

МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет
«Московский институт электронной техники»

Факультет микроприборов и технической кибернетики
Кафедра высшей математики №1

Панфилова Кристина Владимировна

Магистерская диссертация
по направлению 01.04.01 «Прикладная математика»

Компенсация линейного смаза цифровых изображений с помощью метода
Люси-Ричардсона

Студент _____ (Панфилова К.В.)

Научный руководитель,
профессор, д.ф.-м.н _____ (Умняшкин С.В.)

Москва 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Теоретические сведения	6
1.1 Формулировка задачи	6
1.2 Методы компенсации линейного смаза	8
1.2.1 Обзор существующих методов восстановления изображения	9
1.2.2 Сравнение методов деконволюции	14
1.2.3 Обзор существующих модификаций метода Люси-Ричардсона	17
1.3 Оценка параметров искажения	19
1.3.1 Определение параметров шума	19
1.3.2 Определение параметров смаза	20
1.4 Выводы	23
2 Компенсация линейного смаза на основе метода Люси-Ричардсона	25
2.1 Обнаружение искажённого объекта	25
2.1.1 Оценка на основе блочной компенсации движения	26
2.1.2 Сглаживание контура искажённой области	33
2.2 Устранение краевых артефактов	37
2.2.1 Использование весовой функции Хэмминга	39
2.2.2 Размытие границы маски	40
2.3 Определение оптимального числа итераций	41
2.4 Выводы	44
3 Результаты экспериментов	46
3.1 Восстановление изображения без выделения искажённого объекта .	47

3.2	Поиск искажённой области	47
3.3	Подготовка области движущегося объекта	51
3.3.1	Сглаживание контура области	51
3.3.2	Расширение маски	54
3.4	Процесс восстановления искажённого изображения	56
3.5	Программная реализация	59
3.6	Выводы	60
ВЫВОДЫ И ЗАКЛЮЧЕНИЕ		61
Литература		63
Приложение 1. Результаты обработки изображения «Грузовик»		66
Приложение 2. Результаты обработки изображения «Автомобиль»		74

ВВЕДЕНИЕ

Во многих приложениях компьютерного зрения и машинного обучения важно, чтобы на вход алгоритмов поступали изображения высокого качества. Но из-за несовершенства процесса формирования изображений, возникают искажения, которые могут стать причиной снижения информативности кадра. Поэтому необходимо проводить предварительную обработку изображений, т.е. выполнять их восстановление.

На кадрах видеопоследовательностей при движении объекта съёмки относительно системы регистрации изображения возникает искажение типа линейный смаз. Одной из причин такого искажения может быть неправильно заданная длина выдержки. Существуют фото- и видеокамеры с характеристиками, которые позволяют избежать данной проблемы, но они являются дорогостоящими и сложными в использовании. Поэтому задача компенсации линейного смаза является актуальной. Следующей причиной потери качества снимка является шум, который может возникнуть по ряду причин, например, из-за недостаточного освещения.

Как правило искажению подвергается только движущийся объект. Следовательно, нет необходимости восстанавливать весь кадр целиком. Благодаря этим соображениям, можно избавиться от дополнительных артефактов в неискажённой области и уменьшить вычислительные затраты.

В данной работе будет рассмотрен итерационный метод Люси-Ричардсона и его модификации для восстановления изображений. Также будут предложены дополнения и модификации алгоритма для более точной реконструкции искажённого кадра.

Целью данной работы является разработка метода компенсации линейного смаза на изображении, основанного на методе Люси-Ричардсона. Для достижения этой цели необходимо поставить и решить следующие задачи:

- разработать метод оценки числа итераций алгоритма;
- разработать метод оценки параметров искажения;

- разработать метод выделения искажённого объекта;
- реализовать алгоритм, который будет принимать на вход искажённое изображение и возвращать оценку неискажённого изображения.

В первой главе рассмотрены теория, связанная с методами восстановления, существующими модификациями и методами определения параметров искажения.

Во второй главе описывается использование метода Люси-Ричардсона. Например, каким образом выбирать параметры алгоритма, сколько итераций необходимо выполнить для восстановления, как выделить искажённый объект и другие.

В третьей главе приводятся результаты экспериментов. Определяется модифицированный метод Люси-Ричардсона на основе проведённых экспериментов.

Глава 1

Теоретические сведения

1.1 Формулировка задачи

Изображение — это двумерная проекция трёхменой сцены, подлежащей съёмке. В данной работе под изображением мы понимаем двумерный дискретный сигнал $f(x, y)$, где $0 \leq x \leq M, 0 \leq y \leq N$ (M, N — ширина и высота изображения соответственно). Рассматриваем только полуточновые изображения, где $0 \leq f(x, y) \leq 255$ — яркость пикселя в точке с координатами (x, y) .

Пусть $f(x, y)$ — неискажённое изображение; $g(x, y)$ — искажённое изображение; $h(x, y)$ — импульсная характеристика (ИХ) оператора искажения; $\hat{f}(x, y)$ — оценка изображения $f(x, y)$ и $\eta(x, y)$ — некоррелированный шум. Фурье-образы функций $f(x, y), g(x, y), h(x, y), \eta(x, y)$, полученных с помощью дискретного преобразования Фурье [1, с. 284], обозначим соответственно $F(u, v), G(u, v), H(u, v)$ и $N(u, v)$. Допускаем, что процесс формирования искажённого изображения $g(x, y)$ линейный и его можно описать с помощью линейной дискретной системы [1, с. 403] (см. рис. 1.1):

$$g(x, y) = h(x, y) * * f(x, y) + \eta(x, y), \quad (1.1)$$

где $< * * >$ — операция двумерной свёртки изображения $f(x, y)$ размером $M \times N$ с импульсной характеристикой фильтра $h(x, y)$ размером $m \times n$, которая описывается следующим выражением [1, с. 298]:

$$h(x, y) * * f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b h(s, t) f(x - s, y - t), \quad (1.2)$$

где $a = (m - 1)/2$ и $b = (n - 1)/2$.

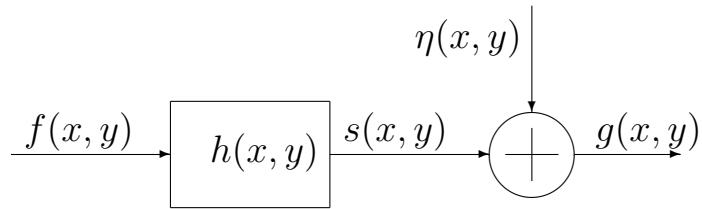


Рис. 1.1: Линейная система искажения изображения

Определение. Двумерная импульсная характеристика (или функция рассеяния точки, ФРТ) — это реакция двумерной дискретной системы на единичный импульс

$$\delta(n, m) = \begin{cases} 1, & n = m = 0 \\ 0, & n \neq 0, m \neq 0 \end{cases}.$$

Таким образом, процесс искажения изображения можно описать как результат взаимодействия исходного изображения $f(x, y)$ с линейно-дискретной системой, изображённой на рисунке 1.1. Сигнал $f(x, y)$ подвергается воздействию оператора смазы с импульсной характеристикой $h(x, y)$, которая обычно не известна заранее. Из-за внешних факторов и несовершенства съёмки к искажению добавляется шум, который считается некоррелированным случайным процессом, также некоррелированным с изображением.

В частотной области процесс искажения (1.1) выглядит следующим образом:

$$G(u, v) = H(u, v) \cdot F(u, v) + N(u, v), \quad (1.3)$$

так как операция свёртки в пространственной области эквивалентна умножению в частотной области [2, с. 39].

Смаз на изображении всегда возникает при относительном движении камеры и объекта. Для определения импульсной характеристики оператора искажения, рассмотрим случай, когда камера перемещается с постоянной горизонтальной скоро-

стью относительно сцены. В дискретном случае [3]:

$$h(x, y) = \begin{cases} \frac{1}{L+1} & 0 \leq y \leq L, x = 0 \\ 0 & \text{в остальных случаях} \end{cases}, \quad (1.4)$$

где L - величина смазы, то есть количество дополнительных точек изображения, соответствующих одной точке объекта. Выражение (1.4) является одним из вариантов функции рассеяния точки.

Частотная характеристика данного оператора искажения определяется выражением [3]:

$$H(u, v) = \frac{1}{L+1} e^{-i(\frac{L\pi}{N}m)} \frac{\sin \frac{\pi(L+1)u}{N}}{\sin \frac{\pi u}{N}}. \quad (1.5)$$

На рисунке 1.2 изображены импульсная и частотная характеристики смазы в 40 пикселей под углом 60^0 . Частотная характеристика состоит из параллельных линий



а — импульсная характеристика, б — амплитудно-частотная характеристика

Рис. 1.2: Смаз величиной в 40 пикселей под углом 60^0

с углом наклона 60^0 . Амплитудно-частотная характеристика (1.5) имеет нули в точках $\frac{N}{L+1}k, v$, где $k = \pm 1, \pm 2, \dots$. Таким образом, на рисунке 1.2б расстояние между параллельными линиями равно $1/40$. Угол наклона линий на рисунке 1.2б характеризует угол смазы [3].

1.2 Методы компенсации линейного смаза

Так как процесс искажения описывается свёрткой, то для восстановления изображения необходимо провести обратную операцию - *деконволюцию*. Методы решения

этой задачи можно разделить в основном на итерационные и неитерационные. Рассмотрим некоторые из них.

Здесь и далее под уровнем искажения DL (Distortion Level) будем понимать следующую величину, выраженную в дБ:

$$DL = 10 \lg \left(\frac{\sum_{(i,j) \in \Omega} (I(i,j) - J(i,j))^2}{\sum_{(i,j) \in \Omega} I(i,j)^2} \right), \quad (1.6)$$

где $I(i,j)$ — значения яркости в пикселе (i,j) оригинального изображения, $J(i,j)$ — яркость в отсчётах искажённого изображения, Ω — анализируемая область. Для оценки уровня шума будем использовать величину SNR (Signal-to-Noise Ratio) — соотношение сигнал/шум:

$$SNR = 20 \lg (\bar{I}/\sigma_\eta), \quad (1.7)$$

где σ_η — среднеквадратичное отклонение шума, а \bar{I} — среднее квадратическое значение для изображения размера $M \times N$: $\bar{I} = \sqrt{\frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N I(x,y)^2}$.

1.2.1 Обзор существующих методов восстановления изображения

Инверсная фильтрация, исходя из информации о формировании изображения и представлении этого процесса в частотной области (1.3), предполагает следующую оценку для спектра восстановленного изображения $\hat{F}(u,v)$ [1, с. 411]:

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}. \quad (1.8)$$

Из (1.3) и (1.8) следует, что

$$\hat{F}(u,v) = F(u,v) + \frac{N(u,v)}{H(u,v)}. \quad (1.9)$$

Инверсная фильтрация обычно непригодна для восстановления зашумлённых изображений: когда спектр $H(u,v)$ искажающего оператора близок к нулю, второе слагаемое в (1.9) вносит очень большой вклад в сумму и искажает изображение ещё сильнее. Видно, что восстановленное изображение 1.3б стало менее качественным

по сравнению с искажённым изображением 1.3а, несмотря на то, что искажение в 6 пикселей под углом 10^0 почти незаметно глазу.



а
а — искажённое изображение ($DL = -65.7$),
б — восстановленное изображение ($DL = 6.64$)

Рис. 1.3: Восстановление изображения размером 450×600 с помощью инверсной фильтрации. Параметры искажения: величина смазы — 6 пикселей, угол — 10^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 10^{-5}$.

От указанного недостатка свободна *винеровская фильтрация*, полученная при решении задачи минимизации среднеквадратичного отклонения e между исходным изображением f и его оценкой \hat{f} [1, с. 414]:

$$e^2 = M\{(f - \hat{f})^2\} \quad (1.10)$$

где $M\{\cdot\}$ — операция математического ожидания. Данный метод в спектральной области описывается следующим соотношением:

$$\hat{F}(u, v) = \left(\frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_\eta(u, v)/S_f(u, v)} \right) G(u, v), \quad (1.11)$$

где $S_f(u, v) = \|F(u, v)\|^2$, $S_\eta(u, v) = M[\|N(u, v)\|^2]$ — энергетические спектры изображения и шума соответственно. Если в искажении не будет участвовать шум (энергетический спектр шума будет равен нулю), то винеровская фильтрация сводится к инверсной.

На практике получить точные значения энергетических спектров шума и исходного изображения не всегда возможно, поэтому в таких случаях используют прибли-

жение формулы (1.11):

$$\hat{F}(u, v) = \left(\frac{1}{H(u, v)} \frac{|H^*(u, v)|^2}{|H(u, v)|^2 + K} \right) G(u, v), \quad (1.12)$$

где K — константа, которая эмпирически подбирается так, чтобы обеспечить наилучшее качество восстановления. Приведём пример работы фильтра Винера в случае искажения с параметрами смазы: 20 пикселей по углом 56° и шумом с параметрами $m_\eta = 0$, $\sigma_\eta = 0.003$ (рис. 1.4а).

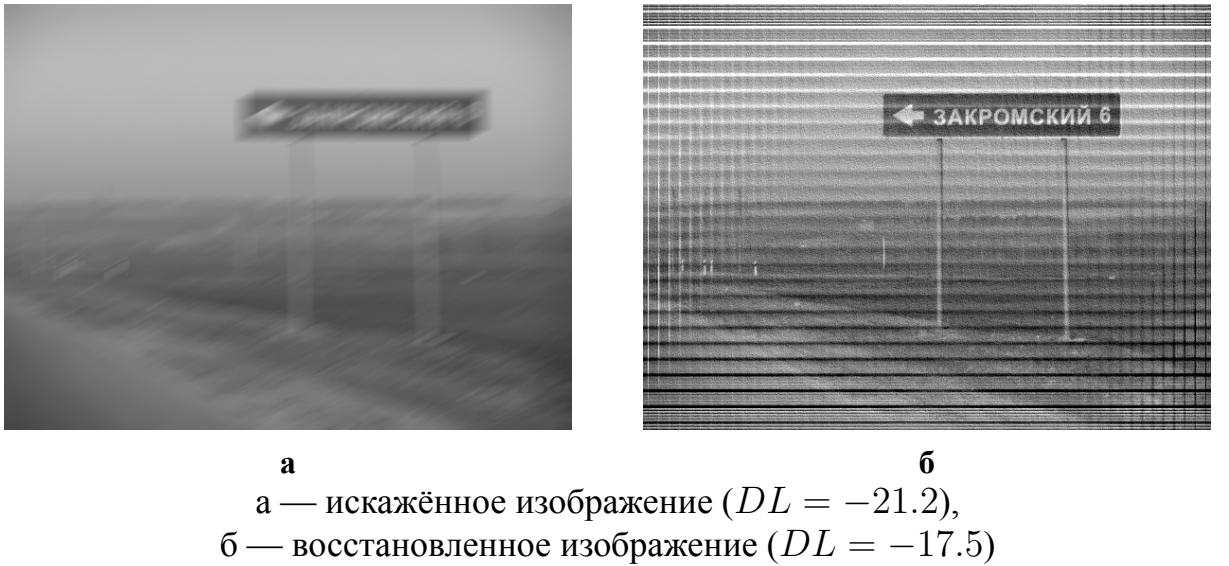


Рис. 1.4: Восстановление изображения размером 450×600 с помощью фильтрации Винера. Параметры искажения: величина смазы — 20 пикселей, угол — 56° ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$ ($SNR = 45$).

К сожалению, с помощью выражения (1.12) не всегда можно получить удовлетворительный результат восстановления, так как значения энергетических спектров сигнала и шума достоверно не известны, а оценка K может оказаться ошибочной. Существуют методы, где достаточно воспользоваться только средним значением шума и его дисперсией. Один из таких методов — *регуляризация по Тихонову*, описываемый в спектральной области соотношением [1, с. 418]:

$$\hat{F}(u, v) = \left(\frac{H^*(u, v)}{|H(u, v)|^2 + \gamma |P(u, v)|^2} \right) G(u, v), \quad (1.13)$$

где от выбора значения параметра регуляризации γ (его выбирают либо эмпирически, либо итеративно) зависит результат восстановления; $P(u, v)$ — двумерное ДПФ

дискретного аналога оператора Лапласа $\nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)$:

$$p(x, y) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \quad (1.14)$$

Если $\gamma = 0$, то фильтрация по Тихонову вырождается в инверсную фильтрацию. Пример работы метода проиллюстрирован на рисунке 1.5.

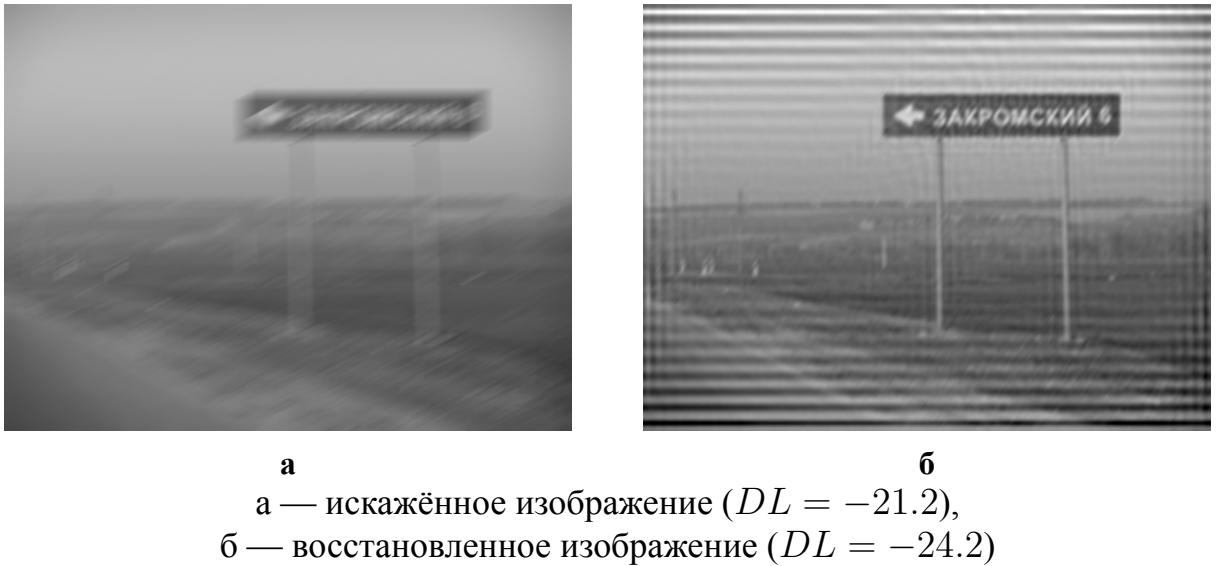


Рис. 1.5: Восстановление изображения размером 450×600 с помощью регуляризации по Тихонову. Параметры искажения: величина смазы — 20 пикселей, угол — 56° ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$ ($SNR = 45$).

Недостатком приведённых неитерационных методов является то, что результат их работы сильно зависит от точности определения ИХ $h(x, y)$. Поэтому чаще используются итерационные методы восстановления, которые в процессе работы уточняют ИХ, хотя и требуют больший объём вычислений. Один из таких методов был предложен независимо друг от друга Л.Б. Люси (1974) [4] и В.Х. Ричардсоном (1972) [5]. Этот подход основан на использовании метода максимального правдоподобия для изображения, моделируемого в виде статистик Пуассона. Формула для оценки неискажённого изображения $\hat{f}_{k+1}(x, y)$ на $(k + 1)$ -ой итерации в *методе Люси-Ричардсона* выглядит следующим образом:

$$\hat{f}_{k+1}(x, y) = \hat{f}_k(x, y) \left(h(-x, -y) * * \frac{g(x, y)}{h(x, y) * * \hat{f}_k(x, y)} \right). \quad (1.15)$$

Пример работы метода Люси-Ричардсона отображён на рисунке 1.6. Заметно, что результат работы итерационного метода превосходит по качеству результаты неитерационных методов.

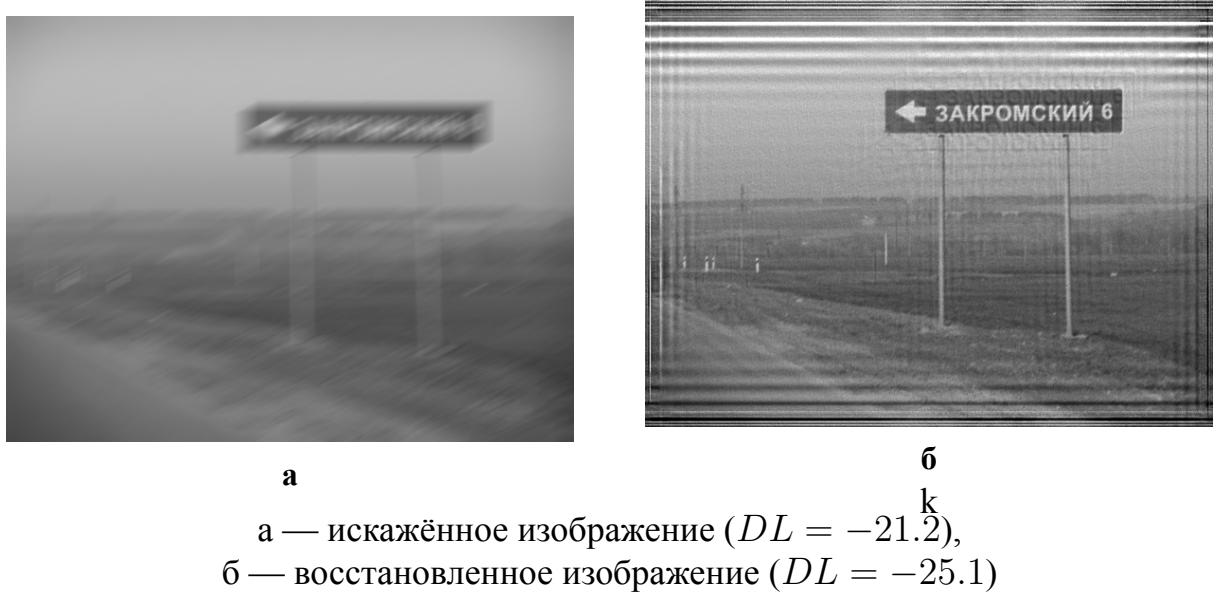
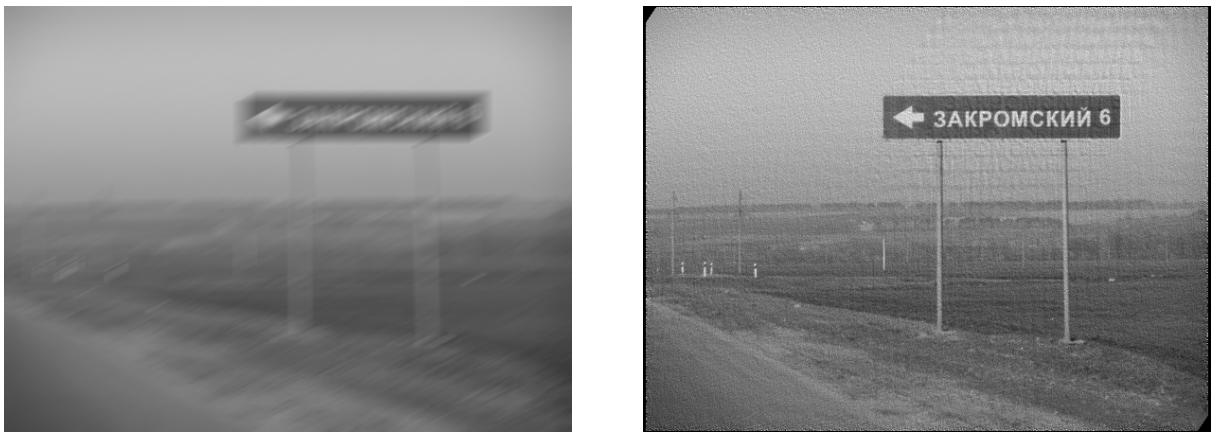


Рис. 1.6: Восстановление изображения размером 450×600 с помощью метода Люси-Ричардсона (50 итераций). Параметры искажения: величина смазы — 20 пикселей, угол — 56^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$ ($SNR = 45$).

Распространение получили также методы *слепой деконволюции* [6, с. 337]. Их основное отличие от метода Люси-Ричардсона заключается в том, что они не требуют знания импульсной характеристики искажающей системы. В основе методов слепой деконволюции также лежит приближение по принципу максимального правдоподобия. Изображение выбирается случайным образом с некоторой вероятностью из семейства других возможных изображений. Необходимо найти максимум функции правдоподобия, которая выражается через функции $g(x, y)$, $f(x, y)$ и $h(x, y)$. Задача оптимизации решается итеративно при выполнении необходимых ограничений и при условии сходимости всей процедуры. Пример работы одного из методов слепой деконволюции, реализованного в расширении среды Matlab *Image Processing Toolbox* [7], изображён на рисунке 1.7.

В MATLAB метод слепой деконволюции на каждой итерации использует восстановление изображение с помощью ядра алгоритма Люси-Ричардсона.



а

б

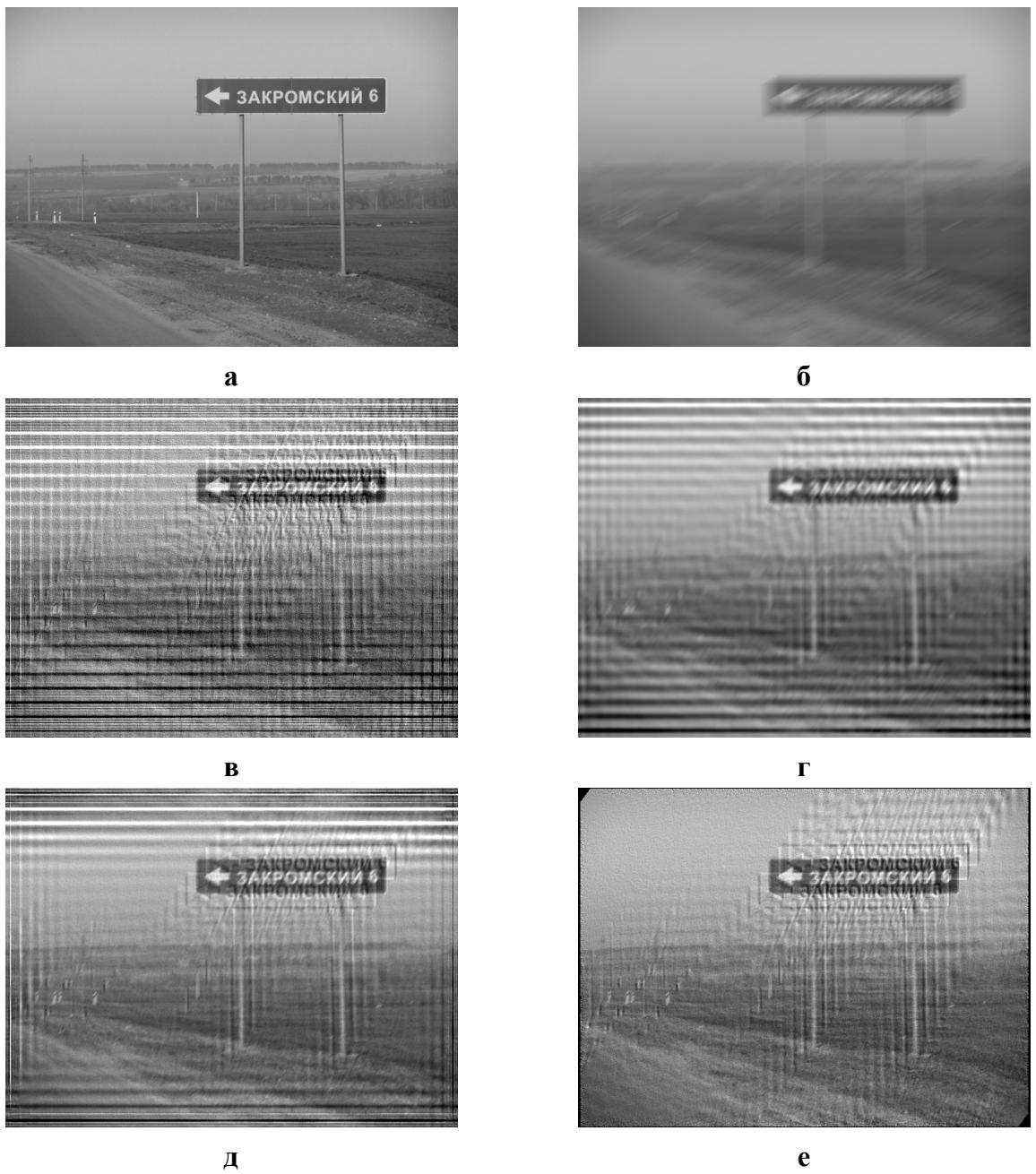
а — искажённое изображение, б — восстановленное изображение

Рис. 1.7: Восстановление изображения размером 450×600 с помощью метода слепой деконволюции (50 итераций). Параметры искажения: величина смазы — 20 пикселей, угол — 56^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$.

1.2.2 Сравнение методов деконволюции

Сравнивая описанные выше методы восстановления изображений на практике, можно сделать вывод, что итерационные методы использовать предпочтительнее, так как они с каждой итерацией улучшают оценку неискажённого изображения $\hat{f}(x, y)$ и могут скомпенсировать погрешности начальных оценок параметров смазы и шума. При работе неитерационных методов эти, даже небольшие, погрешности начальных оценок ухудшают окончательный результат восстановления изображения. Приведём пример работы методов при неточном определении параметров искажения (рис. 1.8). Наблюдаем, что, при восстановлении искажённого изображения (рис. 1.8б) с помощью фильтра Винера (рис. 1.8в), при неточных параметрах искажения дополнительные артефакты затрудняют корректное восприятие информации. При использовании регуляризации по Тихонову (рис. 1.8г) основная информация изображения также потеряна. Итерационные методы (рис. 1.8д и рис. 1.8е) показывают наиболее корректный результат восстановления при неточных параметрах искажения.

По результатам сравнения характеристик пяти упомянутых выше методов восстановления изображения [8, 9, 10], выполненных в среде MATLAB с расширением *Image Processing Toolbox*, для компенсации линейного смаза зашумлённых изображений предлагается использовать метод Люси-Ричардсона. Этот вывод сделан на основании следующих причин:



а — неискажённое изображение, б — искажённое изображение,
в — фильтр Винера, г — регуляризация по Тихонову,
д — метод Люси-Ричардсона (50 итераций), е — метод слепой деконволюции (50 итераций)

Рис. 1.8: Восстановление изображения размером 450×600 . Параметры искажения: величина смазы — 20 пикселей, угол — 56^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$. Параметры искажения, используемые при восстановлении: величина смазы — 23 пикселей, угол — 54^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 0.003$

- Метод позволяет получить более точное восстановление изображения, см. для примера, численные результаты обработки изображения (рис. 1.9б) в таблице 1.1.

- Для реальных изображений процесс восстановления с помощью метода Люси-Ричардсона (1.15) в общем случае не сходится к оригинальному изображению $f(x, y)$. Ограничения метода связаны с необходимостью точного определения ФРТ, влиянии шумов и артефактов, которые присущи итерационному процессу восстановления. Несмотря на это, выполнение N -ого числа итераций метода Люси-Ричардсона (1.15) приближает оценку восстановленного изображения $\hat{f}_{k+1}(x, y)$ к неискажённому $f(x, y)$.
 - Результаты применения методов Люси-Ричардсона и слепой деконволюции зачастую практически идентичны, однако вычислительные затраты могут в разы отличаться в пользу первого метода.



а — неискажённое изображение, б — искажённое изображение

Рис. 1.9: Изображение «Знак» размером 450×600 . Параметры искажения: величина смазы — 30 пикселей, угол — 28^0 ; параметры шума: $m_\eta = 0$, $\sigma_\eta = 10^{-3}$

Таблица 1.1: Результаты восстановления искажённого изображения размером 600×450 со смазом в 30 пикселей под углом 28^0 и с шумом с $m_\eta = 0$, $\sigma_\eta = 10^{-3}$ (SNR = 54дБ). Уровень искажения до восстановления равен -21 дБ

Метод	DL, дБ
Фильтр Винера	-10.7
Регуляризация по Тихонову	-23.2
Алгоритм Люси-Ричардсона (100 итераций)	-25.6
Слепая деконволюция (100 итераций)	-26.1

1.2.3 Обзор существующих модификаций метода Люси-Ричардсона

При оценке результата восстановления искажённого изображения нужно принимать в расчёт не только близость оценки $\hat{f}_{k+1}(x, y)$ к $f(x, y)$, но и необходимые для её получения вычислительные затраты. Существует целый ряд модификаций алгоритма Люси-Ричардсона [4, 5], которые направлены на увеличение его скорости сходимости.

Один из методов увеличения скорости сходимости был предложен Эдвардом Мейнелом [11]. Он заключается в использовании показательного поправочного коэффициента в итерационном процессе (1.15):

$$\hat{f}_{k+1}(x, y) = \hat{f}_k(x, y) \left(h(-x, -y) * * \frac{g(x, y)}{h(x, y) * * \hat{f}_k(x, y)} \right)^k, \quad (1.16)$$

где $k > 1$ — номер итерации.

Другой подход основан на добавлении аддитивного поправочного коэффициента [6, с. 335]

$$\hat{f}_{k+\lambda}(x, y) = \hat{f}_k(x, y) + \lambda g_k(x, y),$$

где $g_k = \varphi(\hat{f}_k) - \hat{f}_k$. Здесь g_k — разностный вектор, используемый в качестве оценки градиента направления, $\varphi(\hat{f}_k)$ — результат $(k + 1)$ -ой итерации алгоритма Люси-Ричардсона (1.15), \hat{f}_k — результат k -ой итерации, λ — коэффициент ускорения. Сложность этого способа заключается в поиске λ , что зачастую может потребовать значительного времени вычисления. Поэтому был предложен более простой метод увеличения скорости сходимости [12]:

$$\hat{f}_k = u_k + \alpha_k h_k, \quad (1.17)$$

где

$$h_k = u_k - u_{k-1}, \quad u_k = \hat{f}_k + g_k, \quad g_k = \varphi(\hat{f}_k) - \hat{f}_k.$$

Таким образом, во время работы алгоритма Люси-Ричардсона (1.17) выполняется экстраполяция: предсказание оценки изображения на текущей итерации основывается на результатах предыдущих. Величина шага α_k определяется на каждой итерации

следующим образом [12]:

$$\alpha_k = \frac{\sum g_{k-1} \cdot g_{k-2}}{\sum g_{k-2} \cdot \sum g_{k-2}}, \quad 0 < \alpha_k < 1, \quad (1.18)$$

где операция « \cdot » — поэлементное умножение, \sum — суммирование по всем элементам массива.

Метод Люси-Ричардсона, как и другие, основанные на методе максимального правдоподобия методы, обычно влечёт усиление шума в восстановленном изображении. Снизить шумовые искажения восстановленного изображения можно несколькими способами: остановить итерационный процесс, когда уровень шума станет недопустимым; выполнить размытие полученного изображения; адаптировать метод, учитывая воздействие шума на процесс восстановления. Более надёжным является последний способ, который сводится к процедуре [6, с. 328]:

$$\hat{f}_k = \hat{f}_k \cdot C, \quad (1.19)$$

где C — модифицированное ядро Люси-Ричардсона. Ядро C находится следующим образом:

$$C = p(x, y)^{\gamma-1} \cdot (\gamma - (\gamma - 1)p(x, y)), \quad (1.20)$$

где

$$p(x, y) = \min \left[-\frac{2}{t^2} \left(g(x, y) \ln \frac{g(x, y)}{J(x, y)} - J(x, y) + g(x, y) \right), 1 \right], \quad (1.21)$$

а t — пороговое значение, которое определяет уровень затухания (удобно, например, выбирать $t = \sigma/10$, где σ — стандарт deviation шума, участвующего в искажении), $J(x, y) = h(x, y) * \hat{f}_k(x, y)$ — оценка искажённого изображения, $g(x, y)$ — искажённое изображение, поступающее на вход алгоритма. Из эмпирических соображений предлагаются принимать $\gamma = 10$ [6, с. 328].

Рассмотренные модификации позволят получить более точную оценку изображения по сравнению со стандартным методом Люси-Ричардсона (1.15). Далее для восстановления изображения будем использовать итерационные процедуры (1.17) и (1.19).

1.3 Оценка параметров искажения

1.3.1 Определение параметров шума

Методы восстановления изображения требуют информацию о параметрах шума, который воздействует на изображение. Для того чтобы определить эти параметры, введём следующие предположения:

- Шум является аддитивным гауссовым белым шумом, то есть его спектр мощности — постоянная величина, а отсчёты некоррелированы между собой.
- Шум не коррелирует с изображением и не зависит от пространственных координат.

Задача состоит в построении оценки для σ_η — среднеквадратичного отклонения шума. Будем решать её в частотной области, используя фурье-спектра изображения. Из рисунка 1.10 видно, что присутствие шума можно обнаружить по спектру изображения даже визуально (пики особенно видны по углам спектра). Но этого недостаточно для конкретного определения параметров шума.

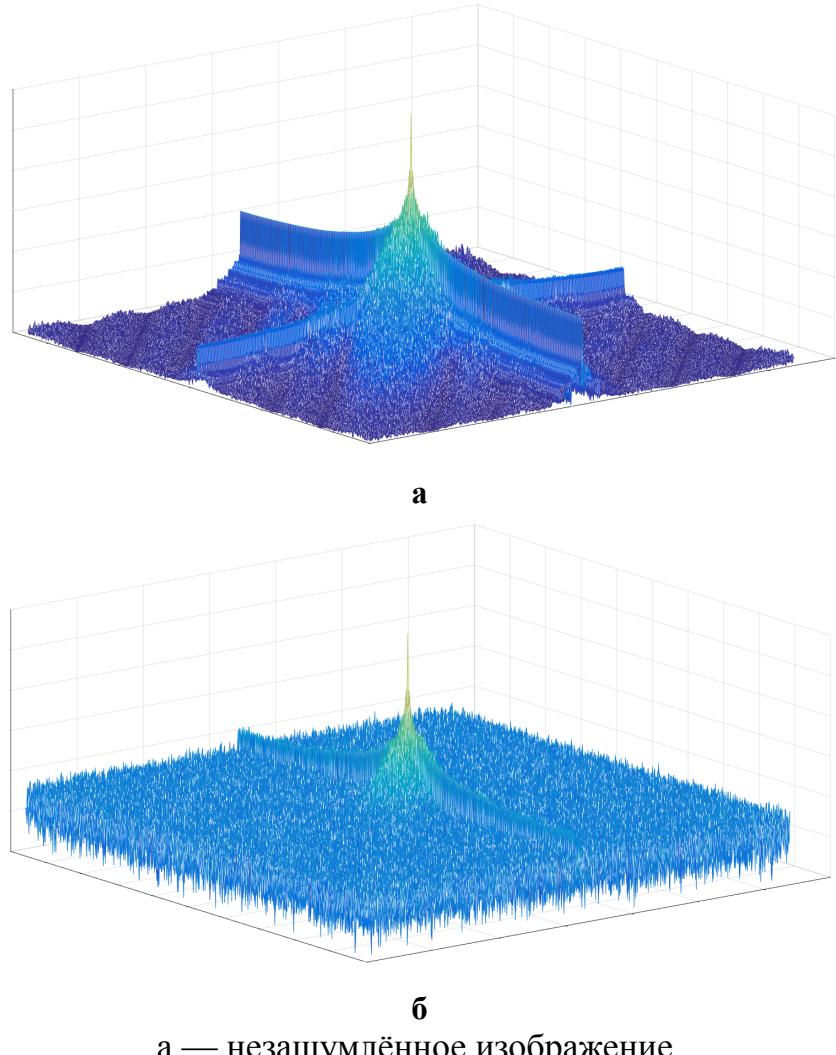
Рассмотрим процесс искажения в частотной области (1.3):

$$G(u, v) = H(u, v) \cdot F(u, v) + N(u, v).$$

Как известно, основная информация изображения сосредоточена в области низких частот (см. рисунок 1.10а). То есть спектр изображения можно считать ограниченным некоторой частотной полосой, а основную энергию спектра локализованной в области низких частот. Тогда значения высокочастотных компонент спектра изображения близки к нулю, $F(u, v) \approx 0$, и вкладом первого слагаемого в (1.3) можно пренебречь в этой области. Поэтому в некоторой области верхних частот Ω : $|G(u, v)|^2 \approx S_\eta(u, v)$ (см. рисунок 1.10б). Известно, что для дискретного белого шума $S_\eta(u, v) = \sigma_\eta^2$, поэтому для оценки можно воспользоваться следующей формулой:

$$\sigma_\eta \approx \sqrt{\frac{1}{|\Omega|} \sum_{(i,j) \in \Omega} |G(i, j)|^2}, \quad (1.22)$$

где Ω — рассматриваемая высокочастотная область, а $G(u, v)$ — спектр искажённого изображения $g(x, y)$.



а — незашумлённое изображение,

б — зашумлённое изображение (параметры шума: $m_\eta = 0$, $\sigma_\eta = 10^{-2}$)

Рис. 1.10: Спектр изображения подверженного оператору смазы: 10 пикселей под углом 45^0

1.3.2 Определение параметров смазы

Рассмотрим искажённое изображение с параметрами смазы: 50 пикселей под углом 30^0 (рис. 1.11б).

Как уже было сказано ранее, по спектру искажённого изображения визуально можно определить и величину и угол смазы. Рассмотрим рисунок 1.12, на котором представлен спектр искажённого изображения 1.11б. Расстояние между чёрными полосами, в которых значение спектра равно нулю, является обратным к величине сма-



a

б

а — оригинальное изображение, б — искажённое изображение

Рис. 1.11: Величина смаза — 50 пикселей, угол — 30^0

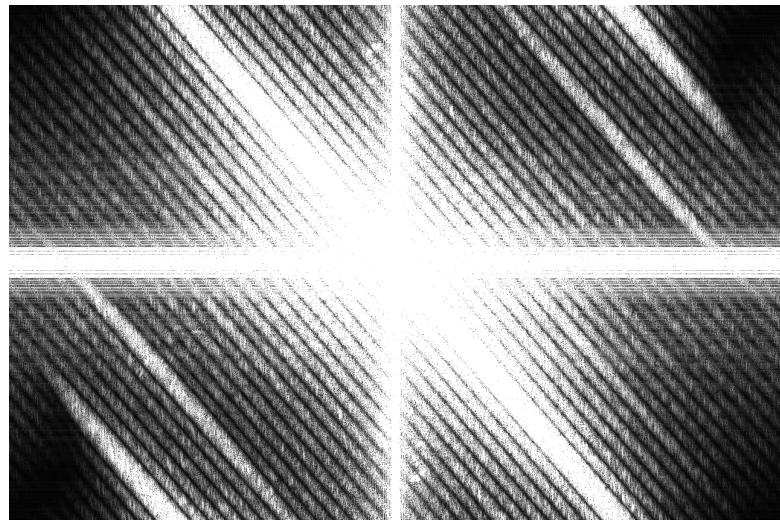


Рис. 1.12: Спектр искажённого изображения с величиной смаза — 50 пикселей и углом — 30^0

за. Также эти полосы ортогональны вектору смаза, угол наклона которого равен 30^0 . Однако, определение точных значений этих величин по частотному представлению изображения затруднительно. Для корректной работы методов восстановления изображения важно как можно точнее оценить ФРТ. Поэтому используют более точные способы определения параметров смаза.

Один из способов оценки параметров оператора линейного смаза основан на использовании кепстрапа изображения $\hat{f}(x, y)$ [3]:

Определение. Кепстр изображения $\hat{f}(x, y)$ — это обратное преобразование Фурье

от логарифма амплитуды спектра изображения.

$$\hat{f}(x, y) = \mathcal{F}^{-1}\{(\log |F(u, v)|)\}, \quad (1.23)$$

где $F(u, v) = \mathcal{F}\{f(x, y)\}$ — частотное представление изображения. Название «кепстр» было получено перестановкой первых четырёх букв слова «спектр».

Одним из основных свойств кепстера является то, что при свёртке двух сигналов их кепстры складываются. То есть, если

$$g(x, y) = h(x, y) * f(x, y), \quad (1.24)$$

то [3]

$$\hat{g}(x, y) = \hat{h}(x, y) + \hat{f}(x, y). \quad (1.25)$$

При линейном горизонтальном размытии частотную характеристику искажения можно записать в виде (1.5):

$$H(u, v) = \frac{1}{L+1} e^{-i(\frac{L\pi}{N}m)} \frac{\sin \frac{\pi(L+1)u}{N}}{\sin \frac{\pi u}{N}}.$$

Она имеет нули в точках $\frac{N}{L+1}k, v$, где $k = \pm 1, \pm 2, \dots$, поэтому $\hat{h}(x, y)$ имеет большой отрицательный пик на расстоянии L от начала координат. Следовательно, кепстр искажённого изображения $\hat{g}(x, y)$ тоже обладает таким пиком, который сообщает о наличии искажения и его расположении.

На рисунке 1.13 изображён кепстр искажённого изображения 1.11б. В центре находится координата $(0, 0)$. Симметрично относительно неё расположены пики, характеризующие параметры смазы. Найдя их координаты, можем рассчитать величину и угол смазы с помощью выражений:

$$r = \sqrt{x^2 + y^2}, \quad (1.26)$$

$$\varphi = \operatorname{arctg} \frac{y}{x}, \quad (1.27)$$

где x — x-координата первого минимума кепстера, y — y-координата первого минимума кепстера, r — величина смазы, φ — угол смазы.

Такой способ позволяет определить искажение достаточно точно, различие с на-

стоящими параметрами в абсолюте не превосходит одной-двух единиц.

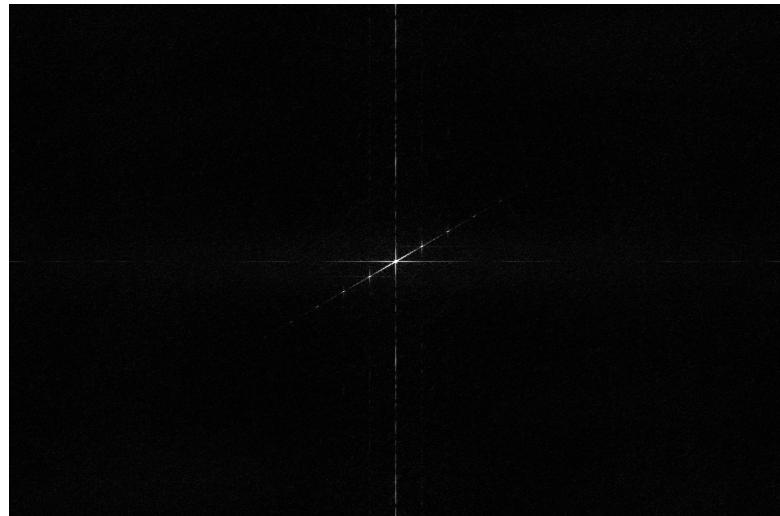


Рис. 1.13: Кепстр изображения с параметрами искажения: величина смаза — 50 пикселей, угол — 30^0

Нахождение параметров смаза с помощью кепстрального представления изображения обладает следующим недостатком: в случае небольшого искажения (длина вектора размытия меньше 5 единиц), точно определить смаз затруднительно. Поэтому, если величина смаза достаточно мала, то определить параметры искажения с помощью кепстрального представления сложно. Такие искажения мы исключаем из рассмотрения, так как они визуально незаметны на изображении. Также при небольшом смазе может возникнуть проблема с определением угла смаза вблизи осей.

Примечание: При больших шумах ($\sigma_\eta \geq 2 * 10^{-2}$) определение параметров размытия по кепстру становится невозможным, так как вклад шума слишком велик и зачастую максимум кепстра приходится именно на случайные составляющие шума, что видно на рисунке 1.14.

1.4 Выводы

Было рассмотрено 5 основных методов восстановления изображения, проведено их сравнение и описаны модификации. Также были рассмотрены методы определения параметров искажения. По каждому из разделов были сделаны выводы — обобщим их:

- неитерационные методы являются малопригодными для восстановления реаль-

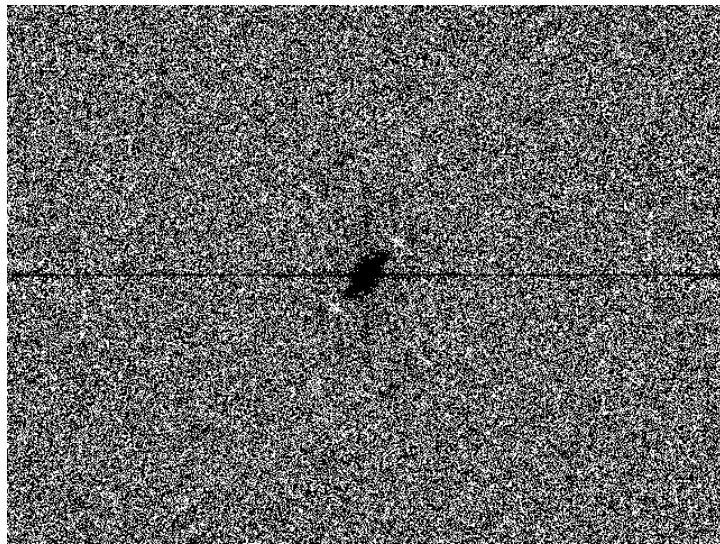


Рис. 1.14: Кепстр изображения с параметрами искажения: величина смазы — 40 пиксейл, угол — 45^0 , шум $\sigma_\eta = 6 * 10^{-2}$

ных искажённых изображений;

- для компенсации линейного смаза искажённых изображений предлагается использовать метод Люси-Ричардсона;
- использование модификаций метода Люси-Ричардсона по увеличению скорости сходимости и снижении воздействия шума позволяет получить более качественный результат восстановления;
- оценка шума основана на том, что шум является белым и не коррелирует с изображением;
- для оценки оператора смаза предлагается использовать кепстр изображения.

Глава 2

Компенсация линейного смаза на основе метода Люси-Ричардсона

2.1 Обнаружение искажённого объекта

Зачастую на изображении искажена только некоторая область, которая соответствует объекту, совершающему движение относительно камеры.



Рис. 2.1: Пример реального искажённого изображения

Например, на рисунке 2.1 искажению типа смаз подвергается только машина, которая находится на переднем плане. Остальная область изображения является ста-

тичной во время съемки кадра и воздействию искажения не подвергается. Предлагается восстанавливать только область соответствующую движущемуся объекту по следующим причинам:

- восстанавливая только фрагмент изображения, можно избежать дополнительных артефактов в ранее неискаженных областях. Например, в результате восстановления изображения 2.1 элементы фона и статичные объекты были искажены (рис. 2.2), что затрудняет анализ изображения;
- восстановление части изображения требует меньших вычислительных затрат, чем восстановление всего изображения.



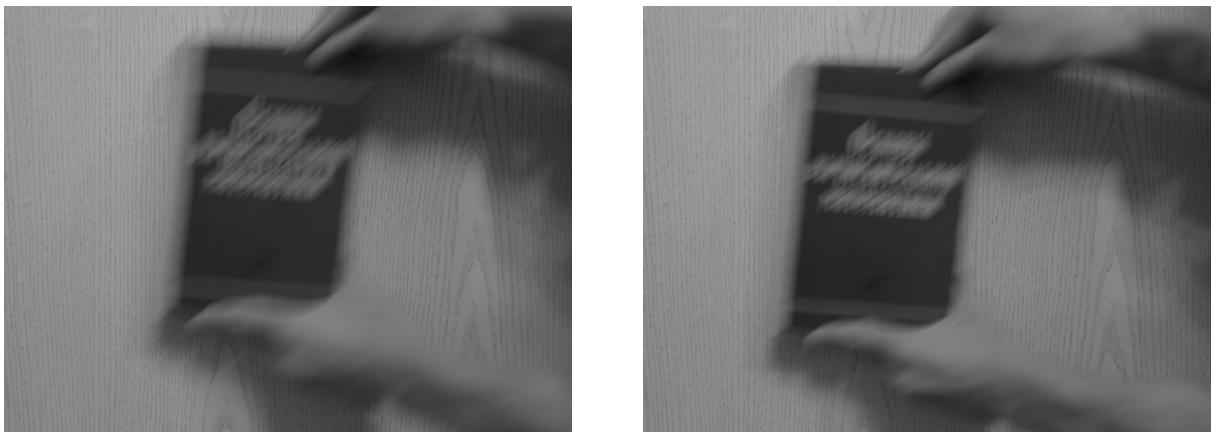
Рис. 2.2: Пример восстановления изображения с параметрами искажения: 15 пикселей под углом 86^0 с шумом $\sigma_\eta = 0.003$

2.1.1 Оценка на основе блочной компенсации движения

Для того чтобы оценить область искажения в кадре, воспользуемся идеей *блочной компенсации движения*, которую используют в алгоритмах видеокомпрессии [13].

В основе данного подхода лежит поиск векторов перемещения для фрагментов изображения, использующий предыдущий кадр в видеопоследовательности (рис. 2.3а и рис. 2.3б).

Текущий кадр (рис. 2.3б) обозначим за I_1 , а опорный кадр, который был перед ним (см. рис. 2.3а) — I_0 . Разобьём I_0 и I_1 на непересекающиеся квадратные блоки размера $S \times S$. Размер блока выбираем в зависимости от размеров изображения: для



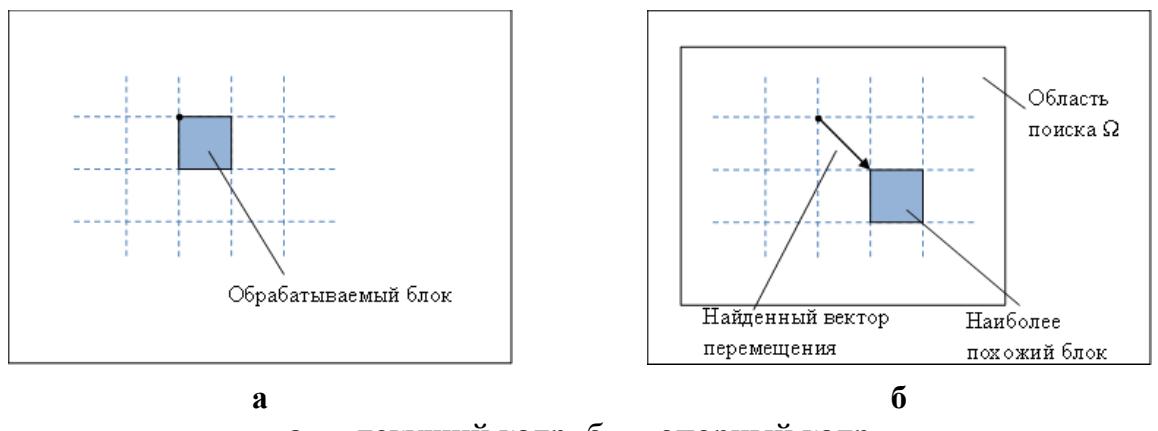
а

б

а — опорный кадр размером 640×480 , б — текущий кадр размером 640×480

Рис. 2.3: Два последовательных кадра видеопоследовательности

изображений размером порядка 320×240 выбираем размер блока 8×8 , 640×480 — 16×16 , 1920×1080 — 32×32 пикселов. Данные зависимости размера блока от размера изображения предлагаются из эмпирических соображений.



а

б

а — текущий кадр, б — опорный кадр

Рис. 2.4: Поиск вектора перемещения для блока из текущего кадра

Для каждого блока $I_1(x, y)$ текущего кадра ищем наиболее похожий блок $I_0(k, l)$ на опорном изображении I_0 (рис. 2.4). Процедура поиска похожего блока проводится в некоторой окрестности Ω текущего положения блока (рис. 2.4б). Степень схожести блоков определяется с помощью функции отклонения $D(I_0(k, l), I_1(x, y))$, которая представляет собой расстояние в пространстве матриц размерности $S \times S$. В качестве

меры D используем сумму модулей разностей пикселов блоков:

$$D(I_0(k, l), I_1(x, y)) = SAD(I_0(k, l), I_1(x, y)) = \\ = \sum_{n=0}^S \sum_{m=0}^S |I_0(Sk + n, Sl + m) - I_1(Sx + n, Sy + m)|, \quad (2.1)$$

где размер блока — $S \times S$.

Чтобы определить наиболее похожий блок необходимо минимизировать функцию отклонения D . Далее, чтобы выделить блоки, соответствующие перемещённым фрагментам опорного кадра, проведём пороговую обработку, рассматривая только те блоки $I_1(x, y)$, для которых в опорном кадре нашёлся некоторый блок $I_0(k, l)$ с модулем вектора перемещения (2.2)

$$r = \sqrt{(Sk - Sx)^2 + (Sl - Sy)^2} > 5. \quad (2.2)$$

В противном случае блок $I_1(x, y)$ считается неперемещённым.

На рисунке 2.5а приведён пример изображения, где белые пиксели соответствуют перемещённым блокам, а остальные — неперемещённым. Выделенный объект на изображении представлен на рис. 2.5б.



Рис. 2.5: Пример обнаружения искажённого объекта

Дополнительные трудности в определении перемещённых блоков могут вызывать тени на изображении и изменение яркости кадра. Например, облако на небе

в солнечную погоду отбрасывает тень на земную поверхность и может влиять на освещённость кадра. Для того, чтобы тени и изменение яркости не оказывало дополнительного воздействия на поиск перемещённых блоков, в процессе вычисления $D(I_0(k, l), I_1(x, y))$ не будем учитывать постоянную составляющую блоков $I_0(x, y)$ и $I_1(k, l)$. Преобразуем выражение (2.1):

$$D(I_0(k, l), I_1(x, y)) = \sum_{n=0}^S \sum_{m=0}^S \left| (I_0(Sk + n, Sl + m) - \bar{I}_0(k, l)) - (I_1(Sx + n, Sy + m) - \bar{I}_1(x, y)) \right|, \quad (2.3)$$

где $\bar{I}(x, y) = \frac{1}{S^2} \sum_{n=0}^S \sum_{m=0}^S I(Sx + n, Sy + m)$ — постоянная составляющая блока $I(x, y)$, $S \times S$ — размер блока.



Рис. 2.6: Пример обнаруженияискажённого объекта неучитываяпостояннуюсоставляющуюблоков (2.3)

В результате получаем более точную оценку области перемещения объекта (см. рис. 2.6).

После отсечения неперемещённых блоков по правилу $r > 5$ на изображении ещё могут присутствовать пиксели относящиеся к другим объектам, поэтому проведём сегментацию движения аналогичную пороговой обработке [1, с. 854].

Определим правило, по которому блок будет относиться к области перемещённого объекта.

Каждый блок имеет вектор перемещения $\mathbf{r} = (x, y)$. После определения пере-

мешённых блоков на текущем кадре I_1 имеем M векторов $\mathbf{r}^i = (x_i, y_i)$. Для того, чтобы рассматривать только блоки перемещённого объекта, необходимо определить свойство, которое будет его характеризовать. Предлагается использовать статистические свойства векторов перемещения блоков. Рассмотрим вектор перемещения $\mathbf{m}_\mathbf{r} = (m_x, m_y)$, где m_x, m_y — медианы векторов \mathbf{r}_x и \mathbf{r}_y соответственно¹. Вектор \mathbf{r}_x состоит из x -компонент векторов \mathbf{r}^i , т.е. $\mathbf{r}_x = (x_1, x_2, x_3, \dots, x_M)$, где $i = 1, 2, \dots, M$. Аналогично, вектор $\mathbf{r}_y = (y_1, y_2, y_3, \dots, y_M)$.

Проведём сегментацию движения следующим образом. К перемещённому объекту будем относить те блоки, для которых выполняется условие:

$$\| \mathbf{r}^i - \mathbf{m}_r \| < \delta \| \mathbf{m}_r \|, \quad (2.4)$$

где параметр δ определяется эмпирически.

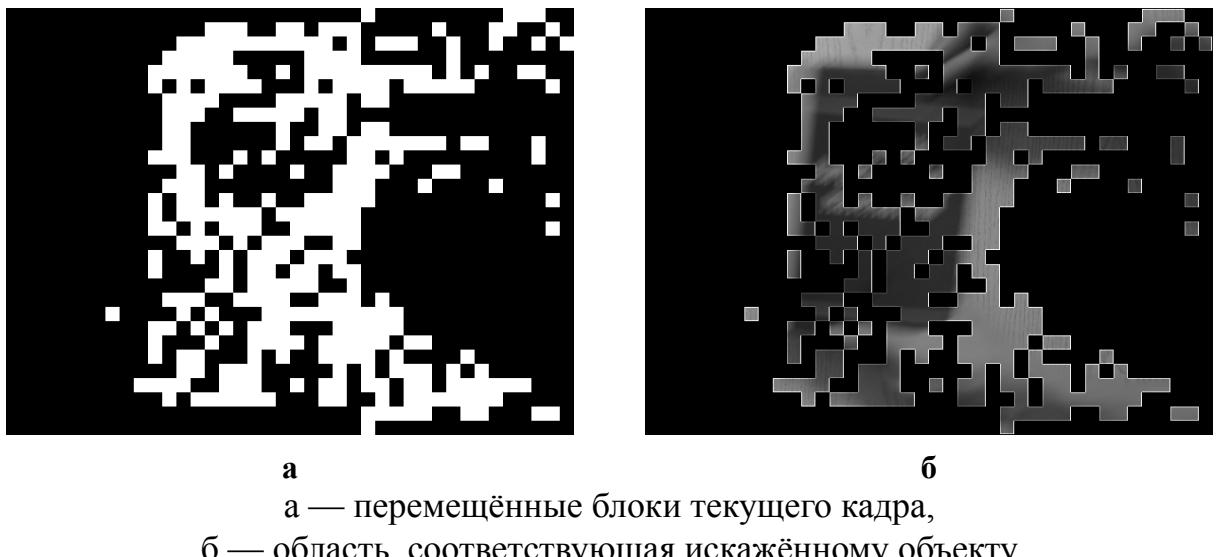


Рис. 2.7: Результат сегментации движения (2.4). $\delta \equiv 3$, $m_r \equiv (16, -16)$

В результате получаем сегментированные области движения (рис. 2.7). Кроме того, сегментация по движению, использующая статистические свойства блоков перемещения, в дальнейшем может позволить находить области с разными искажениями и восстанавливать их независимо друг от друга. То есть, такая сегментация фактически будет решать задачу распознавания движения. Например, если два транспортных средства движутся в разных направлениях (рис. 2.8), результатом сегмен-

¹Медиана — число, характеризующее набор чисел. Медиану можно найти, отсортировав элементы вектора по возрастанию или убыванию и взяв средний элемент.

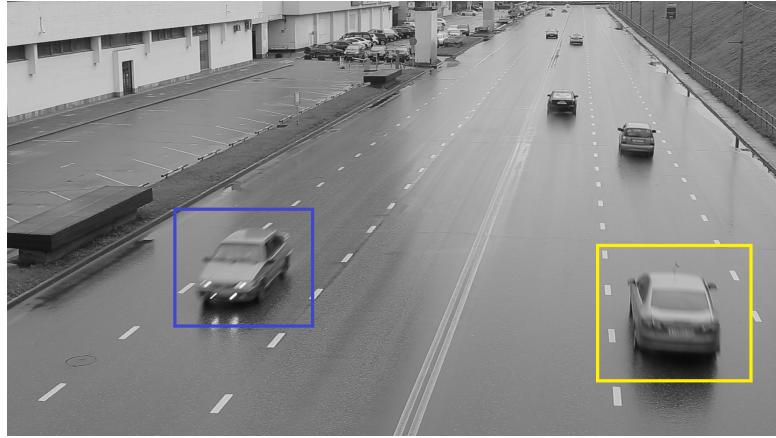


Рис. 2.8: Пример сложной структуры искажения

ции должны быть две области, соответствующие автомобилям. Каждый искажённый фрагмент изображения должен быть восстановлен независимо друг от друга.

Но даже после сегментации, область искажения может быть определена неточно по следующим причинам:

- при больших смазах некоторые блоки не определяются как перемещённые из-за большого размытия изображения, например на рис. 2.3 надпись на книге является искажённой, но не определяется как перемещённая (см. рис. 2.7б);
- если некоторая область изображения имеет сложную текстуру (например, фон на рис. 2.3), то при блочной компенсации движения она может вести себя непредсказуемо из-за своих свойств и шумового воздействия;
- в результате несущественного движения (например, колебание листвы деревьев при ветре) или шумового воздействия могут быть определены ошибочные перемещённые блоки.

Для решения данных трудностей предлагается проводить морфологическую обработку. Для этого формируется вспомогательное бинарное изображение B , каждый пиксель которого принимает значение $B(x, y) = 1$, если соответствующий блок размером $S \times S$ текущего кадра $I_1(x, y)$ считается перемещенным; в противном случае $B(x, y) = 0$. В качестве морфологической обработки предлагается провести морфо-

логическое сглаживание [1, с. 777] с примитивом типа крест размером 3×3 пикселя:

0	1	0
1	1	1
0	1	0

Морфологическое сглаживание представляет собой последовательное применение морфологического размыкания (2.7) и замыкания (2.8) [1, с. 737].

Определение. Эрозия изображения f по плоскому примитиву b в произвольной точке (x, y) [1, с. 772]:

$$(f \ominus b)(x, y) = \min_{(s,t) \in b} f(x + s, y + t). \quad (2.5)$$

Таким образом, для нахождения эрозии f по b необходимо последовательно помещать центр b в каждый пиксель изображения f и находить в этом положении минимальное значение f в точках окрестности, покрываемой примитивом.

Определение. Дилатация изображения f по плоскому примитиву b в произвольной точке (x, y) [1, с. 772]:

$$(f \oplus b)(x, y) = \max_{(s,t) \in b} f(x - s, y - t). \quad (2.6)$$

Определение. Размыкание (*opening*) изображения f по примитиву b обозначается $f \circ b$ и определяется равенством [1, с. 774]:

$$f \circ b = (f \ominus b) \oplus b. \quad (2.7)$$

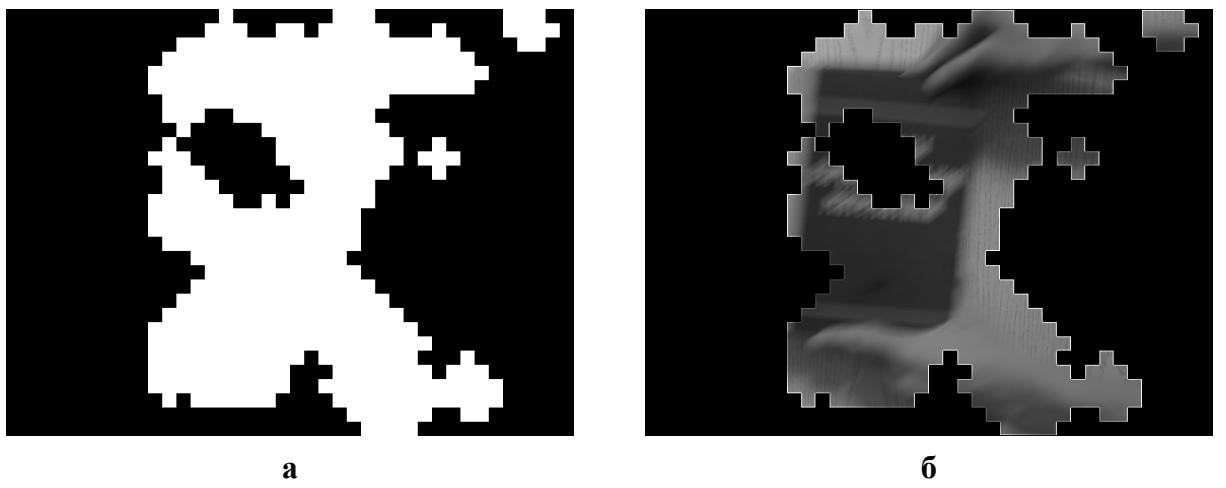
То есть размыкание изображения f по примитиву b строится как эрозия (2.5) f по b , результат которой затем подвергается дилатации (2.6) по тому же примитиву b .

Определение. Замыканием (*closing*) изображения f по примитиву b обозначается $f \bullet b$ и определяется следующим образом [1, с. 774]:

$$f \bullet b = (f \oplus b) \ominus b. \quad (2.8)$$

После процедуры морфологического сглаживания изображение B представляет

собой битовую маску, определяющую области, которые соответствуют движущимся объектам в изображении (рис. 2.9).



а — битовая маска, б — область, соответствующая искажённому объекту

Рис. 2.9: Пример обнаружения искажённого объекта

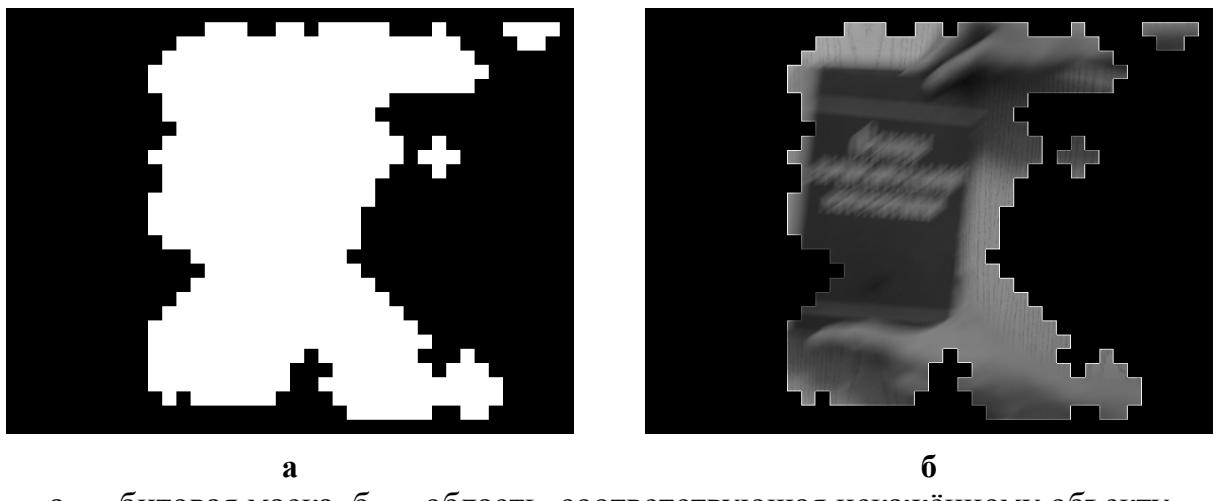
Проблема, которая была описана выше и заключалась в том, что при больших сма-зах некоторые блоки не определяются как перемещённые и не попадают в переме-щённую область, после шага с морфологической обработкой, может быть ещё акту-альна. Предлагается следующее решение: если контур области искажённого объекта замкнут и обладает «дырками» (см. рис. 2.9а), необходимо заполнить их. Для этого также можно использовать алгоритмы морфологической обработки [1, с. 746]. Таким образом, область искажения (2.10б) может быть получена с помощью битовой маски (2.10а).

2.1.2 Сглаживание контура искажённой области

Для того, чтобы уменьшить влияние артефактов на границе восстановленного изображения, необходимо сгладить контур битовой маски (рис. 2.10а). Сделать это можно несколькими способами: с помощью морфологической обработки и/или с по-мощью Фурье-дескрипторов контура.

Сглаживание с помощью морфологической обработки

Используя круглый примитив, в морфологических операциях над бинарным изоб-ражением можно добиться сглаживания прямоугольных областей. Например, резуль-



а — битовая маска, б — область, соответствующая искажённому объекту

Рис. 2.10: Результат обнаружения перемещённого объекта на изображении и его выделения для последующей обработки

тат эрозии (2.5) изображения 2.11а с круглым примитивом размером 5×5 пикселов (2.9) представлен на рис. 2.11б.



а — исходное изображение, б — результирующее изображение

Рис. 2.11: Пример операции эрозии (2.5) с круглым примитивом размером 5×5 пикселов (2.9) (в данном случае он совпадает с примитивом типа ромб).

0	0	1	0	0
0	1	1	1	0
1	1	1	1	1
0	1	1	1	0
0	0	1	0	0

(2.9)

Видно, что край прямоугольной области стал более гладким. Основываясь на этом, будем формировать более гладкий контур битовой маски (рис. 2.10а). В качестве примитива выбираем круг, диаметр которого прямо пропорционален размеру блока $S \times S$, который использовался при блочной компенсации движения. Чтобы не

потерять детали искажённого объекта, необходимо выполнить операцию дилатации (2.6) [1, с. 734], так как она приводит к расширению изображения [1, с. 737]. В результате получаем более гладкую битовую маску (рис. 2.12а), которая определяет область, соответствующую движущемуся объекту на изображении (рис. 2.12б).

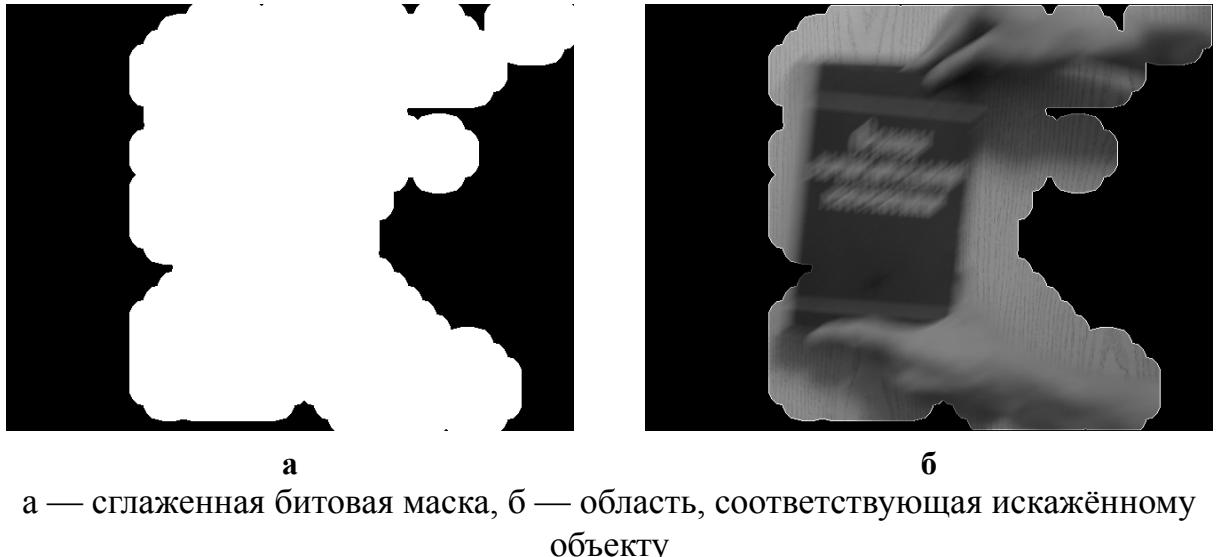


Рис. 2.12: Пример использования сглаженной битовой маски

В процессе поиска искажённой области нам доступна информация о векторах перемещения каждого блока. Предполагается, что при определении угла вектора размытия, его направление соответствует направлению вектора перемещения соответствующего объекта. Для одного и того же объекта значения векторов перемещения принадлежащих ему блоков могут различаться из-за влияния случайных факторов. Поэтому в качестве угла смаза выбираем тот угол перемещения векторов, который чаще всего встречается в искажённой области. Как показали эксперименты, полученный в результате угол вектора перемещения достаточно точно соответствует направлению вектора смаза.

Сглаживание с помощью преобразования Фурье-дескрипторов контура

После того, как область соответствующая искажённому объекту была определена, найдём её контур. Для определения границы области воспользуемся морфологическим алгоритмом выделения контуров [1, с. 745] (рис. 2.13). В результате получим K точек, принадлежащих контуру битовой маски. Координаты точек, которые принадлежат контуру обозначим (x_j, y_j) , где $j = 0, 1, \dots, K - 1$. Эти координаты

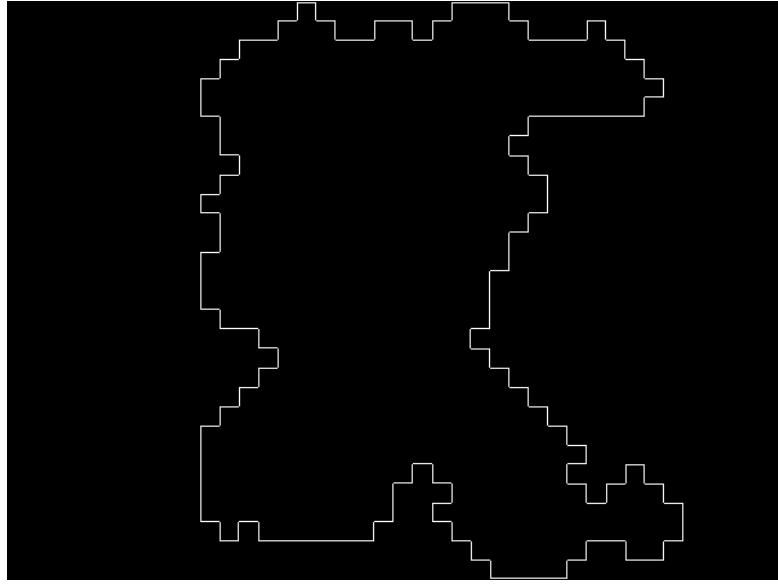


Рис. 2.13: Пример нахождения контура битовой маски

можно записать в форме $x(j) = x_j$ и $y(j) = y_j$. Используя такие обозначения, контур битовой маски можно представить в виде последовательности координатных пар $s(j) = [x(j), y(j)]$, где $j = 0, 1, \dots, K - 1$. Каждую пару координат можно рассматривать как комплексное число:

$$s(j) = x(j) + iy(j), \quad (2.10)$$

для $j = 0, 1, \dots, K - 1$. То есть, x и y — действительная и мнимая оси для последовательности комплексных чисел (2.10). Таким образом, двумерное представление контура сводится к одномерному.

Дискретное преобразование Фурье [2, с. 77] конечной последовательности $s(j)$ имеет вид:

$$a(u) = \sum_{j=0}^{K-1} s(j) e^{-i2\pi u j / K}, \quad (2.11)$$

где $u = 0, 1, \dots, K - 1$.

Введём понятие Фурье-дескрипторов контура, следуя [1, с. 944].

Определение. Фурье-дескрипторы контура — комплексные коэффициенты $a(u)$, которые являются результатом дискретного преобразования Фурье последовательности $s(j)$.

С помощью обратного дискретного преобразования Фурье [2, с. 80], по коэффи-

циентам $a(u)$ можно восстановить контур $s(j)$:

$$s(j) = \frac{1}{K} \sum_{u=0}^{K-1} a(u) e^{i2\pi u j / K}, \quad (2.12)$$

где $j = 0, 1, \dots, K - 1$.

Если для восстановления контура $s(j)$ в (2.12) использовать не все $K - 1$ коэффициенты ряда Фурье $a(u)$, а только их часть $R - 1$ ($R < K$), то в результате получим приближение последовательности $s(j)$:

$$\hat{s}(j) = \frac{1}{K} \sum_{u=0}^{R/2-1} a(u) e^{i2\pi u j / K} + a(u) e^{i2\pi u (j+K/2) / K}, \quad (2.13)$$

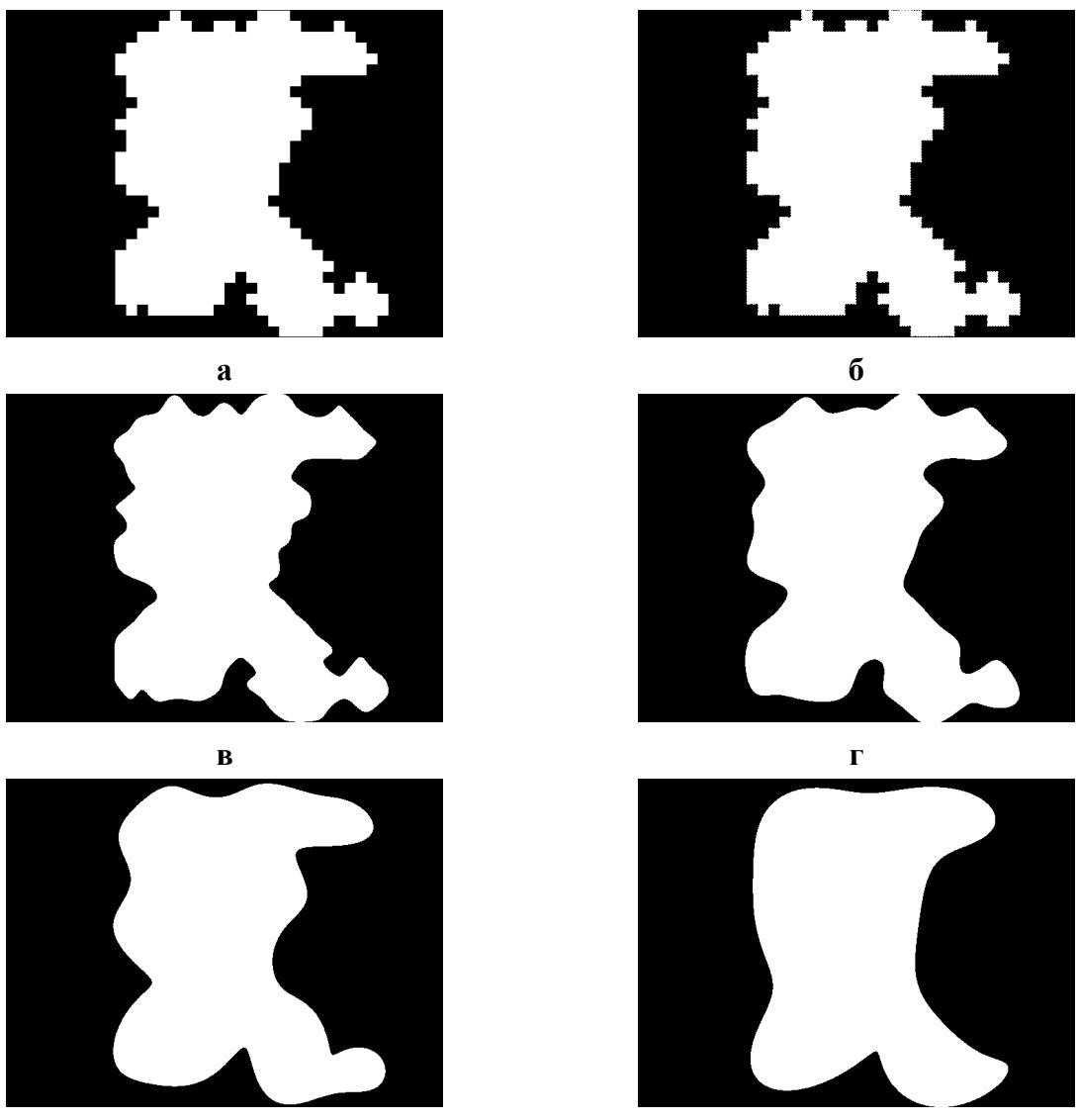
где $j = 0, 1, \dots, K - 1$. Несмотря на то, что в восстановлении будут участвовать не все Фурье-дескрипторы, количество точек контура не изменится.

Известно, что высокочастотные коэффициенты преобразования Фурье описывают резкоизменяющиеся значения сигнала, а низкочастотные — медленно меняющиеся компоненты [1, с. 304]. Таким образом, при уменьшении числа компонент R Фурье-преобразования для восстановления контура, резкие переходы границы теряются и контур становится более гладким.

Рисунок 2.14 иллюстрирует описанные выше выводы. При использовании даже 1% точек, описывающих контур, мы получим область, которая достаточно точно характеризует перемещённый объект. Уменьшая количество компонент Фурье-преобразования, которые принимают участие в восстановлении, мы проводим низкочастотную фильтрацию, тем самым сглаживаем резкий контур битовой маски.

2.2 Устранение краевых артефактов

Метод Люси-Ричардсона является итерационным и обладает следующим недостатком: при большом количестве итераций на изображении вблизи границ появляются артефакты в виде ярких и тёмных вертикальных и горизонтальных полос. Эти полосы «волнами» распространяются от краёв изображения и способствуют образованию «звонка» у границ. Причиной данных искажений выступает «достраивание» изображения за его границы, которое необходимо при выполнении ряда процедур



а — исходная битовая маска (2739 коэффициентов ДПФ),

- б — результат восстановления контура (использование 1369 коэффициентов ДПФ),
- в — результат восстановления контура (использование 100 коэффициентов ДПФ),
- г — результат восстановления контура (использование 50 коэффициентов ДПФ),
- д — результат восстановления контура (использование 30 коэффициентов ДПФ),
- е — результат восстановления контура (использование 15 коэффициентов ДПФ)

Рис. 2.14: Пример использования Фурье-дескрипторов для сглаживания границы битовой битовой маски

восстановления (например, свёртки), приводит к нарушению гладкости, или даже к разрывности функции яркости изображения вдоль его границ. В методе Люси-Ричардсона, после многочисленного вычисления свёрток, захватывающих области за краями изображения, эти граничные явления распространяются внутрь области самого изображения.

Данные искажения можно ослабить, если расширять изображение за его пределы, плавно сводя яркость к значению, близкому к нулевому. Рассмотрим два способа: использование весовой функции Хэмминга и размытие границы.

2.2.1 Использование весовой функции Хэмминга

Предлагается проводить расширение значениями граничных пикселей ненулевой области, умноженными на монотонно убывающую до нуля весовую функцию Хэмминга [1, с. 411]

$$w(x) = 0.54 - 0.46 \cos\left(\frac{2\pi x}{2S - 1}\right), \quad (2.14)$$

где величина S выбирается пропорционально величине смазы r (2.2).

Значение весовой маски в точке (x, y) определяется выражением:

$$\text{mask}(x, y) = w(\text{distance}(x, y)); \quad (2.15)$$

где $\text{distance}(x, y)$ — расстояние пикселя (x, y) до ближайшего единичного пикселя, $w(i)$ определяется с помощью выражения (2.14) и $0 \leq \text{mask}(x, y) \leq 1$.

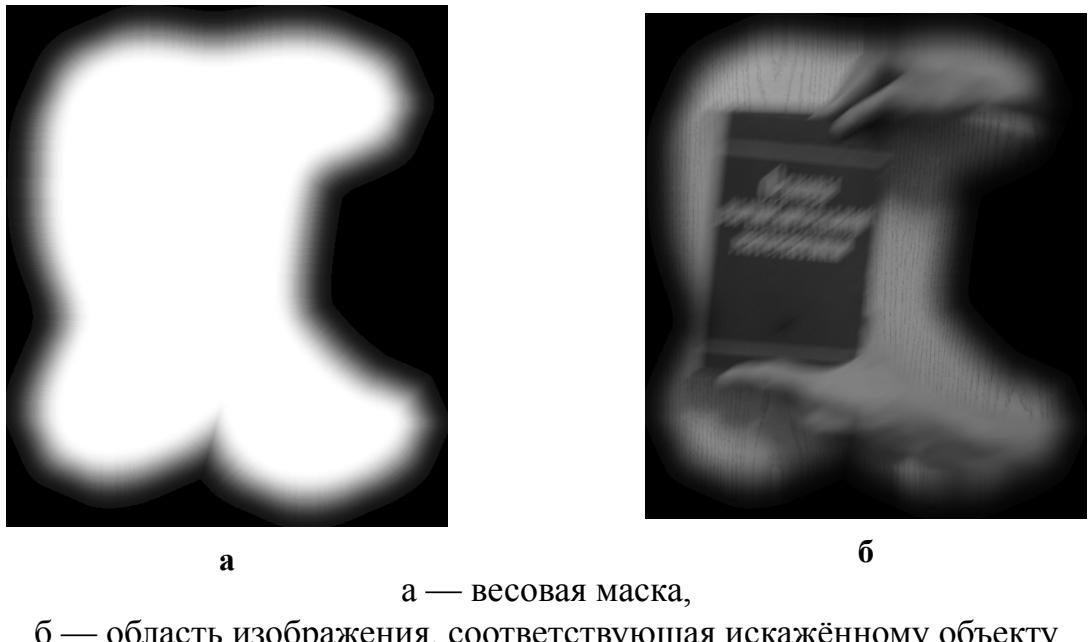


Рис. 2.15: Пример использования маски, полученной с помощью функции Хэмминга

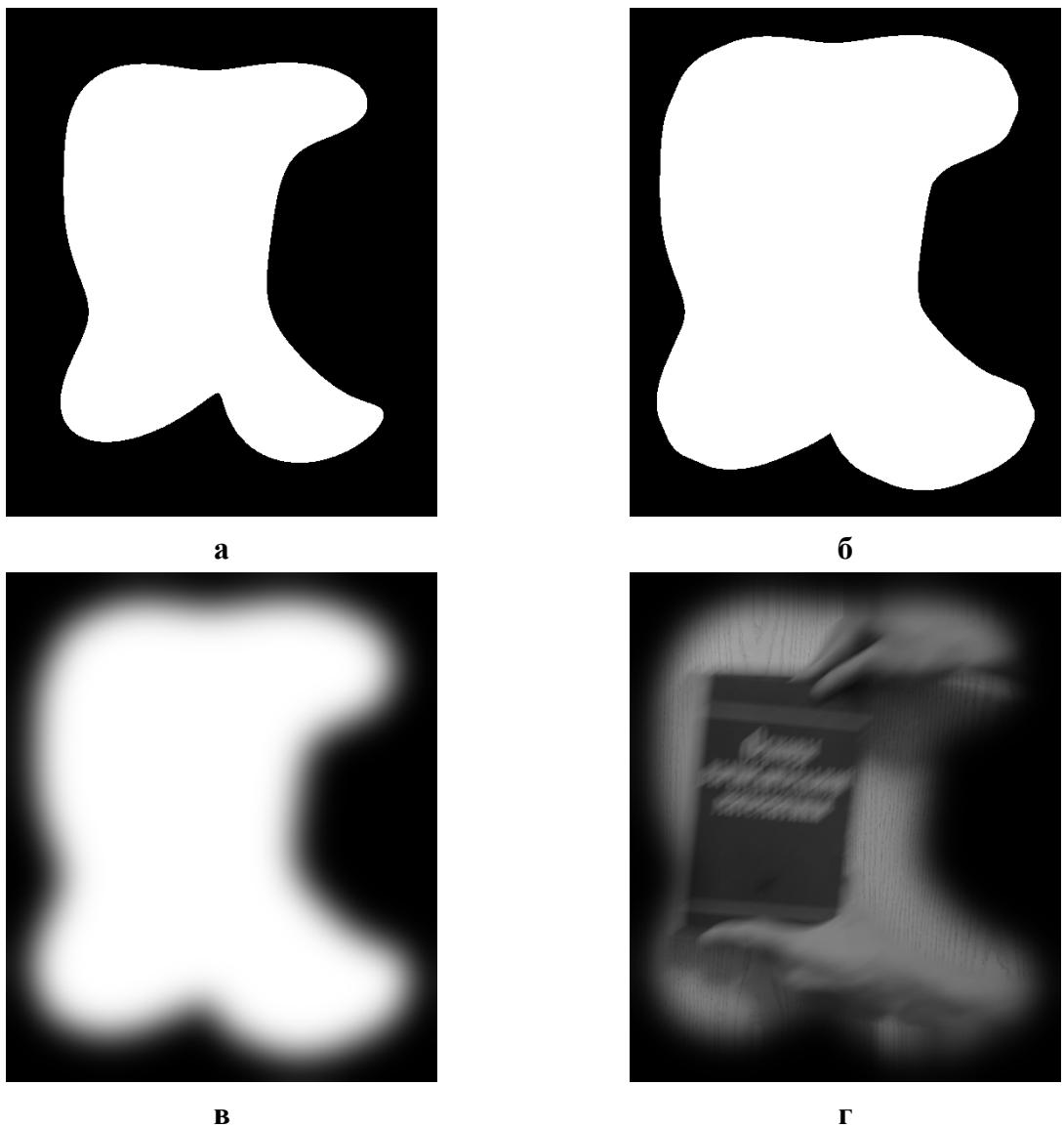
На рис. 2.15а изображена маска сформированная с помощью функции Хэмминга (2.14) по правилу (2.15). Выполняя поэлементное умножение mask с текущим

кадром I_1 получаем область, которую необходимо восстановить с помощью метода Люси-Ричардсона по формулам (1.17), (1.19) (рис. 2.15б).

2.2.2 Размытие границы маски

Следующий способ сведения яркости к нулю заключается в использовании фильтрации Гаусса [1, с. 194].

Для начала необходимо расширить маску единичными пикселями на расстояние пропорциональное величине смазы (рис. 2.16б). Чтобы получить монотонный пере-



а — исходная маска, б — расширенная маска, в — размытая маска,
г — область изображения, соответствующая искажённому объекту

Рис. 2.16: Пример использования маски, полученной с помощью размытия по Гауссу

ход от единичных пикселей к нулевым, выполним гауссову фильтрацию. ИХ фильтра Гаусса, который выполняет размытие, выглядит следующим образом:

$$h(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}}, \quad (2.16)$$

где σ — параметр, определяющий степень размытия.

Схема расширения маски с помощью фильтрации Гаусса, которая позволяет монотонно свести яркость на границе восстанавливаемой области, представлен в алгоритме (1).

Алгоритм 1: Схема монотонного сведения яркости к нулю

Входные данные: Искаажённое изображение g ; маска, характеризующая искажённую область b

Выходные данные: Область изображения, подлежащая восстановлению \hat{g}

- 1 Расширение маски на величину пропорциональную величине смазы.

$$t = \text{extension}(g);$$

- 2 Фильтрация Гаусса полученной маски

$$\hat{b} = t * * h,$$

где h определяется выражением (2.16), а операция « $* *$ » описывает свёртку (1.2).

- 3 Вычисляем область изображения, которая подлежит восстановлению

$$\hat{g}(x, y) = g_* * \hat{b},$$

где « $*.$ » — операция поэлементного перемножения массивов.

2.3 Определение оптимального числа итераций

Сложность практического применения итерационных методов заключается в определении оптимального числа итераций K , которое даёт наиболее качественную оценку \hat{f} оригинального изображения f . Также желательно исключить участие человека из процесса восстановления. Поэтому был предложен [8, 9, 14, 15] критерий останова итерационного процесса (1.17), (1.19).

Если оригинальное изображение f известно, то оптимальное количество итераций можно определить по норме разности

$$\| f - \hat{f} \|^2 = \sum_{(i,j)} f(i,j) - \hat{f}(i,j) \quad (2.17)$$

между ним и восстановленным изображением \hat{f} . Когда значение нормы будет минимально, необходимо завершить работу алгоритма восстановления. На практике оригинальное изображение f неизвестно, поэтому будем использовать евклидову норму разности восстановленных изображений на текущей (k -ой) и предыдущих итерациях:

$$S_k = \| \hat{f}_k - \hat{f}_{k-1} \| . \quad (2.18)$$

Чтобы краевые артефакты оказывали меньшее влияние на работу критерия останова, будем рассматривать только $1/3$ часть пикселей, лежащих в центральной области и которую обозначим Ω . Эксперименты показали, что чем ближе число проделанных повторений (1.17), (1.19) к оптимальному числу итераций, тем меньше значение нормы S_k (см. рис. 2.17б). Следовательно, найдя ту итерацию, которая принесёт наименьшее значение нормы S_k , можно прекратить итерационный процесс восстановления. Для определения такой итерации предлагается использовать алгоритм 2.

На рисунке 2.17 приведён пример определения оптимального числа итераций при восстановлении искажённого изображения (рис. 2.17а).

Свой первый минимум величина S_k (2.18) достигает при числе итераций алгоритма Люси-Ричардсона равном 36. Таким образом, в соответствии с алгоритмом 2, итерационный процесс восстановления будет завершён на 62 итерации. Видно, что после 62 итераций, оценка \hat{f} искажённого изображения f корректна (рис. 2.17д). На последующих итерациях уже не происходит уточнение оценки \hat{f} (рис. 2.17е). Поэтому завершение итерационного процесса Люси-Ричардсона на 62 итерации является обоснованным. Использование алгоритма 2 позволяет решить следующие проблемы:

- не требуется участие человека в процессе восстановления изображения;
- при большём числе итераций оценка \hat{f} не уточняется, но тем не менее подвержена влиянию краевых артефактов. Поэтому при завершении итерационного процесса на оптимальном числе итераций позволяет избежать дополнительных

искажений.

Алгоритм 2: Схема алгоритма Люси-Ричардсона с дополнением для определения оптимального числа итераций

Входные данные: Искажённое изображение g

Выходные данные: Оценка оригинального изображения \hat{f}

- 1 Найдём нормировочный коэффициент (ско) для области Ω , состоящей из центральных пикселей изображения:

$$\sigma_\Omega = \sqrt{\frac{1}{|\Omega|} \sum_{(i,j) \in \Omega} (g(i,j) - A)^2}, \quad \text{где } A = \frac{1}{|\Omega|} \sum_{(i,j) \in \Omega} g(i,j).$$

- 2 Выполняем 5 итераций алгоритма Люси-Ричардсона (1.17, 1.19). Полагаем $k = 6$.
- 3 Переходим к следующей (k -ой) итерации и находим оценку изображения в соответствии с (1.17, 1.19).
- 4 Находим норму разности $S_k = \| \hat{f}_k - \hat{f}_{k-1} \|$.
- 5 Чтобы избежать резких изменений S_k , проведём сглаживание:

$$S_k \leftarrow \frac{1}{4}S_{k-2} + \frac{1}{2}S_{k-1} + \frac{1}{4}S_k.$$

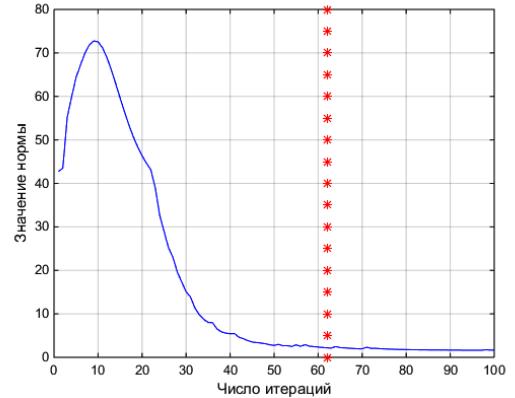
- 6 **если** ($\max(S_{k-4}, S_{k-3}, S_{k-2}, S_{k-1}, S_k) < \sigma_\Omega \cdot 10^{-2}$) **тогда**
- 7 **если** величина смазы $r < 15$ **тогда**
- 8 останавливаемся на $k + 26$ итерации.
- 9 **иначе**
- 10 останавливаемся на $k + 1$ итерации.
- 11 **конец**
- 12 **иначе**
- 13 увеличиваем k на единицу 1 и переходим к шагу 2
- 14 **конец**

2.4 Выводы

- для получения более корректной оценки неискажённого изображения, предлагаются восстанавливать только фрагмент изображения, который соответствует искажённому объекту;
- используя идею блочной компенсации движения, становится возможным найти искажённый объект на изображении;
- предлагается сглаживать контур искажённой области, чтобы устранить краевые артефакты;
- расширение области за её границы и монотонное сведение яркости к нулевому значению позволяет существенно ослабить «звон» на краях восстановленного изображения;
- для оценки необходимого числа итераций метода Люси-Ричардсона был предложен алгоритм, который основан на том, что норма разности восстановленных изображений на соседних итерациях стабилизируется, когда оценка изображения наиболее точная.



а



б



в



г



д

е

- а — искажённое изображение размером 100×100 ,
б — зависимость нормы разности (2.18) от числа итераций,
в — результат восстановления после 10 итераций, г — результат восстановления после 40 итераций,
д — результат восстановления после 62 итераций (оптимальное число итераций),
е — результат восстановления после 100 итераций

Рис. 2.17: Пример определения оптимального числа итераций

Глава 3

Результаты экспериментов

Рассмотрим результаты применения метода Люси-Ричардсона, его адаптаций (1.17), (1.19) и предложенных модификаций на примере 3х изображений. Результаты восстановления одного изображения приведены в данной главе, остальные даны в Приложении 1 и Приложении 2.

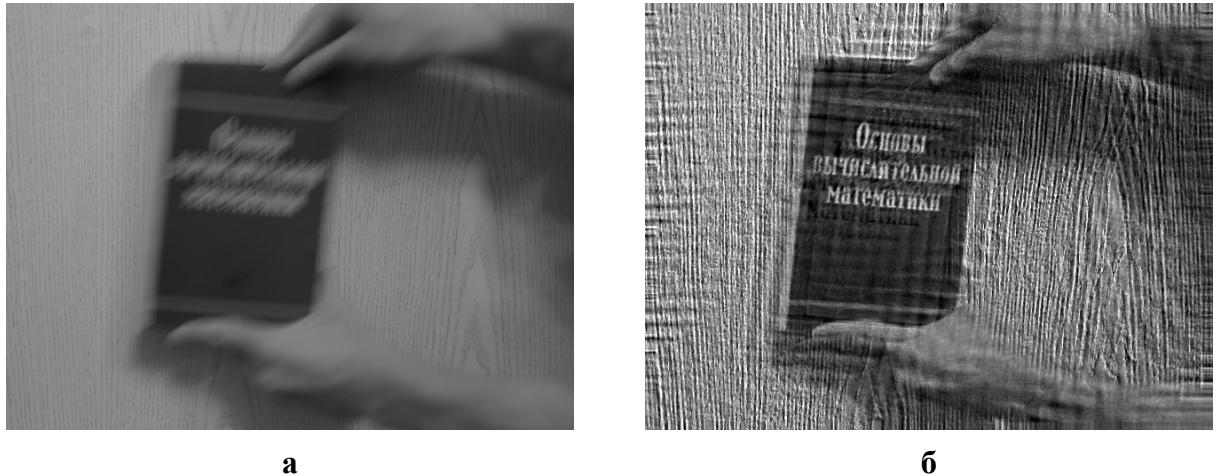
Рассмотрим изображение размером 640×480 (рис. 3.1).



Рис. 3.1: Исходное изображение

3.1 Восстановление изображения без выделения искажённого объекта

На изображении 3.1 присутствует искажённая область, а именно книга. Воспользуемся алгоритмом Люси-Ричардсона (1.17), (1.19) для восстановления изображения.



а — искажённое изображение, б — восстановленное изображение

Рис. 3.2: Результат работы метода Люси-Ричардсона для изображения размером 604×480 . Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.003$

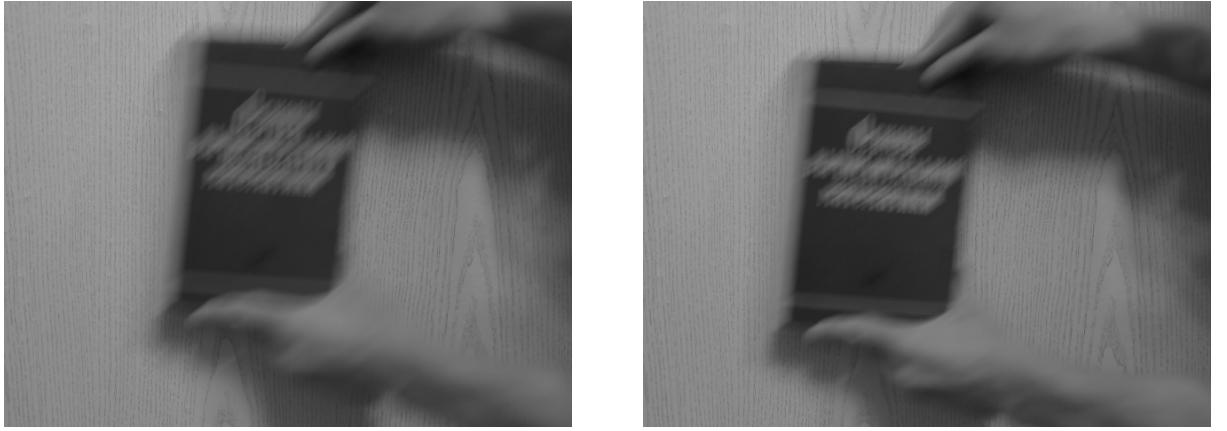
Количество итераций, необходимых для восстановления изображения (3.1), было найдено по алгоритму 2 (см. раздел 2.3) и равно 43.

Несмотря на то, что часть информации на изображении 3.2б была восстановлена (можно прочесть название учебника), существует недостаток восстановления всего изображения. Если бы на фоне присутствовали статичные объекты, то в результате восстановления информация о них была бы утеряна. Из-за текстурного фона в результате восстановления изображение 3.2б стало более зашумлённым ($\sigma_\eta = 0.02$). Поэтому предлагается восстанавливать только искажённый фрагмент изображения.

3.2 Поиск искажённой области

Воспользуемся идеей блочной компенсации движения для обнаружения искажённого объекта на изображении 3.3б.

Выберем размер блока $S = 16$ в соответствии с описанным в разделе 2.1 эмпи-



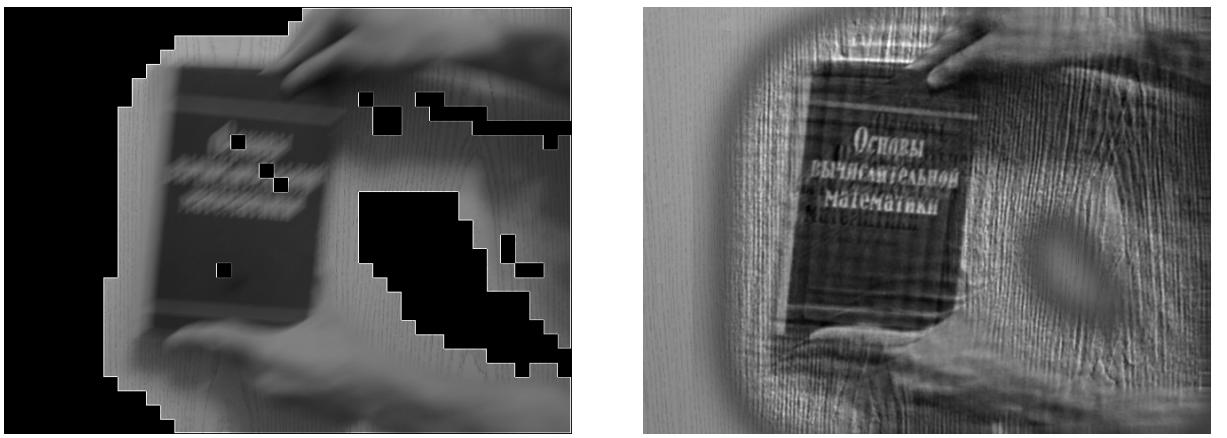
а

б

а — предыдущий кадр видеопоследовательности,
б — текущий кадр видеопоследовательности.

Рис. 3.3: Искажённые изображения

рическим правилом (размер исходного изображения 640×480). Радиус области Ω выберем равным 5. Выполним поиск перемещённых блоков, модуль вектора перемещения r которых удовлетворяет условию (2.2): $r > 5$. В качестве меры схожести блоков для начала используем (2.1). Найденная область представлена на рис. 3.4



а

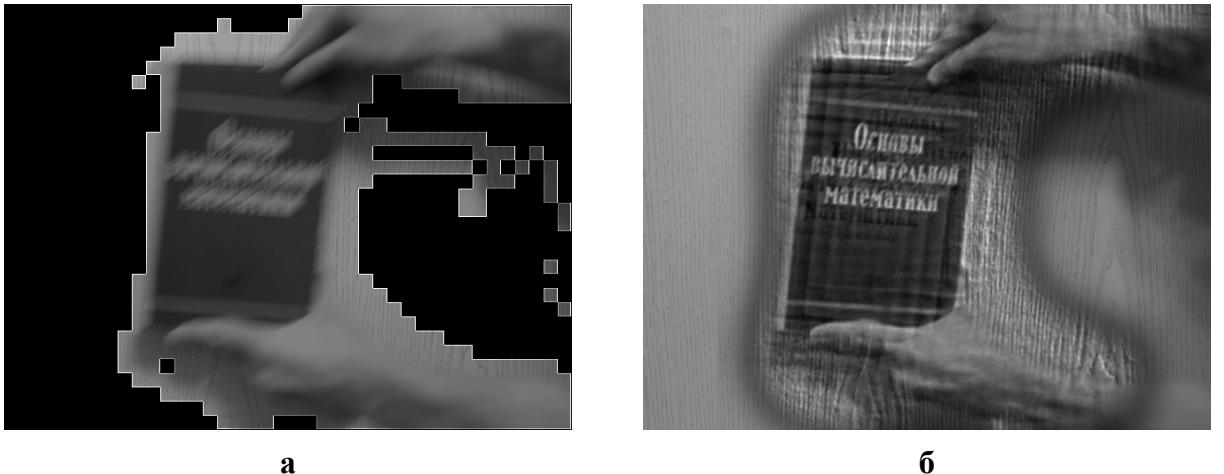
б

а — искажённый фрагмент изображения, полученный путём поиска блоков с помощью выражений (2.1), (2.2), б — восстановленное изображение

Рис. 3.4: Результат работы метода Люси-Ричардсона для фрагмента изображения. Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 48

Из рисунков (3.3а) и (3.3б) видно, что помимо книги движение совершают руки и тень от них. В результате, эти области тоже считаются искажённым, но они могут иметь отличные от истинных векторы перемещения. Чтобы исключить влияние

теней и перепадов яркости на определение перемещённых блоков воспользуемся выражением (2.3) и получим более корректный результат.



а — искажённый фрагмент изображения, полученный путём поиска блоков с помощью выражений (2.3), (2.2), б — восстановленное изображение

Рис. 3.5: Результат работы метода Люси-Ричардсона для фрагмента изображения. Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 62

При использовании выражения (2.3) получаем более точную оценку искажённой области 3.5а по сравнению с областью, найденной ранее (3.4а).

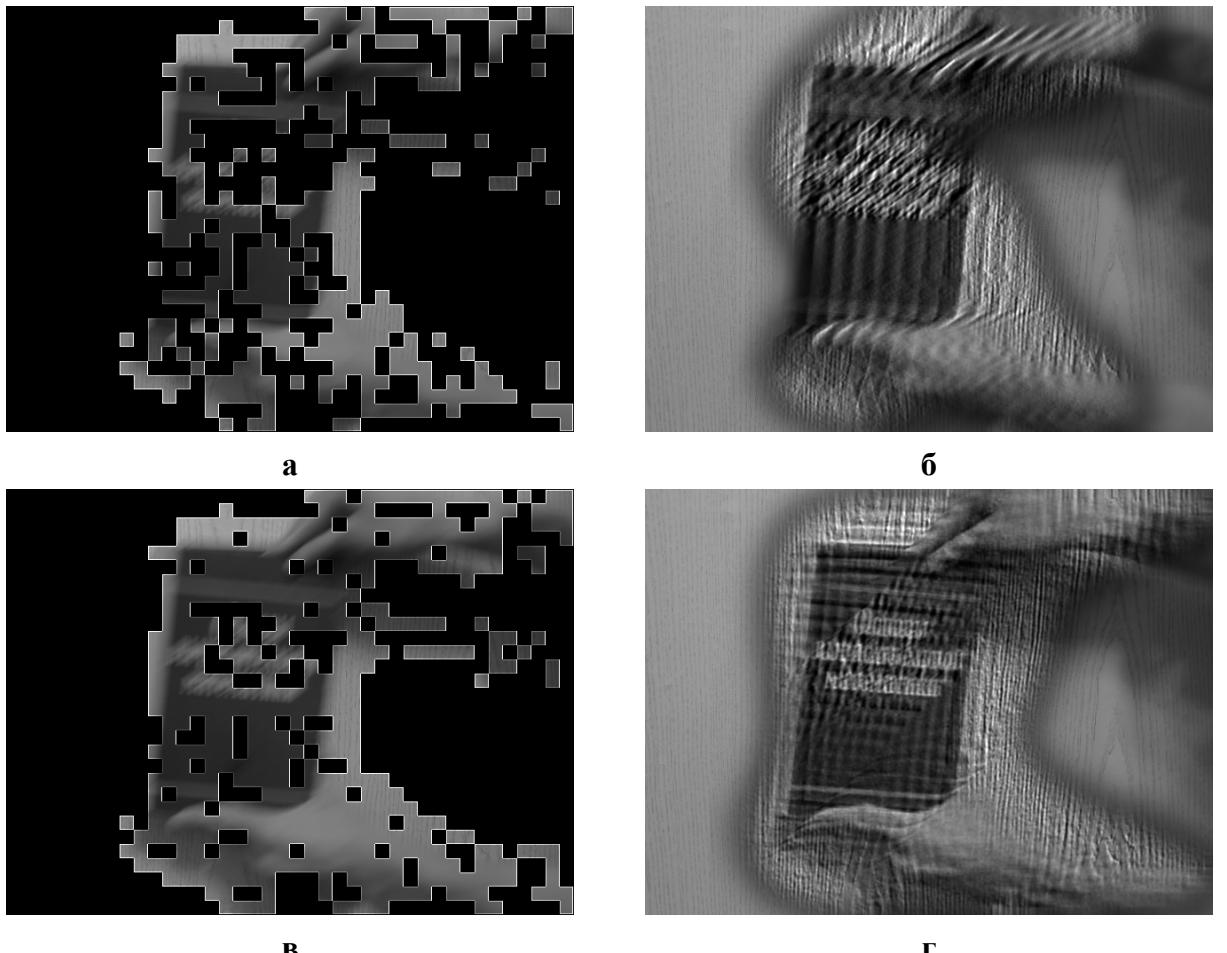
Затем проведём сегментацию движения с помощью выражения (2.4) и получим более точную оценку искажённой области изображения 3.3б:

$$\| \mathbf{r}^i - \mathbf{m}_r \| < \delta \| \mathbf{m}_r \|,$$

Оценим параметр δ . Выберем $\delta = 3$ и $\delta = 5$ (рис. 3.6).

После сегментации движения, на изображении (3.6а, 3.6в) к искажённому объекту ещё относятся шумовые блоки. Чтобы решить эту проблему, проведём морфологическое сглаживание, то есть заполним небольшие «дыры» и избавимся от «шумовых блоков». В результате получаем более точно выделенные области искаженного объекта (рис. 3.7).

Видно, что область выделяется точнее (меньше ошибочно определённых блоков перемещения) при сегментации движения (2.4) с $\delta = 3$. На изображении 3.7а отсутствуют ошибочные блоки, относящиеся к перемещению тени или текстуры фона. Выбирать параметр δ следует наименьшим из возможных, то есть если блоки характеризуют искажённую область в случае $\delta = 3$ и в случае $\delta = 5$, то следует выбирать

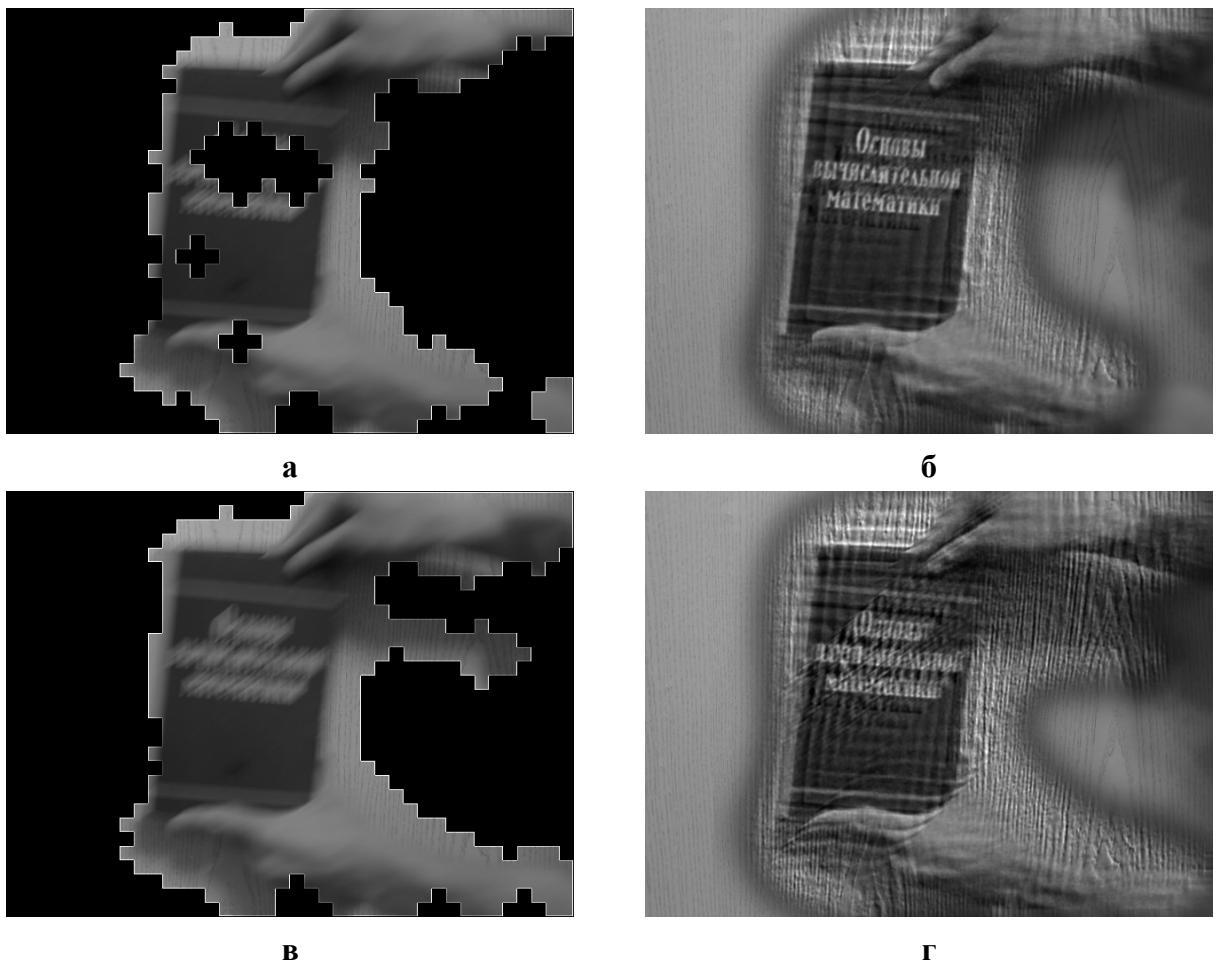


а — искажённый фрагмент изображения, полученный с помощью сегментации движения $\delta = 3$ (2.4),
 б — результат восстановления фрагмента изображение 3.6а. Количество итераций: 63,
 в — искажённый фрагмент изображения, полученный с помощью сегментации движения $\delta = 5$ (2.4),
 г — результат восстановления фрагмента изображение 3.6в. Количество итераций: 59

Рис. 3.6: Результат работы метода Люси-Ричардсона для фрагмента изображения. Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$

сегментацию движения (2.4) с $\delta = 3$. Затем проводить морфологическое сглаживание маски, которая определяет искажённую область 3.7а.

Для того, чтобы отнести искажённые блоки, но неперемещённые (рис. 3.7а) к области восстановления, будем учитывать в процессе восстановления область полностью, если её контур замкнут. Таким образом, область искажения, которая будет подаваться на вход алгоритму Люси-Ричардсона, выглядит следующим образом (см. 3.8а).



а — искажённый фрагмент изображения, полученный с помощью сегментации движения $\delta = 3$ (2.4),
 б — результат восстановления фрагмента изображение 3.7а. Количество итераций: 50,
 в — искажённый фрагмент изображения, полученный с помощью сегментации движения $\delta = 5$ (2.4),
 г — результат восстановления фрагмента изображение 3.7в. Количество итераций: 53

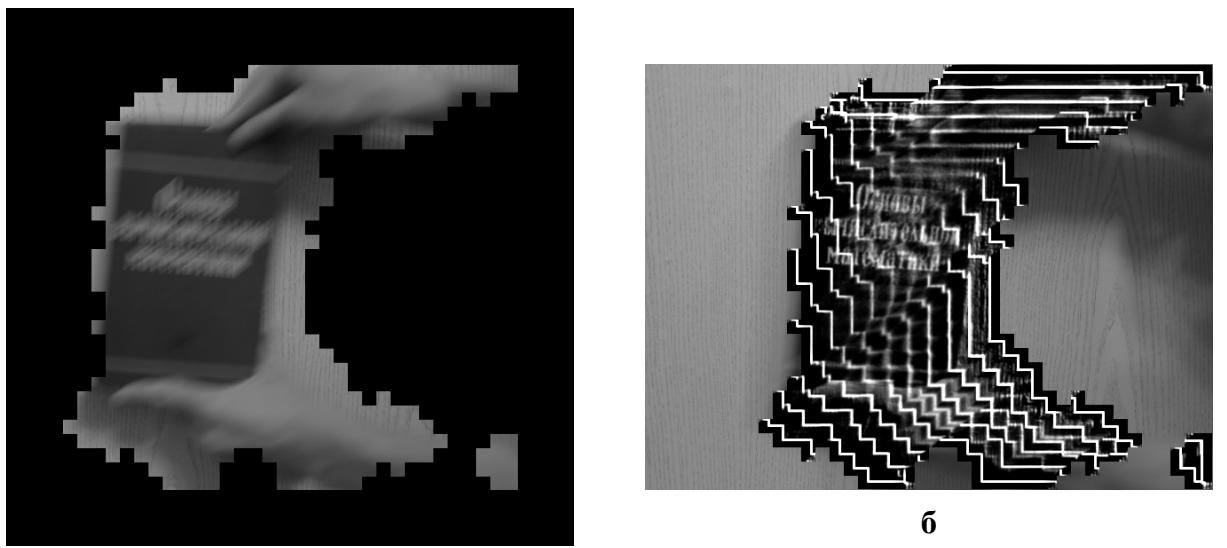
Рис. 3.7: Результат работы метода Люси-Ричардсона для фрагмента изображения. Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$

3.3 Подготовка области движущегося объекта

3.3.1 Сглаживание контура области

Если на вход алгоритма Люси-Ричардсона подать изображение (3.8а), то в результате получим ещё более искажённую оценку изображения (3.8б). Такой результат считается некорректным. Поэтому необходимо сделать контур восстанавливаемой области более гладким.

Рассмотрим два подхода: морфологическая обработка и преобразование Фурье-дескрипторов контура.

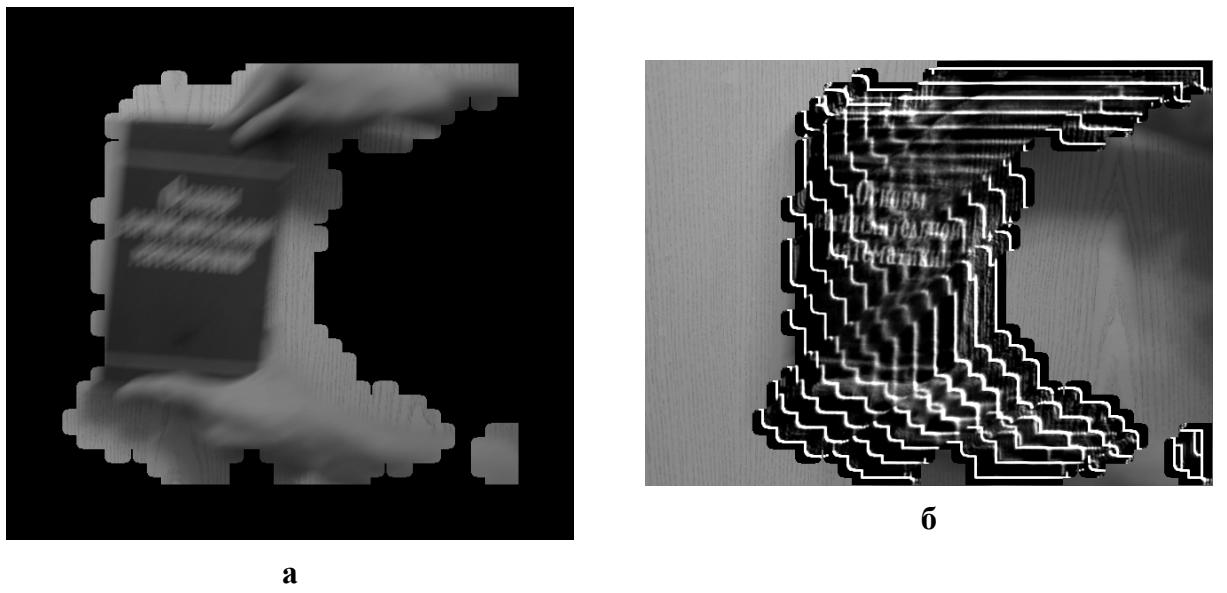


а
а — искажённый фрагмент изображения, б — восстановленное изображение

Рис. 3.8: Пример работы метода после обнаружения искажённого фрагмента изображения Найденные параметры искажения: величина смаза — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 80

Морфологическая обработка

Для маски, характеризующей искажённый фрагмент (3.8а) выполним операцию дилатации (2.6) с круглым примитивом размера 15×15 .



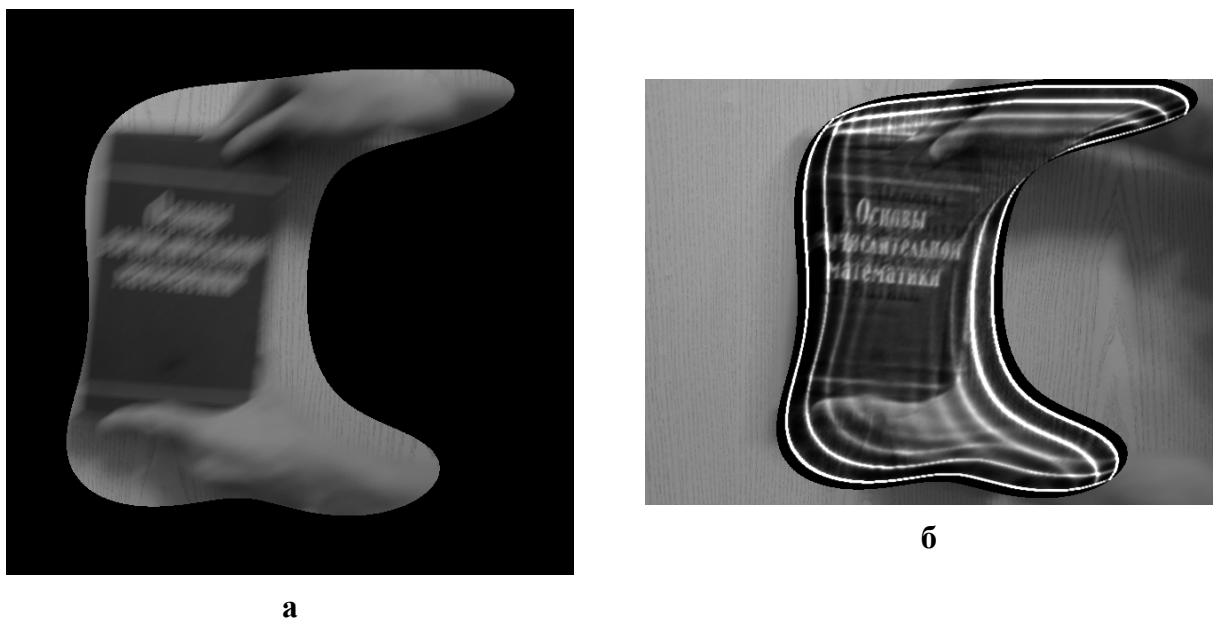
а
а — искажённый фрагмент изображения, б — восстановленное изображение

Рис. 3.9: Сглаживание контура с помощью морфологических операций (дилатации 2.6) Найденные параметры искажения: величина смаза — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 80

В результате получим фрагмент 3.9б, при восстановлении которого не только не получаем видимых улучшений изображения 3.9б, но и искажаем ещё сильнее. Поэтому этот подход в дальнейшем исключается из рассмотрения.

Преобразование Фурье-дескрипторов контура

Использование Фурье-дескрипторов контура позволяет контролировать уровень гладкости контура (см. рис. 2.14). Будем восстанавливать точки контура по формуле (2.13), используя лишь 0.5% коэффициентов ДПФ, принадлежащих Фурье-преобразованию контура. Получаем искажённый фрагмент изображения с гладкой границей (рис. 3.10а).



а — искажённый фрагмент изображения, б — восстановленное изображение

Рис. 3.10: Сглаживание контура с помощью морфологических операций (дилатации 2.6) Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 80

После восстановления изображения (рис. 3.10а) получаем более корректный результат (рис. 3.10б) по сравнению с ранее описанным подходом.

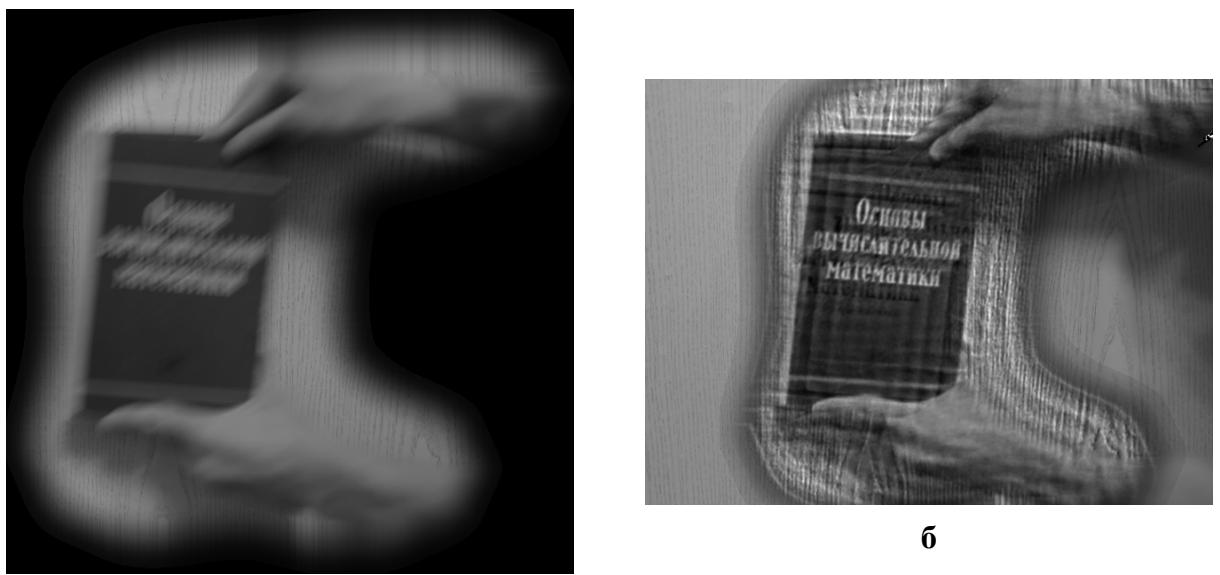
«Звон», возникающий в процессе восстановления, не перекрывает всю область изображения, а дополнительно искажает только края области. Чтобы решить эту проблему будем расширять изображение.

3.3.2 Расширение маски

Для устранения «звонов» (3.10б) на границе будем расширять область искажения за её пределы, монотонно сводя яркость к 0.

Весовая функция Хэмминга

Чтобы свести яркость фрагмента на границах к 0, воспользуемся весовой функцией Хэмминга (2.14). Получим фрагмент изображения (3.11а), который будем подвергать процессу восстановления с помощью метода Люси-Ричардсона.



а — искажённый фрагмент изображения, б — восстановленное изображение

Рис. 3.11: Расширение искажённой с помощью весовой функции Хэмминга. Найденные параметры искажения: величина смаза — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 45

В случае изображения 3.11а использование весовой функции Хэмминга показывает хороший результат. Но в ряде экспериментов этот подход оказался неподходящим. Причина этого заключается в том, что если у маски будет «излом», то описанный подход только усугубит результат восстановления в месте излома: появятся белые и чёрные пиксели, характерные ошибкам восстановления с помощью итерационного процесса. Поэтому было принято решение отказаться от этого подхода в дальнейшем.

Размытие маски по Гауссу

Проведём фильтрацию маски фильтром Гаусса (размер окна пропорционален величине искажения), тем самым проведём её низкочастотную фильтрацию. В результате мы избавимся от резкого перехода маски из 1 в 0 и получим фрагмент изображения, который обладает сглаженным расширением (рис. 3.12а).



а



б

а — искажённый фрагмент изображения, б — восстановленное изображение

Рис. 3.12: Расширение искажённой области с помощью низкочастотной фильтрации Гаусса. Найденные параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$. Количество итераций — 50

После восстановления области изображения 3.12а, получаем более корректную оценку изображения (рис. 3.12б).

В результате экспериментов данный подход показал стабильность в процессе восстановления, поэтому в дальнейшем будем выполнять расширение изображения, монотонно сводя яркость к 0 с помощью низкочастотной фильтрации по Гауссу.

Чтобы шумовая составляющая, которая присутствует на восстановленном изображении (рис. 3.12б), не препятствовала восприятию изображения, предлагается проводить его низкочастотную фильтрацию (рис. 3.13).

В данном случае (рис. 3.13) был использован фильтр Гаусса с размерами окна 3×3 .



Рис. 3.13: Результирующее изображение размером 640×480

3.4 Процесс восстановления искажённого изображения



В результате проведённых экспериментов будем выполнять восстановление искажённого изображения (рис. 3.3б) с помощью метода Люси-Ричардсона (1.15), его адаптаций (1.17), (1.19) и предложенных модификаций (глава 2) по алгоритму 3.

Итогом работы предложенного метода компенсации линейного смаза изображения 3.15, основанном на методе Люси-Ричардсона является изображение 3.16.

Алгоритм 3: Схема алгоритма компенсации линейного смаза цифровых изображений на основе метода Люси-Ричардсона

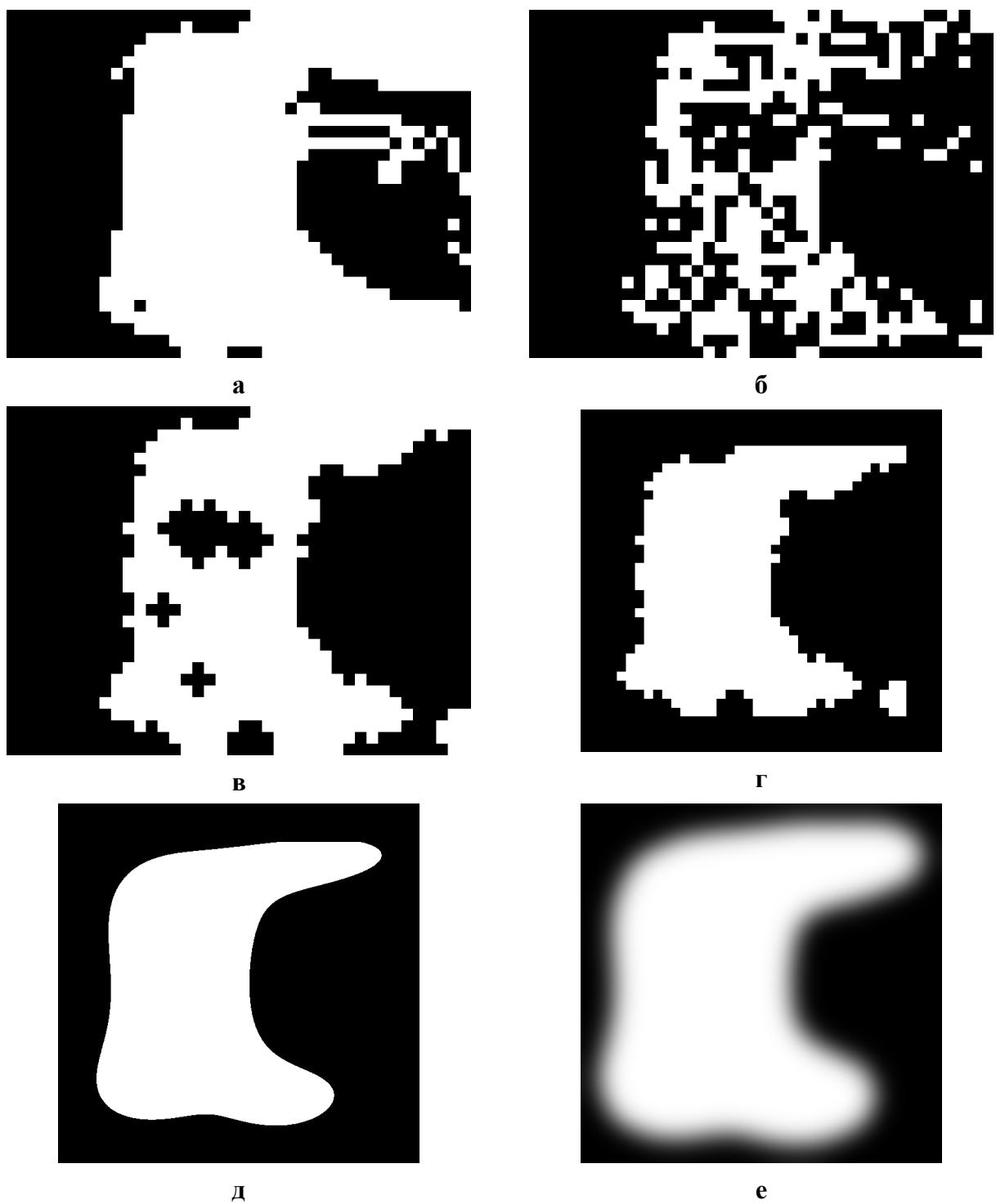
Входные данные: Искажённое изображение f

Выходные данные: Оценка оригинального изображения \hat{f}

// Получим маску, характеризующую искажённую область изображения

- 1 Воспользуемся идеей блочной компенсации движения и выделим перемещённые блоки в соответствии с условием (2.2) (рис. 3.14а).
- 2 Проведём сегментацию движения, используя выражение (2.4) с $\delta = 3$ (рис. 3.14б).
- 3 Выполним морфологические сглаживание маски (рис. 3.14в), полученной на предыдущем шаге (рис. 3.14б).
- 4 Проведём заполнение замкнутого контура области, соответствующей искажённому фрагменту (рис. 3.14г).
- 5 Выполним преобразование Фурье-дескрипторов контура маски (рис. 3.14г), чтобы избавиться от резких границ (рис. 3.14д). Используем 15 элементов Фурье-преобразования для восстановления границы (2.13).
- 6 Проведём расширение маски (рис. 3.14д) за её пределы, используя размытие по Гауссу (рис. 3.14е).
// В результате получим фрагмент изображения (рис. 3.15)
- 7 Фрагмент изображения поступает на вход алгоритма Люси-Ричардсона.
- 8 Найдём параметры искажения по способу описанному в разделе 1.3: величина смазы — 28 пикселей, угол — 36° , $\sigma_\eta = 0.002$.
- 9 Определим оптимальное число итераций N для корректного восстановления изображения 3.15 с помощью алгоритма 2: $N = 50$.
- 10 Выполняем N итераций алгоритма Люси-Ричардсона (1.17), (1.19).
- 11 Проведём низкочастотную фильтрацию полученного изображения с помощью фильтра Гаусса с маской 3×3 .

Результаты применения метода Люси-Ричардсона, его адаптаций (1.17), (1.17) и предложенных модификаций (алгоритм 3) для двух других изображений приведены в Приложении 1 и Приложении 2.



а, б, в — приближение маски, характеризующей искажённую область на изображении,
 г — маска, характеризующая искажённую область на изображении,
 д — маска с гладкой границей, характеризующая искажённую область на изображении,
 е — итоговая маска, характеризующая область искажения на изображении 3.3б

Рис. 3.14: Процесс формирования маски для изображения.



Рис. 3.15: Искажённый фрагмент изображения 3.3б

3.5 Программная реализация

Метод компенсации линейного смаза цифровых изображений на основе алгоритма Люси-Ричардсона был программно реализован на языке C++. Результаты работы на компьютере с процессором Intel Core 2 Duo, 2.20GHz представлены в таблице 3.1.

Таблица 3.1: Характеристики процесса восстановления искажённого изображения размером 640×480 . Параметры искажения: величина смазы — 28 пикселей, угол — 36^0 , $\sigma_\eta = 0.002$

Восстановление изображения	Количество итераций	Время, с
Восстановление всего изображения (см. рис. 3.1, 3.2б)	69	5.3
Восстановление фрагмента изображения (см. рис. 3.15, 3.16)	50	4.72

Как и ожидалось, восстанавливая фрагмент изображения, можно получить не только более точное восстановление, но и сократить вычислительные затраты.



Рис. 3.16: Результирующее изображение размером 640×480

3.6 Выводы

- восстановление фрагмента изображения позволяет получить более корректную оценку неискажённого изображения и сократить вычислительные затраты;
- следует выполнять сегментацию движения, чтобы точнее определить область искажения и сегментировать объекты с разными векторами перемещения;
- для корректной работы итерационного метода Люси-Ричардсона необходимо выполнять сглаживание границы искажённой области с помощью преобразования Фурье-дескрипторов;
- расширять область за её пределы и монотонно сводить яркость к 0 целесообразно с помощью фильтрации бинарной маски по Гауссу.

ВЫВОДЫ И ЗАКЛЮЧЕНИЕ

В данной работе были рассмотрены методы восстановления изображений, подвергнутых линейному смазу, способы определения параметров искажения и адаптации алгоритмов. На основе проведённых исследований был выбран алгоритм Люси-Ричардсона с модификациями, позволяющий восстановить изображение без участия оператора.

Для разработки метода компенсации линейного смаза на изображении, основанного на методе Люси-Ричардсона, были решены следующие задачи:

- разработан метод оценки параметров искажения (раздел 1.3);
- разработан метод оценки числа итераций алгоритма (раздел 2.3);
- разработан метод выделения искажённого объекта (раздел 2.1).

Все предложенные методы были реализованы на языке программирования C++. Результаты их работы продемонстрированы в главе 3, Приложении 1 и Приложении 2.

На основании полученных результатов решения задач сделаны выводы:

- использование метода Люси-Ричардсона и его модификаций позволяет получить более качественный результат восстановления по сравнению с другими рассмотренными методами;
- для повышения точности определения параметров искажения и шума следует использовать как кепстральное представление и частотные свойства изображения, так и информацию о векторах перемещения блоков;
- использование информации об изменении изображения в процессе восстановления позволяет получить оценку оптимального числа итераций алгоритма Люси-Ричардсона;

- выделение и обработка толькоискажённого фрагмента даёт более корректную оценку всего изображения, а также сокращает вычислительные затраты;
- использование сглаживания границ и оконных функций при выделении искажённой области позволяет уменьшить негативное влияние краевых артефактов на последующую обработку.

Основные результаты и положения диссертационной работы опубликованы в 4 работах автора [8, 9, 14, 15], докладывались на конференциях IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EICoRusNW) (февраль 2016), «Цифровая обработка сигналов и её применение» — DSPA-2016 (апрель 2016), 22-ой Всероссийской межвузовской научно-технической конференции студентов и аспирантов «Микроэлектроника и информатика – 2015» (апрель 2015).

Литература

- [1] Гонсалес Р.С., Вудс Р.Е. Цифровая обработка изображений. — М. : Техносфера, 2012. — 1104 с.
- [2] Умняшкин С.В. Основы теории цифровой обработки сигналов: учебное пособие. — М. : Техносфера, 2016. — 528 с.
- [3] Бьемон Ж., Лагендейк Л., Мерсеро Р.М. Итерационные методы улучшения изображений // ТИИЭР. — 1990. — Т. 7, № 5. — С. 58–84.
- [4] Lucy L.B. An iterative technique for the rectification of observed distributions // The Astronomical journal. — 1974. — Vol. 79, no. 6.
- [5] Richardson W.H. Bayesian-Based Iterative Method of Image Restoration // Journal of the Optical Society of America. — 1972. — Vol. 62, no. 1.
- [6] Jansson P.A. Deconvolution of Images and Spectra. — New York : Elsevier, 1997. — 514 p.
- [7] Гонсалес Р.С., Вудс Р.Е., С. Эддинс. Цифровая обработка изображений в среде MATLAB. — М. : Техносфера, 2006. — 621 с.
- [8] Умняшкин С.В, Панфилова К.В. Применение метода Люси-Ричадсона для компенсации линейного смаза цифровых изображений // Сборник научных трудов МИЭТ. Посвящается 70-летию профессора А.С. Поспелова. — 2016. — С. 10–19.
- [9] Панфилова К.В. Компенсация линейного смаза цифровых изображений с помощью метода Люси-Ричардсона // Юбилейная 25-я Международная конференция (GraphiCon2015). — 2015. — Сентябрь. — С. 163–167.

- [10] Разработка методик создания баз изображений и программного обеспечения для тестирования алгоритмов видеоанализа : Отчет о НИР : № гос. регистрации 114071570046 / МИЭТ ; рук. С.В. Умняшкин. — Москва : МИЭТ, 2015. — 125 с.
- [11] Meinel E.S. Optical of linear and non-linear recursive restoration algorithms // Journal of the Optical Society of America. — 1986. — June. — Vol. 3, no. 6. — P. 787–799.
- [12] Biggs D.S.C., Andrews M. Acceleration of iterative image restoration algorithm // Applied Optics. — 1997. — June. — Vol. 36, no. 8. — P. 1766–1775.
- [13] Richardson Iain E. The H.264 Advanced Video Compression Standard. — 2nd edition. — Wiley Publishing, 2010. — 346 p.
- [14] Панфилова К.В. Локальная компенсация смазы изображения на основе метода Люси-Ричардсона // Доклады 18-й Международной конференции «Цифровая обработка сигналов и её применение» — DSPA-2016. — Т. 2. — 2016. — С. 657 – 662.
- [15] Panfilova K., Umnyashkin S. Linear Blur Compensation in Digital Image Using Lucy-Richardson Method // 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW). — 2016. — February. — P. 302 – 304.
- [16] Форсайт Д.А., Понс Ж. Компьютерное зрение. Современный подход. — М. : Издательский дом «Вильямс», 2004. — 928 с.
- [17] Шапиро Л., Стокман Дж. Компьютерное зрение. — М. : БИНОМ. Лаборатория знаний, 2006. — 752 с.
- [18] Яне Б. Цифровая обработка изображений. — М. : Техносфера, 2007. — 584 с.
- [19] Айфичер Э.С., Джервис Б.У. Цифровая обработка сигналов: практический подход. — 2-е изд. изд. — М. : Издательский дом "Вильямс" 2004. — 989 с.
- [20] Грузман И.С., Киричук В.С., др. Цифровая обработка изображений в информационных системах. — Н. : Издательство НГТУ, 2002. — 352 с.
- [21] Pratt W. K. Digital image processing. — NY. : John Willey&Sons, 2001. — 758 p.

[22] Оппенгейм А.В., Шафер Р.В. Цифровая обработка сигналов. — Издание 2-е, исправленное изд. — М. : Техносфера, 2009. — 856 с.

Приложение 1. Результаты обработки изображения «Грузовик»

Таблица 3.2: Характеристики процесса восстановления искажённого изображения 3.17а

Характеристика	Значение
Размер изображения (см. рис. 3.17а)	480×264
Размер блоков компенсации движения (см. рис. 3.23б)	8×8
Величина смаза (см. рис. 3.23б)	8
Угол смаза (см. рис. 3.23б)	8^0
Параметр шума σ_η (см. рис. 3.23б)	0.01
Время восстановление всего изображения (см. рис. 3.17а, 3.17б)	2.18 с.
Количество итераций при восстановлении всего изображения (см. рис. 3.17а, 3.17б)	58
Время восстановление фрагмента изображения (см. рис. 3.23б, 3.24)	1.7 с.
Количество итераций при восстановлении всего изображения (см. рис. 3.23б, 3.24)	32



а



б

а — искажённое изображение, б — восстановленное изображение

Рис. 3.17: Результат работы метода Люси-Ричардсона для изображения размером 480×264 . Найденные параметры искажения: величина смаза — 8 пикселей, угол — 8^0 , $\sigma_\eta = 0.01$



а — маска, соответствующая искажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.18: Приближение маски, характеризующей искажённую область на изображении, полученная после выделения перемещённых блоков в соответствии с условием (2.2). Блоки размером 8×8



а



б

а — маска, соответствующая искажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.19: Приближение маски, характеризующей искажённую область на изображении, полученная после выделения сегментации движения



а



б

а — маска, соответствующаяискажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.20: Приближение маски, характеризующей искажённую область на изображении, полученная после морфологического сглаживания маски 3.19а



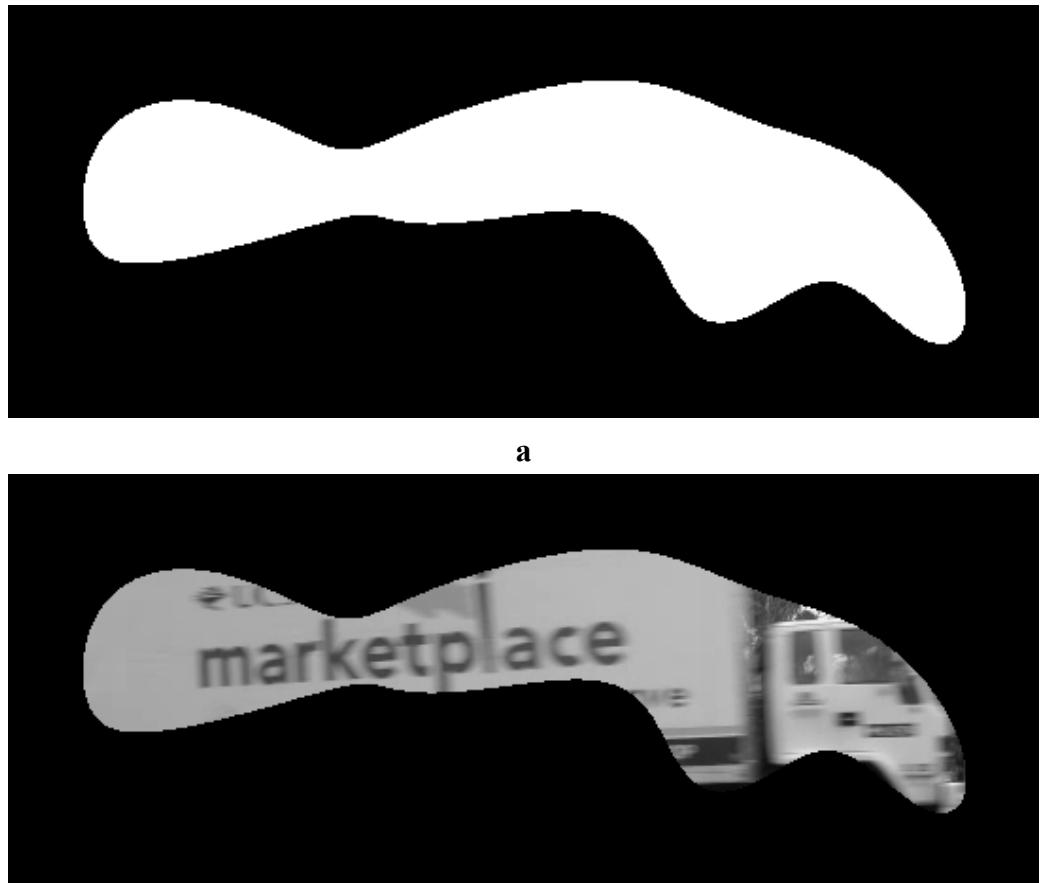
а



б

а — маска, соответствующаяискажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.21: Приближение маски, характеризующей искажённую область на изображении, полученная после заполнения «дырок» в маске 3.20а



б

а — маска, соответствующая искажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.22: Приближение маски, характеризующей искажённую область на изображении, полученная после сглаживания границы контура маски 3.21а



а



б

а — маска, соответствующая искажённой области изображения 3.17а,
б — искажённая область изображения 3.17а

Рис. 3.23: Маска, характеризующая искажённую область на изображении



Рис. 3.24: Результат восстановления искажённого изображения 3.17а

Приложение 2. Результаты обработки изображения «Автомобиль»

Таблица 3.3: Характеристики процесса восстановления искажённого изображения 3.25а

Характеристика	Значение
Размер изображения (см. рис. 3.25а)	1920×1080
Размер блоков компенсации движения (см. рис. 3.31б)	32×32
Величина смаза (см. рис. 3.31б)	18
Угол смаза (см. рис. 3.31б)	77^0
Параметр шума σ_η (см. рис. 3.31б)	0.0015
Время восстановление всего изображения (см. рис. 3.25а, 3.25б)	154.9 с.
Количество итераций при восстановлении всего изображения (см. рис. 3.25а, 3.25б)	120
Время восстановление фрагмента изображения (см. рис. 3.31б, 3.32)	15.2 с.
Количество итераций при восстановлении всего изображения (см. рис. 3.31б, 3.32)	41



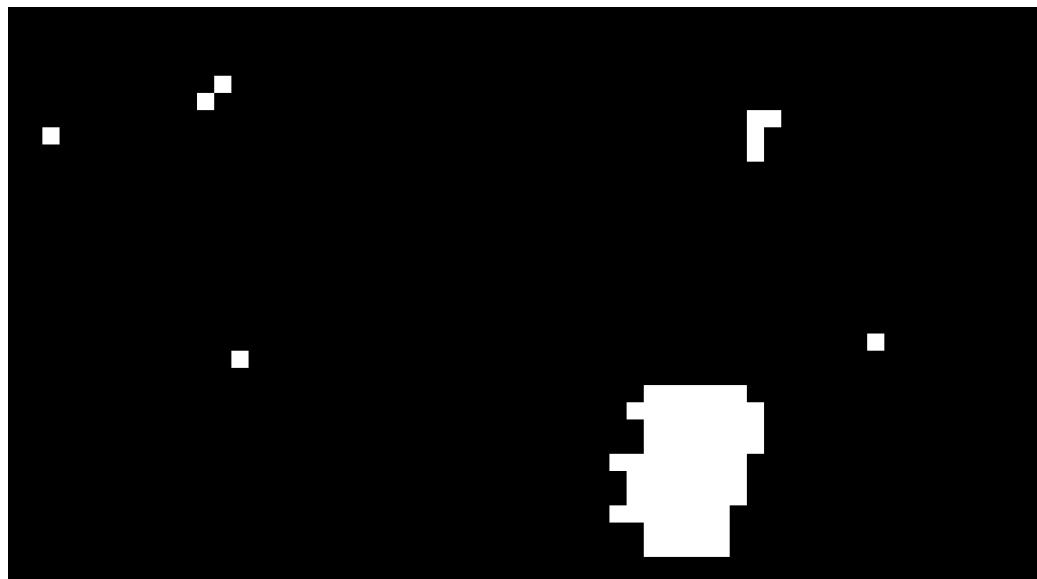
а



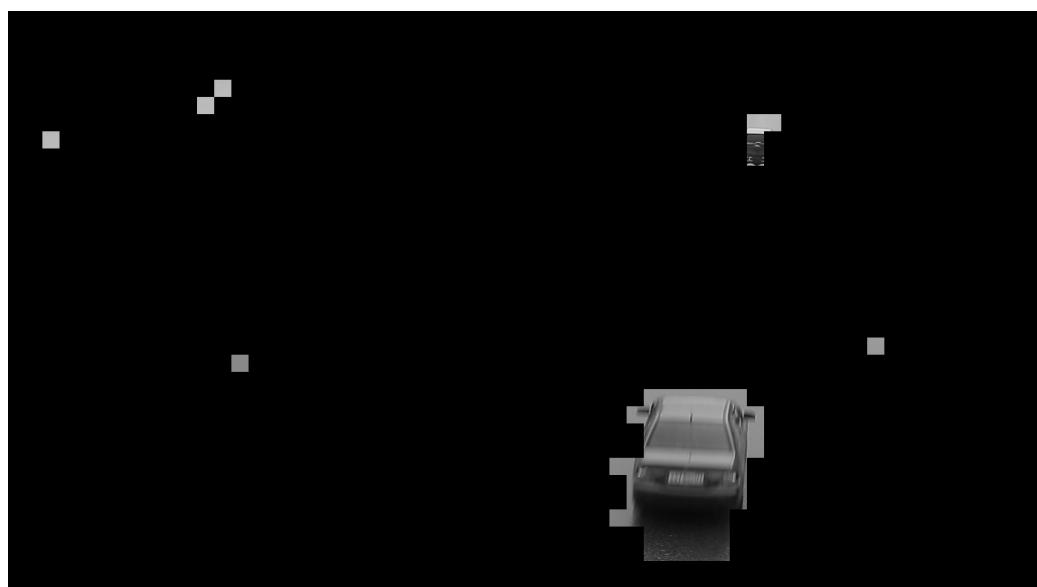
б

а — искажённое изображение, б — восстановленное изображение

Рис. 3.25: Результат работы метода Люси-Ричардсона для изображения размером 1920×1080 . Найденные параметры искажения: величина смазы — 18 пикселей, угол — 77^0 , $\sigma_\eta = 0.0015$



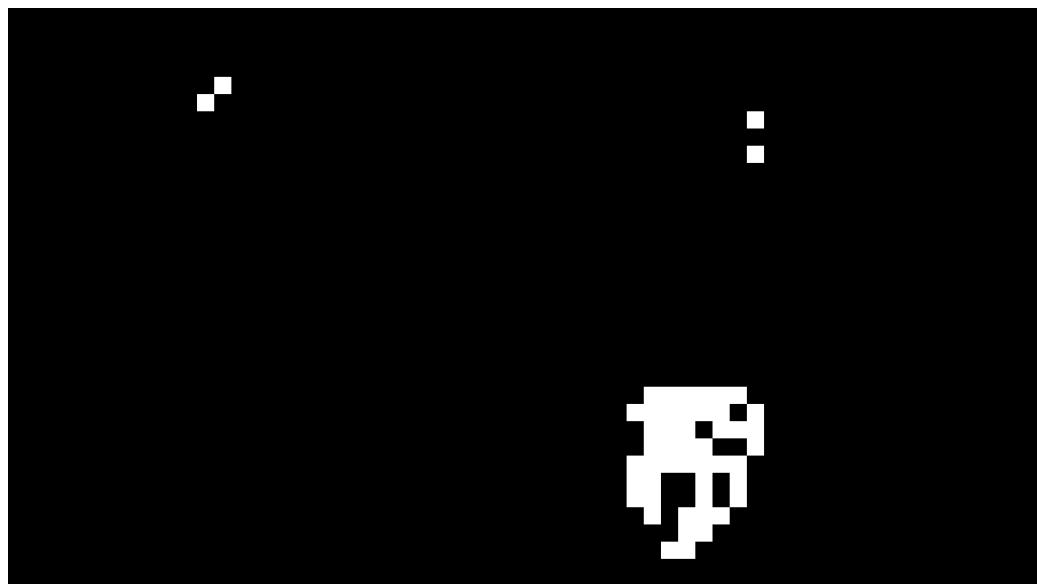
а



б

а — маска, соответствующаяискажённой области изображения 3.25а,
б — искажённая область изображения 3.25а

Рис. 3.26: Приближение маски, характеризующей искажённую область на изображении, полученная после выделения перемещённых блоков в соответствии с условием (2.2). Блоки размером 32×32



а



б

а — маска, соответствующая искажённой области изображения 3.25а,
б — искажённая область изображения 3.25а

Рис. 3.27: Приближение маски, характеризующей искажённую область на изображении, полученная после выделения сегментации движения



а



б

а — маска, соответствующая искажённой области изображения 3.25а,
б — искажённая область изображения 3.25а

Рис. 3.28: Приближение маски, характеризующей искажённую область на изображении, полученная после морфологического сглаживания маски 3.27а



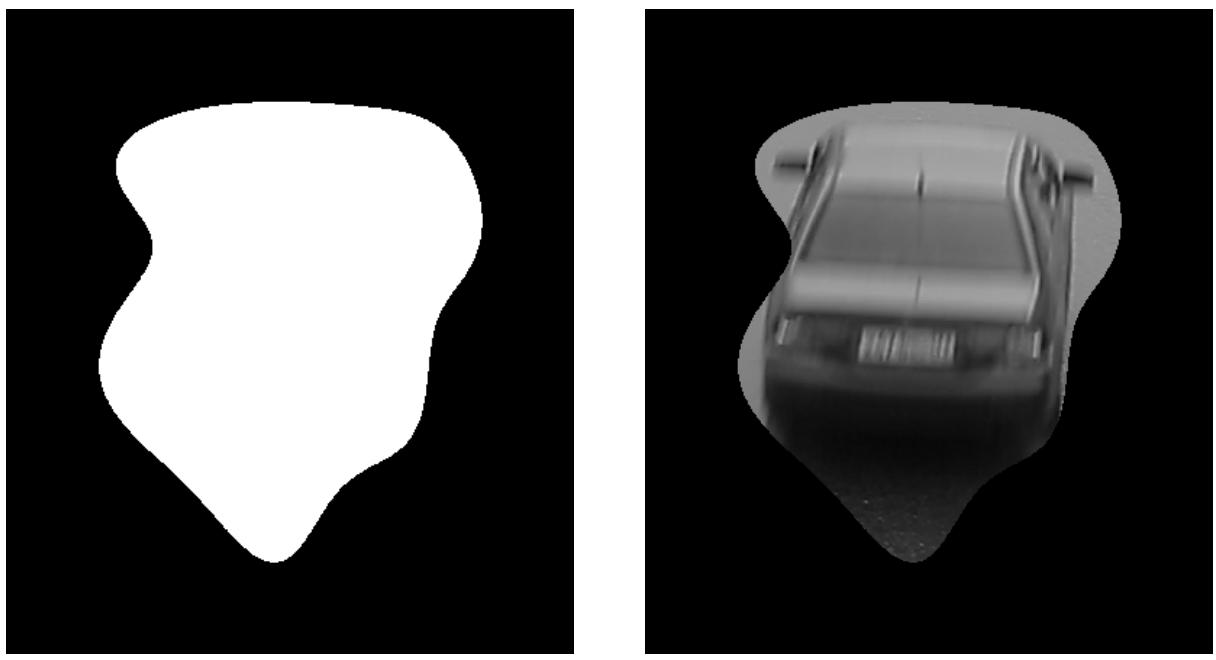
а



б

а — маска, соответствующая искажённой области изображения 3.25а,
б — искажённая область изображения 3.25а

Рис. 3.29: Приближение маски, характеризующей искажённую область на изображении, полученная после заполнения «дырок» в маске 3.28а

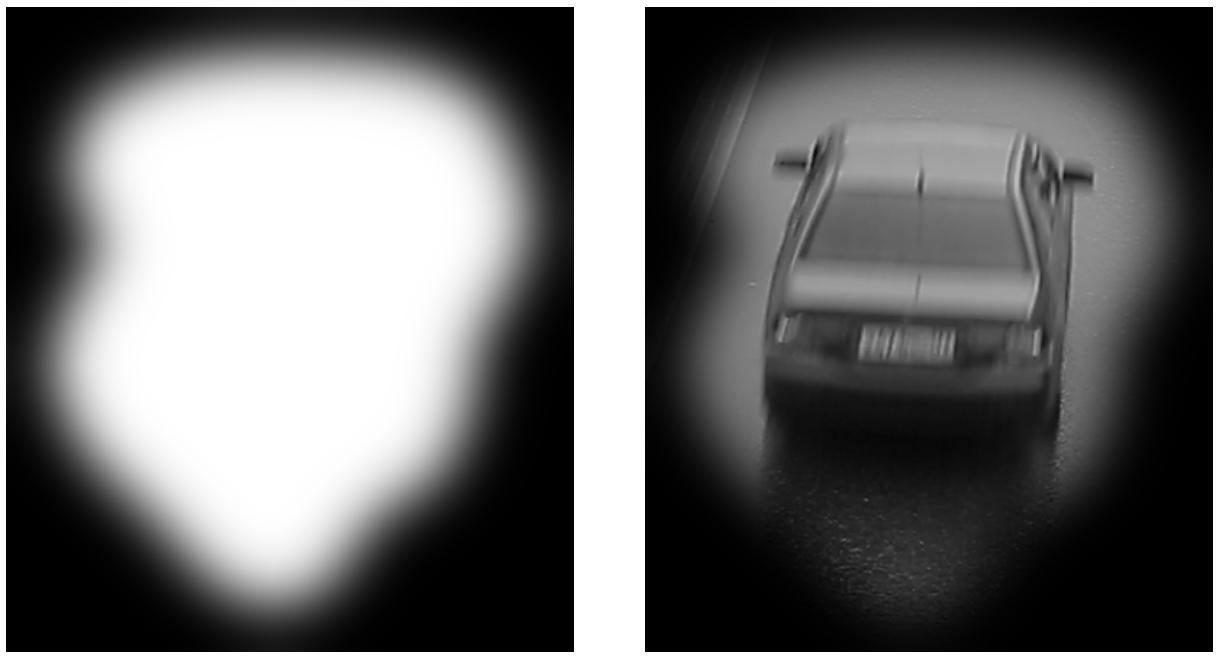


а

б

а — маска, соответствующаяискажённой области изображения 3.25а,
б —искажённая область изображения 3.25а

Рис. 3.30: Приближение маски, характеризующейискажённую область на изображении, полученная после сглаживания границы контура маски 3.29а



а

б

а — маска, соответствующаяискажённой области изображения 3.25а,
б —искажённая область изображения 3.25а

Рис. 3.31: Маска, характеризующейискажённую область на изображении



Рис. 3.32: Результат восстановления искажённого изображения 3.25а