

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**МОСКОВСКИЙ ФИЗИКО - ТЕХНИЧЕСКИЙ ИНСТИТУТ
(ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ)**

**КОМПЛЕКС АЛГОРИТМОВ УЛУЧШЕНИЯ ИЗОБРАЖЕНИЙ И
КОМПЕНСАЦИИ ИХ ГЕОМЕТРИЧЕСКИХ ИСКАЖЕНИЙ**

**Магистерская диссертация студента 915 группы ФРТК
Слынько Юрия Вячеславовича**

Научный руководитель: кандидат физико-математических наук, доцент, старший научный сотрудник Лагуткин Владимир Николаевич

Рецензент: кандидат физико-математических наук, доцент Лукьянов Александр Петрович

Москва 2005

1 Введение	3
2 Постановка задачи	3
3 Алгоритмы определений геометрических искажений	5
3.1 Общие положения	5
3.2 Метод корреляции	5
3.3 Принципиальная блок-схема алгоритма определения параметров трансформации	7
3.4 Метод выделения корреляционных окон	9
3.5 Метод поиска минимума с крупным шагом	12
3.6 Определение параметров искажений с использованием эллипсов ошибок	14
3.7 Метод спуска.....	15
3.8 Использование эталонов	16
3.9 Методы определения сдвига менее одного пикселя	18
3.10 Метод поиска сдвига по сжатым кадрам.....	18
4 Алгоритмы выбора параметров отображения	20
4.1 Общие положения	20
4.2 Выбор параметров отображения путем аппроксимации параметрическими функциями.....	22
4.3 Выбор параметров отображения эмпирическими функциями	22
4.4 Выбор параметров отображение минимизацией критерия качества.....	24
5 Алгоритмы определения побочных параметров видеопоследовательности	26
5.1 Общие положения	26
5.2 Шумоподобные кадры	27
5.3 Кадры с линейной структурой	28
5.4 Кадры с черезстрочной развёрткой	28
5.5 Смазанные кадры.....	30
6 Программная реализация алгоритмов	31
6.1 Блок схема программы улучшения изображений	31
6.2 Особенности программной реализации	32
7 Анализ качества и скорости работы алгоритмов	34
7.1 Методы исследований.....	34
7.2 Подробный анализ работы алгоритмов.....	35
7.2.1 Анализ алгоритмов определения сдвига	35
7.2.2 Анализ алгоритмов определения всех параметров трансформации	37
7.2.3 Анализ работы при ограничении времени выполнения	39
7.3 Сравнительный анализ работы различных алгоритмов.....	39
8 Выводы	40
9 Литература	42

1 Введение

При съёмке видеоизображений зачастую имеет место вибрация камеры, приводящая к нежелательному дрожанию получаемых изображений. В простейшем случае это небольшие колебания оптической оси камеры. В принципе, возможны существенные колебания, достигающие полукадра, особенно при съёмках с сильным увеличением. Кроме того, возможны и более сложные случаи, например, камера может не только колебаться, но и изменять ориентацию и положение в пространстве.

Тогда встаёт задача компенсации таких колебаний камеры и получения стабилизированных изображений. Эта задача распадается на две подзадачи.

Во-первых, необходимо определить параметры деформации изображения от кадра к кадру. В данной работе мы ограничимся классом линейных (аффинных) деформаций. Несмотря на относительную простоту модели деформации, она позволяет описать такие типичные искажения, как смещение, поворот, сжатие, перекося и их комбинации. Для успешного решения данной задачи также необходимо определить также побочные параметры, в частности: наличие черезстрочной развертки, шумоподобности или смазанности кадров и т.д.

Во-вторых, по оцененным параметрам деформации необходимо провести обратное преобразование входного кадра к прежней системе координат (компенсацию деформации). При этом необходимо отделить регулярные деформации, вызванные целенаправленным движением камеры, от нерегулярных шумовых возмущений и компенсировать только последние.

Основной целью при разработке алгоритмов было обеспечение следующих двух условий.

1. Алгоритмы должны работать устойчиво с самыми разными видеопоследовательностями. При этом последовательности могут быть сняты в сложных условиях и иметь очень плохое качество.
2. Данные алгоритмы должны обеспечивать обработку видеопотоков в реальном времени на современных неспециализированных вычислительных средствах.

2 Постановка задачи

Задача компенсации геометрических искажений распадается на две подзадачи.

А. Задача определения геометрических деформаций.

Для каждого кадра видеопоследовательности необходимо оценить его геометрическую трансформацию относительно предыдущего, полагая при этом, что кадры

являются изображением одного и того же стационарного фона. Под геометрической трансформацией подразумевается аффинное преобразование, т.е. преобразование вида:

$$\vec{r}(t+1) = A(t)\vec{r}(t) + \Delta\vec{r}(t), \text{ где} \quad (2.1)$$

$\vec{r}(t)$ - координаты точки фона на кадре,

$\vec{r}(t+1)$ - координаты той же точки фона на следующем кадре,

$A(t)$ и $\Delta\vec{r}(t)$ – параметры геометрических искажений, которые и надо определить.

При этом в процессе создания и отработки алгоритмов на реальных изображениях были учтены возможные отклонения от данной математической модели:

1. Изображение мало контрастно с малым отношением сигнал-шум.
2. Шумы негауссовы, нестационарные, коррелированные по времени и пространству.
3. Дополнительные искажения изображения. Например: смазы, нелинейные геометрические искажения, нелинейные изменения яркости и цветности по кадру, искажения в оптике камеры и т.д.
4. Искажения изображения за счет черезстрочной развёртки.
5. Искажение изображения за счет использования кодеков
6. Нестационарность самой снимаемой картины, т.е. движение одних её частей относительно других.

Б. Задача отделения регулярных и шумовых геометрических искажений.

Для адекватного отображения видеопоследовательности необходимо компенсировать только шумовую составляющую его геометрических искажений. Регулярную составляющую (например, поступательное смещение камеры) следует по возможности оставлять с минимальными изменениями. Т.е. геометрические искажения представляют собой следующую функцию времени:

$$\begin{cases} \Delta\vec{r}(t) = \Delta\vec{r}_s(t) + \Delta\vec{r}_n(t) \\ A(t) = A_s(t) + A_n(t) \end{cases}, \text{ где} \quad (2.2)$$

$\Delta\vec{r}_s(t)$ и $A_s(t)$ - полезные составляющие геометрических искажений,

$\Delta\vec{r}_n(t)$ и $A_n(t)$ - шумовые составляющие, которые и надо компенсировать.

Данная задача не может быть решена без априорных сведений о полезных составляющих искажений. Но эти сведения, как правило, скудны и не позволяют полностью и точно решить эту задачу. Поэтому в данной статье рассмотрены эмпирические способы решения, которые охватывают как можно большее число возможных ситуаций, а также могут быть настроены в соответствии с конкретной задачей.

3 Алгоритмы определений геометрических искажений

3.1 Общие положения

Для определения преобразования одного кадра относительно другого используется следующая принципиальная формула:

$$Q = \arg \min_Q X(\|F(t-1)_{ij}\|, \|T_Q(F(t))_{ij}\|), \text{ где} \quad (3.1)$$

Q – вектор параметров преобразования, который может включать как сдвиг кадров, так и угол поворота, и коэффициент масштабирования, или шесть аффинных коэффициентов,

X – функция «непохожести», которая должна быть тем меньше, чем сильнее совпадают её первый и второй аргументы,

$F(t)_{ij}$ – кадр момента времени t ,

T_Q – преобразование кадра с заданным вектором параметров преобразования.

Существуют различные методы оценки параметров деформации кадра, в частности, на основе оптического потока внутри кадра [4], на основе выделения характерных точек изображения [2],[3] или напрямую используя яркости точек входного изображения [1].

В данной работе под функцией $X(F_1, F_2)$ подразумевается функция среднеквадратичной невязки или какие-либо её обобщения. Данная функция достаточно быстро вычисляется. При этом вычисления легко распараллеливаемы на современных векторных процессорах.

В данной работе функция $T_Q(F)$ всегда представляется как комбинация функций целочисленного сдвига. Тогда она может быть выполнена всего в несколько операций на один кадр (число операций не зависит от размера кадра) путём изменения формального адреса начала кадра в памяти. Фактически, её вычисление занимает пренебрежимо малое время по сравнению с остальными вычислениями.

3.2 Метод корреляции

В простейшем случае функция $X(F_1, F_2)$ может быть представлена в виде:

$$X(F(t-1), T_Q(F(t))) = \frac{1}{S(M)} \sum_{(i,j) \in M(dx, dy)} (F(t-1)_{ij} - F(t)_{i+dx, j+dy})^2, \quad (3.2)$$

а само преобразование

$$Q = (dx, dy) = \arg \min_{(dx, dy) \in D} X(F(t-1), T_Q(F(t))), \quad (3.3)$$

где $Q=(dx, dy)$ – сдвиги вдоль обеих осей,

D - область возможных сдвигов, по которой производится поиск,

$M(dx,dy)$ – область, по которой производится вычисление невязки между кадрами (например, область, для которой точки (i,j) и $(i+dx,j+dy)$ принадлежат $F(t-1)$, т.е. область пересечения кадров).

$S(M)$ – площадь области M (количество точек).

Такую функцию $X(F_1, F_2)$ будем называть функцией корреляции двух кадров. Она зависит от двух аргументов - (dx, dy) , представляющих собой сдвиги по каждой из осей. В дальнейшем мы будем обозначать ее $X_{(F_1, F_2)}(dx, dy)$.

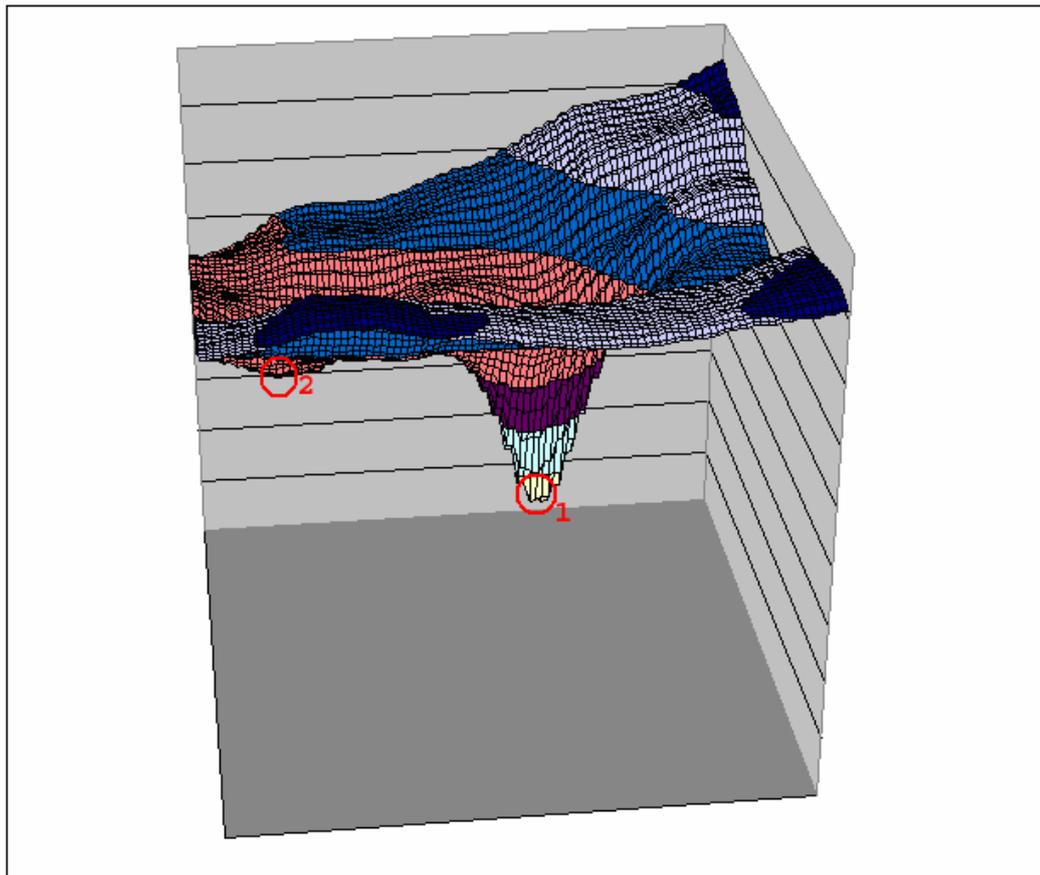


Рис1.

Типичный вид функции корреляции.

Видно два минимума 1 - главный и 2 - побочный.

Данный метод обладает следующими преимуществами:

- Устойчивость, т.к. здесь не используются никакие априорные данные относительно входных изображений. Более того, если входные кадры не соответствуют выбранной модели геометрических искажений, данный метод позволяет получить их разумную приблизительную оценку.
- Высокая скорость работы данного алгоритма, а также возможность параллельного вычисления на нескольких процессорах.

Весьма существенным моментом в данном методе является многомодальность функции (на Рис.1 хорошо видны побочные минимумы) $X_{(F_1, F_2)}(dx, dy)$, и для нахождения сдвига нельзя напрямую использовать методы спуска - необходим перебор всех возможных вариантов. За счет большой скорости вычисления функции корреляции, такой способ на общей скорости работы сказывается не существенно, однако сильно повышает надежность определения сдвига.

Недостатком метода перебора является сложность обобщения данного подхода на случай определения поворота кадра. Ведь оператор поворота кадра, в отличие от оператора целочисленного сдвига, весьма трудоемок, и его использование замедлит алгоритм на порядки. Однако задачу можно решить не используя оператор поворота кадра, а пользуясь только оператором сдвига. Это будет описано ниже.

3.3 Принципиальная блок-схема алгоритма определения параметров трансформации

Принципиальная схема алгоритма представлена на Схеме 1.

Данный алгоритм состоит из двух частей: определение сдвига по сжатым кадрам и определение всех параметров трансформации с использованием окон наибольшей информативности.

Методы с использованием сжатых кадров описаны в главе 3.10. Их вполне достаточно для определения сдвига. Однако, для более точного определения сдвига, а также для определения угла поворота и коэффициента масштабирования, необходимо в дополнение к этому методу использовать метод определения трансформации по окнам с наибольшей информативностью.

Метод с использованием окон с наибольшей информативностью описан в главах 3.4, 3.5, 3.6 (методы перебора) и 3.7 (метод спуска). Его вполне достаточно для точного определения сдвига и других параметров трансформации. Для увеличения скорости и устойчивости работы следует предварительно оценить сдвиг целого кадра с использованием сжатых кадров.

Методы использования эталонов описаны в главе 3.8.

Во всех представленных методах в качестве уточняющего метода используются различные алгоритмы оценки дробного сдвига, описанные в главе 3.9.

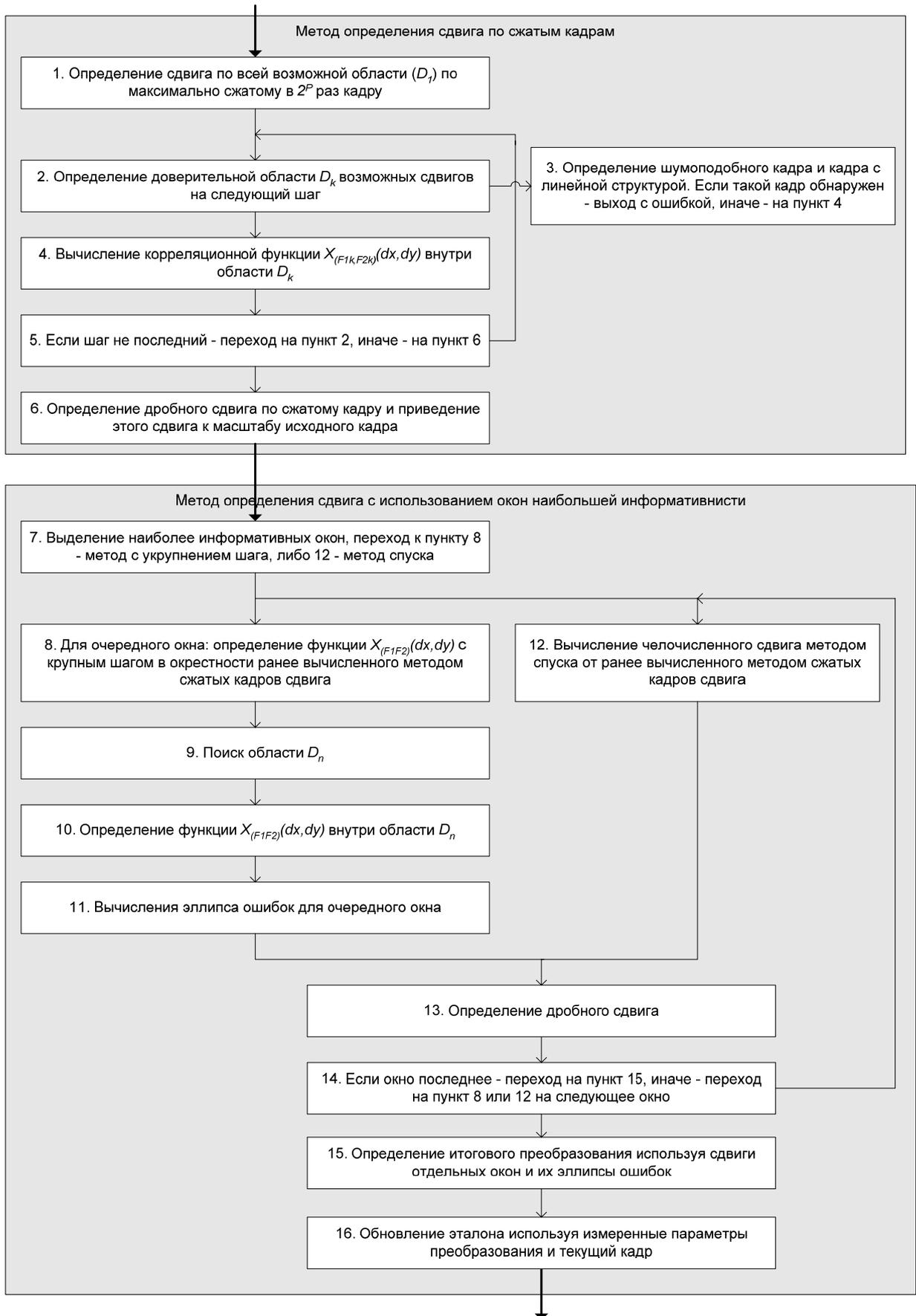


Схема 1. Блок схема алгоритма определения трансформации кадров

3.4 Метод выделения корреляционных окон

Для уменьшения количества вычислений можно использовать не весь кадр, а только некоторую его часть (пункт 7 Схемы 1). Произвольно ее выбирать нельзя, т.к. зачастую в кадрах попадаются неинформативные участки, ошибка определения сдвига по которым, даже полным перебором, может достигать размеров самого сдвига.



Рис 2.

Пример выделения наиболее информативных областей кадра

Удобно выбирать некоторое количество прямоугольных областей с наибольшей информативностью, и только по ним определять невязку. Т.е. под областью M следует понимать объединение нескольких прямоугольников (корреляционных окон). Это позволяет существенно снизить количество выполняемых операций и ускорить алгоритм.

Также это очень удобный способ определения угла поворота и коэффициента масштабирования кадра. Для этого надо подсчитать не сразу сдвиг всего кадра, а сдвиг отдельных окон. Если приписать точке, находящейся в центре окна, сдвиг этого окна, то методом наименьших квадратов [6] (зная сдвиг нескольких точек) нетрудно определить сдвиг, угол поворота и коэффициент масштабирования всего кадра. Т.е. производить вычисление параметров преобразования по следующей формуле (пункт 15 Схемы 1):

$$Q = \arg \min_Q \sum_{k=1}^N (\vec{\rho}_k - \tau_Q(\vec{r}_k))^2, \text{ где} \quad (3.4)$$

\vec{r}_k – координаты центров областей в кадре $F(t-1)$,

$\vec{\rho}_k$ - координаты тех же областей в кадре $F(t)$ (вычисленные по формулам (3.2), (3.3))

N – количество прямоугольников,

$\tau_Q(\vec{r}_k)$ - функция преобразования координат. Если необходимо определить только сдвиг, то эта функция принимает вид:

$$\tau_Q(\vec{r}_k) = \vec{r}_k + (dx, dy). \quad (3.5)$$

Если же надо определить также коэффициент масштабирования и поворота, то можно записать её в таком виде:

$$\tau_Q(\vec{r}_k) = \begin{pmatrix} a & b \\ -b & a \end{pmatrix} \cdot \vec{r}_k + \begin{pmatrix} dx \\ dy \end{pmatrix}, \text{ тогда} \quad (3.6)$$

угол поворота $\varphi = \text{arctg}(b/a)$, а коэффициент масштабирования $\kappa = \sqrt{a^2 + b^2}$.

Такая запись в обоих случаях позволяет свести задачу нахождения минимума к поиску решения системы линейных уравнений 2-го и 4-го порядка соответственно. Возможно обобщить данный метод на случай 6 аффинных коэффициентов (это приведёт к системе 6 линейных уравнений). Однако практическая ценность такого обобщения невелика, т.к. на практике кадры редко деформируются по всем 6 аффинным параметрам.

Поиск минимума не обязательно проводить по всей области. Можно начать двигаться от центра области D по «спирали», т.е. по квадратам, постепенно увеличивая их сторону. При нахождении минимума целесообразно обойти еще несколько более крупных витков спирали. Если в них не окажется точки с меньшим значением невязки, то поиск можно остановить. Так можно добиться уменьшения количества операций в случаях малых сдвигов, оставляя при этом возможность определять и большие сдвиги. Однако такой подход может привести к ошибкам вследствие выбора не глобального, а локального минимума.

В целом, метод выбора окон с наибольшей информативностью обладает следующими преимуществами:

- Позволяет существенно снизить количество вычислений, а следовательно, повысить скорость.
- Позволяет уменьшить влияние шумов, за счет исключения из рассмотрения участков с низким отношением сигнал-шум, которые не только не прибавляют точности определения сдвига, но и уменьшают ее.
- Позволяет обобщить корреляционный подход на случай определения углов поворота и масштабирования. По формулам, аналогичным (3.5) и (3.6) можно находить и более сложные преобразования кадров, такие как полные аффинные, квадратичные, проективные и др.
- Позволяет избежать краевых эффектов, т.к. наиболее информативные окна можно выбирать вдали от краев изображения.

- Позволяет осуществлять слежение за объектом, движущимся относительно фона, т.к. окна имеют относительно небольшой размер, но в тоже время позволяют точно определять смещение. Тогда сдвиг, определяемый по такому окну, будет уже не сдвигом кадра в целом, а сдвигом конкретного объекта.
- Позволяет избежать влияния небольших нестационарных (т.е. движущихся относительно фона) объектов на качество определения сдвига кадра в целом. Если одно из окон попадает на такой объект, то его можно исключить из формулы (3.4), когда сдвиг по нему дает большую невязку по сравнению со сдвигами по другим окнам.

Корреляционные окна выбираются из следующих соображений. Во-первых, они должны обладать максимальной информативностью. Во-вторых, они должны отстоять как можно дальше друг от друга. Т.е. критерий выбора k -ой точки следующий:

$$\vec{r}_k = \arg \max_{\vec{r}_k} \left(I(\vec{r}_k) \cdot \left(\min_{i=1..k-1} |\vec{r}_k - \vec{r}_i| + C \right) \right), \text{ где} \quad (3.7)$$

$I(\vec{r}_k)$ - информативность точки \vec{r}_k ,

C – некоторая константа, подбираемая экспериментально (можно взять порядка половины размера кадра).

Информативность окна вычисляется как минимум автокорреляционной функции вдоль нескольких направлений, т.е. по формуле:

$$I(\vec{r}_k) = \min_{l=1..L} \frac{1}{\sqrt{dx^2(l) + dy^2(l)}} \sum_{(i,j) \in W(\vec{r}_k)} (F_{ij} - F_{i+dx(l),j+dy(l)})^2, \text{ где} \quad (3.8)$$

$(dx(l), dy(l))$ – набор направлений,

$W(\vec{r}_k)$ - корреляционное окно с центром в точке \vec{r}_k ,

L – количество направлений.

В данной работе использовались следующие параметры:

$L=4$,

$(dx(1), dy(1))=(0,1)$,

$(dx(2), dy(2))=(1,1)$,

$(dx(3), dy(3))=(1,0)$,

$(dx(4), dy(4))=(1,-1)$.

Выбор автокорреляционной функции в качестве критерия информативности обоснован тем, что в качестве целевой функции используется функция невязки, а следовательно, ее крутизна пропорциональна точности оценки сдвига.

Здесь необходимо заметить, что после того, как был вычислен сдвиг первого окна, за область D удобно принять окрестность вычисленного сдвига. Тогда можно существенно уменьшить её площадь по сравнению с априорной, т.е. сократить объём вычислений.

3.5 Метод поиска минимума с крупным шагом

Для уменьшения количества операций можно применять следующую схему. Сначала перебрать всю область D с крупным шагом (см. Рис 3), т.е. перебирать не все точки, а пропускать их с каким-то шагом. В данной работе был выбран шаг 4 (пункт 8 Схемы 1). После этого надо обойти область вблизи вычисленного минимума.

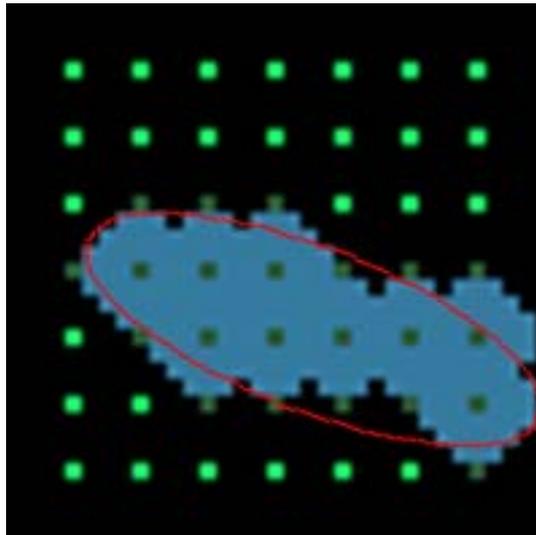


Рис 3.

Область, по которой вычислялась функция $X_{(F_1 F_2)}(dx, dy)$. Зеленым цветом обозначены точки, в которых вычислялась функция при переборе с крупным шагом. Темно зеленым – область \tilde{D}_n , светло зеленым – точки, не попавшие в область \tilde{D}_n . Синим – область D_n . Красный эллипс – эллипс ошибок.

Однако здесь есть определённые проблемы. Т.к. вид функции $X_{(F_1 F_2)}(dx, dy)$ от координат может быть очень сложный, а ее минимум слабо выраженным и сильно испорченным шумом, то зачастую нельзя ограничиться окрестностью одной точки. Приходится обходить окрестность нескольких точек. Эти точки выбираются так, что значения функции $X_{(F_1 F_2)}(dx, dy)$ в них может отличаться от минимума не за счет неправильного определения сдвига, а за счет шумовых флуктуаций.

Таким образом, для выделения области с «подозрительными» значениями функции $X_{(F_1 F_2)}(dx, dy)$ необходимо определить доверительную область ее значений в минимуме, т.е. математическое ожидание и дисперсию, предполагая, что ее значения определяются только шумом на изображении.

Предполагается, что шум в пикселях нормальный некоррелированный. Тогда значения функции $X_{(F_1F_2)}(dx, dy)$ имеют χ^2 -распределение, а её значение в минимуме будем считать математическим ожиданием. Это не вполне верно, однако более точную оценку математического ожидания сделать трудно, т.к. неизвестно, какие именно значения функции $X_{(F_1F_2)}(dx, dy)$ обусловлены шумом, а какие – разностью полезных сигналов.

Тогда дисперсия шума в отдельных пикселях будет:

$$\sigma_p = \sqrt{\frac{X_{\min}}{N_p}}, \text{ где} \quad (3.9)$$

X_{\min} – значение функции $X_{(F_1F_2)}(dx, dy)$ в минимуме,

N_p – количество точек в области $M(dx, dy)$, соответствующей минимуму.

Тогда математическое ожидание и дисперсия значений функции $X_{(F_1F_2)}(dx_i, dy_i)$

будет следующим:

$$m_i = \frac{X_{\min}}{N_p} \cdot N_i$$

$$\sigma_i = \frac{X_{\min}}{N_p} \cdot \sqrt{2N_i}, \text{ где} \quad (3.10)$$

m_i – математическое ожидание,

σ_i - дисперсия,

N_i – количество точек в области $M(dx_i, dy_i)$, по которой считалась функция $X_{(F_1F_2)}(dx_i, dy_i)$.

Тогда доверительная область, в которой могут лежать значения функции $X_{(F_1F_2)}(dx_i, dy_i)$, обусловленные шумом, будет следующая:

$$\tilde{D}_n = \{dx_i, dy_i : X_{(F_1F_2)}(dx_i, dy_i) < m_i + B \cdot \sigma_i\}, \text{ где} \quad (3.11)$$

B – выбираемый экспериментально порог (~ 3).

Затем область \tilde{D}_n дополняется до связанной области D_n (см. Рис 3 и пункт 9 Схемы 1). В ней вычисляются значения функции $X_{(F_1F_2)}(dx, dy)$ и находится ее минимум, который и соответствует искомому сдвигу (пункт 10 Схемы 1).

Данный подход обладает следующими полезными свойствами:

- Позволяет существенно уменьшить количество вычислений за счет не полного перебора всех вариантов, не увеличивая при этом вероятности попадания в локальный минимум.

- Позволяет учесть не только возможность существования побочных минимумов (т.е. производится обход всей области, пусть и с крупным шагом), но и возможность существования зашумленных кадров с плохо обусловленным минимумом корреляционной функции.
- Устойчиво работает с шумами, свойства которых существенно отличаются от тех, на основе которых выводились формулы (3.9), (3.10) и (3.11). А на практике очень часто встречаются не гауссовы, коррелированные по пространству и по времени, нестационарные и т.д. шумы. Эта устойчивость достигается за счет выбора параметра B , который может быть подобран для адекватной работы с широким классом шумов.

3.6 Определение параметров искажений с использованием эллипсов ошибок

Т.к. в ходе поиска максимума были вычислены значения функции $X_{(F_1, F_2)}(dx, dy)$ в довольно большой области, то эту информацию можно использовать для улучшения качества оценки параметров преобразования.

В частности, на её основе можно вычислить ковариационную матрицу ошибок оценивания сдвигов отдельных окон (пункт 11 Схемы 1). Для этого область D_n необходимо пересчитать для нового значения X_{min} по формулам 3.9, 3.10 и 3.11 и с помощью метода наименьших квадратов [6] аппроксимировать эллипсом вида (см. Рис 3):

$$A(x-x_{min})^2 + 2B(x-x_{min})(y-y_{min}) + C(y-y_{min})^2 = 0, \text{ где} \quad (3.12)$$

(x_{min}, y_{min}) – точка минимума функции $X_{(F_1, F_2)}(dx, dy)$.

Тогда матрица

$$C^{-1} = \begin{vmatrix} A & B \\ B & C \end{vmatrix} \text{ является оценкой ковариационной матрицы ошибок оценивания}$$

сдвигов отдельных окон.

Для оценки параметров аффинных искажений можно преобразовать формулу (3.4) так, чтобы в ней использовались вычисленные ковариационные матрицы:

$$Q = \arg \min_Q \sum_{k=1}^N (\bar{\rho}_k - \tau_Q(\bar{r}_k)) C_k^{-1} (\bar{\rho}_k - \tau_Q(\bar{r}_k))^T \quad (3.13)$$

Данная схема позволяет:

- Улучшить точность оценки параметров искажений.

- Не учитывать (или учитывать с малым весом) неточные измерения. Если процедура поиска минимума попала в локальный минимум, то эллипс ошибок обычно велик. Это позволяет не учитывать плохие измерения.
- Исключать из рассмотрения шумовые окна, т.е. окна, в которых отсутствует полезный сигнал. Для таких окон эллипс ошибок велик, и по этому признаку их можно исключать из обработки.

3.7 Метод спуска

Помимо метода полного перебора возможно также использование метода спуска (пункт 12 Схемы 1).

Он работает по следующему итеративному алгоритму:

1. На первом шаге выбирается некоторая начальная точка (dx_0, dy_0) . Ею может быть либо точка $(0, 0)$, либо некоторое предсказание сдвига, полученное из оценки сдвига, либо по предыдущим окнам, либо по методу с использованием сжатых кадров.
2. Перебираются все точки в некоторой окрестности D_i текущей точки (dx_i, dy_i) . Размеры окрестности выбираются экспериментально – например, в данной работе в качестве такой окрестности был выбран квадрат размером 3 на 3 пикселя с центром в точке (dx_i, dy_i) .
3. В выбранной области D_i считается значение функции $X_{(F_1 F_2)}(dx, dy)$ и выбирается ее минимум. Если точка минимума находится на границе области D_i , то она выбирается в качестве точки (dx_{i+1}, dy_{i+1}) и переходит на пункт 2. Иначе точка минимума считается точкой глобального минимума, т.е. искомой.



Рис 4.

Вид функции $X_{(F_1 F_2)}(dx, dy)$, вычисленной методом спуска. Более яркие точки соответствуют большим значениям функции.

Результат вычисления функции $X_{(F_1, F_2)}(dx, dy)$ по данному методу представлен на Рис.4. Здесь для наглядности размер области D_i был установлен 5 на 5 пикселей.

Данный алгоритм работает значительно быстрее алгоритма с перебором, но и не исключает возможности попадания в побочный локальный минимум. Для устранения такой возможности необходимо, чтобы сдвиг относительно точки (dx_0, dy_0) был небольшим. Это достигается либо при небольших колебаниях изображения, либо при достаточно точных предсказаниях сдвига. Как показывает практика, предсказания сдвига, сделанные на основе метода с использованием сжатых кадров, достаточно точны и в подавляющем большинстве случаев позволяют точно определить точку глобального минимума функции $X_{(F_1, F_2)}(dx, dy)$ методом спуска.

3.8 Использование эталонов

Для улучшения качества работы алгоритмов можно использовать эталонные кадры. Т.е. сравнивать текущий кадр не с предыдущим, а с эталонным. Эталонный кадр создается путем накопления кадров с учетом измеренных сдвигов.

При этом целесообразно хранить эталон не всего кадра, а только выбранных окон наибольшей информативности. Это позволяет не только уменьшить количество выполняемых операций, но и улучшить качество эталонов. Связанно это с тем, что целиком кадр плохо подчиняется выбранной модели искажений, да и эталон всего кадра накопить сложно. А в небольшом окне отклонения от модели менее существенны и ими можно пренебречь.

В качестве начального эталона окна можно взять окно из первого кадра в последовательности. Далее, после определения сдвига очередного окна необходимо обновить эталон (пункт 16 Схемы 1) с учетом текущего кадра и измеренного сдвига. Обновление эталона производится по следующей формуле:

$$E_{ij}(t+1) = (1 - W) \cdot E_{ij}(t) + W \cdot T_Q(F(t))_{ij}, \text{ где} \quad (3.14)$$

$E_{ij}(t)$ – кадр эталона,

$T_Q(F(t))$ – преобразование текущего кадра в соответствии с измеренными параметрами трансформации Q ,

W – весовой коэффициент, определяемый формулой

$$W = \frac{1}{\max\{t - t_0, T_w\}}, \text{ где} \quad (3.15)$$

T_w – время накопления эталона,

t_0 – время начала хранения эталона.

Время накопления T_w выбирается экспериментально, исходя из точности соответствия исходной видеопоследовательности выбранной модели. Если соответствие хорошее, то целесообразно выбирать большое время накопления. Если же имеют место некоторые несоответствия модели и реальной последовательности (например, наличие движущихся объектов в кадре, сложные геометрические преобразования от кадра к кадру и т.д.), то при большом времени накопления эталон может размазываться и становиться непригодным к определению сдвига по нему.

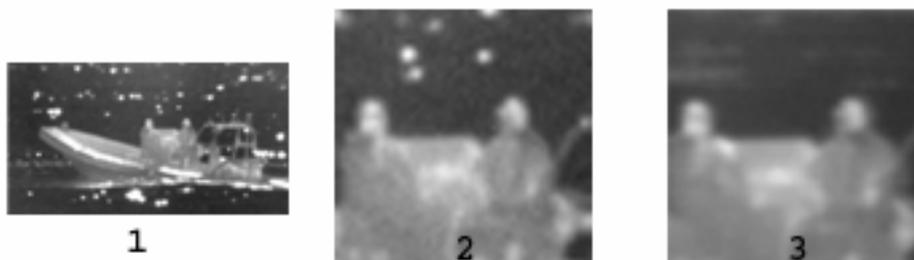


Рис 5.

1 - Исходный кадр – лодка, плывущая на фоне ярких бликов воды, 2 – фрагмент этого кадра – хорошо видны сильные шумы и мешающие объекты фона, 3- эталон этого фрагмент – шумы сильно подавлены, а мешающий фон смазан

Следует обратить внимание на то, что в данном случае функция $T_Q(F(t))$ не является функцией целочисленного сдвига, как всюду в этой статье. Для создания качественного эталона эта функция должна точно учитывать все измеренные параметры преобразования, т.е. дробные сдвиги, угол поворота и т.д.

Для использования эталона для определения сдвига окна необходимо вместо сдвига относительно опорного кадра вычислять сдвиг относительно эталона, т.е. вычислять не функцию $X_{(F_1, F_2)}(dx, dy)$, а функцию $X_{(EF_2)}(dx, dy)$.

Использование эталонов дает целый ряд преимуществ по сравнению с определением сдвига между последовательными кадрами (см. Рис 5):

- Снижается ошибка определения сдвига, т.к. в эталоне уровень шума значительно ниже, чем в исходной последовательности.
- Снижается ошибка определения сдвига между кадрами с моментами времени, далеко отстоящими друг от друга. Она появлялась за счет интегрирования ошибки сдвига между последовательными кадрами. Избежать её трудно, если сравнивать кадр не с предыдущим, а с отстоящим дальше по времени. Т.к. если кадр изменяется (пусть даже медленно), то кадр, отстоящий далеко по

времени уже плохо согласуется с текущим. В эталонах используется информация и из последних кадров, в то время как ошибка определения сдвига не интегрируется.

- Если стоит задача стабилизировать не весь кадр, а некоторый объект, который смещается относительно всего остального кадра, то введение эталонов позволяет уменьшить влияние фона на определение сдвига. В этом случае удерживаемый объект накапливается хорошо, в то время как фон размывается и становится малоинформативным, а следовательно, слабо влияет на определение сдвига.

3.9 Методы определения сдвига менее одного пикселя

Для улучшения работы алгоритма желательно определять сдвиг с точностью более одного пикселя (пункт 13 Схемы 1). Это особенно важно в рассмотренном ниже методе поиска сдвига по сжатым кадрам, т.к. там дробный сдвиг сжатого кадра может соответствовать большому целочисленному сдвигу исходного изображения.

Различные способы отыскания дробного сдвига подробно рассмотрены в статье [5]. Каждый из них можно использовать для уточнения сдвига и в данных алгоритмах.

Здесь, однако, есть одна особенность. Она заключается в том, что в ходе определения целочисленного сдвига уже были вычислены значения функции $X_{(F_1, F_2)}(dx, dy)$ в окрестности минимума. Т.к. сдвиг соответствует минимуму этой функции, то можно аппроксимировать её значения в дискретных точках гладкой поверхностью и найти минимум на этой поверхности.

Удобно брать 9 точек вокруг максимума и методом наименьших квадратов [6] находить параметры параболоида $z = ax^2 + bxy + cy^2 + dx + ey$. Тогда минимум этого параболоида и соответствует минимуму функции $X_{(F_1, F_2)}(dx, dy)$, а значит и искомому сдвигу. Этот метод дает несколько меньшую точность, чем рассмотренные в статье [5], но, в силу указанных выше причин, работает существенно быстрее.

3.10 Метод поиска сдвига по сжатым кадрам

Другим способом улучшения скорости (а зачастую и качества) работы алгоритма является процедура оценки сдвига по сжатым кадрам.

Предварительно следует из исходных кадров получить сжатые. Т.к. здесь необходимо из одного кадра получить сразу последовательность кадров разного масштаба (Рис 6), то удобнее всего использовать сжатие в 2 раза. Тогда:

$$F^{k+1} = \Sigma(F^k), \text{ где} \tag{3.16}$$

F^k – кадр, сжатый в k раз (т.е. имеющий в k раз меньший линейный размер, чем исходный),

$\Sigma(F)$ - некоторая функция сжатия в 2 раза.

Для получения ряда кадров с разным масштабом надо последовательно применять функцию сжатия сначала к исходному кадру, а потом и к сжатым.



Рис 6.

Последовательность сжатых в 2 раза кадров

В качестве функции сжатия в данном случае выбрана следующая:

$$\Sigma(F)_{ij} = \sum_{l,k=0}^1 F_{i+l,j+k} \cdot \quad (3.17)$$

После получения сжатых кадров можно запускать процедуру определения сдвига с их использованием. Работает она по следующей итеративной схеме:

1. Для первого шага выбирается область D_1 как прямоугольник заданных размеров (практика показывает, что сдвиги в пол кадра определяются достаточно надёжно). Внутри этой области считается функция $X_{(F_1^P F_2^P)}(dx, dy)$ - т.е. корреляционная функция двух сжатых кадров. Величина P выбирается такой, чтобы сжатые в 2^P раз кадры имели размер сторон 15–30 точек (см. Схему 1, пункт 1).
2. Выбирается область D_k по формулам (3.9), (3.10) и (3.11), только вместо функции $X_{(F_1 F_2)}(dx, dy)$ используется $X_{(F_1^{P-k} F_2^{P-k})}(dx, dy)$ - пункт 2 Схемы 1.
3. На k -ом шаге берутся кадры F_1^{P-k} и F_2^{P-k} - сжатые в $P-k$ раз исходные кадры. Для этих кадров вычисляется функция $X_{(F_1^{P-k} F_2^{P-k})}(dx, dy)$ внутри области D_{k+1} (см. Схему 1, пункт 4).
4. Если шаг k – не последний – переход на следующий шаг $k+1$ на пункт 2. Иначе – переход на пункт 5. Здесь следует отметить, что данную процедуру целесообразно прерывать задолго до достижения шага, на котором используются кадры исходного масштаба, т.к. сравнение целого кадра требует очень большого количества операций. Удобнее сделать 2–3 шага, а затем

перейти к описанной выше процедуре определения параметров преобразования с использованием нескольких окон (см. Схему 1, пункт 5).

5. Для последнего шага определяется дробный сдвиг (см главу 3.9) и все сдвиги приводятся к масштабу исходного кадра – пункт 6 Схемы 1.

Метод с использованием сжатия кадра обладает существенными преимуществами по сравнению с другими вышеописанными методами:

- Он позволяет вычислять сдвиги до полукадра, что невозможно другими способами ввиду большого количества требуемых операций.
- Вероятность ошибок существенно меньше, т.к. можно полностью проверить все возможные варианты сдвигов, а значит нельзя попасть в локальный минимум. Также существенно, что здесь сравнивается весь кадр, а не какая-то его часть. Это опять-таки приводит к уменьшению количества ошибок.
- Метод работает несколько быстрее описанных выше.
- Он гораздо лучше работает со смазанными кадрами или кадрами, в которых преобладают низкие пространственные частоты.

Следует заметить, что, несмотря на то, что при сжатии не определяются угол поворота и коэффициент масштабирования, они вычисляются в процедуре уточнения, а их наличие в кадре не сказывается на работоспособности алгоритма определения сдвига с помощью масштабирования.

4 Алгоритмы выбора параметров отображения

4.1 Общие положения

После определение параметров трансформации методами, описанными в главе 3, необходимо отобразить стабилизированную видеопоследовательность, т.е. компенсировать эти искажения.

Однако геометрические искажения могут быть вызваны не только мешающими (шумовыми) воздействиями, но и регулярными перемещениями камеры. Тогда встает задача отделения тряски камеры от ее движения за счет управления.

Т.е. надо решить следующую задачу:

$$\begin{cases} \Delta\vec{r}_n(t) = F_r(\Delta\vec{r}(t), \Delta\vec{r}(t-1), \Delta\vec{r}(t-2)...), \\ A_n(t) = F_A(A(t), A(t-1), A(t-2)...), \end{cases} \text{ где} \quad (4.1)$$

$\Delta\vec{r}_n(t)$ и $A_n(t)$ - шумовые составляющие, которые надо компенсировать (см. (2.2)),

$A(t)$ и $\Delta\vec{r}(t)$ – параметры геометрических искажений (см. (2.1)), которые были определены методами, описанными в главе 3,

F_r и F_A – некоторые функции, которые должны быть подобраны такими, чтобы наиболее адекватно отделять шумовые составляющие геометрических искажений от не шумовых.

Для решения этой задачи удобно сделать следующие замены переменных:

- представлять матрицу A как комбинацию двух параметров – угла поворота φ и коэффициента масштабирования k :

$$A = \begin{vmatrix} a & b \\ -b & a \end{vmatrix} \Leftrightarrow \begin{vmatrix} \varphi = \operatorname{arctg}\left(\frac{b}{a}\right) \\ k = \sqrt{a^2 + b^2} \end{vmatrix} \quad (4.2)$$

- перейти от относительных преобразований двух последовательных кадров к абсолютным преобразованиям $A_a(t)$ и $\Delta\vec{r}_a(t)$ последнего кадра относительно первого (опорного):

$$\begin{aligned} A_a(t) &= \prod_{\tau=t}^0 A(\tau) \\ \Delta\vec{r}_a(t) &= \Delta\vec{r}(t) + \sum_{k=t-1}^0 \left(\prod_{\tau=t}^{k+1} A(\tau) \right) \cdot \Delta\vec{r}(k) \end{aligned} \quad (4.3)$$

Шумовая и полезная составляющие геометрических искажений вычисляются по аналогичным формулам.

Данная задача принципиально не формализуема, т.к. данные об управлении камерой в подавляющем большинстве недоступны либо недостаточно точны. Поэтому необходимо искать не точное решение, а приближенное. Здесь возможны три подхода:

1. Задаться некоторыми априорными сведениями относительно полезных составляющих $\Delta\vec{r}_s(t)$ и $A_s(t)$ искажений (глава 4.2). Например, аппроксимировать их какой-либо параметрической функцией.
2. Задаться некоторой эмпирической формой функций F_r и F_A (глава 4.3) с некоторым числом параметров, и подобрать эти параметры для наиболее адекватного решения задачи.
3. Задаться критерием качества (глава 4.4), который может включать в себя разнообразные факторы: например, скорость смещения результирующего изображения, степень несовпадения его с оригиналом и т.д. Тогда искомые параметры находятся путем минимизации этого критерия.

4.2 Выбор параметров отображения путем аппроксимации параметрическими функциями

В соответствии с подходом 1 задачу выбора параметров отображения (т.е. шумовых составляющих искажений $\Delta\vec{r}_n(t)$ и $A_n(t)$, которые надо компенсировать) можно решать путем аппроксимации параметров $\Delta\vec{r}_s(t)$ и $A_s(t)$ какими-либо параметрическими функциями $f_A(t, a_1, a_2, \dots)$ и $\vec{f}_r(t, r_1, r_2, \dots)$. Тогда, (с учетом (4.3)):

$$\begin{aligned} F_r(\Delta\vec{r}_a(t), \Delta\vec{r}_a(t-1), \Delta\vec{r}_a(t-2), \dots) &= \Delta\vec{r}_a(t) - \hat{\Delta\vec{r}}_s(\Delta\vec{r}_a(t), \Delta\vec{r}_a(t-1), \Delta\vec{r}_a(t-2), \dots), \\ F_A(A_a(t), A_a(t-1), A_a(t-2), \dots) &= A_a(t) - \hat{A}_s(A_a(t), A_a(t-1), A_a(t-2), \dots) \end{aligned}, \text{ где (4.4)}$$

$\hat{\Delta\vec{r}}_s$ и \hat{A}_s - оценки полезных составляющих, вычисляемые по формулам:

$$\begin{aligned} \hat{\Delta\vec{r}}_s(t) &= \vec{f}_r(t, \hat{r}_1, \hat{r}_2, \dots) \\ \hat{A}_s(t) &= f_A(t, \hat{a}_1, \hat{a}_2, \dots) \end{aligned} \quad (4.5)$$

$$\begin{aligned} (\hat{r}_1, \hat{r}_2, \dots) &= \arg \min_{r_1, r_2, \dots} \sum_{\tau=0}^t (\vec{r}_a(\tau) - \vec{f}_r(\tau, r_1, r_2, \dots))^2 \\ (\hat{a}_1, \hat{a}_2, \dots) &= \arg \min_{a_1, a_2, \dots} \sum_{\tau=0}^t (A_a(\tau) - f_A(\tau, a_1, a_2, \dots))^2 \end{aligned}, \text{ где (4.6)}$$

$\hat{r}_1, \hat{r}_2, \hat{a}_1, \hat{a}_2, \dots$ - оцениваемые параметры. Оценку этих параметров можно проводить либо методом наименьших квадратов, либо фильтром Калмана [6].

В качестве функций $f_A(t, a_1, a_2, \dots)$ и $\vec{f}_r(t, r_1, r_2, \dots)$ удобно брать полиномиальные функции небольших степеней. Это позволяет сгладить шум и достаточно точно описать поведение полезных составляющих геометрических искажений в простых случаях.

Однако на практике движения камеры зачастую не подчиняются простому полиномиальному закону, особенно, если камера управляется оператором вручную. В этом случае описание искажений с помощью таких функций становится неудовлетворительным.

4.3 Выбор параметров отображения эмпирическими функциями

В соответствии с подходом 2 функции F_r и F_A можно выбрать эмпирически, а потом найти такие параметры, которые будут описывать ситуацию наиболее адекватно.

В данной работе рассматривалось несколько таких функций:

Простейшая линейная функция:

$$\begin{cases} \Delta\bar{r}_{na} = \Delta\bar{r}_a \cdot \left(1 - \frac{1}{N_r}\right) \\ k_{na} = 1 \\ \varphi_{na} = \varphi_a \cdot \left(1 - \frac{1}{N_\varphi}\right) \end{cases}, \text{ где} \quad (4.7)$$

k_{na} и φ_{na} - представление шумовой матрицы A_{na} в виде (4.2),

φ_a - компонент матрица A_a ,

N_r и N_φ - параметры, выбираемые экспериментально (можно взять порядка 20 – 200).

Шумовая компонента коэффициента масштабирования здесь принята за единицу, так как крайне редко можно встретить случай, когда масштаб изображения колеблется случайным образом.

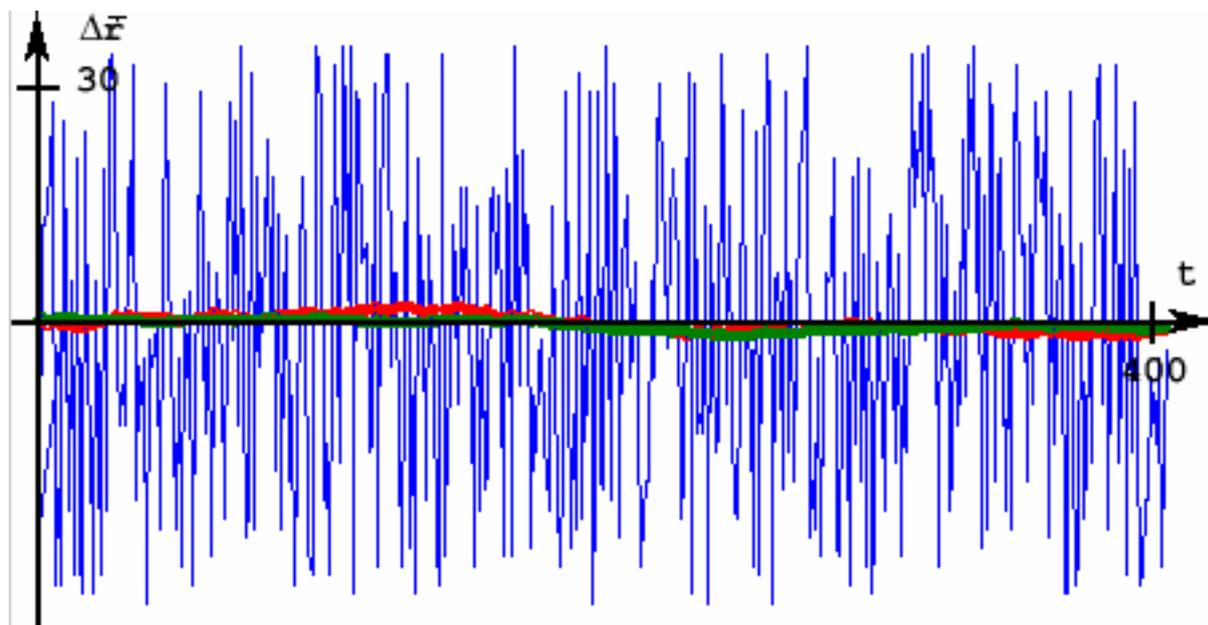


Рис 7.

Пример работы функций выбора параметров при колебании изображения вокруг некоторого положения. Синим цветом обозначена функция $\Delta\bar{r}$, зеленым - $\Delta\bar{r}_s$, вычисленная первым способом, красным – вторым. Видна высокая степень подавления шумовой компоненты.

Данный подход позволяет сгладить шумовые колебания, в то время как медленное поступательное и вращательное движения камеры будет отслеживаться (Рис 7). Однако если движение будет достаточно быстрым, то в конечном итоге входное изображение

может полностью уйти из выходного кадра, т.е. сместиться на столько, что не будет видно на выходном кадре (Рис 8).

Экспоненциальная функция:

$$\begin{cases} \Delta \bar{r}_{na} = \Delta \bar{r}_a \cdot \exp\left(-\frac{1}{N_{r1}} - \frac{|\Delta \bar{r}|}{N_{r2}}\right) \\ k_{na} = 1 \\ \varphi_{na} = \varphi_a \cdot \left(1 - \frac{1}{N_\varphi}\right) \end{cases}, \text{ где} \quad (4.8)$$

N_{r1} и N_{r2} – дополнительные настроечные параметры (в данном случае $N_{r1} = 20-200$, $N_{r2} = 0.05 N_{r1}$).

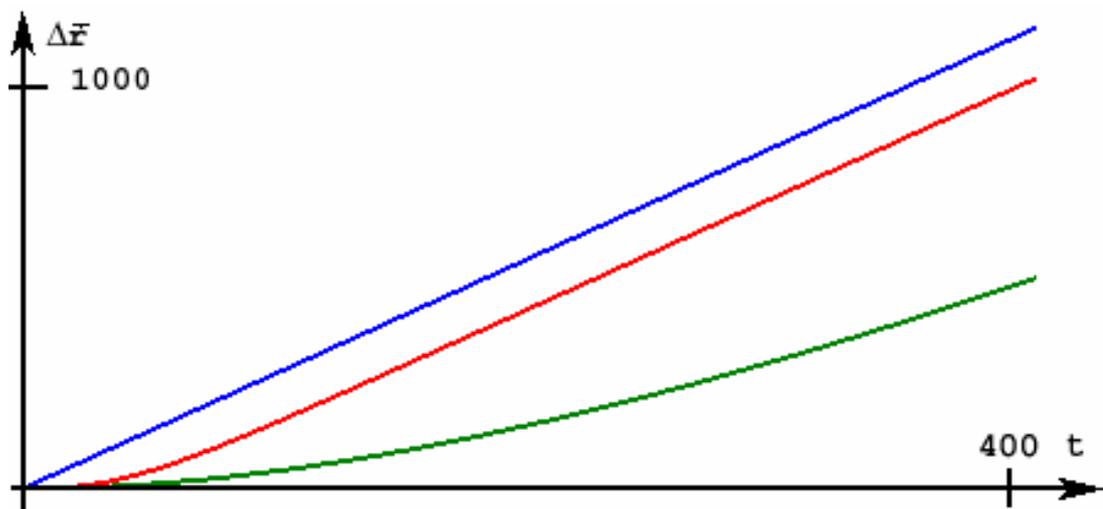


Рис 8.

Пример работы функций выбора параметров при поступательном смещении изображения. Обозначения аналогично Рис 7. Видно, что первая функция не вполне отслеживает поступательное движение изображения.

Данный подход при небольших сдвигах работает аналогично предыдущему (Рис 7), но при больших обеспечивает незначительное «отставание» выходного изображения от входного (Рис 8).

4.4 Выбор параметров отображение минимизацией критерия качества

В соответствии с подходом 3 функции F_r и F_A можно находить путем минимизации критерия качества. При этом критерий удобнее записывать не через шумовые составляющие, а через полезные. Тогда:

$$\begin{cases} A_{na}(t) = A_a(t) \hat{A}_{sa}^{-1}(t) \\ \Delta \vec{r}_{na}(t) = \Delta \vec{r}_a(t) - A_n(t) \cdot \Delta \vec{r}_{sa}(t) \\ (\hat{A}_{sa}(t), \Delta \vec{r}_{sa}(t)) = \arg \min_{A_{sa}, \Delta \vec{r}_{sa}} J(A_a(t), A_a(t-1), \dots, A_{sa}(t), A_{sa}(t-1), \dots), \text{ где} \\ \Delta \vec{r}_a(t), \Delta \vec{r}_a(t-1), \dots, \Delta \vec{r}_{sa}(t), \Delta \vec{r}_{sa}(t-1), \dots \end{cases} \quad (4.9)$$

$\hat{A}_{sa}(t)$ и $\Delta \vec{r}_{sa}(t)$ - оценки полезных составляющих геометрических искажений (относительно начального кадра),

$J(\dots)$ – критерий качества.

Данный критерий качества может включать в себя различные характеристики, которые должны быть минимальны для более качественного отображения выходного кадра. В данной работе использовался следующий критерий:

$$\begin{aligned} J(\dots) = & (\Delta \vec{r}_a(t) - \Delta \vec{r}_{sa}(t))^4 + \\ & \alpha (\Delta \vec{r}_{sa}(t) - \Delta \vec{r}_{sa}(t-1))^2 + \\ & \beta |\varphi_a(t) - \varphi_{sa}(t)|_\varphi^2 + \quad , \text{ где} \quad (4.10) \\ & \gamma |\varphi_{sa}(t) - \varphi_{sa}(t-1)|_\varphi^2 + \\ & \delta \cdot D^2(t, \Delta \vec{r}_{sa}(t), \varphi_{sa}(t)) \end{aligned}$$

$|\dots|_\varphi$ - модуль разности углов, учитывающий их цикличность,

$D^2(t, \Delta \vec{r}_{sa}(t), \varphi_{sa}(t))$ - количество точек на выходном кадре, на которые не попала информация с входного кадра,

$\alpha, \beta, \gamma, \delta$ - весовые коэффициенты, которые удобно выбирать в следующих пределах:

$$\begin{cases} \alpha = 5 \cdot 10^4 \div 5 \cdot 10^5 \\ \beta = 0.5 \div 5 \\ \gamma = \beta \cdot (10 \div 100) \\ \delta = 10^5 \div 10^6 \end{cases} \quad (4.11)$$

В данном критерии первое слагаемое $(\Delta \vec{r}_a(t) - \Delta \vec{r}_{sa}(t))^4$ не позволяет выходному изображению слишком сильно отдаляться от входного, тем самым обеспечивая отслеживание быстро сдвигающегося изображения. Второе слагаемое $\alpha (\Delta \vec{r}_{sa}(t) - \Delta \vec{r}_{sa}(t-1))^2$ не позволяет выходному изображению быстро колебаться, тем самым сглаживая шумовую составляющую сдвигов. Вместе эти два слагаемых определяют оптимальное сглаженное слежение за полезным сдвигом и подавление шумового сдвига.

Третье и четвертое слагаемые $(\beta|\varphi_a(t) - \varphi_{sa}(t)|_\varphi^2)$ и $\gamma|\varphi_{sa}(t) - \varphi_{sa}(t-1)|_\varphi^2)$ осуществляют аналогичное поведение выходного изображения относительно угла поворота.

Последнее слагаемое $\delta \cdot D^2(t, \Delta \vec{r}_{sa}(t), \varphi_{sa}(t))$ характеризует количество точек на выходном кадре, на которые не попала информация с входного кадра. Данное слагаемое имеет смысл, только если выходной кадр строится не из одного входного кадра, а из всей последовательности (см главу 6.2). Например, если сначала из всей входной последовательности, с учетом оцененных сдвигов, строится общая мозаика большого размера, а затем из этой мозаики «вырезается» выходной кадр меньшего размера. При этом в выходной кадр могут попасть точки мозаики, в которых не была записана информация ни с одного входного кадра. Количество таких точек необходимо свести к минимуму.

Задача минимизации критерия качества является нелинейной трехмерной задачей минимизации. А из-за присутствия последнего члена она не может быть решена аналитически. Для ее решения необходимо применять методы нелинейной многомерной оптимизации, например, метод Пауэла [7].

Данный метод выбора параметров отображения обладает следующими преимуществами:

- В критерий качества можно включать произвольное число параметров, которые позволяют синтезировать критерий произвольного качества и с произвольными свойствами. Причем каждое слагаемое имеет простой и понятный смысл, что позволяет синтезировать критерии с желаемыми свойствами.
- Все остальные описанные выше подходы могут быть представлены через минимизацию критерия качества путем выбора соответствующего критерия.

5 Алгоритмы определения побочных параметров видеопоследовательности

5.1 Общие положения

Зачастую входные кадры искажены так, что определить сдвиг по ним не представляется возможным. Иногда при таких искажениях сама задача определения сдвига между кадрами становится бессмысленной, т.е. последовательные кадры перестают даже приблизительно представлять собой один и тот же кадр, искаженный

геометрически. Или же имеет место слабая обусловленность задачи, тогда приходится ее оценивать и принимать соответствующее решение.

Среди возможных исключительных случаев при обработке видеопоследовательностей следует выделить следующие:

1. Шумоподобные кадры (глава 5.2), т.е. кадры, на которых отсутствует полезный сигнал. Это самый распространенный пример слабо обусловленной задачи.
2. Кадры с линейной структурой (глава 5.3). Это пример задачи, слабо обусловленной по одной переменной.
3. Кадры с черезстрочной разверткой (глава 5.4). Эффект черезстрочной развертки может привести к раздвоению изображения, а значит оно перестает соответствовать предложенной модели.
4. Смазанные кадры (глава 5.5). На таких кадрах изображение тоже может раздваиваться или искажаться еще сильнее, что существенно затрудняет их обработку.
5. Колебания яркости, например, при включении или выключении света в помещении, либо при сбое в работе камеры. Для их устранения можно производить выравнивание средней яркости кадров к одному уровню, однако при медленном изменении яркости в этом нет необходимости.

5.2 Шумоподобные кадры

Зачастую на вход алгоритма подаётся видеопоследовательность монотонных кадров. Т.е. кадров одного цвета (возможно зашумлённых). Тогда, в силу случайности шумов, функция $X_{(F_1 F_2)}(dx, dy)$ имеет минимум, который в подавляющем большинстве случаев находится не в точке $(dx, dy) = (0, 0)$. Это приводит к сильным колебаниям выходного изображения при статичном входном.

Простое сравнение входного и выходного кадров не даёт результата, т.к. при наличии шумов они не абсолютно идентичны. Тогда встаёт проблема нахождения порога, по которому отсеиваются подобные последовательности. Задача выбора такого порога, работающего адекватно на любых последовательностях, не может быть решена простыми способами без вычисления уровней шумов.

Отсеивание шумоподобных кадров можно производить как на этапе определения сдвига по сжатым кадрам, так и на этапе работы с окнами. В обоих случаях необходимо анализировать корреляционную функцию ($X_{(F_1 F_2)}(dx, dy)$ или $X_{(F_1^p F_2^p)}(dx, dy)$). Для нее по формулам (3.9), (3.10) и (3.11) определяется область D_k . Если эта область велика (т.е.

порядка 5-10 пикселей в диаметре), то кадр (или окно) признается шумоподобным и должно быть исключено из обработки, т.к. ошибка определения сдвига по такому кадру будет слишком велика.

5.3 Кадры с линейной структурой

Часто на вход алгоритма подаётся видеопоследовательность кадров, имеющих линейную структуру. Например, это может быть стандартная телевизионная заставка. Здесь та же проблема, что и в шумоподобных кадрах, но ошибка измерения сдвига может быть большой только вдоль одного направления.

Здесь, как и в предыдущем случае, надо по формулам (3.9), (3.10) и (3.11) определить область D_k . Затем, если эта область имеет линейный вид, принимается решение о линейности входных кадров (или окон).

Здесь возможны два варианта. Во-первых, можно прекратить обработку такой последовательности и на выходе выдавать то же, что и на входе. Во-вторых, можно подсчитать только сдвиг в направлении, перпендикулярном к линейной структуре в кадре и угол поворота.

В случае определения сдвига по сжатым кадрам можно просто приравнять сдвиг вдоль выделенного направления нулю

В случае работы с окнами можно достичь более существенных результатов. Для этого надо увеличить (в пределе до бесконечности) полуось эллипса ошибок вдоль линейной структуры. Затем, в реализации функции нахождения минимума (3.13) (т.е. в функции решения соответствующей линейной системы), ввести регуляризацию, описанную, например, в [8].

Это позволяет использовать для определения параметров трансформации даже окна с линейной структурой, фактически, используя их информацию о сдвиге лишь вдоль одной из осей. За счет этого можно также обеспечить определение всех искомым параметров, (даже если все выделенные на изображении окна имеют линейную структуру), если выделенное направление в окнах различается. Если же выделенное направление во всех окнах одно и то же, то сдвиг вдоль него за счет регуляризации будет приравнен к 0, что и является в данном случае наиболее оптимальным результатом.

5.4 Кадры с черезстрочной развёрткой

Зачастую кадры видеопоследовательности снимаются черезстрочной развёрткой, т.е. четные и нечетные строчки снимаются в разные моменты времени. При сильной тряске камеры это приводит к искажениям изображения (Рис 9), когда принятая математическая модель вовсе перестаёт быть справедливой. Однако полукадры, собранные только из

четных (или нечетных) строк, имеют только обычные, указанные в главе 2 (2.1), деформации.

Определение наличия эффекта черезстрочной развертки можно проводить путем сравнения корреляции между соседними строками и строками, идущими через одну, т.е. путем сравнения следующих величин:

$$\begin{cases} I_1 = \sum_{k=0}^N (F_{i(k),j(k)} - F_{i(k),j(k)+1})^2 \\ I_2 = \sum_{k=0}^N (F_{i(k),j(k)} - F_{i(k),j(k)+2})^2 \end{cases}, \text{ где} \quad (5.1)$$

N – количество точек, по которым определяется наличие черезстрочной развертки (может быть выбрано 1-10% от количества точек в кадре),

$i(k)$ и $j(k)$ – некоторые функции, согласно которым выбирается очередная точка. Их можно выбирать так, чтобы точки, участвующие в определении черезстрочной развертки располагались в узлах прямоугольной сетки.

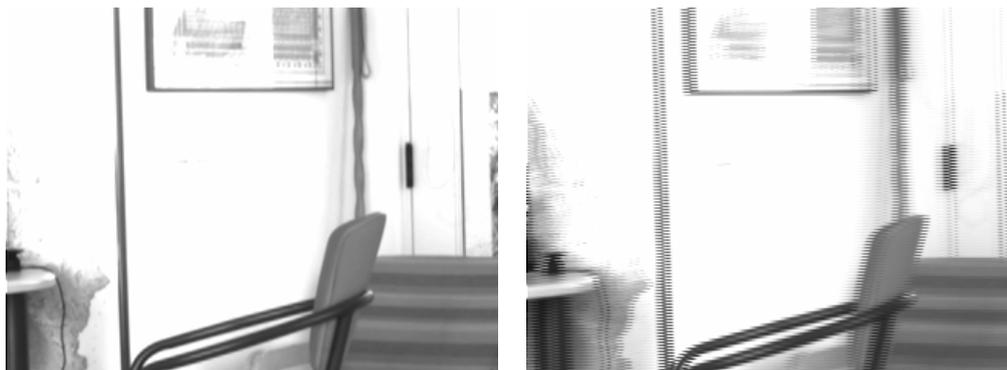


Рис 9.

Слева – кадр, на котором эффект черезстрочной развертки отсутствует, справа – присутствует. Видно, что изображение справа сильно искажено и раздвоено.

Однако на кадре попадают регулярные периодические структуры (вызванные, например, кодированием видеозображения), которые могут «резонировать» с такой сеткой. Для устранения этого эффекта в данной работе функции $i(k)$ и $j(k)$ были выбраны псевдослучайными.

Для вынесения решения о наличии черезстрочной развертки производится сравнение:

$$I_1 / I_2 > T, \text{ где} \quad (5.2)$$

T - некоторая константа (в данном случае была выбрана равной 1.2).

Если неравенство (5.2) верно, значит эффект черезстрочной развертки присутствует.

Для получения качественного выходного изображения можно использовать только один полукадр, либо последовательно на вход алгоритма подавать сначала кадр из нечетных, затем кадр из четных строк. Тем самым будет решена задача стабилизации изображения, но при этом уменьшится формальная разрешающая способность вдоль вертикальной оси. Однако в силу того, что на входном кадре изображение было размазано с амплитудой колебания оптической оси, фактическая разрешающая способность выходного кадра повышается.

5.5 Смазанные кадры

Входные кадры могут быть смазаны за счет сильных колебаний камеры (или испорчены каким-либо другим образом – Рис 10). Но, т.к. эти искажения носят случайный характер и сильно изменяются по степени от кадра к кадру, то для улучшения качества выходного изображения и качества работы алгоритма определения деформации можно пропускать кадры, испорченные сильнее других. Также можно попытаться избавиться от смаза и восстановить изображение, как описано, например в [9]. Однако алгоритмы восстановления смазанных кадров работают, как правило, медленно и неустойчиво.



Рис 10.

Слева – несмазанный кадр, справа – смазанный. На правом кадре видно искаженное раздвоенное изображение

Критерием качества здесь может считаться информативность кадра, вычисленная по формуле (3.8). Если максимальная информативность очередного кадра ниже средней, то такой кадр считается смазанным. Сам же критерий пропуска выбирается из других соображений, например, сколько кадров можно пропустить подряд, насколько сильно может ухудшаться изображение и т.д.

6 Программная реализация алгоритмов

6.1 Блок схема программы улучшения изображений

В Схеме 2 представлена блок схема программы улучшения изображений.

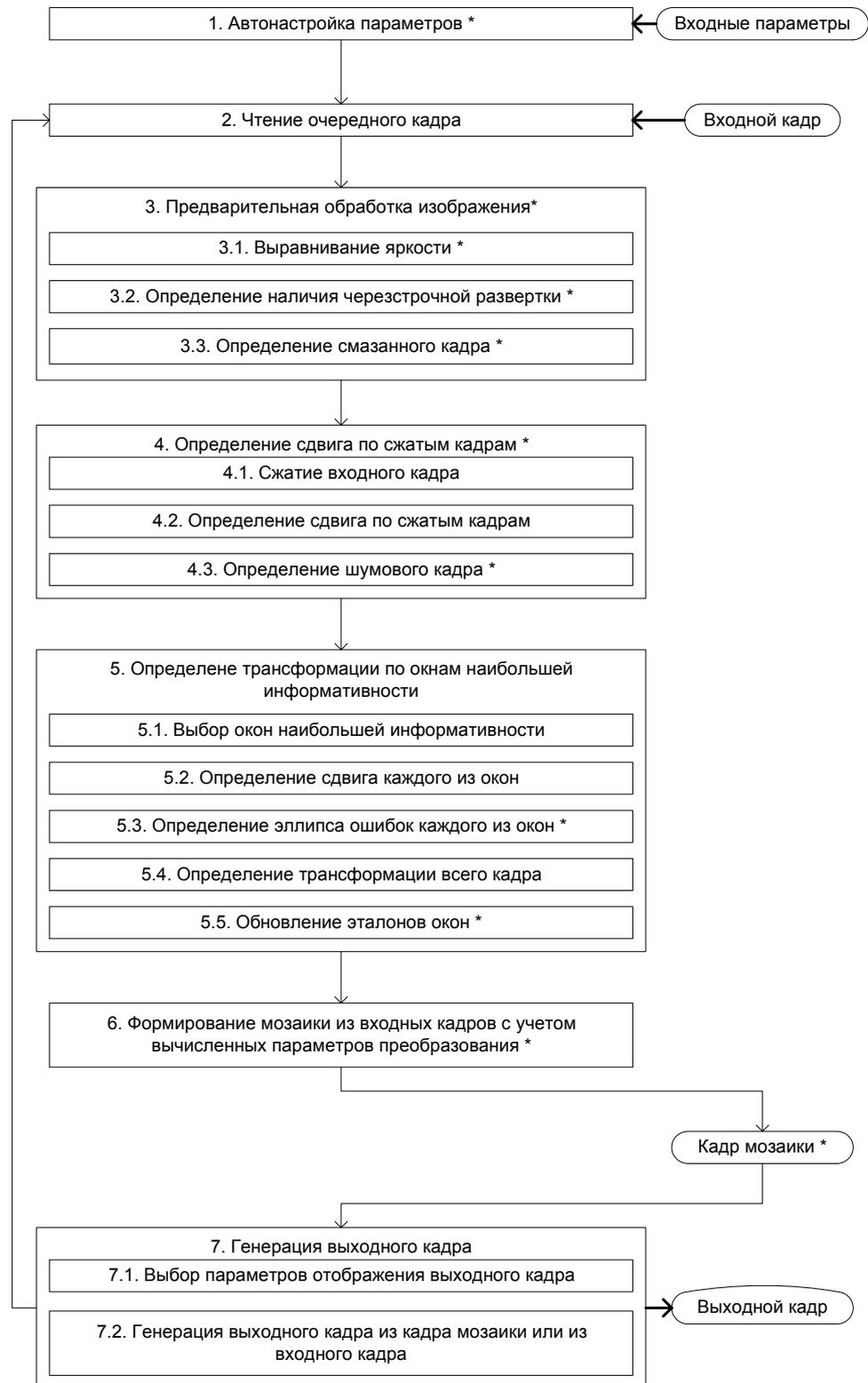


Схема 2.

Блок схема программы улучшения изображений и компенсации геометрических искажений. * - необязательные блоки программы.

Данная программа помимо описанных выше алгоритмов улучшения изображений и компенсации геометрических искажений содержит в себе некоторые дополнительные служебные блоки.

Программа была написана на языке C++ со следующими параметрами:

1. Операционная система Microsoft® Windows™ 95/98/ME/XP 2000/2003.
2. Среда разработки Microsoft® Development Environment 2003 v7.1.
3. Компилятор Intel® C++ Compiler v7.1.
4. Дополнительная библиотека функций Intel® Integrated Performance Primitives 3.0.

6.2 Особенности программной реализации

В программной реализации следует выделить несколько особенностей.

«Ускорение» алгоритмов.

Для увеличения скорости работы, помимо чисто алгоритмических, были предприняты следующие программные ходы:

1. Использование SSE2 инструкций.
2. Использование библиотеки функций Intel® Integrated Performance Primitives 3.0.
3. Использование Intel® Vtune™ Performance Analyzer 7.0 для подробного анализа скорости работы.

Блок автонастройки параметров.

В описанных выше алгоритмах используется большое количество настроечных параметров: размеры окон наибольшей информативности, степень сжатия, пороги и т.д. Оптимальные их значения могут зависеть от различных входных параметров, в том числе: размер изображения, наличие или отсутствие черезстрочной развертки, включения режима определения сдвига по сжатым кадрам и т.д.

Для настройки внутренних параметров алгоритмов, исходя из внешних параметров программы, используется блок автонастройки. Все внутренние параметры алгоритмов и их зависимость от внешних были подобраны экспериментально на основе многочисленных экспериментов с широким спектром входных видеопоследовательностей.

Более того, была создана система автоматического тестирования алгоритмов и выявления оптимальных параметров алгоритмов. Эта система в автоматическом режиме перебирала все комбинации внутренних параметров алгоритмов и на основе тестовых последовательностей оценивала характеристики этих алгоритмов, такие как скорость, точность оценок и др.

Блок создания мозаики.



Рис 11.

Создание мозаики – сверху – два кадра исходной видеопоследовательности, внизу – результирующий кадр мозаики.

При генерации выходного кадра можно без существенных затрат решить еще одну попутную задачу – генерацию мозаики.

При этом все кадры исходной видеопоследовательности объединяются в один кадр мозаики (Рис 11) с учетом измеренных сдвигов. Далее этот кадр подается на модуль генерации выходного кадра и именно из него «вырезается» выходной кадр.

Введение модуля создания мозаики помимо получения кадра мозаики позволяет также улучшить качество выходных изображений. Если входное изображение сильно сдвигается от кадра к кадру, то на выходном кадре будет большое количество точек, на которых не попадает информация с входных кадров – «черных точек». Если же выходной кадр генерируется из мозаики, то такие «черные точки» могут быть заполнены информацией не с текущего кадра, а с предыдущих. При этом их количество резко уменьшается.

Для создания мозаики можно использовать методы, изложенные, например, в [10]. Однако все точные методы создания мозаики требуют знания фокусного расстояния объектива камеры, ее углового разрешения и т.д. В данной работе такие данные не доступны. И тогда самым оптимальным способом создания мозаики является простое

наложение очередного кадра, преобразованного согласно вычисленным параметрам преобразования, на кадр мозаики.

Ограничение времени выполнения.

Данный комплекс алгоритмов предназначен для обработки видеопоследовательности в режиме реального времени.

Но время обработки разных кадров может сильно отличаться друг от друга. И если для определенной последовательности большинство кадров может быть обработано в реальном времени, некоторые из них требуют для своей обработки несколько большее время. Чтобы не выходить за пределы реального времени, можно ограничить время обработки каждого кадра. При этом, если кадр за это время не был обработан полностью, можно в качестве оценки параметров его преобразования взять текущую.

В ходе работы алгоритма практически всегда существует некоторая оценка сдвига. В частности:

- В методе определения сдвига по сжатым кадрам можно прекратить обработку на любом масштабе кадров. А при вычисление функции $X_{(F_1^{P-k} F_2^{P-k})}(dx, dy)$ можно провести не по всей области D_n . При этом увеличивается вероятность попадания в побочный минимум, однако в целом, такая оценка сдвига состоятельна. После этого можно определить дробный сдвиг и привести его к масштабу исходного кадра. Получившаяся оценка сдвига не всегда точна, но в большинстве случаев ошибка не превышает 1-2 пикселей.
- В методе определения параметров трансформации по окнам можно остановить выполнение уже после определения сдвига по одному окну. Далее вычислить общее преобразование по формуле (3.13) с учетом только просчитанных окон.

Таким образом, при прерывании обработки кадра на каком-то внутреннем шаге алгоритма можно получить оценку параметров трансформации, которая в большинстве случаев достаточно точна.

7 Анализ качества и скорости работы алгоритмов

7.1 Методы исследований

Все описанные выше алгоритмы исследовались с целью выяснения качества их работы. При этом считались С.К.О. ошибок определения параметров трансформации, (т.е. сдвигов, углов поворота и коэффициентов масштабирования), вероятность сбоя и скорость работы.

Качество работы алгоритмов оценивалось тремя способами:

1. На основе модельных кадров. Т.е. одно и то же изображение преобразовывалось с заданными параметрами и подавалось на вход алгоритма. При этом можно точно оценить ошибки определения параметров преобразования. В то же время, такое моделирование не отражает многих проблем, с которыми приходится сталкиваться в реальности.
2. На реальных видеопоследовательностях. По ним можно определить сдвиг лишь приближенно и субъективно, и здесь основным критерием должно являться вероятность сбоя, т.е. «очевидной» ошибки, превышающей несколько пикселей.
3. Работа в условиях ограничения времени обработки одного кадра. При этом оценивались средние ошибки определения сдвигов и углов поворота на синтетических последовательностях.

Исследование работы алгоритмов проводилось на ПК со следующими характеристиками:

- Процессор Intel® Pentium® 4 2.4ГГц.
- Память 512Мб.
- Видеокарта NVIDIA GeForce4® MX 440.
- Операционная система Microsoft® Windows™ 2000.

Также был проведен сравнительный анализ различных алгоритмов, как теоретически описанных в литературе, так и коммерческих продуктов.

7.2 Подробный анализ работы алгоритмов

7.2.1 Анализ алгоритмов определения сдвига

В Таблице 1 даны результаты работы различных алгоритмов определения сдвига. Сдвиги моделировались случайные по обеим осям, равномерно распределенные в интервале от -30 до $+30$ пикселей.

В данном случае, под скоростью работы подразумевается только скорость работы алгоритмов определения сдвига, в то время как чтение кадров, генерация выходного кадра и их отображение не учитывается.

Особенно сильно различаются скорости работы различных алгоритмов. Алгоритм сравнения всего кадра работает на 3 порядка медленнее остальных предложенных. Следует заметить, что трудоемкость первого алгоритма растёт пропорционально количеству точек кадра (т.е. квадратично относительно длины стороны). Трудоемкость же второго и третьего растёт существенно медленнее.

Также следует заметить, что максимально возможные сдвиги первых двух алгоритмов составляли в данном случае 60 пикселей. При увеличении этого параметра скорость алгоритмов падает пропорционально его квадрату.

Размер изображения (пиксели)	Сравнение всего кадра (глава 3.2)		Только метод корреляционных окон (главы 3.4 – 3.9)		Полнофункциональный алгоритм		
	С.К.О. определения сдвига (пиксели)	Количество кадров в секунду	С.К.О. определения сдвига (пиксели)	Количество кадров в секунду	С.К.О. определения сдвига (пиксели)	Количество кадров в секунду	Максимальный сдвиг (пиксели)
320x240	0.07	0.9	0.04	206	0.08	714	120
640x480	0.07	0.17	0.03	152	0.07	232	260
800x600	0.07	0.1	0.03	116	0.08	180	330
1024x768	0.07	0.055	0.03	83	0.11	129	400
1152x864	0.07	0.044	0.02	42	0.11	87	480
1280x1024	0.07	0.038	0.04	37	0.08	75	540

Таблица 1.

Результаты работы алгоритмов определения сдвига.

В полнофункциональном алгоритме максимальный сдвиг, который мог быть оценен алгоритмом, был выбран размером в полкадра. Максимальный же сдвиг, который реально может быть оценен, определялся на отдельной тестовой последовательности как сдвиг, при котором точность алгоритма падает вдвое.



Рис 12.

Пример кадров из «реальной» тестовой последовательности. Слева – кадр относительно хорошего качества, справа – типичный кадр для данной последовательности.

Качество же работы приблизительно одинаковое. Более того, метод оценки с использованием только корреляционных окон дает примерно вдвое меньшую ошибку определения сдвига, чем даже метод сравнения всего кадра.

Сравнение всего кадра (глава 3.2)		Только метод корреляционных окон (главы 3.4 – 3.9)		Полнофункциональный алгоритм	
Вероятность сбоя	С.К.О. по сдвигу при сбоях (пиксели)	Вероятность сбоя	С.К.О. по сдвигу при сбоях (пиксели)	Вероятность сбоя	С.К.О. по сдвигу при сбоях (пиксели)
0.3%	7	1.3%	20	0%	0

Таблица 2.

Вероятность сбоя на реальных последовательностях без поворотов.

Также алгоритмы тестировались на реальных последовательностях. (Таблица 2). При этом использовалась последовательность, снятая низкокачественной камерой с большой тряской. Изображения в таких условия получались смазанные, перекошенные (причем и величина, и направление перекосов и смазов менялась от кадра к кадру), сильно зашумленные, с заметными искажениями за счет неидеальной оптики камеры (Рис 12).

Для каждого сбоя считалась его величина путем сопоставления двух последовательных кадров «вручную».

В результате полнофункциональный алгоритм показал идеальные результаты, в то время как остальные давали небольшое количество заметных сбоев.

7.2.2 Анализ алгоритмов определения всех параметров трансформации

В Таблице 3 даны результаты работы различных алгоритмов определения сдвига, поворота и угла масштабирования.

Сдвиги моделировались случайные по обеим осям, равномерно распределенные в интервале от -10 до $+10$ пикселей. Поворот также выбирался случайным, равномерно распределённым в интервале от -2° до $+2^\circ$. Коэффициент масштабирования менялся линейно так, что на 100-м кадре последовательности масштаб был в 4 раза больше, чем на первом.

Размер изображения (пиксели)	Только метод корреляционных окон (главы 3.4 – 3.9)			Полнофункциональный алгоритм			Максимальный угол поворота (градусы)
	С.К.О. определения угла (градусы)	С.К.О. коэффициента масштабирования	Количество кадров в секунду	С.К.О. определения угла (градусы)	С.К.О. коэффициента масштабирования	Количество кадров в секунду	
320x240	0.034	0.2%	104	0.036	0.2%	161	10
640x480	0.036	0.18%	84	0.038	0.18%	123	8
800x600	0.041	0.15%	63	0.040	0.15%	75	8

Таблица 3.

Точности определения угла поворота.

Здесь скорость работы заметно ниже, чем без определения угла поворота, однако по-прежнему остается достаточно высокой даже для кадров 800 на 600 пикселей.

Что касается реальных, а не модельных последовательностей, то здесь имеет место та же ситуация, что и в предыдущей главе (Таблица 4). Алгоритмы тестировались на той же последовательности, что и в главе 7.2.1 (Рис 12).

Сбои в работе были замечены лишь в редких случаях, особенно у полнофункционального алгоритма.

Только метод корреляционных окон (главы 3.4 – 3.9)			Полнофункциональный алгоритм		
Вероятность сбоя	С.К.О. по сдвигу при сбоях (пиксели)	С.К.О. по углу при сбоях (градусы)	Вероятность сбоя	С.К.О. по сдвигу при сбоях (пиксели)	С.К.О. по углу при сбоях (градусы)
3%	30	10	0.6%	15	10

Таблица 4.

Вероятность сбоя на реальных последовательностях с поворотом.

7.2.3 Анализ работы при ограничении времени выполнения

Особое внимание уделялось исследованию функционирования алгоритмов при ограничении скорости работы (см. главу 6.2).

Тестирование проводилось на синтетических последовательностях. Средний сдвиг составлял 20 пикселей.

При этом исследовалась скорость не алгоритма определения сдвига, а программы в целом, включая чтение из файла, собственно алгоритмы стабилизации и генерации выходного кадра, и отображение на экран.

Размер изображения (пиксели)	Количество точек на изображении (мегапиксели)	С.К.О. определения сдвига (пиксели)
800x600	0.48	0.07-0.08
1024x768	0.79	0.07-0.09
1152x864	1.00	0.3-0.4
1280x1024	1.31	0.3-0.4
1600x900	1.44	0.8-1.0
1600x1024	1.64	обработка отключается

Таблица 5.

Точности определения сдвига при ограничении скорости работы.

Результаты исследования приведены в Таблице 5.

Частота кадров была 25 в секунду. Вплоть до размера 1.44 мегапикселя изображение стабилизировалось с приемлемым качеством.

7.3 Сравнительный анализ работы различных алгоритмов

Особый интерес представляет сравнение данного алгоритма с уже существующими.

При этом сравнение велось как с теоретически описанными алгоритмами, так и с коммерческими программными продуктами. Результаты представлены в Таблице 6.

Здесь скорость работы и вероятность сбоя измерялись по алгоритмам, определение угла поворота и коэффициента масштабирования в которых отключено. Под скоростью работы подразумевалась совокупная скорость работы алгоритмов определения параметров трансформации и генерации кадров.

В некоторых случаях не удалось определить точные значения измеряемых величин. Для коммерческих продуктов это трудно сделать, поскольку они не предоставляют необходимой информации. Удалось оценить необходимые параметры лишь приближенно. Также некоторые алгоритмы не определяют углы поворота.

Название алгоритма	С.К.О. определения сдвига (пиксели)	Максимальный сдвиг (пиксели)	С.К.О. определения угла (градусы)	Максимальный угол (градусы)	Вероятность сбоя	Количество кадров в секунду
Описанный выше алгоритм.	0.08	120	0.036	10	0%	233
DynaPel SteadyHands DV 2,2,0,2 [11].	<2	~30	<0.2	10	100%	14
DeShaker 1.6 [12].	0.08	90	0.02	10	0.3%	6
Video Stabilizer 2.6.0.0 [13].	~3	10	∅	∅	100%	40
Спектральный метод компенсации смещений [14].	0.05	7	∅	∅	100%	21
Projection-Based Image Registration in the Presence of Fixed-Pattern Noise [15].	0.06	80	∅	∅	1.3%	97
Полупиксельный пирамидальный алгоритм [5].	0.03	45	0.015	30	2%	44
Вейвлет метод [16].	0.04	30	∅	∅	10%	0.12

Таблица 6.

Сравнение различных алгоритмов компенсации геометрических искажений.

Как видно – предложенные алгоритмы работают существенно быстрее, чем все остальные исследуемые алгоритмы. Кроме того, описанные алгоритмы имеют преимущество по надежности работы и максимальному определяемому сдвигу.

Точности же работы большинства алгоритмов в целом очень близки.

8 Выводы

1. Были созданы и реализованы различные алгоритмы компенсации геометрических линейных искажений кадров видеопоследовательности. При этом использовались только алгоритмы, основанные на вычислении среднеквадратичной невязки кадров или их частей.
2. Были созданы и реализованы различные алгоритмы отделения случайной составляющей геометрических трансформаций от регулярной, связанной с управляемым движением камеры.
3. Были созданы и реализованы алгоритмы оценки и компенсации таких параметров входных изображений, как наличие черезстрочной развертки, шумоподобность, смазанность и т.д.

4. Был создан и реализован алгоритм создания мозаики – т.е. кадра большого размера, полученного путем объединения нескольких кадров видеопоследовательности в один с учетом оцененных сдвигов.
5. Была проведена оптимизация разработанных алгоритмов. В результате были достигнуты следующие показатели:
 - a. Скорость работы данных алгоритмов - до 230 кадров в секунду при размере кадра 320x240.
 - b. Скорость работы программы полной обработки изображений, включающей в себя, помимо алгоритмов определения параметров деформации, также считывание исходных данных и отображение на экран - до 25 кадров размером в 1.44 мегапикселя в секунду.

Однако, при использовании специализированных сигнальных процессоров или многопроцессорных систем возможно существенное увеличение производительности за счет хорошей распараллеливаемости данных алгоритмов.

6. Была создана система автоматического подбора внутренних параметров алгоритмов. Она позволяет в автоматическом режиме выбрать оптимальные параметры алгоритмов для достижения наилучших показателей по точностям оценок параметров геометрических трансформаций и скорости работы.
7. Было проведено исследование качества работы алгоритмов и получены следующие результаты:
 - a. На модельных последовательностях разработанные алгоритмы дают ошибку измерения сдвигов кадров не более 0.08 пикселей, при максимальном значении сдвига порядка полукадра, а ошибку определения углов поворота – менее 0.04 при максимальном угле поворота в 8-10°. Точность определения коэффициента масштабирования составляет 0.2%.
 - b. На реальных последовательностях, даже снятых в сложных условиях и при искажениях, не описываемых используемой математической моделью, разработанные алгоритмы работают устойчиво. При этом сбои стабилизации происходят с вероятностью не большей 0.3%, даже для низкокачественных последовательностей.
8. Был проведен сравнительный анализ различных алгоритмов компенсации геометрических трансформаций. При этом исследовались как теоретически описанные алгоритмы, так и коммерческие продукты. В целом,

разработанный алгоритм показал по крайней мере не худшие результаты, чем остальные исследованные, а по некоторым показателям и превзошел их.

В целом, разработка и тестирование комплекса алгоритмов показали принципиальную возможность реализации обработки больших потоков видеoinформации в реальном времени на современных, неспециализированных вычислительных средствах.

9 Литература

1. B. K. P. Horn and E. J. Weldon, Jr. Direct methods for recovering motion. *International Journal of Computer Vision*, 2:51–76, 1988.
2. R. Chipolla, Y. Okamoto, and Y. Kuno, “Robust Structure from Motion Using Motion Parallax,” *Int’l Conf. Computer Vision*, pp. 374–382, Berlin, May 1993.
3. F. Lustman, O.D. Faugeras, and G. Toscani, “Motion and Structure from Motion from Point and Line Matching,” *Proc. First Int’l Conf. Computer Vision*, pp. 25–34, London, 1987.
4. S. Negahdaripour and S. Lee, “Motion Recovery from Image Sequences Using First-Order Optical Flow Information,” *IEEE Workshop Visual Motion*, pp. 132–139, Princeton, N.J., Oct. 1991.
5. А.К. Ким, А.Е. Колесса, В.Н. Лагуткин, А.В. Лотоцкий, В.Г. Репин. Алгоритмы идентификации и подавления нестационарного мешающего фона и повышение разрешающей способности в оптическом сенсоре с хаотически колеблющейся оптической осью и динамическими аберрациями. *Радиотехника*, 1998, №12
6. Брамер К., Зиффлинг Г. Фильтр Калмана-Бьюси. – пер. с англ. «Наука», 1982.
7. Дж. Дэннис, Р. Шнабель. Численные методы безусловной оптимизации и решения нелинейных уравнений. Пер. с англ. Москва «Мир», 1988
8. Й. Абаффи, Э. Спендикато. Математические методы для линейных и нелинейных уравнений. Проекционные ABS-алгоритмы. Мир, 1996.
9. Alex Rav-Acha, Shmuel Peleg. Restoration of Multiple Images with Motion Blur in Different Directions, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, proceedings of the conference, 2000.
10. Richard Szeliski, Heung-Yeung Shum. Creating Full View Panoramic Image Mosaics and Environment Maps. Microsoft Research, 1997.
11. <http://steadyhand.dynapel.com>
12. <http://home.bip.net/gunnart/video/deshaker.htm>
13. <http://members.aol.com/HuberMM/index.html>

14. В.Г. Репин, В.Н. Лагуткин. Спектральный метод компенсации смещений и повышения разрешения изображений при обработке последовательности смещающихся цифровых кадров. Радиотехника и электроника, том 45, №11, с 1359-1364.
15. Stephen C. Cain, Majeed M. Hayat and Ernest E. Armstrong. Projection-Based Image Registration in the Presence of Fixed-Pattern Noise ,IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 10, NO. 12, DECEMBER 2001.
16. В.Н. Лагуткин, Ю.В. Радченко. Применение вейвлет-преобразования в задаче оценки смещения объекта. Вопросы радиоэлектроники, сер. РЛТ, 2004, вып 1.