n} \rho(x_n, M) = \sup_{x_n \in M_n} \left(\inf_{x \in M} \rho(x_n, x) \right).$$

Отметим, что если последовательность $\{M_n\}$ бета-сходится к некоторому множеству M , то она является бета-сходящейся к любому множеству M' , содержащему M .

Утверждение 1. Пусть множества M_n являются подмножествами компакта M_0 в банаховом пространстве, а множество M_P — множество всех предельных точек всевозможных последовательностей $\{x_n\}$, составленных из элементов множеств $M_n: x_n \in M_n$. Тогда последовательность $\{M_n\}$ является бета-сходящейся к M_P .

Доказательство. Докажем от противного. Пусть последовательность $\{M_n\}$ не является бета-сходящейся к M_P . Тогда последовательность $\{y_n\}$, где

$$y_n = \sup_{x_n \in M_n} \rho(x_n, M_P)$$

не сходится к нулю при $n \rightarrow \infty$, а значит, существует $\varepsilon > 0$, для которого можно выделить подпоследовательность $\{y_{n_k}\}$ такую, что

$$y_{n_k} = \sup_{x_{n_k} \in M_{n_k}} \rho(x_{n_k}, M_P) > \varepsilon$$

для всех k .

В каждом из множеств M_{n_k} возьмём элемент $x'_{n_k} \in M_{n_k}$ такой, что

$$\rho(x'_{n_k}, M_P) \geq \varepsilon.$$

Так как M_0 — компакт, то из последовательности $\{x'_{n_k}\}$ можно выделить сходящуюся подпоследовательность $\{x'_{n_{km}}\} \rightarrow \tilde{x}$, причём

$$\lim_{m \rightarrow \infty} \rho(x'_{n_{km}}, M_P) = \rho(\tilde{x}, M_P) \geq \varepsilon,$$

а значит, предельная точка \tilde{x} не принадлежит множеству M_P . Но по условию последовательность $\{x'_{n_{km}}\}$ является подпоследовательностью некоторой последовательности, составленной из элементов множеств M_n , и все её предельные точки, включая \tilde{x} , принадлежат M_P . Мы пришли к противоречию, значит последовательность $\{M_n\}$ является бета-сходящейся к M_P . Утверждение доказано. ■

Будем считать задачу (1.21) корректно поставленной, если для любого u_0 из множества допустимых значений правой части множество решений $A^{-1}u_0$ непусто и устойчиво к малым изменениям правой части, то есть для любого $\varepsilon > 0$ существует $\delta > 0$ такое, что

$$\beta(A^{-1}u_\delta, A^{-1}u_0) = \sup_{f \in A^{-1}u_\delta} \rho_F(f, A^{-1}u_0) < \delta$$

для любых u_δ таких, что $\rho_U(u_0, u_\delta) \leq \varepsilon$.

Предположим, что вместо точного значения правой части $u = u_0$ уравнения (1.21) известно его приближённое значение u_δ . Множество точных решений уравнения (1.21) не является устойчивым к малым изменениям u , поэтому нельзя использовать множество $A^{-1}u_\delta$ в качестве приближённого решения уравнения (1.21). Альтернативой этому является исполь-

зование метода регуляризации Тихонова, заключающегося в рассмотрении задачи, близкой к (1.21), решение которой обладает устойчивостью и бета-сходится к точному решению уравнения (1.21) при $\delta \rightarrow 0$.

Задачу нахождения приближённого решения для (1.21) можно поставить одним из следующих способов [25]:

I. Нахождение $\inf (\|Af - u_\delta\|_U + \lambda\Psi[f])$, $\lambda > 0$.

II. Нахождение $\inf \Psi[f]$ для $\|Af - u_\delta\|_U \leq \delta$, $\delta > 0$.

III. Нахождение $\inf \|Af - u_\delta\|_U$ для $\Psi[f] \leq C$, $C > 0$.

Также будем использовать следующую постановку:

Ia. Нахождение $\inf (\|Af - u_\delta\|_U^2 + \lambda\Psi[f])$, $\lambda > 0$.

Мы рассмотрим построение регуляризующего метода в постановке II, а затем применим теоремы эквивалентности для перехода к постановкам I, Ia и III.

Используем для уравнения (1.21) с неединственным решением понятие *регуляризующего оператора*

$$f_\delta = R(u_\delta, \delta), \quad u_\delta \in U, \quad f_\delta \in F,$$

обладающего свойствами:

1) существует такое число $\delta_1 > 0$, что оператор $R(u_\delta, \delta)$ определён для всякого δ , $0 \leq \delta \leq \delta_1$ и любого $u_\delta \in U$ такого, что

$$\rho_U(u_\delta, u_0) \leq \delta;$$

2) для всякого $\varepsilon > 0$ существует $\delta_0 = \delta_0(\varepsilon, u_\delta) \leq \delta_1$ такое, что из неравенства

$$\rho_U(u_\delta, u_0) \leq \delta \leq \delta_0$$

следует неравенство

$$\beta(f_\delta, A^{-1}u_0) = \sup_{f \in f_\delta} \rho_F(f, A^{-1}u_0) \leq \varepsilon.$$

Далее используем понятие *стабилизирующего функционала*, или *стабилизатора* — неотрицательного функционала $\Psi[f]$, определённого на

всюду плотном в F подмножестве F_1 множества F и удовлетворяющего условиям:

1) элементы $f \in A^{-1}u_0$ принадлежат его области определения;

2) для всякого числа $d > 0$ множество $F_{1,d} = \{f \in F_1 : \Psi[f] \leq d\}$ компактно в F .

Определим функционал $\tilde{R}(u_\delta, \delta)$. Пусть для некоторого u_δ существует непустое множество элементов \tilde{f}_δ , минимизирующих $\Psi[f]$ на множестве

$$Q_\delta = \{f : f \in F_1, \rho_U(Af, u_\delta) \leq \delta\}.$$

Множество f_δ можно рассматривать как применение к правой части уравнения (1.21) некоторого оператора \tilde{R} , зависящего от параметра δ , т.е.

$$\tilde{f}_\delta = \tilde{R}(u_\delta, \delta) = \arg \min_{f \in Q_\delta} \Psi[f] = \{f \in Q_\delta : \Psi[f] = \min_{f' \in Q_\delta} \Psi[f']\}. \quad (1.22)$$

Это множество является подмножеством компактного в F множества $F_{1,d}$ при $d = \min_{f \in Q_\delta} \Psi[f]$.

Предположим, что такое множество существует для любых $\delta \geq 0$ и $u_\delta \in U$. Тогда справедлива

Теорема 1. $\tilde{R}(u_\delta, \delta)$ (1.22) является регуляризирующим оператором для уравнения (1.21).

Доказательство теоремы аналогично доказательству соответствующей теоремы из [6] со следующими отличиями:

1) Вместо сходимости $\tilde{R}(u_\delta, \delta)$ к единственному точному решению уравнения $Az = u_0$ при $\delta \rightarrow 0$ происходит β -сходимость к множеству точных решений $A^{-1}u_0$.

2) Множества значений $\tilde{R}(u_\delta, \delta)$ являются подмножествами компактного в F множества

$$F_{1,0} = \{f \in F_1 : \Psi[f] \leq \min_{f_0 \in A^{-1}u_0} \Psi[f_0]\},$$

таким образом, $\tilde{R}(u_\delta, \delta)$ бета-сходится к компактному подмножеству множества точных решений $A^{-1}u_0 \cap F_{1,0}$.

Рассмотренный подход, заключающийся в минимизации стабилизатора $\Psi[f]$ на множестве приближённых решений $\rho_U(Af, u_\delta) < \delta$ (постановка II), неудобен с вычислительной точки зрения. Для задачи повышения разрешения изображений эффективнее ставить задачу регуляризации в виде минимизации регуляризирующего функционала (постановка I):

$$R(u_\delta, \lambda) = \arg \min_{f \in F_1} (\|Af - u_\delta\|_U + \lambda\Psi[f]), \quad (1.23)$$

где λ — параметр регуляризации.

Для случая единственности решения уравнения (1.21) эквивалентность регуляризирующих методов в вариантах постановки I, II и III достигается при выполнении следующих условий [25]:

1) Пространства F и U — нормированные, метрика порождена нормой пространства

$$\rho_U(u_1, u_2) = \|u_1 - u_2\|_U,$$

область определения $D(A)$ оператора A — линейное многообразие.

2) $\Psi[f]$ — выпуклый функционал с областью определения $D(A)$.

3) Для любого $f \in D(A)$, $f \neq 0$ функция $\gamma(\lambda) = \Psi[\lambda f]$ является строго возрастающей функцией параметра $\lambda \geq 0$.

Для случая неединственности решения уравнения (1.21) мы используем видоизменённое третье условие и накладываем дополнительное условие на регуляризирующий функционал:

3) Для любого $f \in D(A)$, $f \neq 0$, $\Psi[f] \neq 0$ функция $\gamma(\lambda) = \Psi[\lambda f]$ является строго возрастающей функцией параметра $\lambda \geq 0$.

4) Если множество $f_\lambda \subset F$ — решение задачи регуляризации в постановке I, то для любых $f_1, f_2 \in f_\lambda$ выполняется

$$\|Af_1 - u_\delta\|_U = \|Af_2 - u_\delta\|_U.$$

В этом случае эквивалентность регуляризирующих методов в постановках I и II достигается при выполнении условий теоремы:

Теорема 2. Пусть A — линейный оператор с областью определения F_1 ($\overline{F_1} = F$) и областью значений $AF_1 \subseteq U$, F и U — нормированные

пространства, выполнено неравенство

$$\|u_\delta - u_0\|_U < \delta, \quad 0 < \delta < \frac{1}{2}\|u_0\|_U, \quad u_0 \neq 0,$$

а также выполнены вышеописанные условия.

Тогда если f_δ — решение II, причём $\Psi[f_\delta] > 0$, то существует положительное число $\lambda_\delta > 0$ такое, что F_δ минимизирует функционал

$$M[f; u_\delta, \lambda_\delta] = \|Af - u\|_U + \lambda_\delta \Psi[f],$$

причём $\|Af_\delta - u_\delta\|_U = \delta$. И наоборот, если f_δ минимизирует функционал I для некоторого $\lambda_\delta > 0$ и $\delta = \|Af_\delta - u_\delta\|_U$, $\Psi[f_\delta] > 0$, то f_δ — решение II.

Доказательство теоремы аналогично доказательству соответствующей теоремы из [25].

Замечание. От условия 4 можно избавиться, если рассматривать эквивалентность постановок Ia и II. В этом случае на множестве f_δ выполняется

$$\|Af - u\|_U^2 + \lambda \Psi[f] \equiv \text{const.}$$

Отсюда следует, что квадрат нормы невязки и стабилизатор являются одновременно и выпуклыми, и вогнутыми функционалами на f_δ , а это возможно только в том случае, если они являются линейными функционалами на f_δ . Но квадратичный функционал является линейным только в том случае, если он — константа. Отсюда следует, что равенство

$$\|Af_1 - u_\delta\|_U = \|Af_2 - u_\delta\|_U$$

выполняется для любых $f_1, f_2 \in f_\delta$.

Таким образом, в случае выполнения условия 4, регуляризирующие методы для задачи повышения разрешения изображений в постановках Ia и II будут эквиваленты. В этом случае мы будем использовать постановку, наиболее удобную для минимизации регуляризирующего функционала.

Аналогично формулируется и доказывается теорема эквивалентности регуляризирующих методов в постановках Ia, II и III.

Применение метода регуляризации к задаче повышения разрешения изображений

Для задачи повышения разрешения изображений (1.13)

$$A_h f = D_h H_{\sigma_0 h} f = u, \quad f \in F, \quad u \in U$$

определим оператор

$$f_\lambda = R_h[u; \lambda] = \arg \min_{f \in F} (h^2 \|A_h f - u\|_U + \lambda \Psi[f]) \quad (1.24)$$

Здесь был добавлен нормировочный коэффициент h^2 к невязке, чтобы избавиться от зависимости λ от h .

Достаточным условием того, что $R_h[u; \lambda]$ является регуляризирующим оператором для задачи повышения разрешения изображений, является выполнение условий теорем 1 и 2. В частности, теорема 1 требует, чтобы $\Psi[f]$ являлся стабилизатором, для которого выполнено условие компактности множества $\{f : \Psi[f] \leq d\}$ для всех $d > 0$. Это условие не выполняется в пространстве $F = L(\mathbb{R}^2)$ для используемых в данной работе стабилизаторов и норм пространств F и U .

Наложим ограничения на пространства F и U . Размеры матрицы сенсоров камеры, с помощью которой регистрируются реальные изображения низкого разрешения, конечны. Получаемые с помощью камеры изображения \tilde{u} определены на конечномерной сетке

$$\tilde{\Omega}_h = \{(ih, jh) : 0 \leq i \leq N_x, 0 \leq j \leq N_y\}.$$

Значения пикселей изображения $u \in \ell(\Omega_h)$ с индексами (i, j) , лежащими за пределами прямоугольника $[0, N_x] \times [0, N_y]$, вычисляются с помощью чётного продолжения (1.2). Обозначим как $\tilde{\ell}(\Omega_h)$ пространство изображений u , полученных из $\tilde{u} \in \ell(\tilde{\Omega}_h)$ путём продолжения на сетку Ω_h .

В этом случае логично ввести предположение о том, что изображение высокого разрешения $f \in L(\mathbb{R}^2)$, соответствующее изображению низкого

разрешения из $\tilde{\ell}(\Omega_h)$, также является чётным продолжением изображения $\tilde{f} \in L([0, R_x] \times [0, R_y])$, заданного на прямоугольнике $[0, R_x] \times [0, R_y]$, $R_x = hN_x$, $R_y = hN_y$:

$$f(x + 2R_x p, y + 2R_y q) = \begin{cases} \tilde{f}(x, y) & \text{при } 0 \leq x \leq R_x, 0 \leq y \leq R_y, \\ \tilde{f}(2R_x - x, y) & \text{при } R_x \leq x < 2R_x, 0 \leq y < R_y, \\ \tilde{f}(x, 2R_y - y) & \text{при } 0 \leq x < R_x, R_y \leq y < 2R_y, \\ \tilde{f}(2R_x - x, 2R_y - y) & \text{при } R_x \leq x < 2R_x, R_y \leq y < 2R_y, \end{cases}$$

$p, q \in \mathbb{Z}$.

Обозначим пространство чётных продолжений изображений из $L([0, R_x] \times [0, R_y])$ на \mathbb{R}^2 как $\tilde{L}(\mathbb{R}^2)$.

Значения норм элементов в пространствах $\tilde{\ell}(\Omega_h)$ и $\tilde{L}(\mathbb{R}^2)$ определим через значения норм соответствующих им элементов из пространств $\ell(\tilde{\Omega}_h)$ и $L([0, R_x] \times [0, R_y])$:

$$\begin{aligned} \|u\|_{\tilde{\ell}(\Omega_h)} &= \|\tilde{u}\|_{\ell(\tilde{\Omega}_h)}, \\ \|f\|_{\tilde{L}(\mathbb{R}^2)} &= \|\tilde{f}\|_{L([0, R_x] \times [0, R_y])}. \end{aligned}$$

Использование пространств $F = \tilde{L}(\mathbb{R}^2)$ и $U = \tilde{\ell}(\Omega_h)$ вместо $L(\mathbb{R}^2)$ и $\ell(\Omega_h)$ для задачи повышения разрешения изображений (1.13) не накладывает ограничений на её практическое применение. В данных пространствах метод минимизации (1.24) принимает вид

$$\begin{aligned} f_\lambda &= \arg \min_{f \in F} (h^2 \|A_h f - u\|_U + \lambda \Psi[f]), \\ F &= \tilde{L}(\mathbb{R}^2), U = \tilde{\ell}(\Omega_h). \end{aligned} \tag{1.25}$$

В разделе 1.3.2 будет произведён выбор стабилизаторов и норм пространств F , U , при которых метод (1.25) является регуляризирующим для задачи повышения разрешения изображений.

Для нахождения приближённого решения задачи повышения разрешения изображений в дискретном случае (1.16) будем использовать сле-

дующий метод:

$$\begin{aligned} z_\lambda &= \arg \min_{z \in Z} (s^2 \|A_s z - u\|_U + \lambda \Psi[z]), \\ Z &= \tilde{\ell}(\Omega_{h/s}), U = \tilde{\ell}(\Omega_h). \end{aligned} \quad (1.26)$$

Применение метода регуляризации к задаче суперразрешения

Для задачи суперразрешения основной проблемой являются ошибки вычисления векторов движения, причём решения систем уравнений (1.19), (1.20) являются неустойчивыми к малым изменениям векторов движения. Также нельзя априори определить, являются ли системы уравнений (1.19), (1.20) совместными.

Приближённое решение задачи суперразрешения будем искать в виде решения задачи минимизации

$$f_\lambda = \arg \min_{f \in F} \left(\sum_{k=1}^K \frac{h^2}{K} \|A_h^k f - u\|_U + \lambda \Psi[f] \right). \quad (1.27)$$

в непрерывной постановке и

$$z_\lambda = \arg \min_{z \in Z} \left(\sum_{k=1}^K \frac{s^2}{K} \|A_s^k z - u\|_U + \lambda \Psi[z] \right). \quad (1.28)$$

в дискретной постановке [32–34].

Методы, основанные на минимизации (1.27) и (1.28), не являются регуляризирующими, тем не менее, их практическое применение даёт хорошие субъективные результаты.

1.3.2 Оптимизация параметров регуляризирующих методов

Следующим шагом является выбор стабилизатора Ψ и норм пространств F , Z , U для задач повышения разрешения изображений и суперразрешения и проверка условий того, что методы (1.25) и (1.26) являются регуляризирующими при данном выборе стабилизаторов и норм пространств.

Выбор норм пространств и стабилизатора

Мы используем нормы L_1 и L_2 :

$$\|f\|_1 = \int_0^{R_x} \int_0^{R_y} |f(x, y)| dx dy,$$

$$\|f\|_2^2 = \int_0^{R_x} \int_0^{R_y} |f(x, y)|^2 dx dy$$

и их дискретные аналоги

$$\|z\|_1 = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} |z_{i,j}|,$$

$$\|z\|_2^2 = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} |z_{i,j}|^2.$$

Это наиболее распространённый выбор норм при решении задач ре-самплинга. Мы выбираем норму, исходя из априорной информации о типе шума на изображении низкого разрешения. Так, в случае присутствия импульсного шума предпочтительнее использовать норму L_1 , а в случае шума с распределением, близким с гауссовскому — L_2 .

В работе рассматриваются следующие варианты выбора стабилизатора:

- 1) Квадратичный стабилизатор $\|\Delta f\|_2^2$.
- 2) Функционал полной вариации.
- 3) Функционал билатеральной полной вариации, представляющий собой адаптацию функционала полной вариации для задачи повышения разрешения изображений в дискретном случае.

Квадратичный стабилизатор

Квадратичные стабилизаторы [6] являются одними из наиболее часто используемых в методах регуляризации стабилизаторов. В наших расчётах

используется стабилизатор вида

$$\Psi[f] = \|\Delta f\|_2^2 = \int_0^{R_x} \int_0^{R_y} \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right)^2 dx dy.$$

Построим уравнение Эйлера для задачи минимизации (1.25) при $\Psi[f] = \|\Delta f\|_2^2$:

$$(h^2 A^* A + \lambda \Delta^* \Delta) f_\lambda = A^* u, \quad (1.29)$$

где сопряжённый к A оператор A^* определяется из условия $(Af, u) = (f, A^*u)$, для любых $f \in F$, $u \in U$. В силу линейности оператора A сопряжённый к нему оператор A^* будет линейным, а, значит, и оператор $(h^2 A^* A + \lambda \Delta^* \Delta)$ из (1.29) также будет линейным.

Это означает, что при выборе квадратичного стабилизатора и квадратичной нормы результату повышения разрешения изображений присущи артефакты линейных методов, а именно при больших λ возникает эффект размытия, а при малых λ появляется эффект Гиббса, проявляющийся в виде ореолов возле контуров объектов.

Функционал полной вариации

Важнейшей частью изображений являются контуры. Алгоритмы повышения разрешения изображений не должны удалять, либо добавлять контуры. Так, в линейных методах ресамплинга контуры теряются при размытии изображения, а новые контуры добавляются вследствие ложного оконтуривания и алиасинга. Для контроля этих артефактов будем использовать функционал полной вариации [35].

Полная вариация дифференцируемой функции $f(x) \in \tilde{L}(\mathbb{R}^2)$ определяется в виде

$$\|f\|_V = \int_0^{R_x} \int_0^{R_y} |\vec{\nabla} f(x, y)| dx dy,$$

где

$$|\vec{\nabla} f(x, y)| = \left(\left(\frac{\partial f(x, y)}{\partial x} \right)^2 + \left(\frac{\partial f(x, y)}{\partial y} \right)^2 \right)^{1/2}.$$

Для недифференцируемых функций полная вариация вычисляется в виде

$$\|f\|_V = \lim_{h \rightarrow 0} \int_0^{R_x} \int_0^{R_y} \left(\left(\frac{f(x+h, y) - f(x, y)}{h} \right)^2 + \left(\frac{f(x, y+h) - f(x, y)}{h} \right)^2 \right)^{1/2} dx dy.$$

В дискретном случае для $z \in \tilde{\ell}(\Omega_h)$ полная вариация определяется как

$$\|z\|_V = h \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \left((z_{i+1, j} - z_{i, j})^2 + (z_{i, j+1} - z_{i, j})^2 \right)^{1/2}.$$

С вычислительной точки зрения более эффективно использовать определение полной вариации в виде

$$\|z\|_{TV} = h \sum_{i=1}^{N_x} \sum_{j=0}^{N_y} |z_{i, j} - z_{i-1, j}| + h \sum_{i=0}^{N_x} \sum_{j=1}^{N_y} |z_{i, j} - z_{i, j-1}|. \quad (1.30)$$

Мы будем пользоваться этим определением полной вариации для задач повышения разрешения изображений и суперразрешения. Для него справедливо

$$\|z\|_V \leq \|z\|_{TV} \leq \sqrt{2} \|z\|_V,$$

так как $a^2 + b^2 \leq (|a| + |b|)^2 \leq 2(a^2 + b^2)$.

Важнейшим свойством полной вариации является её взаимосвязь с осцилляциями Гиббса, возникающими при действии идеального фильтра высоких частот [35]. Пусть $f_\xi = f * g_\xi$, где g_ξ — ядро идеального частотного фильтра со значениями преобразования Фурье

$$\hat{g}_\xi(w) = 1_{[-\xi, \xi]} = \begin{cases} 1, & |w| \leq \xi, \\ 0, & |w| > \xi. \end{cases}$$

Если $f(x) \in L_2(\mathbb{R}^2)$, то $\lim_{\xi \rightarrow +\infty} \|f - f_\xi\|_2 = 0$. Однако если $\|f\|_V < +\infty$, то $\|f_\xi\|_V = +\infty$ для любого $0 < \xi < \infty$. Таким образом, для предотвращения возникновения эффекта Гиббса можно наложить дополнительное условие на ограниченность полной вариации.

Теорема 3. Метод, основанный на минимизации (1.25), является регуляризирующим для стабилизатора $\Psi[f] = \|f\|_V$ в пространствах $F = \widetilde{L}_1(\mathbb{R}^2)$, $U = \widetilde{\ell}_1(\Omega_h)$ — пространствах $\widetilde{L}(\mathbb{R}^2)$ и $\widetilde{\ell}(\Omega_h)$ с нормами L_1 и ℓ_1 соответственно.

Доказательство. Для доказательства теоремы достаточно показать, что множество

$$M_C = \{f \in F : \|f\|_V \leq C, \|A_h f - u\|_U \leq \delta\}$$

является компактным в F для любого $C > 0$. Из ограниченности полной вариации следует ограниченность разности между наибольшим

$$\sup f = \sup_{[0, R_x] \times [0, R_y]} f(x, y)$$

и наименьшим

$$\inf f = \inf_{[0, R_x] \times [0, R_y]} f(x, y)$$

значениями функции f :

$$\sup f - \inf f \leq C.$$

Оператор A_h не увеличивает разность между наибольшим $\inf f$ и наименьшим $\sup f$ значениями функции f , поэтому

$$\inf f \leq [A_h f]_{i,j} \leq \sup f, \quad (i, j) \in \Omega_h,$$

откуда следует

$$0 \leq [A_h f]_{i,j} - \inf f \leq \sup f - \inf f \leq C, \quad (i, j) \in \Omega_h.$$

При вычислении нормы неравенство сохраняется

$$\begin{aligned} \|A_h f - \inf f\|_U &= \sum_{i,j} ([A_h f]_{i,j} - \inf f) \leq \\ &\leq (\sup f - \inf f)(R_x + 1)(R_y + 1) \leq C(R_x + 1)(R_y + 1). \end{aligned}$$

С другой стороны,

$$|\inf f|(R_x + 1)(R_y + 1) = \|\inf f\|_U \leq \|A_h f - \inf f\|_U + \|A_h f\|_U,$$

откуда следует, что

$$|\inf f| \leq C + \frac{\|A_h f\|_U}{(R_x + 1)(R_y + 1)}.$$

Из условия $\|A_h f - u\|_U \leq \delta$ получаем $\|A_h f\| \leq \|u\|_U + \delta$. Таким образом

$$|\inf f| \leq C + \frac{\|u\|_U + \delta}{(R_x + 1)(R_y + 1)} = C_1.$$

Из $\sup f - \inf f \leq C$ получаем неравенство для $\sup f$:

$$|\sup f| \leq C_1 + C.$$

Отсюда следует, что значения функции f ограничены константой

$$C_2 = \max(C_1, C_1 + C) = C_1 + C,$$

значит множество M_C является ограниченным.

Для доказательства компактности M_C воспользуемся теоремой Арцела [36], построив конечную ε -сеть множества M_C . Разобьём $[0, R_x] \times [0, R_y]$ на $N \times N$ прямоугольников

$$\begin{aligned} P_{i,j} &= \left[\frac{(i-1)R_x}{N}, \frac{iR_x}{N} \right) \times \left[\frac{(j-1)R_y}{N}, \frac{jR_y}{N} \right), \quad 1 \leq i, j < N, \\ P_{N,j} &= \left[\frac{(N-1)R_x}{N}, R_x \right] \times \left[\frac{(j-1)R_y}{N}, \frac{jR_y}{N} \right), \quad 1 \leq j < N, \\ P_{i,N} &= \left[\frac{(i-1)R_x}{N}, \frac{iR_x}{N} \right) \times \left[\frac{(N-1)R_y}{N}, R_y \right], \quad 1 \leq i < N, \\ P_{N,N} &= \left[\frac{(N-1)R_x}{N}, R_x \right] \times \left[\frac{(N-1)R_y}{N}, R_y \right] \end{aligned}$$

и запишем функционал полной вариации в виде суммы полных вариаций на этих прямоугольниках:

$$\|f\|_V = \sum_{i=1}^N \sum_{j=1}^N \|f\|_{V, P_{i,j}}.$$

Рассмотрим кусочно-постоянную функцию

$$g(x, y) = \inf_{(x', y') \in P_{i,j}} f(x', y'), \quad \text{для } (x, y) \in P_{i,j}.$$

Для неё будет справедливо $\|f - g\|_{V, P_{i,j}} = \|f\|_{V, P_{i,j}}$ на каждом отдельном прямоугольнике $P_{i,j}$. Вычислим $\|f - g\|_F$ на всём множестве:

$$\begin{aligned} \|f - g\|_F &= \int_0^{R_x} \int_0^{R_y} |f(x, y) - g(x, y)| dx dy = \\ &= \sum_{i=1}^N \sum_{j=1}^N \int_{P_{i,j}} |f(x, y) - g(x, y)| dx dy = \\ &= \sum_{i=1}^N \sum_{j=1}^N \int_{P_{i,j}} (f(x, y) - \inf_{P_{i,j}} f(x, y)) dx dy \leq \\ &\leq \frac{R_x R_y}{N^2} \sum_{i=1}^N \sum_{j=1}^N (\sup_{P_{i,j}} f(x, y) - \inf_{P_{i,j}} f(x, y)) \leq \\ &\leq \frac{R_x R_y}{N^2} \sum_{i=1}^N \sum_{j=1}^N \|f\|_{V, P_{i,j}} = \frac{R_x R_y}{N^2} \|f\|_V = \frac{R_x R_y C}{N^2}. \end{aligned}$$

Выберем N таким, чтобы

$$\frac{R_x R_y C}{N^2} \leq \frac{\varepsilon}{2}.$$

Ранее было показано, что значения f ограничены по модулю значением C_2 . Определим конечное множество значений

$$\frac{C_2 m}{M}, \quad m = 0, \pm 1, \dots, \pm M,$$

которые могут принимать кусочно-постоянные функции. При выборе M таким, чтобы

$$\frac{R_x R_y C_2}{2M + 1} \leq \frac{\varepsilon}{2},$$

норма разности функции g и ближайшей кусочно-постоянной функцией с дискретными значениями не будет превышать $\frac{\varepsilon}{2}$.

Таким образом, мы получили ε -сеть из функций, покрывающих M_C . Значит, множество M_C является компактным, а метод минимизации (1.22)

со стабилизатором $\Psi[f] = \|f\|_V$ — регуляризирующим. Теорема доказана. ■

Следствие 1. Теорема справедлива и для пространств $F = \widetilde{L}_2(\mathbb{R}^2)$, $U = \widetilde{\ell}_2(\Omega_h)$ с квадратичной нормой, а также для стабилизатора $\Psi[f] = \|f\|_{TV}$.

При использовании в качестве стабилизатора функционала полной вариации удобно использовать регуляризирующий метод в постановке III:

$$f_C = \arg \min_{\|f\|_V \leq C} \|A_h f - u\|_U. \quad (1.31)$$

В отличие от постановок I и II, где регуляризирующие параметры λ и δ априори неизвестны, в постановке III есть возможность оценить значение C вследствие связи полной вариации с суммарной длиной линий уровня функции f .

Обозначим

$$\Theta_t = \{(x, y) \in \mathbb{R} : f(x, y) > t\}.$$

Если $f(x, y)$ непрерывна, то граница $\partial\Theta_t$ множества Θ_t является линией уровня всех точек (x, y) таких, что $f(x, y) = t$. Пусть $H^1(\Theta_t)$ — длина $\partial\Theta_t$. Справедлива теорема [37], что если $\|f\|_V < +\infty$, тогда

$$\|f\|_V = \int_{-\infty}^{+\infty} H^1(\Theta_t) dt.$$

При понижении разрешения изображений теряются мелкие детали изображения, поэтому полная вариация изображения низкого разрешения будет ниже полной вариации изображения высокого разрешения, из которого было получено изображение низкого разрешения. Однако при повышении разрешения изображений нет возможности восстановить эти детали, поэтому поставим условие, чтобы полная вариация изображения высокого разрешения была равна полной вариации изображения низкого разрешения:

$$C = \|u\|_V.$$

Функционал билатеральной полной вариации

В дискретном случае использование (1.30) в качестве полной вариации даёт лишь приближённую информацию о полной вариации изображения. Для повышения точности будем использовать функционал билатеральной полной вариации [1]:

$$\|z\|_{BTV} = h \sum_{s,t=-p}^p \left(\gamma_{s,t} \sum_{i,j} |z_{i+s,j+t} - z_{i,j}| \right),$$

вычисляющий взвешенную сумму разностных производных на изображении z . Параметр p ограничивает количество направлений. Чем больше значение p , тем выше точность метода, но и выше его вычислительная сложность. Параметр $\gamma_{s,t}$ задаёт веса для каждого из направлений. В данной работе используется значение $\gamma_{s,t} = 0.8^{|s|+|t|}$.

Анализ выбора стабилизатора и нормы для задачи повышения разрешения изображений

Регуляризирующий метод повышения разрешения изображений был протестирован на базе из 124 фотографий, включающих в себя изображения природы и зданий. Для оценки качества метода эталонные изображения из базы были уменьшены в некоторое число раз, используя модель (1.15), затем увеличены с помощью разработанного регуляризирующего метода до исходного размера и сравнены с эталонными изображениями с помощью метрики структурного подобия *SSIM* [38]:

$$SSIM(u, v) = \frac{(2\mu_u\mu_v + c_1)(2\sigma_{uv} + c_2)}{(\mu_u^2 + \mu_v^2 + c_1)(\sigma_u^2 + \sigma_v^2 + c_2)}, \quad (1.32)$$

где μ_u и μ_v — средние значения u и v соответственно, σ_u^2 и σ_v^2 — дисперсии, σ_{uv} — ковариация u и v . Значения $c_1 = (0,01L)^2$, $c_2 = (0,03L)^2$, где L — динамический диапазон интенсивностей. Для компьютерных изображений с 8-битным значением интенсивности используется $L = 255$. Данная метрика хорошо коррелирует с восприятием человека [38].

Был проведён экспериментальный анализ выбора наилучшей комбинации нормы пространств Z , U , F и стабилизатора Ψ для увеличения изображений в 2, 3 и в 4 раза с использованием данной метрики [39]. Для каждого изображения значение параметра регуляризации λ выбиралось таким, чтобы значение $SSIM$ было максимальным. В качестве критерия оценки качества результатов для конкретного выбора нормы и стабилизатора использовалось среднее значение $SSIM$ по всей базе при наилучшем выборе λ .

Также анализ был проведён для случая зашумлённых изображений низкого разрешения. Был использован шум с нормальным распределением со стандартным отклонением 10.

Увеличение	в 2 раза		в 3 раза		в 4 раза	
Норма	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$
Стабилизатор	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$
$\ \Delta z\ _2^2$	0,9849	0,9852	0,9732	0,9738	0,9652	0,9653
$\ z\ _{TV}$	0,9849	0,9844	0,9731	0,9731	0,9654	0,9649
$\ z\ _{BTV}(p = 1)$	0,9850	0,9845	0,9734	0,9736	0,9658	0,9654
$\ z\ _{BTV}(p = 2)$	0,9839	0,9829	0,9736	0,9737	0,9658	0,9655
$\ z\ _{BTV}(p = 3)$	0,9833	0,9820	0,9740	0,9739	0,9659	0,9655
Среднее	0,9844	0,9838	0,9735	0,9736	0,9656	0,9653

Таблица 1.1: Средние значения $SSIM$ при использовании различных норм пространств и стабилизаторов при повышении разрешения незашумлённых изображений в 2, 3 и в 4 раза.

В таблицах 1.1 и 1.2 приведены средние значения $SSIM$ для используемых нами норм пространств и стабилизаторов для различных коэффициентов масштабирования. На основе полученных результатов можно сделать следующие наблюдения:

1. Для случая незашумлённых изображений нормы $\|\cdot\|_1$ и $\|\cdot\|_2$ дают практически одинаковые значения $SSIM$ с незначительным преимуществом $\|\cdot\|_1$, а для зашумлённых наилучшие результаты показывает норма $\|\cdot\|_2$. Использование нормы $\|\cdot\|_3$ даёт неудовлетворительные результаты (в таблице не показаны).

Увеличение	в 2 раза		в 3 раза		в 4 раза	
Норма	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$
Стабилизатор						
$\ \Delta z\ _2^2$	0,9614	0,9704	0,9454	0,9565	0,9262	0,9449
$\ z\ _{TV}$	0,9692	0,9732	0,9538	0,9603	0,9451	0,9492
$\ z\ _{BTV}(p=1)$	0,9687	0,9732	0,9556	0,9607	0,9458	0,9499
$\ z\ _{BTV}(p=2)$	0,9683	0,9711	0,9548	0,9600	0,9448	0,9495
$\ z\ _{BTV}(p=3)$	0,9683	0,9687	0,9546	0,9596	0,9444	0,9491
Среднее	0,9672	0,9713	0,9528	0,9594	0,9413	0,9485

Таблица 1.2: Средние значения $SSIM$ при использовании различных норм пространств и стабилизаторов при повышении разрешения зашумлённых изображений в 2, 3 и в 4 раза.

2. Для коэффициента масштабирования $s = 2$ и нормы $\|\cdot\|_2$ наилучшие результаты достигаются при использовании стабилизаторов $\|z\|_V$, $\|z\|_{TV}$, $\|z\|_{BTV}(p=1)$.

3. Для коэффициентов масштабирования $s = 3, 4$ и нормы $\|\cdot\|_2$ все рассмотренные стабилизаторы дают примерно одинаковые результаты. С вычислительной точки зрения эффективнее использовать стабилизаторы с наименьшим числом операций. Такими стабилизаторами являются $\|z\|_{TV}$ и $\|z\|_{BTV}(p=1)$.

Обобщая эти наблюдения, приходим к выводу: для повышения разрешения изображений с помощью разработанного регуляризирующего метода предпочтительнее использование нормы $\|\cdot\|_2$ и стабилизатора $\|z\|_{BTV}$ при $p=1$.

Анализ выбора стабилизатора и нормы для задачи суперразрешения

Аналогичный анализ был произведён для задачи суперразрешения на изображениях из той же базы. Для каждого эталонного изображения высокого разрешения z было сгенерировано 32 изображения низкого разрешения u_k в соответствии с моделью (1.18) с операторами движения T_k , состоящими из случайного сдвига в пределах ± 5 пикселей по горизонтали и вертикали и случайного поворота в пределах ± 10 градусов.

Основной проблемой в задаче суперразрешения являются ошибки вычисления операторов движения T_k , поэтому после моделирования изображений низкого разрешения к операторам T_k добавлялась ошибка. Для четверти операторов T_k ошибка представляла собой равномерно распределённое случайное смещение каждого пикселя в интервале $[-n_0, n_0]$, для остальных операторов ошибка находилась в интервале $[-n_0/4, n_0/4]$.

Тестирование методов проводилось как в условиях недостатка данных, когда число неизвестных в уравнении (1.20) больше числа уравнений, то есть при $s^2 > K$, так и в условиях избытка данных $s^2 < K$. Увеличение производилось в 2, 3 и 4 раза при числе изображений низкого разрешения $s^2/2$, s^2 и $2s^2$. Рассматривалось три уровня шума: $n_0 = 0.1, 0.5, 1$.

Полученные результаты были аналогичны результатам для задачи повышения разрешения одиночного изображения.

1.3.3 Регуляризирующий метод нахождения соответствий между изображениями в задаче суперразрешения

Операторы движения T_k (1.17) в задачах суперразрешения (1.19), (1.20) не являются известными и должны быть предварительно вычислены.

Задача построения оператора движения $T = (T_x, T_y)$ для пары изображений u и v заключается в минимизации $\|Tu - v\|_2^2$ при наложении дополнительных ограничений на оператор T . Наиболее часто в качестве такого ограничения используется предположение, что изображение состоит из участков, на которых движение одинаково во всех пикселях.

Часто задача оценки движения ставится в виде:

$$T_\lambda = \arg \min_T (\|Tu - v\|_2^2 + \lambda \Psi[T]),$$

где $\Psi[T]$ — ограничение, накладываемое на вектора движения T_x, T_y .

Функционал невязки $\|Tu - v\|_2^2$ не является выпуклым, поэтому его использование на практике приводит к неудовлетворительным результа-

там. Для устранения этого недостатка используется вариационный подход [40, 41]. Пусть u, v — дифференцируемые функции, а значения векторов движения T_x, T_y достаточно малы. Тогда справедливо

$$\begin{aligned} v(x, y) &= u(x + T_x(x, y), y + T_y(x, y)) = \\ &= u(x, y) + u_x(x, y)T_x(x, y) + u_y(x, y)T_y(x, y) + o(T_x, T_y). \end{aligned}$$

Отсюда следует, что

$$u_x(x, y)T_x(x, y) + u_y(x, y)T_y(x, y) = v(x, y) - u(x, y) = w(x, y).$$

Левая часть уравнения является линейным оператором $M_u[T]$. Задачу нахождения оператора движения поставим в виде нахождения минимума регуляризирующего функционала

$$T_\lambda = \arg \min_T (\|M_u[T] - w\|_2^2 + \alpha(\Psi[T])).$$

В качестве стабилизатора мы используем $\Psi[T] = \|T_x\|_{TV} + \|T_y\|_{TV}$.

1.3.4 Неитерационный метод решения задачи суперразрешения

Применение регуляризирующих методов для решения задачи суперразрешения (1.28) также не всегда является приемлемым из-за высоких вычислительных затрат. В связи с этим возникает задача быстрого нахождения приближённого решения задачи суперразрешения.

Рассмотрим оператор понижения разрешения (1.20)

$$A_s^k z = D_s H_{\sigma_0 \sqrt{s^2 - 1}} T_k, \quad k = 1, 2, \dots, K,$$

и оператор

$$\bar{A}_s^k z = D_s T_k H_{\sigma_0 \sqrt{s^2 - 1}}, \quad k = 1, 2, \dots, K.$$

Операторы T_k и $H_{\sigma_0 \sqrt{s^2 - 1}}$ не являются коммутирующими, поэтому $A_s^k \neq \bar{A}_s^k$. В частном случае, когда T_k является комбинацией сдвига и поворота, эти операторы равны $A_s^k = \bar{A}_s^k$.

Предположим, что значения векторов движения оператора T_k изменяются достаточно слабо в пространстве, то есть в небольшой окрестности

каждого пикселя движение можно считать близким к комбинации поворота и сдвига. В этом случае будем искать приближённое решение задачи суперразрешения как решение системы уравнений

$$\overline{A}_s^k z' = D_s T_k z' = u^k, \quad k = 1, 2, \dots, K \quad (1.33)$$

с последующим повышением резкости изображения

$$H_{\sigma_0 \sqrt{s^2 - 1}} z = z'.$$

Найдём соответствие между координатами пикселей размытого изображения высокого разрешения z' и изображения низкого разрешения u_k :

$$u_{i,j}^k = [D_s T_k z']_{i,j} = [T_k z'](si, sj) = z'(si + T_x^k(si, sj), sj + T_y^k(si, sj)).$$

Обозначим

$$\begin{aligned} x_{i,j}^k &= si + T_x^k(si, sj), \\ y_{i,j}^k &= sj + T_y^k(si, sj), \\ w_{i,j}^k &= u_{i,j}^k. \end{aligned}$$

В этом случае систему уравнений (1.33) можно записать в виде

$$\begin{aligned} z'(x_{i,j}^k, y_{i,j}^k) &= w_{i,j}^k, \quad k = 1, 2, \dots, K, \\ i &= 0, 1, \dots, N_x, \quad j = 0, 1, \dots, N_y, \end{aligned}$$

Переходя от трёхмерной индексации i, j, k к одномерной, получаем

$$z'(x_m, y_m) = w_m, \quad m = 1, 2, \dots, M.$$

Таким образом, мы пришли к задаче построения изображения z' при известных значениях пикселей с координатами, вообще говоря, не являющимися узлами сетки изображения z . Изображение z' можно построить с помощью гауссовской интерполяции [42, 43]:

$$z'_{i,j} = \frac{\sum_m w_m G_\sigma(i - x_m, j - y_m)}{\sum_m G_\sigma(i - x_m, j - y_m)}, \quad (1.34)$$

где

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}.$$

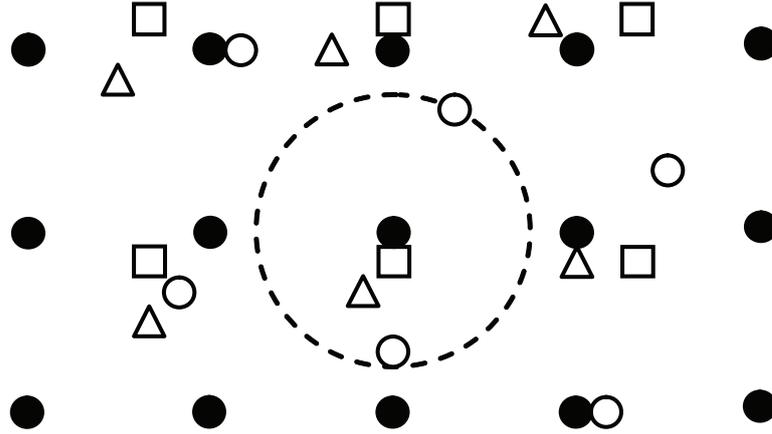


Рис. 1.5: Иллюстрация алгоритма выбора пикселей изображения низкого разрешения, используемых для интерполяции значения пикселя изображения высокого разрешения. Пиксели изображений низкого разрешения показаны как \square , \circ и \triangle . Пиксели изображения высокого разрешения обозначены как \bullet .

Параметр фильтра Гаусса σ выбирается экспериментально в зависимости от коэффициента увеличения s и точности задания векторов движения. Ядро фильтра Гаусса $G_\sigma(x, y)$ имеет бесконечный носитель, но при больших значениях x, y он принимает пренебрежимо малые значения, что позволяет использовать конечный носитель [44]. Мы будем использовать для интерполяции только пиксели, находящиеся в круговой окрестности интерполируемого пикселя (см. рис. 1.5) с радиусом $R = 2\sigma$.

В [45] рассмотрено использование медианной фильтрации для суперразрешения, в [46] предложен алгоритм, заключающийся в медианном усреднении всех пикселей низкого разрешения, попавших в рассмотренную круговую окрестность.

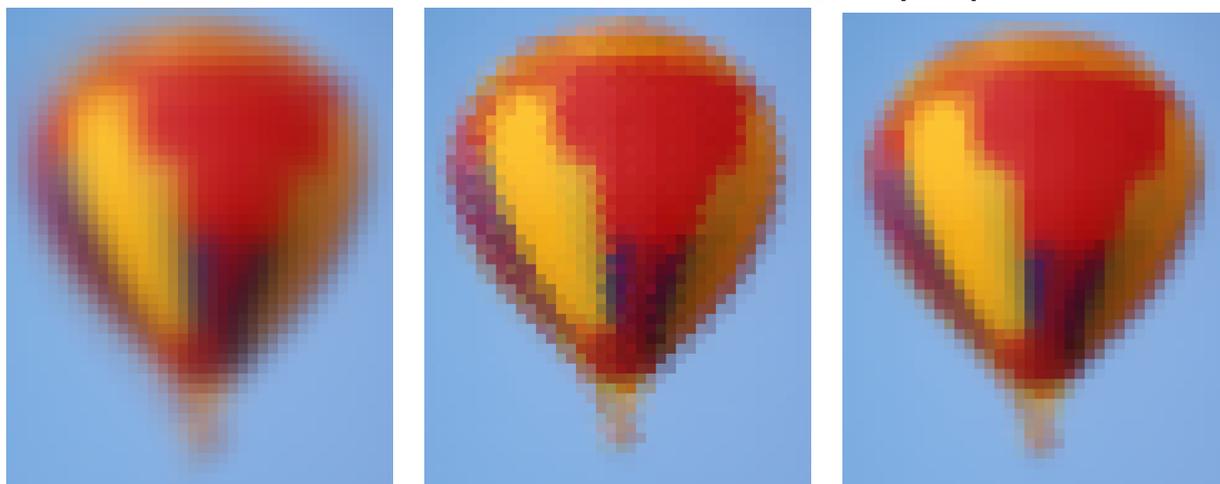
В случае точно известных векторов движения, метод, основанный на гауссовской фильтрации, и метод, основанный на медианном усреднении, дают хорошие результаты при небольших значениях R и σ . В случае неточно найденных векторов движения на изображении появляются артефакты, с которыми можно бороться, увеличивая R и σ , при этом изображение, полученное с помощью гауссовского усреднения, становит-

ся сильно размытым, тогда как контуры на изображении, полученном с помощью медианного усреднения значений пикселей в круговой окрестности, остаются резкими, но неровными.

Альтернативой этим двум подходам является применение взвешенной медианной фильтрации [47, 48], сочетающей в себе свойства как гауссовской, так и медианной фильтрации. Взвешенная медианная фильтрация с целыми весами, при которой значение пикселя берётся определённое число раз, равное его весу, применяется для фильтрации изображений [49]. В нашем случае используется взвешенная медианная фильтрация с вещественными весами, равными весам в гауссовском усреднении (1.34).

Примеры применения неитерационных методов суперразрешения изображения в случае неточно вычисленных векторов движения приведены на рис. 1.6. Для изображения высокого разрешения на основе модели (1.20) было построено 16 изображений низкого разрешения с коэффициентом масштабирования $s = 2$, где операторами движения являлись случайные сдвиги изображения. Затем к вектору движения для каждого пикселя каждого оператора движения было добавлено некоторое случайное значение, и были применены рассмотренные неитерационные методы суперразрешения. Параметры методов σ и R выбирались так, чтобы значение $SSIM$ было наибольшим.

Результаты при добавлении к векторам движения равномерно
распределённой случайная величина из отрезка $[-2, 2]$:

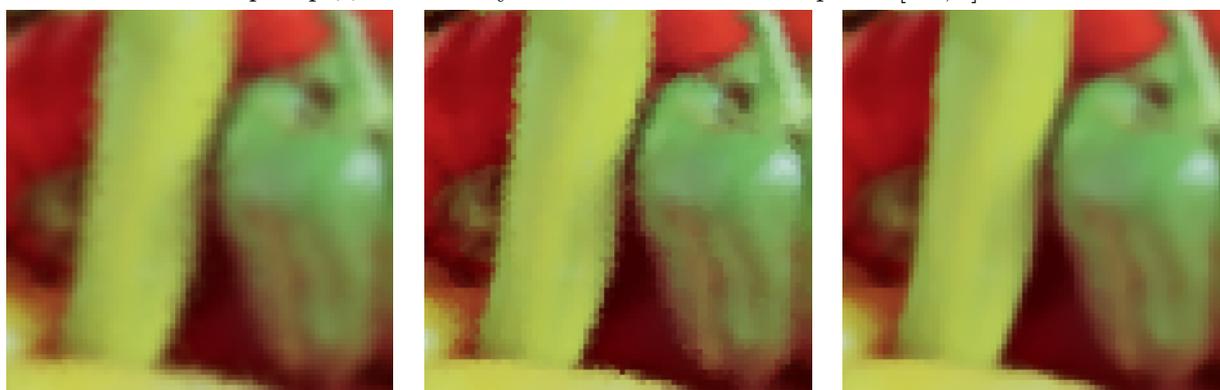


гауссовское усреднение,
 $\sigma = 0,8$

медианная фильтрация,
 $R = 1,2$

взвешенная медианная
фильтрация, $\sigma = 0.8$.

Результаты при добавлении к векторам движения равномерно
распределённой случайная величина из отрезка $[-1, 1]$:



гауссовское усреднение,
 $\sigma = 0,8$

медианная фильтрация,
 $R = 1,2$

взвешенная медианная
фильтрация, $\sigma = 1.1$

Рис. 1.6: Примеры применения неитерационных методов суперразрешения изображения в случае неточно вычисленных векторов движения для коэффициента увеличения $s = 2$ и 16 изображений низкого разрешения.

Глава 2

Методы анализа и постобработки для задачи повышения разрешения изображений

Методы повышения разрешения изображений вносят артефакты в получаемые изображения высокого разрешения, такие как эффект Гиббса, размытие, ступенчатость контуров. Данная глава посвящена оценке качества изображений, обнаружению и подавлению эффекта Гиббса и размытия на изображениях, возникающих после повышения разрешения.

В первом разделе главы описывается метод оценки качества алгоритмов повышения разрешения изображения, основанный на нахождении на эталонном изображении двух областей, в которых наиболее сильно проявляются артефакты алгоритмов повышения разрешения изображений: множества точек базовых контуров и окрестности множества точек базовых контуров. Рассматривается применение адаптивных метрик для оценки качества изображения по отдельности в каждом из этих множеств, а также применение полученного метода оценки качества алгоритмов для построения метода повышения разрешения изображений, комбинирующего результаты нескольких алгоритмов повышения разрешения изображений.

Во втором и третьем разделах рассматриваются задачи обнаружения и подавления артефактов, наиболее часто возникающих в результате по-

вышения разрешения изображений — эффекта Гиббса и размытия. Для обнаружения эффекта Гиббса производится многомасштабный анализ полной вариации профилей контуров при различных параметрах эффекта Гиббса. На основании этого анализа строится критерий определения наличия эффекта Гиббса на изображении по профилям базовых контуров. Подавление эффекта Гиббса после повышения разрешения изображений производится с помощью метода проекции на множество изображений с ограниченной полной вариацией.

2.1 Метрики для базовых контуров на изображениях

Для того, чтобы объективно оценивать качество изображений, используются метрики, в численном виде оценивающие близость изображения к эталонному. Основной проблемой при выборе метрики является требование соответствия численной оценки, выдаваемой метрикой, субъективному качеству изображения. Популярные метрики, основанные на вычислении среднеквадратичного отклонения, плохо коррелируют с визуальным восприятием изображения: так, незначительные изменения средней интенсивности и контраста незаметны человеческому глазу, тогда как они сильно влияют на значение метрики. Типичным представителем подобных метрик является пиковое отношение сигнала к шуму (peak signal-to-noise ratio, PSNR):

$$\rho_{PSNR}(u, v) = 10 \log_{10} \frac{255^2(N_x + 1)(N_y + 1)}{\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} (u_{i,j} - v_{i,j})^2},$$

где $(N_x + 1)$ и $(N_y + 1)$ — ширина и высота изображения соответственно. Данная метрика определена, только если изображения u и v являются различными.

Наибольшую корреляцию с субъективной оценкой имеют адаптивные методы, основанные на моделировании визуальной системы человека [50, 51], учитывающие особенности человеческого восприятия. Данные

модели редко используются на практике из-за их сложности и дополнительных требований к положению наблюдателя.

Промежуточную позицию занимает описанная в предыдущем разделе метрика структурного подобия SSIM (1.32), сочетающая простоту вычисления с хорошей связью с субъективным восприятием изображений.

Вышеописанные метрики являются универсальными метриками, позволяющими оценить близость между двумя произвольными изображениями. Для оценки качества изображений, полученных с помощью определённых алгоритмов обработки изображений, применяются специальные метрики, ориентированные, в первую очередь, на обнаружение и оценку характерных для данных алгоритмов обработки изображений артефактов. Данные метрики вычисляют близость реконструированного изображения к эталонному. Например, для оценки качества изображений с помощью алгоритма JPEG2000 применяются метрики обнаружения эффекта Гиббса и размытия [52], аналогичные метрики применяются для оценки качества изображений в задаче обращения свёртки [53].

Задача повышения разрешения изображений заключается в восстановлении потерянной при понижении разрешения высокочастотной информации. Алгоритмы повышения разрешения изображений производят восстановление потерянной высокочастотной информации, используя сохранившуюся низкочастотную информацию, но при этом восстановленная высокочастотная информация отличается от исходной. Искажения в высокочастотной информации проявляются, в основном, возле контуров на изображениях в виде эффекта Гиббса и размытия контуров.

Если удалить из изображения всю информацию с частотой, большей $\frac{1}{2d_0}$ Гц, например, обнулив соответствующие коэффициенты при преобразовании Фурье, то на изображении возникнет эффект Гиббса, а контуры будут размыты. При этом ширина одиночной осцилляции эффекта Гиббса и ширина контура будут равны d_0 . Пример эффекта Гиббса и размытия при пороговом удалении высокочастотной информации приведён

на рис. 2.1.

Такой же результат получается при повышении разрешения изображений с помощью идеальной (1.6) интерполяции

$$f(x, y) = \sum_{i,j} u_{i,j} \operatorname{sinc} \left(\frac{x}{h} - i \right) \operatorname{sinc} \left(\frac{y}{h} - j \right),$$

$$f \in L(\mathbb{R}^2), u \in \ell(\Omega_h).$$

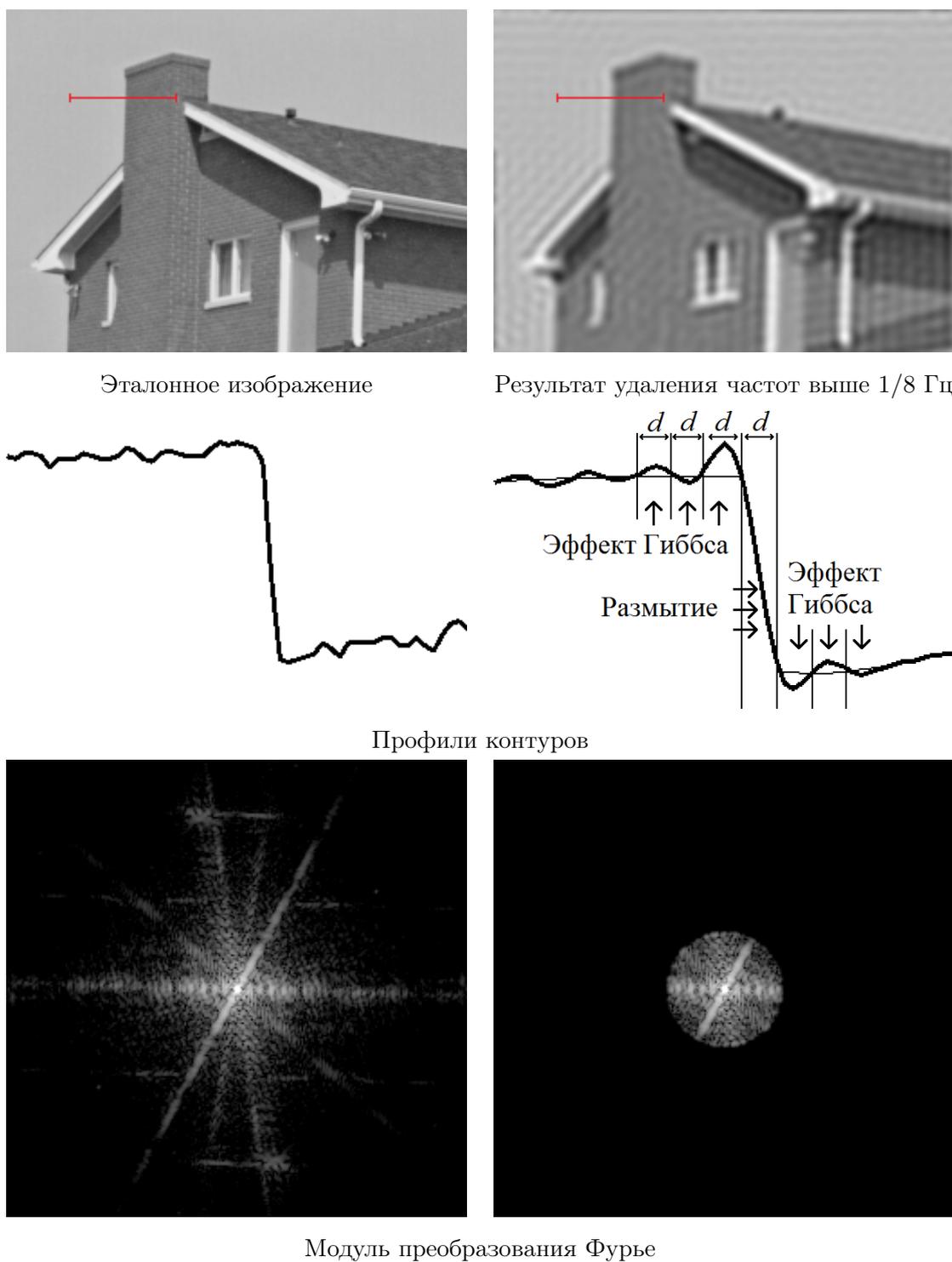
Параметр d_0 мы будем называть *коэффициентом отсечения*.

При использовании других алгоритмов повышения разрешения изображений производится восстановление высокочастотной информации, поэтому эффект Гиббса становится менее выраженным или же полностью отсутствует. При этом, если эффект Гиббса присутствует, то область его локализации не зависит от используемого метода повышения разрешения изображений, а коэффициент отсечения зависит только от шага сетки изображения низкого разрешения: $d_0 = h$. При ресамплинге с коэффициентом масштабирования s удобно шаг сетки изображения высокого разрешения положить равным 1: $\frac{h}{s} = 1$. В этом случае коэффициент отсечения равен коэффициенту масштабирования $d_0 = s$ [54].

Таким образом, на изображении, полученном с помощью алгоритмов повышения разрешения, можно выделить две области: область, где возможно появление эффекта Гиббса, и область, где возможно размытие контуров. Эти области не зависят от выбора метода повышения разрешения изображений, а зависят только от самого изображения и коэффициента масштабирования s .

Помимо повышения разрешения изображений, задача восстановления высокочастотной информации возникает при подавлении эффекта Гиббса и при повышении резкости изображений. В [55] рассмотрена задача построения метрики оценки качества изображений после повышения резкости для гауссовского размытия, при этом в качестве коэффициента отсечения использовалось значение радиуса фильтра Гаусса $d_0 = \sigma$.

В данном разделе рассматривается построение эталонной метрики оцен-



Эталонное изображение

Результат удаления частот выше 1/8 Гц

Профили контуров

Модуль преобразования Фурье

Рис. 2.1: Пример размытия и эффекта Гиббса на изображении при пороговом удалении высокочастотной информации.

ки качества изображений после применения алгоритмов восстановления высокочастотной информации на изображениях. Входными данными для метрики является тройка (z, z_0, d_0) , где z_0 — эталонное изображение, z — реконструированное изображение. Результатом метрики является набор значений, оценивающих качество изображения в целом, а также качество изображения в области, соответствующей возможному эффекту Гиббса, и области, соответствующей возможному эффекту размытия.

2.1.1 Метод нахождения базовых контуров на изображении и их окрестностей

Рассмотрим задачу нахождения областей на эталонном изображении, соответствующих артефактам эффекта Гиббса и размытия при известном коэффициенте отсечения d_0 . Будем искать эти области в районе контуров — перепадов интенсивности на изображениях.

Не все контуры на изображении подходят для анализа качества. Мы будем искать контуры на эталонном изображении высокого разрешения, которые сохраняются при деградации изображения (понижение разрешения, размытие, пороговое отсечение высоких частот), и его последующего восстановления.

Мы используем для анализа только контуры, удовлетворяющие следующим условиям:

1) Значения модуля градиента в пикселях контура должны быть выше определённого порога. Данное требование обусловлено наличием шума на изображении.

2) Отсутствие поблизости других контуров с близким или значительно большим модулем градиента. Например, при потере высокочастотной информации контуры с низким модулем градиента могут быть маскированы близкими контурами с большим модулем градиента, тогда как близко расположенные контуры с примерно одинаковым модулем градиента могут либо сместиться ближе друг к другу, либо, наоборот, удалить-

ся друг от друга. Также близко расположенные контуры не подходят для анализа эффекта Гиббса из-за взаимного наложения осцилляций.

Контуры, удовлетворяющие данным условиям, будем называть *базовыми контурами*.

Предлагаемый алгоритм нахождения базовых контуров состоит из трёх этапов [55]:

- 1) нахождение контуров и устранение маскированных контуров;
- 2) нахождение пикселей контуров, удалённых от других контуров;
- 3) нахождение прилегающей к базовым контурам окрестности.

Нахождение контуров и их маскирование

Первый этап основан на применении метода Канни нахождения контуров [56] с последующим устранением маскированных контуров. Для изображения z мы вычисляем поле градиентов $\nabla z = \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \right)$, находим локальные максимумы по строкам, столбцам и двум диагоналям изображения, применяя метод подавления немаксимальных значений (pop-maximum suppression) как в методе Канни. Затем находим пиксели контуров, для которых выполнено условие того, что они не являются маскированными

$$|\nabla z_{i_0, j_0}| \geq \max_{i, j} \left(|\nabla z_{i, j}| \phi(\sqrt{(i - i_0)^2 + (j - j_0)^2}) \right). \quad (2.1)$$

В качестве функции $\phi(d)$ будем использовать функцию

$$\phi(d) = \phi_0 e^{-\frac{d^2}{2\sigma_E^2}},$$

где $\phi_0 = \frac{1}{2}$, $\sigma_E = d_0$.

Использование данной функции показывает хорошие результаты как для анализа эффекта Гиббса, возникающего после повышения разрешения изображений [57, 58], так и для анализа эффекта размытия [55, 59]. Преимуществом данной функции является возможность эффективного вычисления

$$\max_{i, j} \left(|\nabla z_{i, j}| \phi(\sqrt{(i - i_0)^2 + (j - j_0)^2}) \right)$$

с использованием техники динамического программирования [55].

Далее мы применяем отсечение по порогу градиента, оставляя только пиксели, прошедшие условие

$$|\nabla z_{i_0, j_0}| \geq g_{min}, \quad (2.2)$$

где g_{min} — пороговое значение модуля градиента для контура.

В результате мы получаем множество пикселей немаскированных контуров M_E , удовлетворяющих (2.1) и (2.2).

Нахождение базовых контуров

На втором этапе с помощью методов математической морфологии [60] мы находим множество пикселей базовых контуров — подмножество $M_{BE} \subset M_E$ пикселей контуров, удалённых от других контуров более, чем на R_K с помощью алгоритма [55]:

1. Применение морфологического сужения к множеству пикселей, не являющимися пикселям контуров, с круговым структурным элементом с радиусом $\frac{R_K}{2}$. Если контур удалён более, чем на R_K от других контуров, тогда полученное множество будет лежать по обе стороны относительно контура на расстоянии $\frac{R_K}{2}$.

2. Применение морфологического расширения к полученному на первом шаге множеству с радиусом $\frac{R_K}{2} + \varepsilon$. Параметр $\varepsilon > 0$ используется для того, чтобы алгоритм не был чувствителен к небольшим ошибкам нахождения контуров. Мы используем $\varepsilon = 2$.

3. Затем применяем морфологическое сужение с радиусом 2ε пикселей. Мы используем значение $R_K = 3d_0$. Множество M_{BE} представляет собой пересечение полученного множества и множества пикселей контуров M_E .

Данный алгоритм проиллюстрирован на рис.2.2.

Пример выделения базовых контуров для $d_0 = 4$ приведён на рис. 2.3.

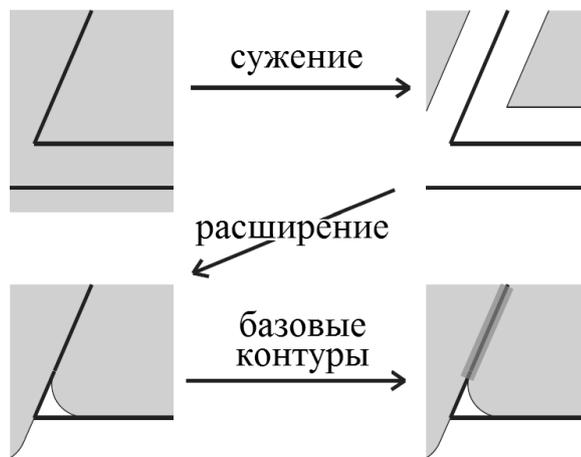


Рис. 2.2: Морфологический метод нахождения базовых контуров.



а) Эталонное изображение



б) Результат нахождения базовых контуров

Рис. 2.3: Пример выделения базовых контуров на изображении с параметром $d_0 = 4$. Базовые контуры обозначены толстым белыми линиями, небазовые контуры, прошедшие условие маскирования (2.1) — тонкими светло-серыми линиями, небазовые контуры, не прошедшие условие маскирования — тонкими тёмно-серыми линиями.

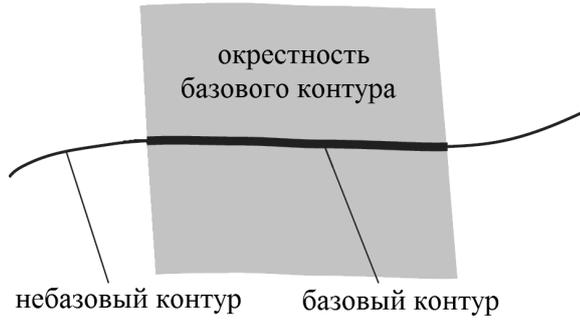


Рис. 2.4: Нахождение прилегающей к базовым контурам окрестности M_P .

Нахождение окрестности, прилегающей к базовым контурам

На последнем этапе мы с помощью преобразования евклидового расстояния [61] находим для каждого пикселя $P = (i, j)$ расстояние до ближайшего контура и до ближайшего базового контура:

$$\rho(P, M_E) = \min_{Q \in M_E} \rho(P, Q)$$

$$\rho(P, M_{BE}) = \min_{Q \in M_{BE}} \rho(P, Q).$$

В случае, если $\rho(P, M_E) = \rho(P, M_{BE})$, это значит, что ближайший к P контур является базовым контуром. Обозначим за M_P множество всех пикселей, для которых $\rho(P, M_E) = \rho(P, M_{BE})$ — *окрестность базовых контуров*. Пример такого множества приведён на рис. 2.4.

2.1.2 Метрики точек базовых контуров и окрестностей базовых контуров

Перейдём к построению интересующих нас множеств, в которых наиболее сильно проявляются артефакты методов повышения разрешения изображений.

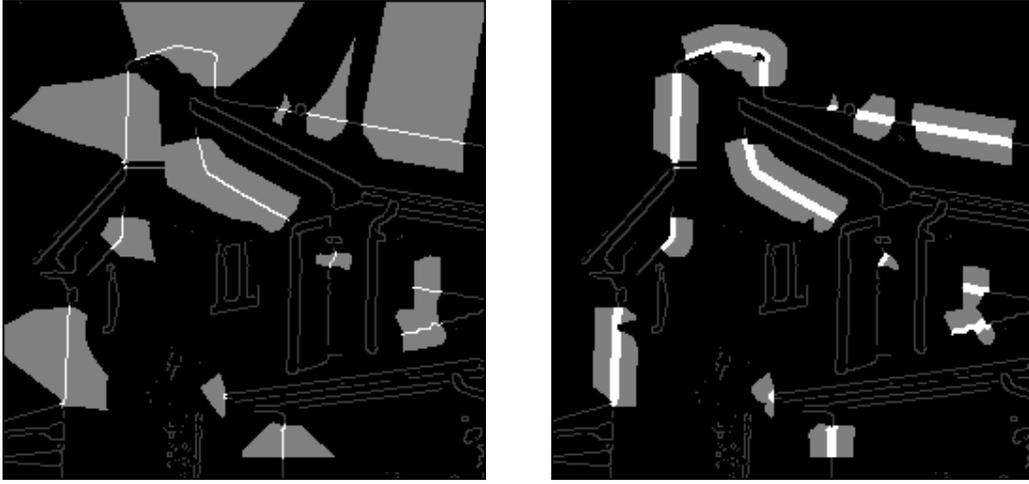
На множестве прилегающих к базовым контурам пикселей M_P выделим два подмножества: ближнюю окрестность базовых контуров

$$M_{NBE} = \{P \in M_P : \rho(P, M_{BE}) < r\}$$

и дальнюю окрестность множества пикселей базовых контуров

$$M_{FBE} = \{P \in M_P : r \leq \rho(P, M_{BE}) \leq R\}.$$

Эффект размытия проявляется в области шириной d_0 (см. рис. 2.1), поэтому для нахождения области, соответствующей эффекту размытия, будем использовать $r = \frac{d_0}{2}$. Для нахождения области, соответствующей эффекту Гиббса, мы используем $R = \frac{5d_0}{2}$. Данный выбор обусловлен тем, что в результате применения алгоритмов восстановления высокочастотной информации обычно наблюдается не более двух осцилляций эффекта Гиббса. Пример нахождения множеств M_{NBE} и M_{FBE} для изображения `house.png` приведён на рис. 2.5.



а) Визуализация множества M_P б) Нахождение множеств M_{NBE} и M_{FBE}

FF

Рис. 2.5: Пример нахождения множеств M_{NBE} и M_{FBE} для $d_0 = 4$. Белыми линиями обозначены базовые контуры, тёмно-серыми — небазовые контуры, прошедшие условие маскирования (2.1). Слева: серым цветом обозначено множество M_P . Справа: белым цветом обозначено множество M_{NBE} , тёмно-серым цветом — множество M_{FBE} .

Введём на основе метрики ρ_{SSIM} (1.32) метрики на множествах M_{NBE} и M_{FBE} :

$$\begin{aligned}\rho_{NBE}(z_0, z) &= \rho_{SSIM(M_{NBE}(z_0))}(z_0, z), \\ \rho_{FBE}(z_0, z) &= \rho_{SSIM(M_{FBE}(z_0))}(z_0, z),\end{aligned}\tag{2.3}$$

где z_0 — эталонное изображение, по которому строятся множества M_{NBE} и M_{FBE} . Метрика $\rho_{SSIM(M)}$ — это метрика $SSIM$, в которой значения $\mu_u, \mu_v, \sigma_u^2, \sigma_v^2$ и σ_{uv} вычисляются по множеству M , а не по всему изображению.

Метрики ρ_{NBE} и ρ_{FBE} являются метриками оценки качества изображений после применения алгоритмов восстановления высокочастотной информации. Мы будем их использовать для оценки качества результатов алгоритмов повышения разрешения изображений, повышения резкости изображений и подавления эффекта Гиббса.

2.1.3 Применение метрик базовых контуров для задачи комбинирования результатов нескольких методов повышения разрешения изображений

Проиллюстрируем применение предложенных метрик ρ_{NBE} и ρ_{FBE} (2.3) для задачи комбинирования результатов двух методов повышения разрешения изображений, где первый метод даёт сильно выраженный эффект Гиббса, а второй — сильный эффект размытия. В качестве первого метода можно использовать, например, интерполяцию Ланцоша или sinc-интерполяцию, в качестве второго — билинейную интерполяцию, бикубическую интерполяцию или интерполяцию с помощью фильтра Гаусса.

Будем использовать подход, основанный на контроле значения полной вариации изображения. Предположим, что значение полной вариации результата первого метода $\|z_1\|_V$ выше, а значение полной вариации результата второго метода $\|z_2\|_V$ ниже, чем значение полной вариации изображения низкого разрешения $\|u\|_V$. Тогда можно взять линейную комбинацию z_1 и z_2 так, чтобы значение полной вариации комбинации изображения $z_c = (1 - c)z_1 + cz_2$ стало равно $\|u\|_V$. Функция $\|z_c\|_V$ непрерывно зависит от c , причём $\|z_c\|_V = \|z_1\|_V$ при $c = 0$ и $\|z_c\|_V = \|z_2\|_V$ при $c = 1$. В силу того, что $\|z_1\|_V \leq \|u\|_V \leq \|z_2\|_V$ существует такое c , $0 \leq c \leq 1$, что $\|z_c\|_V = \|u\|_V$. Мы используем приближение

$$c = \frac{\|u\|_V - \|z_1\|_V}{\|z_2\|_V - \|z_1\|_V}.$$

Недостаток такого подхода заключается в том, что комбинация линейных методов остаётся линейным методом с присущими им недостатками. Глобальная комбинация методов не учитывает локальные особенности

методов. Так, эффект Гиббса проявляется в виде ореолов в окрестностях контуров, тогда как размытие портит сами контуры.

Простая комбинация результатов, при которой в качестве значения пикселя, лежащего на контуре, берётся соответствующее значение пикселя результата первого метода, а в качестве значения пикселя из окрестности контура — значение пикселя результата второго метода, даёт артефакт в виде стыков на границах контуров и их окрестностей. Также такой подход требует применения алгоритмов поиска контуров на изображении, что может существенно повысить вычислительную сложность метода.

Чтобы уменьшить эти недостатки, будем использовать адаптивную комбинацию результатов на основе значений локальной полной вариации, вычисляемой в небольшой окрестности пикселя [62]. Для вычисления значения каждого пикселя $z_{i,j}$ изображения высокого разрешения z вычислим полную вариацию в квадратной окрестности пикселя (i, j) для изображений z_1, z_2 и соответствующей ей окрестности на изображении низкого разрешения u :

$$\begin{aligned} \|z_1\|_{V(i,j)} &= \int_{i-r}^{i+r} \int_{j-r}^{j+r} \left(\left(\frac{\partial \bar{z}_1}{\partial x} \right)^2 + \left(\frac{\partial \bar{z}_1}{\partial y} \right)^2 \right)^{1/2} dx dy, \\ \|z_2\|_{V(i,j)} &= \int_{i-r}^{i+r} \int_{j-r}^{j+r} \left(\left(\frac{\partial \bar{z}_2}{\partial x} \right)^2 + \left(\frac{\partial \bar{z}_2}{\partial y} \right)^2 \right)^{1/2} dx dy, \\ \|u\|_{V(i,j)} &= s \int_{\frac{i-r}{s}}^{\frac{i+r}{s}} \int_{\frac{j-r}{s}}^{\frac{j+r}{s}} \left(\left(\frac{\partial \bar{u}}{\partial x} \right)^2 + \left(\frac{\partial \bar{u}}{\partial y} \right)^2 \right)^{1/2} dx dy, \end{aligned}$$

где $\bar{z}_1, \bar{z}_2, \bar{u}$ — непрерывные изображения, полученные в результате билинейной интерполяции, s — коэффициент увеличения, $r > 0$ — параметр, задающий размер квадратной окрестности и определяемый комбинируемыми методами повышения разрешения изображений.

В случае, если r целое, то для вычисления значений интегралов (2.1.3)

нет необходимости в построении непрерывных изображений \bar{z}_1 , \bar{z}_2 , \bar{u} . Значения $\|z_1\|_{V(i,j)}$ и $\|z_2\|_{V(i,j)}$ при этом равны сумме длин градиентов пикселей, попавших в квадратную окрестность пикселя (i, j) , а значение $\|u\|_{V(i,j)}$ равно $\|u_s\|_{V(i,j)}$, где u_s — результат билинейной интерполяции u с коэффициентом увеличения s . При этом для эффективного вычисления значений полной вариации со сложностью $O(1)$, не зависящей от r , можно, пользуясь свойством аддитивности интеграла, использовать подход, основанный на вычислении интегрального изображения [63].

Далее вычисляется весовая функция

$$c_{i,j} = \frac{\|u\|_{V(i,j)} - \|z_1\|_{V(i,j)}}{\|z_2\|_{V(i,j)} - \|z_1\|_{V(i,j)}}$$

и строится значение пикселя комбинированного изображения

$$z_{i,j} = (1 - c_{i,j})z_{1,i,j} + c_{i,j}z_{2,i,j}.$$

При этом, если $c_{i,j} < 0$, например, если $\|u\|_{V(i,j)} < \|z_1\|_{V(i,j)}$, то используется $c_{i,j} = 0$. Аналогично, если $c_{i,j} > 1$, то используется $c_{i,j} = 1$.

Пример комбинации методов идеальной интерполяции и бикубической интерполяции приведён на рис.2.6. Комбинированный метод даёт значения метрик, близкие к значениям метрик наилучшего из методов, участвующих в комбинации, при этом эффект Гиббса незначителен.

2.2 Задача подавления эффекта Гиббса на изображениях

2.2.1 Алгоритм обнаружения и оценки эффекта Гиббса

В разделе 1.3.2 была показана взаимосвязь между эффектом Гиббса и полной вариацией. В случае возникновения эффекта Гиббса при действии идеального фильтра на некоторую функцию из $L_2(\mathbb{R})$ с ограниченной полной вариацией, например, при свёртке с функцией sinc, полная вариация этой функции становится бесконечно большой. Ещё одной проблемой является наличие шума, например, равномерного шума, при котором полная вариация также становится бесконечно большой [35].

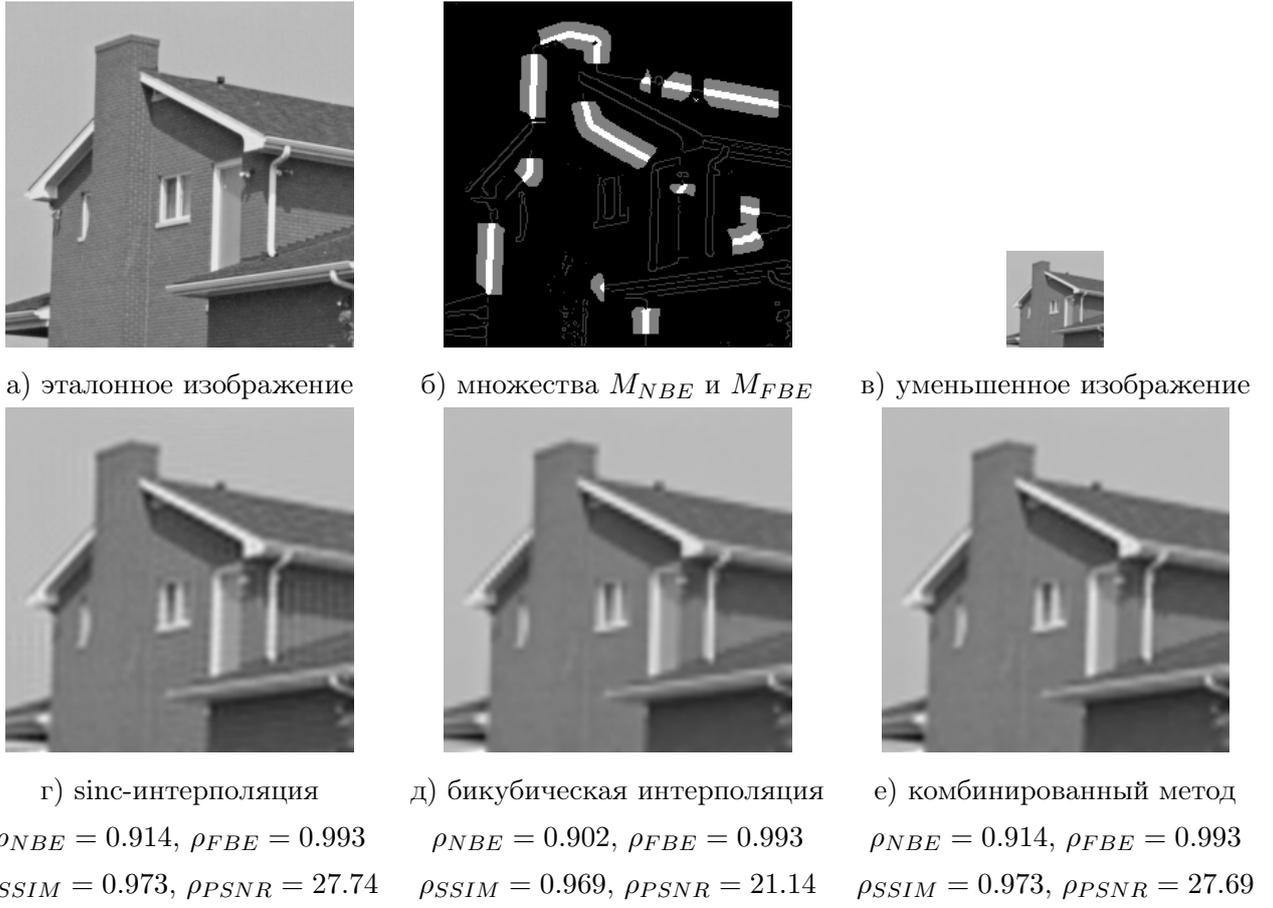


Рис. 2.6: Пример комбинации методов идеальной (sinc) интерполяции и бикубической интерполяции при повышении разрешения в 4 раза.

Размерность реальных изображений конечна, а вместо идеального фильтра обычно используется его аппроксимация с финитным ядром, поэтому полная вариация изображения высокого разрешения становится значительно выше полной вариации изображения низкого разрешения, но при этом она остаётся конечной.

С субъективной точки зрения, наиболее значимыми являются несколько первых осцилляций эффекта Гиббса возле контуров, тогда как последующие осцилляции либо отсутствуют, либо их амплитуда ниже уровня шума. Для того, чтобы учитывать только значимые осцилляции, будем использовать *взвешенную полную вариацию*

$$WTV(f, w) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |\nabla f(x, y)| w(x, y) dx dy, \quad f \in L(\mathbb{R}^2), \quad (2.4)$$

где $w(x, y) \geq 0$ — весовая функция. В дискретном случае определим взвешенную полную вариацию как

$$WTV(u, w) = h \sum_{i,j=-\infty}^{+\infty} (|u_{i+1,j} - u_{i,j}|^2 + |u_{i,j+1} - u_{i,j}|^2)^{1/2} w(ih, jh) dx dy,$$

$$u \in \ell(\Omega_h).$$

Для выбора весовой функции и критериев определения наличия эффекта Гиббса, нами был произведён многомасштабный анализ эффекта Гиббса при повышении разрешения изображения, содержащего только один ступенчатый контур.

Многомасштабный анализ профилей контуров

Осцилляции эффекта Гиббса располагаются параллельно породившему их контуру, поэтому для определения наличия эффекта Гиббса возле контура достаточно использовать профили контура — сечения контура прямой, перпендикулярной направлению контура.

Рассмотрим задачу определения наличия эффекта Гиббса при повышении разрешения ступенчатого контура, профиль которого представлен функцией

$$p_0(x) = \begin{cases} 1, & x > 0, \\ 0, & x \leq 0. \end{cases}$$

Профиль $p_u[i]$ ступенчатого контура на изображении низкого разрешения на сетке с узлами $\{x_i : x_i = hi - \frac{h}{2}\}$ будет иметь вид

$$p_u[i] = \begin{cases} 1, & i > 0, \\ 0, & i \leq 0. \end{cases}$$

При повышении разрешения изображения в s раз выберем узлы сетки профиля $p_z[i]$, соответствующего профилю $p_u[i]$, следующим образом: $\{y_i : y_i = \frac{hi}{s} - \frac{h}{2}\}$.

Целью исследования является выделение характеристик, по которым можно определить, присутствует ли эффект Гиббса в окрестности произвольного контура при известном коэффициенте увеличения s на основе анализа профиля контура $p_z[i]$. При этом метод, с помощью которого было получено изображение высокого разрешения, является неизвестным.

В качестве базового метода, не приводящего к возникновению эффекта Гиббса, будем использовать билинейную интерполяцию. При этом профиль ступенчатого контура принимает вид

$$p_b[i] = \begin{cases} 0, & i \leq 0, \\ \frac{i}{s}, & 0 < i < s, \\ 1, & i \geq s, \end{cases}$$

а в качестве базового метода, после применения которого появляется эффект Гиббса — идеальную (sinc) интерполяцию, с профилем контура

$$p_s[i] = \sum_{j=1}^{+\infty} \text{sinc}(i - js).$$

Для моделирования произвольных методов повышения разрешения изображений будем прибавлять равномерно распределённый шум со случайным параметром к результатам базовых методов:

$$p_{b,n_0}[i] = p_b[i] + \xi_n[i],$$

$$p_{s,n_0}[i] = p_s[i] + \xi_n[i],$$

где $\xi_n[i]$ — случайная величина с равномерным распределением, принимающая значения из отрезка $[-\frac{n}{2}, \frac{n}{2}]$. Значение уровня шума n постоянно для каждого профиля контура и выбирается случайным образом из отрезка $[0, n_0]$, где n_0 — максимальный рассматриваемый уровень искажений. Такой подход позволяет учесть при анализе как контуры с малым уровнем искажения, так и контуры с большим уровнем искажения.

Для фиксированных s и n_0 было сгенерировано 10000 профилей контуров без эффекта Гиббса p_{b,n_0} и 10000 профилей контуров с эффектом

Гиббса p_{s,n_0} . Для каждого из контуров были вычислены значения взвешенной полной вариации профиля контура p , взятого на разных разрешениях σ

$$TV(p, \sigma, w_{\alpha s}) = WTV(p * G_{\sigma}, w_{\alpha s}), \quad (2.5)$$

где

$$w_{\alpha s}(x) = e^{-\frac{x^2}{2(\alpha s)^2}},$$

при различных α и σ . Параметр α определяет количество учитываемых осцилляций эффекта Гиббса.

Пример значений $TV(p, \sigma, w_{\alpha s})$ для $n_0 = 25$, $\alpha = 3$ и $s = 10$ приведён на рис. 2.7. На данном примере видно, что существует множество значений σ , при которых по значению $TV(p, \sigma, w_{\alpha s})$ можно определить, был ли контур с профилем p с эффектом Гиббса или нет.

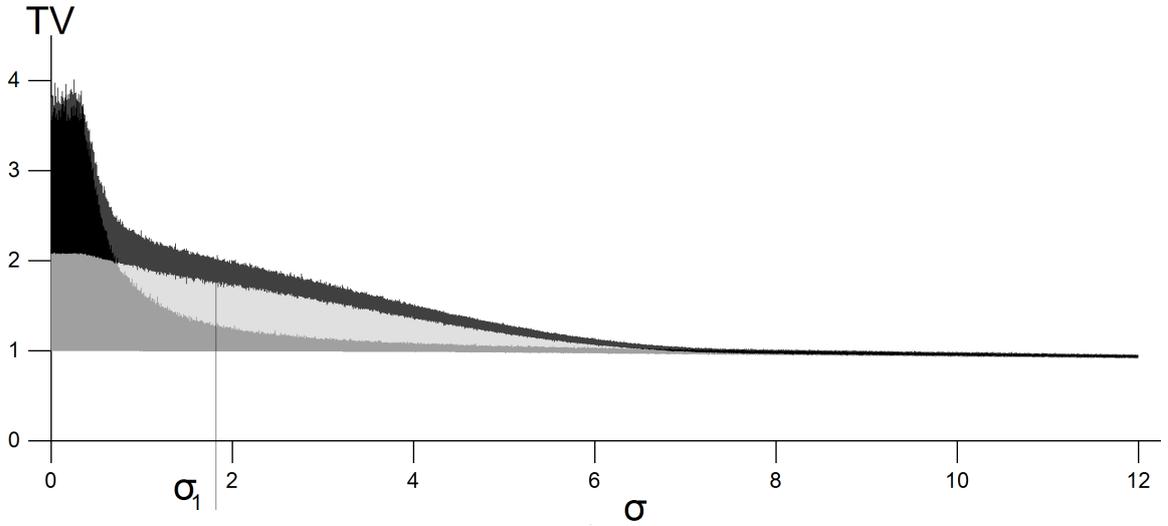


Рис. 2.7: Значения $TV(p, \sigma, w_{\alpha s})$ для $n_0 = 25$, $\alpha = 3$ и $s = 10$. Серым цветом обозначено множество значений $TV(p, \sigma, w_{\alpha s})$ для профилей контуров без эффекта Гиббса, тёмно-серым — для профилей контуров с эффектом Гиббса. Пересечение этих множеств обозначено чёрным цветом. Светло-серым цветом обозначено множество значений, которые $TV(p, \sigma, w_{\alpha s})$ не принимает для рассматриваемых профилей контуров.

Введём обозначения:

$$g^*(s, \sigma, n_0, \alpha) = \min_{p=p_{s,n_0}} TV(p, \sigma, w_{\alpha s}),$$

$$g_*(s, \sigma, n_0, \alpha) = \max_{p=p_{b,n_0}} TV(p, \sigma, w_{\alpha s}).$$

Функция $g^*(s, \sigma, n_0, \alpha)$ представляет собой нижнюю границу множества значений $TV(p, \sigma, w_{\alpha s})$ для профилей контуров с эффектом Гиббса (тёмно-серая область на рис. 2.7), а $g_*(s, \sigma, n_0, \alpha)$ — верхнюю границу соответствующего множества значений для профилей контуров без эффекта Гиббса (серая область на рис. 2.7).

Если при некотором σ значение $g^*(s, \sigma, n_0, \alpha)$ больше, чем $g_*(s, \sigma, n_0, \alpha)$, то это σ можно использовать для определения наличия эффекта Гиббса по значению $TV(p, \sigma, w_{\alpha s})$. Найдём σ_1 , при котором различие между $g^*(s, \sigma, n_0, \alpha)$ и $g_*(s, \sigma, n_0, \alpha)$ наибольшее:

$$\sigma_1(s, n_0, \alpha) = \arg \max_{\sigma} (g^*(s, \sigma, n_0, \alpha) - g_*(s, \sigma, n_0, \alpha)).$$

На рис. 2.8 показана зависимость $\sigma_1(s, n_0, \alpha)$ от s при $n_0 = 25$, $\alpha = 3$. Экспериментальный анализ для тестового множества сгенерированных профилей контуров показал, что функцию $\sigma_1(s, n_0, \alpha)$ можно аппроксимировать линейной функцией

$$\sigma_2(s, n_0, \alpha) = m(n_0, \alpha)s,$$

при этом значения σ_2 можно использовать для определения наличия эффекта Гиббса при тех же s , что и σ_1 . Для $n_0 = 25$ и $\alpha = 3$ мы используем $m(n_0, \alpha) = 0,19$, для $n_0 = 50$ и $\alpha = 3$ — $m(n_0, \alpha) = 0,25$. Также было отмечено, что при $\alpha > 2$ значения $m(n_0, \alpha)$ практически не зависят от α , поэтому можно считать функцию $m(n_0, \alpha)$ зависящей только от n_0 :

$$\sigma_2(s, n_0, \alpha) = \sigma_2(s, n_0) = m(n_0)s.$$

Таким образом, для определения наличия эффекта Гиббса на контуре с профилем p , полученном из одномерного ступенчатого контура при повышении разрешения в s раз, достаточно вычислить значение

$$TV_R(p, n_0, \alpha) = TV(p, m(n_0)s, w_{\alpha s})$$

и сравнить его с $G^*(s, n_0, \alpha)$ и $G_*(s, n_0, \alpha)$, где

$$G^*(s, n_0, \alpha) = g^*(s, m(n_0)s, n_0, \alpha),$$

$$G_*(s, n_0, \alpha) = g_*(s, m(n_0)s, n_0, \alpha).$$

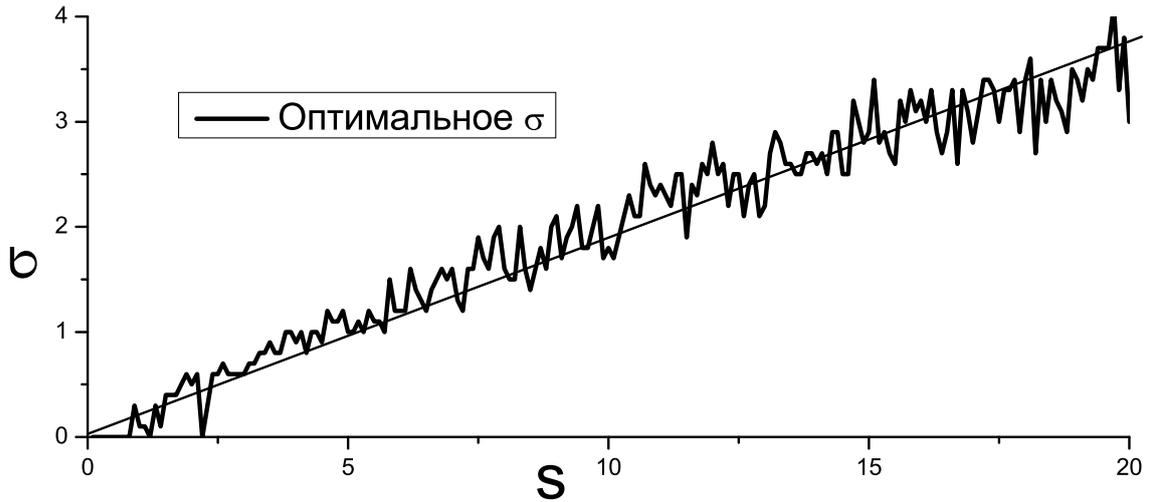


Рис. 2.8: Зависимость значений $\sigma_1(s, n_0, \alpha)$ от s при $n_0 = 25$, $\alpha = 3$. Тонкой прямой показана линейная аппроксимация функции $\sigma_1(s, n_0, \alpha)$.

На рис. 2.9 приведены графики функций $G^*(s, n_0, \alpha)$ и $G_*(s, n_0, \alpha)$ при $n_0 = 25$, $\alpha = 3$.

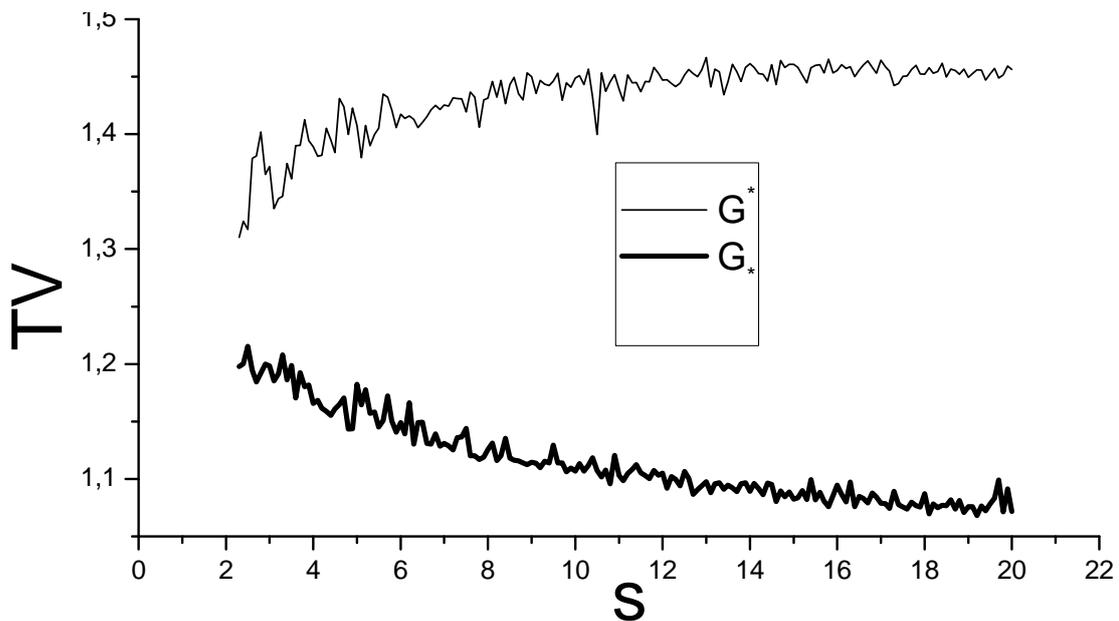


Рис. 2.9: Зависимость значений $G^*(s, n_0, \alpha)$ и $G_*(s, n_0, \alpha)$ от s при $n_0 = 25$, $\alpha = 3$.

Оценка уровня эффекта Гиббса

Многомасштабный анализ был проведён для контура с величиной перепада, равной 1, тогда как величина перепада контуров на реальных изображениях не является фиксированной. В этом случае произведём нор-

мирование контура так, чтобы величина перепада получившегося контура стала равна 1. Для этого воспользуемся результатами проведённого многомасштабного анализа. Заметим, что значения $TV(p, \sigma, w_{\alpha s})$ стремятся к значению перепада контура при $\sigma \rightarrow \infty$. Реальные контуры же имеют конечные размеры, поэтому для оценки перепада необходимо использовать как можно меньшее σ . Хорошие результаты получаются при аппроксимации высоты значением $TV(p, \sigma, w_{\alpha s})$ при $\sigma = s$.

Таким образом, для произвольного контура с профилем p значение TV_R вычисляется в виде

$$TV_R(p, s, n_0, \alpha) = \frac{TV(p, m(n_0)s, w_{\alpha s})}{TV(p, s, w_{\alpha s})}. \quad (2.6)$$

Введём значение R_E — *оценку уровня эффекта Гиббса*:

$$R_E(p, s, n_0, \alpha) = 2 \frac{TV_R(p, s, n_0, \alpha) - G_*(s, n_0, \alpha)}{G^*(s, n_0, \alpha) - G_*(s, n_0, \alpha)} - 1. \quad (2.7)$$

Для контуров с эффектом Гиббса значения R_E будут ≥ 1 , для контуров без эффекта Гиббса $R_E \leq -1$. Появление промежуточных значений $-1 < R_E < 1$ возможно в случае, если значение n_0 оказалось меньше, чем реальный уровень искажения контура.

Для изображений, содержащих несколько контуров, мы выбираем базовые контуры с наибольшим значением модуля градиента, затем строим несколько профилей для отобранных контуров и вычисляем среднее значение R_E [57].

2.2.2 Регуляризирующий метод подавления эффекта Гиббса

В разделе 1.3.2 первой главы была показана связь функционала полной вариации с эффектом Гиббса. Появление эффекта Гиббса приводит к значительному увеличению значения полной вариации. В связи с этим поставим задачу подавления эффекта Гиббса на изображении z в виде задачи проектирования изображения z на множество изображений с

ограниченной полной вариацией

$$z_C = pr_C z = \arg \min_{z': \|z'\|_{TV} \leq C} \|z' - z\|_2, \quad (2.8)$$

где параметр C отвечает за силу подавления эффекта Гиббса.

Задача подавления эффекта Гиббса в общем случае

Рассмотрим задачу подавления эффекта в общем случае, когда дано изображение z с эффектом Гиббса, но нет информации об изображении низкого разрешения. В этом случае для выбора параметра C в (2.8) будем пользоваться оценкой R_E (2.7) для определения наличия и уровня эффекта Гиббса. В качестве параметра (2.8) выбирается наибольшее значение C , при котором $R_E(z_C) \leq \beta$, где $-1 \leq \beta \leq 1$. Неоднозначность выбора β вызвана тем, что метод оценки эффекта Гиббса даёт однозначный результат только для $R_E < 1$ и $R_E > 1$. Таким образом, задача подавления эффекта Гиббса в общем случае принимает вид

$$z_1 = pr_{C_1} z,$$

где

$$C_1 = \max_{R_E(pr_C z) \leq \beta} C.$$

В общем случае неизвестны значения s , n_0 , используемые при вычислении R_E . В этом случае мы используем фиксированное значение n_0 , а коэффициент s оцениваем с помощью анализа ширины контуров [57].

Пример подавления эффекта Гиббса в общем случае для различных β приведён на рис. 2.10.

Задача подавления эффекта Гиббса после повышения разрешения изображения

При подавлении эффекта Гиббса после повышения разрешения изображений параметр C в (2.8) может быть оценён через изображение низкого разрешения u [64, 65]. Будем использовать тот факт, что полная вариация равна сумме длин линий уровня [35]. Так как алгоритм повышения



Изображение с эффектом Гиббса

$$\rho_{SSIM} = 0,9500, \|\cdot\|_{TV} = 6,872$$

$$\rho_{NBE} = 0,9523, \rho_{FBE} = 0,9725$$



Результат подавления эффекта
Гиббса для $\beta = -1$

$$\rho_{SSIM} = 0,9456, \|\cdot\|_{TV} = 4,123$$

$$\rho_{NBE} = 0,9481, \rho_{FBE} = 0,9718$$



Результат подавления эффекта
Гиббса для $\beta = 0$

$$\rho_{SSIM} = 0,9495, \|\cdot\|_{TV} = 4,948$$

$$\rho_{NBE} = 0,9519, \rho_{FBE} = 0,9733$$



Результат подавления эффекта
эффекта Гиббса для $\beta = 1$

$$\rho_{SSIM} = 0,9505, \|\cdot\|_{TV} = 5,839$$

$$\rho_{NBE} = 0,9529, \rho_{FBE} = 0,9735$$

Рис. 2.10: Пример подавления эффекта Гиббса для различных β .

разрешения изображений не должен добавлять или удалять контуры, то и длина линий уровня не должна меняться при переходе на более мелкую сетку. Значит, полная вариация изображения высокого разрешения

должна быть равна полной вариации изображения низкого разрешения

$$C = \|u\|_{TV}.$$

Так как изображение может состоять из достаточно разнородных участков, глобальное подавление полной вариации может привести к тому, что в одних участках помимо эффекта Гиббса могут быть подавлены мелкие детали, тогда как в других участках эффект Гиббса не будет полностью подавлен. Для того, чтобы учесть локальные особенности изображений, мы используем блочный подход [66]. В этом случае изображение z разбивается на перекрывающиеся блоки, подавление эффекта Гиббса производится независимо для каждого блока, при этом в качестве параметра C используется значение полной вариации соответствующих блоков на изображении u . Затем из блоков восстанавливается изображение высокого разрешения.

Пример подавления эффекта Гиббса после повышения разрешения изображений приведён на рис. 2.11.

2.3 Задача повышения резкости изображений

2.3.1 Постановка задачи повышения резкости изображений

Поставим задачу повышения резкости изображений как обратную задачу для задачи размытия изображений:

$$Hz = u, \quad z, u \in \ell(\Omega_h),$$

где z — искомое изображение, u — известное размытое изображение, H — оператор размытия, являющийся неизвестным. В общем случае для нахождения оператора H используются специальные методы [67]. Мы будем работать со случаем, когда $H = G_\sigma$ является фильтром Гаусса с неизвестным радиусом σ .

Для нахождения её решения будем использовать метод регуляризации



Изображение низкого разрешения

$$TV = 8,6515$$



Увеличение изображения в 2 раза

$$TV = 17,24, R_E = 8, \rho_{SSIM} = 0,9779$$



Глобальный метод

$$TV = 8,927, R_E = -0,7, \rho_{SSIM} = 0,9827$$



Блочный метод

$$TV = 8,736, R_E = -1,5, \rho_{SSIM} = 0,9812$$

Рис. 2.11: Пример подавления эффекта Гиббса после повышения разрешения изображений.

со стабилизатором в виде функционала полной вариации [68]:

$$z_C = \arg \min_{z: \|z\|_{TV} \leq C} \|G_\sigma z - u\|_2^2. \quad (2.9)$$

Для оценки параметра σ по размытому изображению u нами было введено понятие ширины контура и проведён анализ зависимости ширины ступенчатого контура при размытии с помощью фильтра Гаусса с

разными радиусами [59].

Вычисление ширины контура по его профилю $y = y(x)$ производилось по следующему алгоритму [57]: находились минимальное y_0 и максимальное y_1 значения интенсивности в окрестности контура, затем находились точки пересечения функции $y(x)$ с $y = \frac{3}{4}y_0 + \frac{1}{4}y_1$ и с $y = \frac{1}{4}y_0 + \frac{3}{4}y_1$ с аргументами x'_0 и x'_1 соответственно. В случае нескольких точек пересечения бралось их среднее. Через точки $(x'_0, y(x'_0))$ и $(x'_1, y(x'_1))$ проводилась прямая до пересечения с прямыми $y = y_0$ в x_0 и $y = y_1$ в x_1 . Значение $|x_1 - x_0|$ принималось за определение ширины границы. Данный алгоритм проиллюстрирован на рис. 2.12.

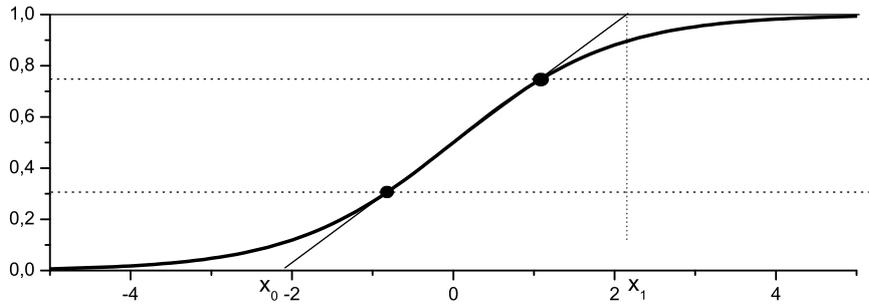


Рис. 2.12: Вычисление ширины контура по его профилю. Для приведённого контура ширина равна 4, 2.

Анализ зависимости ширины ступенчатого контура w от радиуса гауссовского размытия σ на представительной базе реальных изображений показал линейную взаимосвязь между шириной границы и параметром размытия [57]:

$$\frac{\sigma}{w} = k_0,$$

где $k_0 = 0,45$.

Для определения параметра размытия на изображении производится выделение контуров, отбор контуров с наибольшим значением модуля градиента и вычисление их средней ширины [59].

Значение параметра C в (2.9) принимается равным полной вариации

размытого изображения $C = \|u\|_{TV}$.

На рис. 2.13 приведены результаты работы данного метода повышения резкости.

2.3.2 Регуляризирующий метод адаптивного повышения резкости низкочастотной части изображения

Недостатком метода (2.9) является низкое качество результатов в случае, когда размытые изображения зашумлены. В этом случае будем производить разделение размытого изображения на структурную и шумовую компоненты.

Для выделения структурной компоненты применим оператор размытия H к исходному размытому изображению

$$u_s = Hu,$$

тогда шумовая компонента примет вид

$$u_n = u - u_s = u - Hu.$$

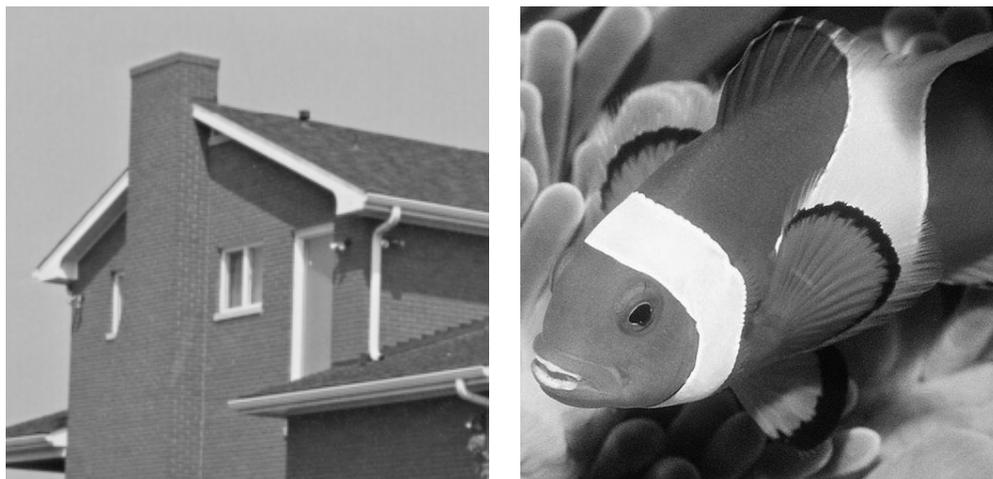
Применим для структурной компоненты метод повышения резкости (2.9)

$$z_s = \arg \min_{z: \|z\|_{TV} \leq \|u_s\|_{TV}} \|HHz - u_s\|_2^2. \quad (2.10)$$

Результирующее изображение с повышенной резкостью представляет собой сумму обострённой структурной компоненты и исходной шумовой компоненты

$$z = z_s + u_n. \quad (2.11)$$

Результаты работы данного алгоритмы приведены на рис. 2.13



Эталонные изображения



Множества метрик базовых контуров:
 белые области — M_{NBE} , серые - M_{FBE}

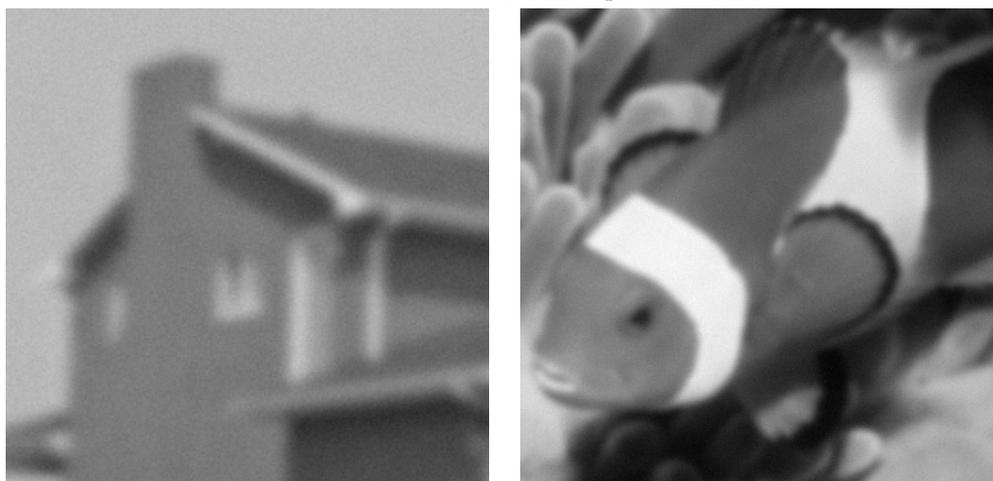
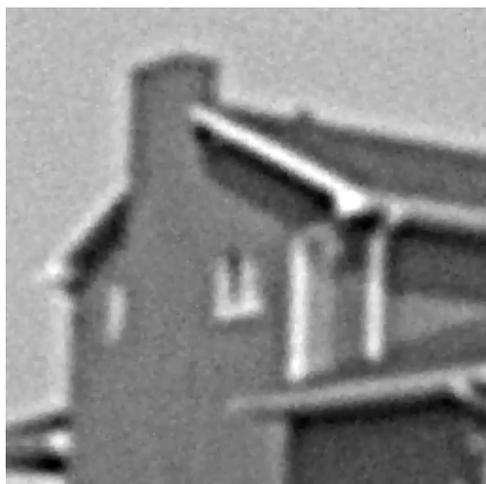
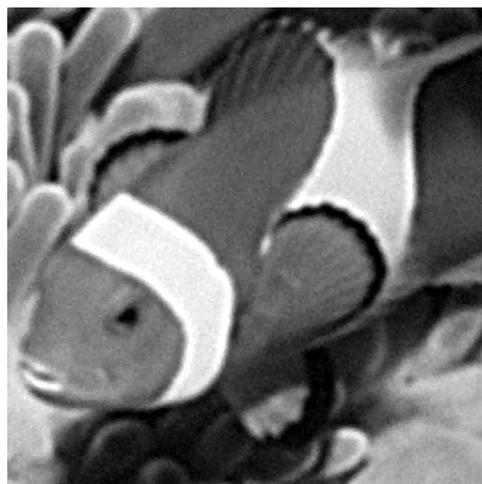

 $\rho_{SSIM} = 0,9150$
 $\rho_{SSIM} = 0,9791$
Размытые изображения с $\sigma = 4$

Рис. 2.13: Результаты повышения резкости изображений с помощью методов (2.9, 2.11).

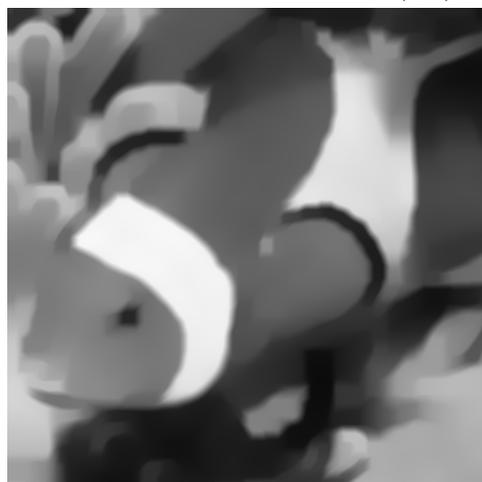


$$\rho_{SSIM} = 0,9421$$



$$\rho_{SSIM} = 0,9788$$

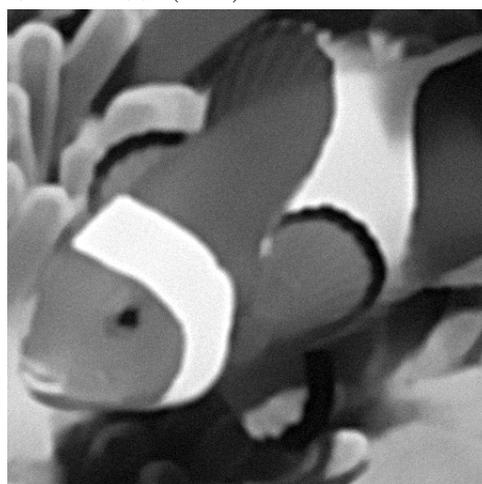
Результат повышения резкости изображений с помощью метода (2.9)



Результат повышения резкости структурной
компоненты с помощью метода (2.10)



$$\rho_{SSIM} = 0,9434$$



$$\rho_{SSIM} = 0,9797$$

Результат повышения резкости изображений с помощью метода (2.11)

Рис. 2.13: Результаты повышения резкости изображений с помощью методов (2.9, 2.11).

Глава 3

Программный комплекс

В работе предложен ряд методов, решающих актуальные задачи повышения разрешения изображений. Наряду с построением регуляризирующих алгоритмов, практические задачи требуют эффективной программной реализации предложенных алгоритмов. В данной главе приводятся описание программной реализации и примеры результатов работы разработанных методов.

Предложенные методы повышения разрешения и постобработки изображений реализованы в виде программного комплекса для обработки и анализа растровых изображений. Программы были созданы в среде разработки Microsoft Visual Studio 2010 под операционные системы Microsoft Windows. Программный комплекс построен на основе технологии .NET, основная часть программы написана на языке C#. Пользовательский интерфейс реализован с помощью библиотеки Windows Forms, а сами алгоритмы, для увеличения скорости расчетов, реализованы в виде отдельной библиотеки DLL, написанной на языке C++. Для повышения производительности на современных многоядерных процессорах было произведено частичное распараллеливание кода с помощью библиотеки Intel Threading Building Blocks (TBB) и оптимизация с использованием векторных инструкций SSE и AVX [69]. Подключение библиотеки к основной программе происходит с помощью стандартных процедур .NET Interop.

Все реализованные в программе алгоритмы оптимизированы для ра-

боты с изображениями в градациях серого. При повышении разрешения цветных изображений производится автоматическое разложение изображения на изображение в градациях серого Y и цветоразностное изображение $(R - Y, G - Y, B - Y)$. При этом используется формула

$$Y = 0.299R + 0.587G + 0.114B,$$

где R соответствует интенсивности красного канала, G — зеленого канала, B — синего канала.

Повышение разрешения изображения в градациях серого производится с помощью разработанных методов, а повышение цветоразностного изображения — с помощью бикубической интерполяции. Результирующее изображение высокого разрешения получается путём суммирования изображения в градациях серого с цветоразностным изображением.

3.1 Используемые численные методы и их программная реализация

3.1.1 Субградиентный метод минимизации регуляризирующих функционалов

Для нахождения решения задач повышения разрешения изображений (1.26), суперразрешения (1.28), подавления эффекта Гиббса (2.8), повышения резкости изображений (2.9, 2.10) требуется произвести минимизацию функционалов вида

$$z^* = \arg \min_{z \in Z} J(z),$$

где $J(z) = \|Az - u\| + \lambda\Psi[z]$.

При использовании регуляризирующих методов со стабилизатором $\Psi[z]$ в виде полной вариации или билатеральной полной вариации регуляризирующий функционал не является дифференцируемым. Для того, чтобы иметь возможность использовать градиентные методы недифференцируемых функционалов, мы используем понятие субградиента [70].

Определение. Пусть функционал $J(z)$ определён на множестве Z из E^n . Вектор $c = c(z_0) \in E^n$ называется *субградиентом* функционала $J(z)$ в точке z_0 , если

$$J(z) \geq J(z_0) + (c(z_0), z - z_0)$$

выполняется для любого $z \in E^n$.

Множество всех субградиентов функционала в точке z_0 называется *субдифференциалом* и обозначается через $\partial J(z_0)$.

Субградиент обладает следующими свойствами [70]:

1. Если $J(z)$ — выпуклый функционал, а Z — выпуклое множество, то субдифференциал является непустым для любого $z_0 \in Z$.

2. Если функционал $J(z)$ является дифференцируемым в точке $z_0 \in Z$, то его градиент $J'(z_0)$ является субградиентом, а если z_0 — внутренняя точка Z , то его субдифференциал содержит только один вектор, равный градиенту $J'(z_0)$.

Для минимизации $J(z)$ мы используем субградиентный метод [71]

$$z^{(k+1)} = z^{(k)} - \alpha_k g^{(k)},$$

где $g^{(k)}$ — один из субградиентов функционала $J(z)$ в точке $z = z^{(k)}$.

В [71] рассматриваются следующие стратегии выбора шага α_k и для них доказывається сходимость субградиентного метода:

1. Постоянный коэффициент шага $\alpha_k = \alpha$, либо постоянная длина шага $\alpha_k = \frac{\gamma}{\|g^{(k)}\|}$. В этом случае сходимость является ε -оптимальной [71], то есть начиная с некоторой итерации все приближённые решения $z^{(k)}$ находятся в ε -окрестности точного решения z^* , при этом ε зависит только от α .

2. Последовательность α_k , удовлетворяющая условиям

$$\alpha_k \geq 0, \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty, \quad \sum_{k=1}^{\infty} \alpha_k = \infty.$$

3. Несуммируемая убывающая последовательность коэффициентов ша-

га

$$\alpha_k \geq 0, \quad \lim_{k \rightarrow \infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} \alpha_k = \infty,$$

либо несуммируемая убывающая последовательность длин шага

$$\gamma_k \geq 0, \quad \lim_{k \rightarrow \infty} \gamma_k = \infty, \quad \sum_{k=1}^{\infty} \gamma_k = \infty,$$

$$\alpha_k = \frac{\gamma_k}{\|g^{(k)}\|}.$$

Скорость сходимости промежуточного решения $z^{(k)}$ субградиентного метода к точному решению z^* при наилучшем выборе $\alpha_k = (R/G)/\sqrt{k}$ может быть оценена с помощью формулы

$$\|z^{(k)} - z^*\| \leq \frac{RG}{\sqrt{k}}, \quad (3.1)$$

где $R = \|z^{(0)} - z^*\|$, $\|g^{(k)}\| \leq G$.

Вычисление субградиента регуляризирующего функционала

Рассмотрим задачу минимизации функционала вида

$$J(z) = \|Az - u\|_Z^n + \lambda\Psi[z].$$

Определим для функции $\phi(x) = |x|$ субградиент при $x = 0$ равным нулю: $\partial\phi(x)|_{x=0} = 0$. В этом случае получим следующие субградиенты для невязки $\|Az - u\|_Z^n$:

$$\begin{aligned} \partial(\|Az - u\|_1) &= A^* \operatorname{sign}(Az - u), \\ \partial(\|Az - u\|_1^2) &= 2\|Az - u\|_1 \cdot A^* \operatorname{sign}(Az - u), \\ \partial(\|Az - u\|_2) &= \frac{1}{\|Az - u\|_2} A^*(Az - u), \\ \partial(\|Az - u\|_2^2) &= 2A^*(Az - u), \end{aligned}$$

где $\operatorname{sign} u$ — оператор, действующий из U в U и представляющий из себя

покомпонентное применение функции sign :

$$[\text{sign } u]_{i,j} = \begin{cases} 1, & u_{i,j} > 0, \\ 0, & u_{i,j} = 0, \\ -1, & u_{i,j} < 0. \end{cases}$$

Для стабилизатора $\|\Delta z\|_2^2$:

$$\partial\|\Delta z\|_2^2 = 2\Delta^* \Delta z.$$

Субградиенты стабилизаторов $\|z\|_{TV}$ и $\|z\|_{BTV}$ можно найти, используя их представление в виде

$$\begin{aligned} \|z\|_{TV} &= h (\|S_x z - z\|_1 + \|S_y z - z\|_1), \\ \|z\|_{BTV} &= h \left(\sum_{s,t=-p}^p \gamma_{s,t} \|S_x^s S_y^t z - z\|_1 \right), \end{aligned}$$

где S_x^s и S_y^t — операторы сдвига изображения по горизонтали и по вертикали на s и t пикселей соответственно.

Для операторов S_x^s и S_y^t субградиент выглядит следующим образом

$$\begin{aligned} \partial\|S_x^s z - z\|_1 &= S_x^{-s} \text{sign}(S_x^s z - z), \\ \partial\|S_y^t z - z\|_1 &= S_y^{-t} \text{sign}(S_y^t z - z). \end{aligned}$$

Найдём в явном виде сопряжённые операторы для $A_h = D_h H_{\sigma_0 h}$, $A_s = D_s H_{\sigma_0 \sqrt{s^2-1}}$ и Δ^* .

Оператор $H_\sigma^* = H_\sigma$ как в непрерывном, так и в дискретном случае. В

самом деле,

$$\begin{aligned}
(Hf, g) &= \int [Hf](x, y)g(x, y)dxdy = \\
&= \int \left(\int \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-x')^2 + (y-y')^2}{2\sigma^2}\right) f(x', y')dx'dy' \right) g(x, y)dxdy = \\
&= \int \int \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-x')^2 + (y-y')^2}{2\sigma^2}\right) f(x', y')g(x, y)dx'dy'dxdy = \\
&= \int \int f(x', y') \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x'-x)^2 + (y'-y)^2}{2\sigma^2}\right) g(x, y)dx'dy'dxdy = \\
&= \int f(x', y') \left(\int \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x'-x)^2 + (y'-y)^2}{2\sigma^2}\right) g(x, y)dxdy \right) dx'dy' = \\
&= \int f(x', y')[Hg](x', y')dx'dy' = (f, Hg)
\end{aligned}$$

Аналогично доказывается что $\Delta^* = \Delta$.

Далее построим $v = D_h^*u$:

$$\begin{aligned}
(D_h f, u) &= \sum_{i,j} f(ih, jh)u_{i,j} = \\
\int f(x, y) \sum_{i,j} u_{i,j} \delta(x - ih) \delta(y - jh) dx dy &= (f, D_h^* u),
\end{aligned}$$

откуда следует, что

$$v(x, y) = \sum_{i,j} u_{i,j} \delta(x - ih) \delta(y - jh).$$

В дискретном случае

$$[D_s^* u]_{i,j} = \begin{cases} u_{i/s, j/s}, & \text{если } i \text{ и } j \text{ кратны } s, \\ 0, & \text{иначе.} \end{cases}$$

Программная реализация субградиентного метода

Субградиентный метод имеет низкую скорость сходимости (3.1). При программной реализации субградиентного метода особое внимание было уделено выбору начального приближения $z^{(0)}$ и выбору коэффициентов α_k .

При повышении разрешения изображений основной задачей является не нахождение точного решения задачи минимизации, а нахождение приближённого решения приемлемой точности за минимальное количество итераций. Анализ, проведённый в [22], показал, что для повышения разрешения изображения можно использовать шаг вида

$$\alpha_k = \frac{N\gamma_k}{\|g^{(k)}\|_2}, \quad \gamma_k = \gamma_0 q^k, \quad (3.2)$$

где N — число пикселей изображения. Основным параметром здесь является q , отвечающий за скорость уменьшения длины шага.

Субградиентный метод останавливается, когда $\gamma_k < \varepsilon$. При этом количество итераций является фиксированным и зависит только от γ_0 , q и ε . С ростом q точность метода должна повышаться. Однако анализ зависимости метрики ρ_{PSNR} от q показал, что, начиная с некоторого $q = q_0$, значение ρ_{PSNR} перестаёт повышаться. Это связано с тем, что точное решение задачи минимизации отличается от эталонного изображения. Поэтому для получения изображения наилучшего качества достаточно использовать значение $q = q_0$.

В программном комплексе был реализован выбор шага (3.2) с возможностью ручного задания параметров γ_0 и q . При это в качестве значений по умолчанию использовались $\gamma_0 = 25$, $q = 0.875$ и $\varepsilon = 0.1$. Такой выбор последнего параметра обусловлен тем, что при выводе изображения на экран значения пикселей округляются до целого значения.

Анализ выбора зависимости результата от начального приближения показал, что чем больше значение ρ_{PSNR} для начального приближения $z^{(0)}$, тем больше значение ρ_{PSNR} для результата субградиентного метода. Однако с ростом q зависимость от начального приближения снижается.

В общем случае нельзя задать параметры γ_0 , q и начального приближения $z^{(0)}$, которые были бы оптимальными для всех изображений.

3.1.2 Метод минимизации квадратичных функционалов на множестве функций с ограниченной полной вариацией

Рассмотрим задачу минимизации квадратичного функционала

$$F(z) = \|Az - u\|_2^2,$$

где A — линейный непрерывный оператор с $\det A \neq 0$, на множестве функций с ограниченной полной вариацией

$$M_C = \{z : \|z\|_{TV} \leq C\},$$

используемую в задачах подавления эффекта Гиббса (2.8) и повышения резкости изображений (2.9, 2.10).

Для нахождения её решения будем использовать метод минимизации, основанный на методе условного градиента [72].

Метод условного градиента

Метод условного градиента применяется для минимизации функционала $F(z)$ на множестве M — выпуклом многограннике с вершинами $\{T_j\}$.

В методе условного градиента минимизирующая $z^{(k)}$ и вспомогательная $\bar{z}^{(k)}$ последовательности строятся согласно следующему алгоритму:

1. Выбирается некоторое начальное приближение $z^{(0)} \in M$.
2. Пусть известно приближение $z^{(k)}$, $k = 0, 1, \dots$. Тогда вспомогательное приближение $\bar{z}^{(k)}$ полагается равным решению задачи:

$$(F'(z^{(k)}), \bar{z}^{(k)}) = \min_{z \in M} (F'(z^{(k)}), z).$$

Так как функционал $(F'(z^{(k)}), z)$ является линейным, а множество M — выпуклым, то $\bar{z}^{(k)}$ следует искать на границе множества M . Для случая выпуклого многогранника достаточно перебрать все его вершины T_n .

3. Пусть значение $\bar{z}^{(k)}$ найдено. Далее $z^{(k+1)}$ строится по формуле

$$z^{(k+1)} = z^{(k)} + \lambda_k (\bar{z}^{(k)} - z^{(k)}),$$

где $\eta_k \in [0, 1]$ является решением задачи минимизации

$$F(z^{(k+1)}) = F(z^{(k)} + \eta_k(\bar{z}^{(k)} - z^{(k)})) = \min_{\eta \in [0,1]} F(z^{(k)} + \eta(\bar{z}^{(k)} - z^{(k)})).$$

Применение метода условного градиента для задачи подавления эффекта Гиббса

Метод условного градиента неприменим в явном виде к задаче минимизации

$$z_1 = \arg \min_{z \in M_C} F(z), \quad (3.3)$$

так как множество M_C не является ограниченным.

Рассмотрим задачу минимизации

$$z_2 = \arg \min_{z \in M} F(z), \quad (3.4)$$

где M — некоторый выпуклый многогранник.

Построим такое множество $M \subset M_C$, при котором задачи (3.3) и (3.4) будут эквивалентны, в одномерном случае. Решения задач (3.3) и (3.4) существуют и единственны в силу строгой выпуклости минимизируемого функционала и выпуклости множеств M и M_C . Для доказательства эквивалентности достаточно показать, что $z_1 = z_2$. Это равенство выполняется в случае, если $z_1 \in M$.

Используя рассуждения, аналогичные рассуждениям в теореме 3 из раздела 1.3.2, можно показать, что значение $\|Az_1\|_2$ ограничено некоторой константой C_1 , зависящей от $\|u\|$ и C . Обратный оператор A^{-1} существует и ограничен, так как $\det A \neq 0$, а пространство конечномерно. Отсюда следует, что решение (3.3) ограничено:

$$\|z_1\|_2 \leq C_2 = C_1 \|A^{-1}\|$$

и для него справедливо

$$|z_{1,i}| \leq B, \quad i = 1, 2, \dots, n,$$

где B — константа, зависящая от A , $\|u\|$ и C .

Таким образом, мы получили, что z_1 принадлежит ограниченному множеству

$$M_z = \{z \in Z : \max_i |z_i| \leq B\}.$$

Построим в явном виде выпуклый многогранник M , такой, что $M_z \subset M \subset M_C$. В [72] рассмотрено построение вершин многогранника для множества M_C^0 , состоящего из элементов $z \in Z$, удовлетворяющих условию

$$\begin{aligned} |z_2 - z_1| + |z_3 - z_2| + \dots + |z_n - z_{n-1}| &\leq C, \\ z_n &= 0. \end{aligned}$$

Для данного множества вершины $T^{(j)}$, $j = -(n-1), \dots, -1, 1, \dots, n-1$ имеют вид

$$\begin{aligned} T_i^{(j)} &= \begin{cases} C, & i \leq j, \\ 0, & i > j, \end{cases} \quad j = 1, 2, \dots, n-1, \\ T_i^{(-j)} &= -T_i^{(j)}, \quad j = 1, 2, \dots, n-1. \end{aligned}$$

Далее избавимся от условия $z_n = 0$, используя множество M_z^0 , состоящее из элементов $z \in Z$, удовлетворяющих условию

$$\begin{aligned} z_1 = z_2 = \dots = z_n, \\ |z_n| \leq B. \end{aligned}$$

Рассмотрим множество $M = M_C^0 \oplus M_z$. Это множество состоит из всех элементов z , для которых $\|z\|_{TV} \leq C$ и $|z_n| \leq B$, и представляет собой призму с вершинами

$$\begin{aligned} T_i^{(j)} &= \begin{cases} C - B, & i \leq j, \\ -B, & i > j, \end{cases} \quad j = 1, 2, \dots, n-1, \\ T_i^{(j+n-1)} &= \begin{cases} C + B, & i \leq j, \\ B, & i > j, \end{cases} \quad j = 1, 2, \dots, n-1, \\ T_i^{(-j)} &= -T_i^{(j)}, \quad j = 1, 2, \dots, 2n-2. \end{aligned}$$

Очевидно, что $M \subset M_C$ и $M_z \subset M$, а значит, задача (3.4) эквивалентна задаче (3.3). Таким образом, мы свели нахождение решения задачи

(3.3) к нахождению решения задачи (3.4), для которой применим метод условного градиента.

Двухмерный случай

В двухмерном случае построение многогранника является достаточно затруднительным и неэффективным с вычислительной точки зрения.

Мы применяем метод условного градиента только для задачи подавления эффекта Гиббса после повышения разрешения изображения, в которой изображение разбивается на перекрывающиеся блоки небольшого размера. При этом, мы вместо квадратных блоков используем горизонтальные и вертикальные блоки с шириной в один пиксель.

Пусть $Q_H[z, u]$ — оператор подавления эффекта Гиббса на изображении высокого разрешения z при известном изображении низкого разрешения u с помощью метода условного градиента с горизонтальными блоками, $Q_V[z, u]$ — с вертикальными блоками.

Тогда результатом подавления эффекта Гиббса является изображение [66]

$$z' = \frac{1}{2} (Q_H(Q_V(z, u), u) + Q_V(Q_H(z, u), u)).$$

3.1.3 Практический выбор параметра регуляризации для задачи повышения разрешения изображений

До сих пор не была затронута задача выбора параметра регуляризации λ для задач повышения разрешения изображений (1.25), (1.26).

Это связано с тем, что в общем случае параметр регуляризации выбрать нельзя, так как значение ошибки приближённо заданного изображения низкого разрешения u неизвестно. Для каждого изображения параметр регуляризации, при котором будут наилучшими значения метрик оценки качества изображений, будет различаться.

Тем не менее, можно осуществить выбор параметра регуляризации, при котором качество результата повышения разрешения изображений будет близко к наилучшему. Мы используем две стратегии выбора параметра регуляризации: предварительное вычисление параметра регуляризации на основе изображений из базы эталонных изображений и адаптивное вычисление параметра регуляризации на основе анализа самого изображения.

Предварительное вычисление параметра регуляризации

В разделе 1.3.2 была рассмотрена задача выбора наилучшей комбинации нормы невязки и стабилизатора. В ней использовалась база эталонных изображений, в которой для каждого изображения низкого разрешения было известно соответствующее ему эталонное изображение высокого разрешения. При этом для каждой пары изображений вычислялось значение параметра регуляризации, максимизирующего ρ_{SSIM} между эталонным изображением высокого разрешения и результатом повышения разрешения изображения низкого разрешения.

Для выбора параметра регуляризации мы используем предположение, что значение параметра регуляризации, максимизирующего ρ_{SSIM} , будет одинаково для всех изображений из одного класса. Таким образом, достаточно вычислить наилучшее значение параметра регуляризации для базы изображений определённого класса, а затем использовать полученное значение при повышении разрешения всех изображений этого класса.

В таблицах 3.1 и 3.2 приведены средние геометрические значения наилучших параметров регуляризации при использовании различных норм пространств и стабилизаторов при повышении разрешения изображений в 2, 3 и в 4 раза, вычисленные для базы, состоящей из 124 фотографий природы и зданий. Данные значения были использованы в программном комплексе как параметры по умолчанию при выборе конкретной нормы невязки, стабилизатора и коэффициента масштабирования.

Увеличение	в 2 раза		в 3 раза		в 4 раза	
Норма	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$
Стабилизатор						
$\ \Delta z\ _2^2$	0,00328	0,01576	0,01215	0,07164	0,00624	0,01696
$\ z\ _{TV}$	0,04936	0,58840	0,21240	0,60741	0,26688	0,35920
$\ z\ _{BTV}(p=1)$	0,04720	0,28412	0,10665	0,37071	0,13152	0,23600
$\ z\ _{BTV}(p=2)$	0,01124	0,11164	0,03636	0,08343	0,03024	0,06928
$\ z\ _{BTV}(p=3)$	0,01060	0,04492	0,02205	0,04968	0,01856	0,03696

Таблица 3.1: Средние геометрические значения наилучших параметров регуляризации при использовании различных норм пространств и стабилизаторов при повышении разрешения незашумлённых изображений в 2, 3 и в 4 раза.

Увеличение	в 2 раза		в 3 раза		в 4 раза	
Норма	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$	$\ \cdot\ _1$	$\ \cdot\ _2$
Стабилизатор						
$\ \Delta z\ _2^2$	0,01556	0,48404	0,03645	1,92033	0,02912	2,15136
$\ z\ _{TV}$	0,66348	6,75636	0,73980	10,24659	1,48672	14,57408
$\ z\ _{BTV}(p=1)$	0,35892	3,88072	0,50553	5,24529	0,79424	6,79312
$\ z\ _{BTV}(p=2)$	0,09628	1,32696	0,12663	1,39761	0,17952	1,80144
$\ z\ _{BTV}(p=3)$	0,05260	0,78588	0,05805	0,68310	0,08016	0,84352

Таблица 3.2: Средние геометрические значения наилучших параметров регуляризации при использовании различных норм пространств и стабилизаторов при повышении разрешения зашумлённых изображений в 2, 3 и в 4 раза.

Адаптивное вычисление параметра регуляризации

Альтернативой предварительного вычисления параметра регуляризации является вычисление параметра регуляризации на основе анализа самого изображения низкого разрешения u . Данный подход аналогичен подходу предварительного вычисления параметра регуляризации, но при этом база изображений состоит только из изображения u , выступающего в роли эталонного, и его уменьшенной версии.

Данный подход позволяет находить параметр регуляризации для изображений произвольного класса, однако вычислительные затраты на на-

хождение параметра регуляризации могут превышать затраты на собственно повышение разрешения изображений u .

В программном продукте была реализована возможность выбора данного способа задания параметра регуляризации.

3.1.4 Алгоритм определения уровня эффекта Гиббса на изображениях

В работе был предложен метод определения наличия эффекта Гиббса на основе анализа профилей базовых контуров. При выделении профилей контуров в явном виде на дискретных изображениях основной проблемой является необходимость интерполяции при вычислении значений пикселей с вещественными координатами.

При реализации в программном комплексе данный алгоритм был оптимизирован для обработки дискретных изображений.

Алгоритм состоит из следующих шагов:

1. Вычисление средней ширины контуров. На данном шаге производится нахождение контуров с помощью метода Канни [56] с фиксированным параметром фильтра Гаусса $\sigma = 2$. Среди найденных контуров отбираются 50 контуров с наибольшим модулем градиента и вычисляется их средняя ширина.

2. Нахождение базовых контуров M_{BE} и множества точек M_P , для которых ближайший контур является базовым. В качестве параметра метода нахождения базовых контуров d_0 берётся среднее значение ширины границ, вычисленное на предыдущем шаге.

3. Интегральное вычисление значения TV_R (2.6). Вместо того, чтобы выделять профили базовых контуров, вычислять для них взвешенную полную вариацию WTV (2.5), а затем вычислять её среднее значение, мы используем интегральный подход (2.4):

$$WTV(z * G_\sigma, w_{\alpha s}) = \sum_{i,j} |\nabla(z * G_\sigma)_{i,j}| w_{i,j},$$

где

$$w_{i,j} = \begin{cases} w_{\alpha s}(\rho((i, j), M_{BE})), & (i, j) \in M_P, \\ 0, & (i, j) \notin M_P. \end{cases}$$

4. Вычисление оценки уровня эффекта Гиббса R_E (2.7).

3.2 Структура программного комплекса

Программный продукт состоит из трёх модулей:

1. Основной модуль, представляющий из себя платформу-независимую библиотеку классов для анализа и обработки изображений на языке C#

2. Динамическая библиотека на языке C++, содержащая функции для выполнения атомарных операций над изображениями с использованием низкоуровневых оптимизаций в виде использования векторных инструкций и распараллеливания участков кода. Данный модуль является платформу-зависимым, для использования программы в 32-битных и 64-битных системах модуль компилируется под соответствующую платформу.

3. Интерфейсный модуль, позволяющий пользователю загружать, обрабатывать и сохранять изображения. Данный модуль контактирует с основным модулем.

Общий вид интерфейса программного комплекса проведён на рис. 3.1, рис. 3.2 и рис. 3.3.

Описание основных пунктов меню программного комплекса

Меню **File**:

- **Open** — открытие изображение из файла. Поддерживаются форматы BMP, PNG, JPG, TIF, GIF, PGM;
- **Save** — сохранение изображения в файл;
- **Exit** — выход из приложения.

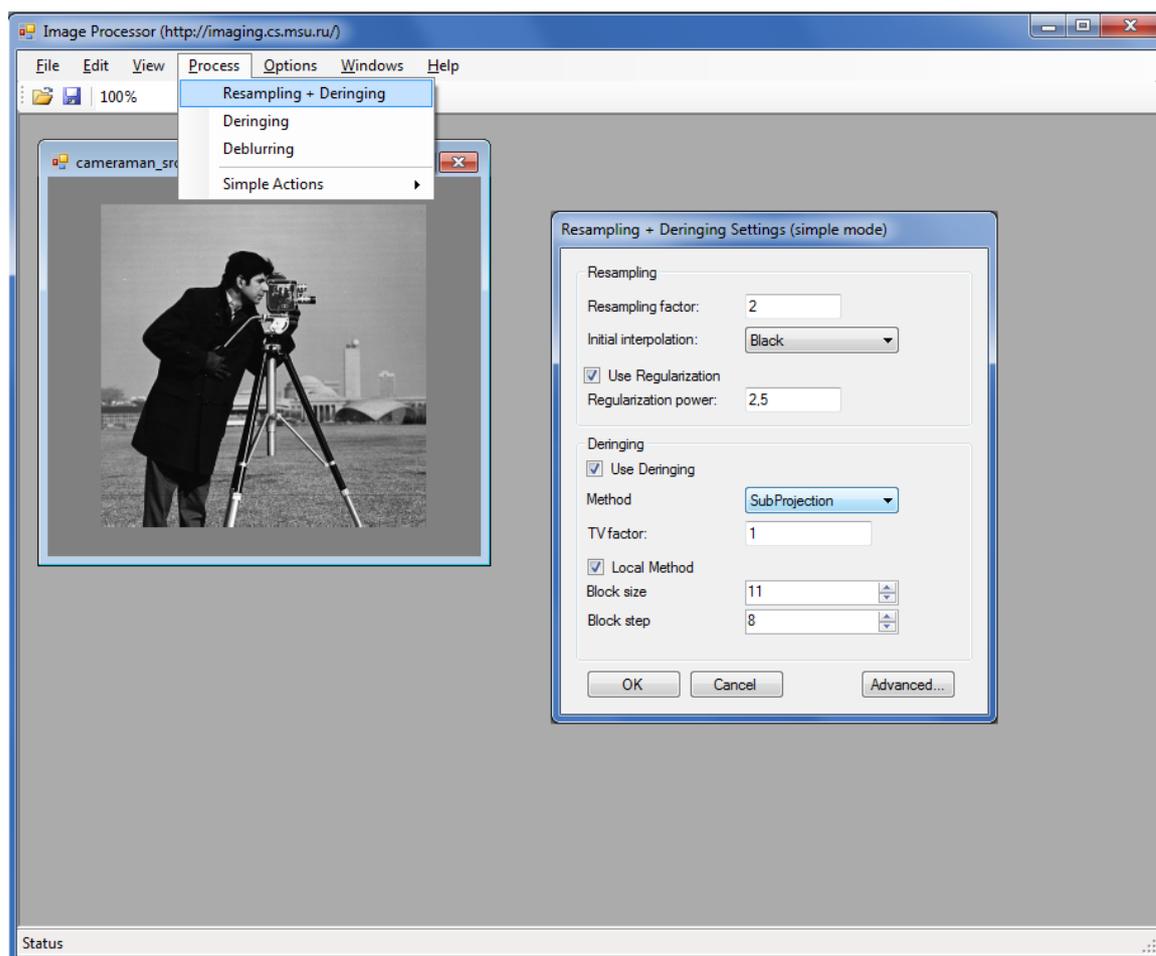


Рис. 3.1: Общий вид интерфейса программного комплекса с открытым окном настроек алгоритма повышения разрешения изображений

Меню **Edit**:

- **Copy** — копирование текущего изображения в буфер обмена;
- **Paste** — создание нового изображения из буфера обмена.

Меню **Process**:

- **Resampling + Deringing** — реализация разработанного алгоритма повышения разрешения изображений с применением алгоритма подавления эффекта Гиббса после повышения разрешения изображений;
- **Deringing** — реализация алгоритма подавления эффекта Гиббса в общем случае;

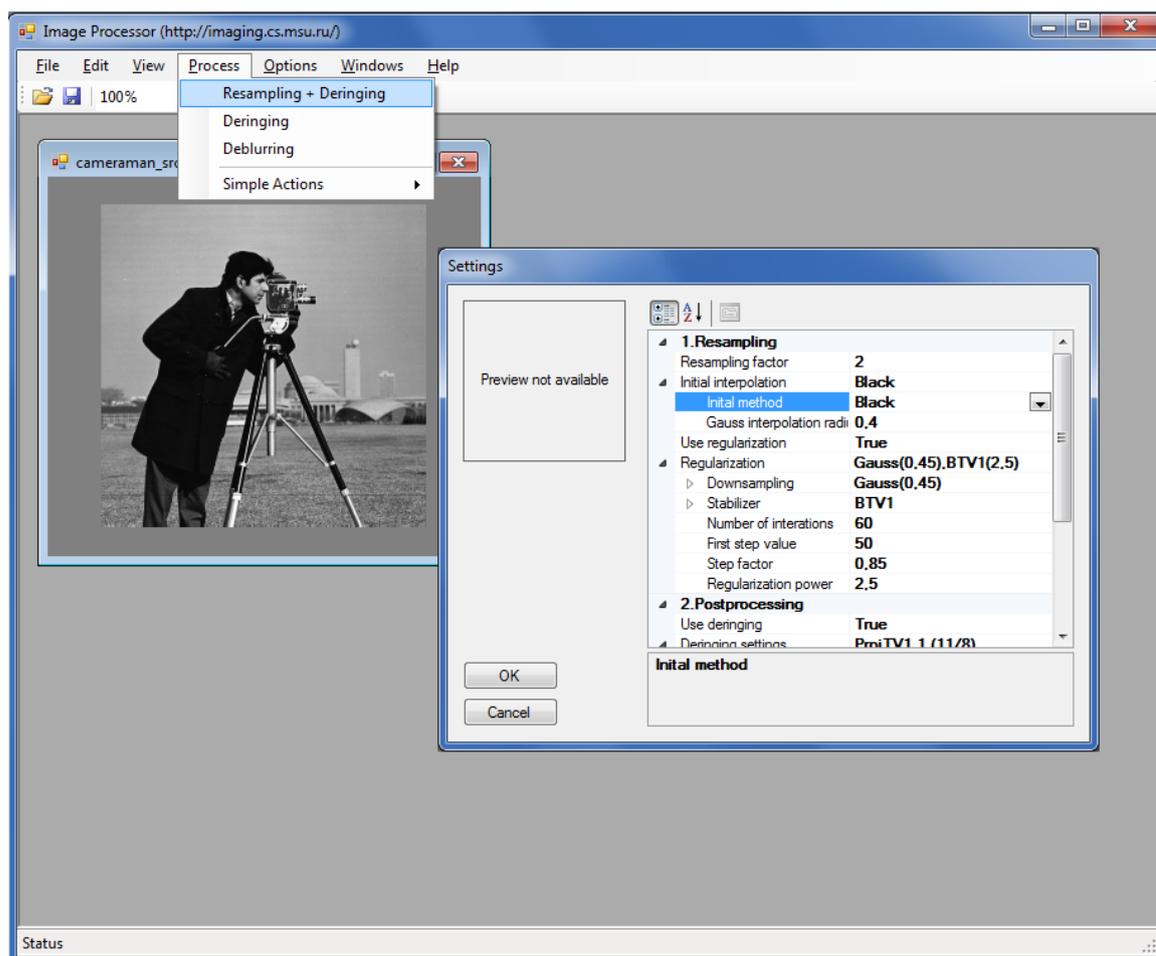


Рис. 3.2: Снимок экрана программного комплекса с открытым окном расширенных настроек алгоритма повышения разрешения изображений

- **Deblurring** — реализация алгоритма автоматического повышения резкости.

Меню **Options**:

- **Multi-thread support** — использование многопоточной реализации используемых алгоритмов, данная настройка включена по умолчанию.

Выбор параметров алгоритмов повышения разрешения изображений

При выборе пункта меню **Resampling + Deringing** открывается окно настроек алгоритма повышения разрешения изображений (см. рис. 3.1), в котором можно выбрать:



Рис. 3.3: Снимок экрана программного комплекса с изображением, обработанными с помощью алгоритма повышения разрешения.

- коэффициент увеличения;
- начальное приближение. В качестве начального приближения доступны: нулевое изображение, метод ближайшего соседа, билинейная интерполяция, бикубическая интерполяция, NEDI, линейная интерполяция с использованием фильтра Гаусса, также можно использовать внешний результат из файла;
- параметр регуляризации;
- использование алгоритма подавления эффекта Гиббса;
- численный метод решения задачи подавления эффекта Гиббса;
- настройки размера и шага окна для метода подавления эффекта

Гиббса после повышения разрешения изображений.

При нажатии на **Advanced** открывается окно расширенных настроек (см. рис. 3.2), в котором можно дополнительно выбрать:

- стабилизатор, по умолчанию используется $BTV(p = 1)$;
- норму невязки;
- число итераций, размера шага;
- параметр σ_0 , используемый при понижении разрешения изображений.

При выборе пункта меню **Deringing** открываются только настройки алгоритма подавления эффекта Гиббса.

Заключение

Основные результаты диссертационной работы заключаются в следующем:

1. Разработаны и обоснованы регуляризирующие алгоритмы повышения разрешения изображений и суперразрешения. Созданы метрики оценки качества алгоритмов повышения разрешения.

2. Предложены и алгоритмически реализованы регуляризирующие методы подавления эффекта Гиббса на изображениях и повышения резкости изображений.

3. Создан программный комплекс для анализа и повышения качества изображений, повышения разрешения и суперразрешения.

Основные результаты докладывались на:

1. 12-й международной конференции и выставке «Цифровая обработка сигналов и её применение» DSPA, Москва, 2010 [48];

2. 10-й международной конференции «Распознавание образов и анализ изображений: новые информационные технологии», Санкт-Петербург, 2010 [55];

3. 20-й международной конференции по распознаванию образов ICPR, Стамбул, Турция, 2010 [47];

4. 16-й международной конференции по обработке изображений ICIP, Каир, Египет, 2009 [39, 58];

5. 5-й международной конференции по обработке изображений и графике ICIG, Сиань, Китай, 2009 [59];

6. 19-й международной конференции по компьютерной графике и зрению Графикон, Москва, 2009 [57, 62];

7. 15-й международной конференции по обработке изображений ICIP, Сан Диего, США, 2008 [66];

8. 8-й международной конференции «Распознавание образов и анализ изображений: новые информационные технологии», Нижний Новгород, 2008 [43];

9. 18-й международной конференции по компьютерной графике и зрению Графикон, Москва, 2008 [33];

10. 9-й международной конференции по обработке сигналов ICSP, Пекин, Китай, 2008 [41];

11. 6-й Курчатовской молодёжной научной школе, Москва, 2008 [65];

12. 17-й международной конференции по компьютерной графике и зрению Графикон, Москва, 2007 [64];

13. 16-й международной конференции по компьютерной графике и зрению Графикон, Новосибирск, 2006 [22].

Основные результаты опубликованы в статьях списка ВАК [14, 32, 34, 42, 73].

Литература

1. Fast and robust multi-frame super-resolution / S. Farsiu, D. Robinson, M. Elad, P. Milanfar // *IEEE Transactions on Image Processing*. — 2004. — Vol. 13, no. 10. — Pp. 1327–1344.
2. Pham T. Q., van Vliet L. J., Schutte K. Robust super-resolution by minimizing a gaussian-weighted ℓ_2 error norm // *Journal of Physics: Conference Series*. — 2008. — Vol. 124, no. 1. — Pp. 12–37.
3. Котельников В. А. О пропускной способности эфира и проволоки в электросвязи // Всесоюзный энергетический комитет. Материалы к I Всесоюзному съезду по вопросам технической реконструкции дела связи и развития слаботочной промышленности. — М.: Управление связи РККА, 1933. — С. 1–19.
4. Zenzo S. D. A note on the gradient of a multi-image // *Journal on Computer Vision, Graphics, and Image Processing*. — 1986. — Vol. 33, no. 1. — Pp. 116–125.
5. Gunturk B. K., Altunbasak Y., Mersereau R. M. Color plane interpolation using alternating projections // *IEEE Transactions on Image Processing*. — 2002. — Vol. 11. — Pp. 997–1013.
6. Тихонов А. Н., Арсенин В. Я. Методы решения некорректных задач. — М.: Наука, 1979. — С. 288.
7. Thevenaz P., Blu T., Unser M. Image interpolation and resampling // Handbook of medical imaging. — Academic Press, Inc., 2000. — Pp. 393–420.
8. Blu T., Thevenaz P., Unser M. Linear interpolation revitalized // *IEEE Transactions on Image Processing*. — 2004. — Vol. 13, no. 5. — Pp. 710–719.
9. Thevenaz P., Blu T., Unser M. Interpolation revisited // *IEEE Transactions on Medical Imaging*. — 2000. — Vol. 19, no. 7. — Pp. 739–758.
10. Turkowski K. Filters for common resampling tasks // Graphics gems. — Academic Press, Inc., 1990. — Pp. 147–165.
11. Young I. T., van Vliet L. J. Recursive implementation of the gaussian filter // *Signal Processing*. — 1995. — no. 44. — Pp. 139–151.

12. Lee Y. J., Yoon J. Nonlinear image upsampling method based on radial basis function interpolation // *IEEE Transactions on Image Processing*. — 2010. — Vol. 19, no. 10. — Pp. 2682–2692.
13. Sun J., Xu Z., Shum H.-Y. Image super-resolution using gradient profile prior // *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2008)*. — 2008. — Pp. 1–8.
14. Krylov A. S., Nasonov A. V. Edge-directed image interpolation using color gradient information // *Lecture Notes in Computer Science*. — 2011. — Vol. 6979. — Pp. 40–49.
15. High accuracy WADI image interpolation with local gradient features / S. Yuan, M. Abe, A. Taguchi, M. Kawamata // *Proceedings of 2005 International Symposium on Intelligent Signal Processing and Communication Systems*. — 2005. — Pp. 85–88.
16. Li X., Orchard M. T. New edge-directed interpolation // *IEEE Transactions on Image Processing*. — 2001. — Vol. 10, no. 10. — Pp. 1521–1527.
17. Leitao J. A., Zhao M., de Haan G. Content-adaptive video up-scaling for high-definition displays // *Proceedings of Image and Video Communications and Processing '2003*. — 2003. — Vol. 5022. — Pp. 612–622.
18. Price J. R., Hayes M. H. Resampling and reconstruction with fractal interpolation functions // *IEEE signal processing letters*. — 1998. — Vol. 5, no. 9. — Pp. 228–230.
19. Wohlberg B., de Jager G. A review of the fractal image coding literature // *IEEE Transactions on Image Processing*. — 1999. — Vol. 8, no. 12. — Pp. 1716–1729.
20. Polidori E., Jean-Luc D. Zooming using iterated function system // *NATO ASI Conf. Fractal Image Encoding and Analysis*. — 1995. — Vol. 5. — Pp. 111–123.
21. Resolution enhancement of images using fractal coding / M. Gharavi-Alkhansari, R. Denardo, Y. Tenda, T. S. Huang // *Visual Communications and Image Processing*. — 1997. — Vol. 3024. — Pp. 1089–1100.
22. Lukin A. S., Krylov A. S., Nasonov A. V. Image interpolation by super-resolution // *Proceedings of GraphiCon'2006*. — 2006. — Pp. 239–242.
23. Aly H. A., Dubois E. Image up-sampling using total-variation regularization with a new observation model // *IEEE Transactions on Image Processing*. — 2005. — Vol. 14, no. 10. — Pp. 1647–1659.
24. Иванов В. К. О линейных некорректных задачах // *ДАН СССР*. — 1962. — Т. 145. — С. 270–272.

25. *Васин В. В.* О связи некоторых вариационных методов приближенного решения некорректных задач // *Математические заметки*. — 1970. — Т. 7, № 3. — С. 265–272.
26. *Rudin L., Osher S. J., Fatemi E.* Nonlinear total variation based noise removal algorithms // *Physica D*. — 1992. — no. 60. — Pp. 259–268.
27. *Irani M., Peleg S.* Super resolution from image sequences // *Proceedings of International Conference on Pattern Recognition (ICPR'90)*. — 1990. — Vol. 2. — Pp. 115–120.
28. *Irani M., Peleg S.* Improving resolution by image registration // *CVGIP: Graphical models and image processing*. — 1991. — Vol. 53, no. 3. — Pp. 231–239.
29. *Ratakonda K., Ahuja N.* POCS based adaptive image magnification // *International Conference on Image Processing*. — 1998. — Vol. 3, no. 3. — Pp. 203–207.
30. *Ozkan M. N., Tekalp A. M., Sezan M. I.* POCS-based restoration of space-varying blurred images // *IEEE Transactions on Image Processing*. — 1994. — Vol. 3, no. 4. — Pp. 450–454.
31. *Танана В. П.* Методы решения операторных уравнений. — М.: Наука, 1981. — С. 156.
32. *Насонов А. В., Крылов А. С., Ушмаев О. С.* Применение метода суперразрешения для биометрических задач распознавания лиц в видеопотоке // *Системы высокой доступности*. — 2009. — Т. 1. — С. 26–34.
33. *Krylov A. S., Nasonov A. V., Sorokin D. V.* Face image super-resolution from video data with non-uniform illumination // *Proceedings of GraphiCon'2008*. — 2008. — Pp. 150–155.
34. *Насонов А. В., Крылов А. С., Ушмаев О. С.* Развитие методов повышения качества изображений лиц в видеопотоке // *Информатика и её применения*. — 2009. — Т. 3, № 1. — С. 19–28.
35. *Малла С.* Вэйвлеты в обработке сигналов. — М.: Мир, 2005. — С. 672.
36. *Колмогоров А. Н., Фомин С. В.* Элементы теории функций и функционального анализа. — М.: Наука, 1972. — С. 496.
37. *Ziemer W. P.* Weakly Differentiable Functions. — New York: Springer-Verlag, 1989.
38. Image quality assessment: from error visibility to structural similarity / *Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli* // *IEEE Transactions on Image Processing*. — 2004. — Vol. 13, no. 4. — Pp. 600–612.

39. Krylov A. S., Lukin A. S., Nasonov A. V. Edge-preserving nonlinear iterative image resampling method // *Proceedings of International Conference on Image Processing (ICIP'09)*. — 2009. — Pp. 385–388.
40. Deriche R., Kornprobst P., Aubert G. Optical-flow estimation while preserving its discontinuities: A variational approach // *Lecture Notes in Computer Science*. — 1996. — Vol. 1035. — Pp. 69–80.
41. Krylov A. S., Nasonov A. V. Fast super-resolution from video data using optical flow estimation // *Proceedings of IEEE International Conference on Signal Processing (ICSP'08), Beijing*. — Pp. 853–856.
42. Krylov A. S., Nasonov A. V., Ushmaev O. S. Video super-resolution with fast deconvolution // *Pattern Recognition and Image Analysis*. — 2009. — Vol. 19, no. 3. — Pp. 497–500.
43. Krylov A. S., Nasonov A. V., Ushmaev O. S. Image super-resolution using fast deconvolution // *Proceedings of 9th Conference on Pattern Recognition and Image Analysis: New Information Technologies, Nizhni Novgorod*. — 2008. — Vol. 1, no. 2. — Pp. 362–364.
44. Sonka M., Hlavac V., Boyle R. *Image Processing, Analysis and Machine Vision*. — Thomson Engineering, Toronto, Canada, 2007.
45. Zomet A., Rav-acha A., Peleg S. Robust super-resolution // *Proceedings of the IEEE Workshop on Applications of Computer Vision*. — 2001. — Pp. 645–650.
46. Sanchez-Beato A., Pajares G. Robust super-resolution using a median filter for irregular samples // *Lecture Notes in Computer Science*. — 2009. — Vol. 5524. — Pp. 298–305.
47. Nasonov A. V., Krylov A. S. Fast super-resolution using weighted median filtering // *Proceedings of International Conference on Pattern Recognition (ICPR'2010)*. — 2010. — Pp. 2230–2233.
48. Насонов А. В., Крылов А. С. Быстрое суперразрешение изображений с использованием взвешенной медианной фильтрации // *Труды 12-й международной конференции и выставки «Цифровая обработка сигналов и её применение» (DSPA'2010)*. — 2010. — Т. 2. — С. 101–104.
49. Brownrigg D. R. K. The weighted median filter // *Communications of the ACM*. — 1984. — Vol. 27, no. 8. — Pp. 807–818.

50. On the performance of human visual system based image quality assessment metric using wavelet domain / A. Ninassi, O. L. Meur, P. L. Callet, D. Barba // *SPIE Human Vision and Electronic Imaging XIII (HVEI'08)*. — 2009. — P. 12.
51. Ginesu F. M. G., Giusto D. A multi-factors approach for image quality assessment based on a human visual system model // *Elsevier Signal Processing: Image Communication*. — 2006. — no. 21. — Pp. 316–333.
52. Perceptual blur and ringing metrics: Application to jpeg2000 / P. Marziliano, F. Dufaux, S. Winkler, T. Ebrahimi // *Signal Processing : Image Communication*. — 2004. — Vol. 19, no. 2. — Pp. 163–172.
53. A ringing metric to evaluate the quality of images restored using iterative deconvolution algorithms / M. Balasubramanian, S. S. Iyengar, J. Reynaud, R. W. Beerman // *Proceedings of the 18th International Conference on Systems Engineering (ICSENG'05)*. — 2005. — Pp. 483—488.
54. Nasonov A. V., Krylov A. S. Edge quality metrics for image enhancement // *Pattern Recognition and Image Analysis*. — 2011. — P. Готовится к публикации.
55. Nasonov A. V., Krylov A. S. Basic edges metrics for image deblurring // *Proceedings of 10th Conference on Pattern Recognition and Image Analysis: New Information Technologies (PRIA2010)*. — 2010. — Vol. 1. — Pp. 243–246.
56. Canny J. A computational approach to edge detection // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. — 1986. — Vol. 8, no. 6. — Pp. 679–714.
57. Nasonov A. V., Krylov A. S. Adaptive image deringing // *Proceedings of GraphiCon'2009*. — 2009. — Pp. 151–154.
58. Nasonov A. V., Krylov A. S. Scale-space method of image ringing estimation // *Proceedings of International Conference on Image Processing (ICIP'09)*. — 2009. — Pp. 2794–2797.
59. Krylov A. S., Nasonov A. V. Adaptive image deblurring with ringing control // *Fifth International Conference on Image and Graphics (ICIG '09)*. — 2009. — Pp. 72–75.
60. Mathematical Morphology and Its Applications to Image and Signal Processing / Ed. by J. Goutsias, L. Vincent, D. S. Bloomberg. — Kluwer Academic Publishers, Boston, 2000. — P. 456.
61. 2D Euclidean distance transforms: A comparative survey / R. Fabbri, L. da F. Costa, J. C. Torelli, O. M. Bruno // *ACM Computing Surveys*. — 2008. — Vol. 40, no. 1. — Pp. 2:1–2:44.

62. Krylov A. S., Nasonov A. V., Chernomorets A. A. Combined linear resampling method with ringing control // *Proceedings of GraphiCon'2009*. — 2009. — Pp. 163–165.
63. Viola P., Jones M. Robust real-time face detection // *International Journal on Computer Vision*. — 2004. — Vol. 57, no. 2. — Pp. 137–154.
64. Nasonov A. V., Krylov A. S., Lukin A. S. Post-processing by total variation quasi-resolution method for image interpolation // *Proceedings of GraphiCon'2007*. — 2007. — Pp. 178–181.
65. Насонов А. В. Программное повышение разрешения и подавление эффекта Гиббса на изображениях // *Сборник аннотаций работ 6-й Курчатовской молодёжной научной школы*. — 2008. — С. 153.
66. Krylov A. S., Nasonov A. V. Adaptive total variation deringing method for image interpolation // *Proceedings of International Conference on Image Processing (ICIP'08)*. — 2008. — Pp. 2608–2611.
67. Chan T. F., Wong C.-K. Total variation blind deconvolution // *IEEE Transactions on Image Processing*. — 1998. — Vol. 7, no. 3. — Pp. 370–375.
68. Oliveira J. a. P., Bioucas-Dias J. M., Figueiredo M. A. T. Review: Adaptive total variation image deblurring: A majorization-minimization approach // *Signal Processing*. — 2009. — Vol. 89. — Pp. 1683–1693.
69. Intel Advanced Vector Extensions Programming Reference. — Intel Ref. 319433-009, 2010.
70. Васильев Ф. П. Численные методы решения экстремальных задач. — М.: Наука, 1988. — С. 552.
71. Boyd S., Xiao L., Mutapcic A. Subgradient methods. — Stanford University: Lecture notes of EE392o, 2003.
72. Регуляризирующие алгоритмы и априорная информация / А. Н. Тихонов, А. В. Гончарский, В. В. Степанов, А. Г. Ягола. — М.: Наука, 1983. — С. 200.
73. Nasonov A. V., Krylov A. S. Finding areas of typical artifacts of image enhancement methods // *Pattern Recognition and Image Analysis*. — 2011. — Vol. 21, no. 2. — Pp. 316–318.