

Санкт-Петербургский государственный университет

ХАРЧЕНКО Данил Сергеевич

Выпускная квалификационная работа

*Построение поля скоростей для последовательностей медицинских изображений
на основе оптимизационного алгоритма*

Уровень образования: магистратура

Направление 01.04.02 «Прикладные математика и информатика»

Основная образовательная программа ВМ.5691 «Прикладная математика и информатика в задачах медицинской диагностики»

Научный руководитель:
доктор физико-
математических наук,
профессор
Котина Елена Дмитриевна

Рецензент:
кандидат физико-
математических наук,
доцент
кафедра медицинской
биофизики СПбГПМУ
Елизарова Марина
Владиславовна

Санкт-Петербург
2023

Содержание

Введение	3
Обзор литературы	4
Постановка задачи	6
Глава 1. Определение поля скоростей	8
1.1. Вариация и градиент функционала	8
1.2. Алгоритм определения поля скоростей	8
Глава 2. Разбиение изображения на подобласти	11
2.1. Алгоритм разбиения изображения	11
2.2. Программная реализация алгоритма разбиения	12
Глава 3. Программная реализация построения поля скоростей 14	
3.1. Случай сдвига	14
3.2. Случай поворота	18
3.3. Случай сдвига и поворота	22
3.4. Последовательность изображений	29
3.5. Анализ результатов для разных моментов времени T	31
Выводы	37
Заключение	38
Список литературы	39

Введение

Существует множество различных способов цифровой обработки изображений, соответствующих разным целям: улучшение качества изображения, распознавание объектов и анализ их движения, восстановление объектов и т.д. Не смотря на это, задача разработки новых алгоритмов всё ещё остается актуальной, так как обработка изображений является востребованной во многих областях, в частности в диагностической медицине – обработка изображений позитронно-эмиссионной томографии, однофотонной эмиссионной компьютерной томографии, гибридных установок и т.п.

Одним из методов цифровой обработки изображений является построение поля скоростей. Такой метод обработки изображений позволяет определить наличие смещения областей интереса между соседними последовательными кадрами, а в случае его наличия определить направление движения. В частности, такой метод применяется при обработке результатов радионуклидных методов исследования.

В данной работе развивается подход, предложенный в работах Котиной Е.Д., Овсянникова Д.А., который позволяет рассматривать не только оптические, но и неоптические потоки [1–3]. Разрабатывается оптимизационный алгоритм и программное обеспечение построения поля скоростей для последовательностей изображений.

Обзор литературы

Задача построения поля скоростей (оптического потока) исследовалась многими авторами. Так, например, алгоритм построения поля скоростей рассматривается в работе Лукаса и Канаде [4]. Данный алгоритм основывается на предположении о постоянстве яркости. Также предполагается, что соседние пиксели смещаются одинаково, вследствие чего поле скоростей ищется для некоторой окрестности. Отыскание происходит путем минимизации целевого функционала методом наименьших квадратов. Данный метод хорошо справляется с поставленной задачей при малых смещениях, но в результате получается неплотный оптический поток.

Самая известная постановка задачи отыскания поля скоростей, в которой используется концепция оптического потока, представлена в работе Хорна и Шанка [5]. В данной работе также используется предположение о постоянстве яркости вдоль траекторий рассматриваемой системы. В качестве дополнительного условия рассматривается предположение о достаточной гладкости оптического потока на всем изображении. Отыскание поля скоростей происходит путем решения уравнений Эйлера-Лагранжа, которые сводятся к разностным уравнениям. В отличие от метода Лукаса-Канаде, данный метод позволяет строить плотный оптический поток [6], но является более чувствительным к шумам, а также требует постановки граничных условий.

Метод, описанный в работе Фарнбека [7], основывается на аппроксимации окрестности каждого пикселя квадратичной формой, описывающей сигнал каждой окрестности. Используется предположение, что в рассматриваемой окрестности все точки смещаются одинаково. Отыскание параметров квадратичной формы, описывающей поле скоростей, происходит с использованием взвешенного метода наименьших квадратов, где весовая функция обычно является гауссовой.

В работе [8] представлен метод построения поля скоростей с использованием фазовой корреляции. В представленном методе изображения разбивается на подобласти. Между соответствующими подобластями рассчитывается функция фазовой корреляции и для каждого пикселя первого

изображения выбирается набор кандидатов, состоящий из пикселей второго изображения, для которых корреляция максимальная. Затем из вектора кандидатов для каждого пикселя первого изображения происходит отыскание соответствующего пикселя на втором изображении с использованием оптимизационных алгоритмов, после чего строится искомое поле скоростей.

Задача построения поля скоростей может быть решена при помощи подходов, основанных на использовании нейронных сетей. В ряде работ [9–11] нейронные сети используются для извлечения признаков, которые затем добавляются в целевой функционал метода [5] как дополнительные члены. Другой подход к использованию нейронных сетей позволяет непосредственно получить поле скоростей, как, например, в работах [12, 13], но к недостаткам такого подхода относится малый объем данных для обучения и относительно низкая скорость работы алгоритмов.

В ряде работ [14, 16–21] рассматривается оптимизационный подход к решению задачи построения поля скоростей. Подходы основываются как на непрерывных системах [14, 16], так и на дискретных [17–20].

В данной работе развивается подход к решению задачи отыскания поля скоростей, описанный в работе [14], с введением разбиения исходных изображений на подобласти. Разбиение изображений производится с использованием алгоритма, основывающегося на морфологической обработке изображений.

Постановка задачи

Рассмотрим систему дифференциальных уравнений

$$\dot{x} = f(t, x, u), \quad (1)$$

где t – время, x – пространственный вектор координат, $x \in R^n$, u – вектор параметров, $u \in R^r$.

Введем в рассмотрение функцию $\rho = \rho(t, x)$ – плотность распределения, которая играет роль плотности массы или заряда в различных задачах механики и электродинамики. В нашем случае это количественная характеристика изображения (яркость), зависящая от вектора координат и времени, или интенсивность распределения радиофармпрепарата в случае обработки результатов радионуклидных исследований. Будем полагать, что яркость является постоянной вдоль траекторий системы (1), то есть

$$\left. \frac{d\rho}{dt} \right|_{(1)} = 0.$$

Мы предполагаем, что функция $\rho = \rho(t, x)$ задана, и нам нужно восстановить функцию $f(t, x, u)$, которая определяет искомое поле скоростей. Здесь вектор параметров $u = (u_1, u_2, \dots, u_r)$ считается неизвестным.

Будем рассматривать задачу восстановления вектора параметров u как задачу оптимизации.

Пусть $M_0 \in R^n$ – множество начальных значений для системы (1). Мы предполагаем, что множество M_0 является замкнутым и имеет ненулевую меру Лебега. Решения системы (1) обозначим через

$$x(t) = x(t, x_0, u), x_0 \in M_0. \quad (2)$$

Данный набор решений образует пучок траекторий, исходящих из множества начальных значений M_0 для фиксированного вектора параметров u .

Обозначим через

$$M_{t,u} = \{x(t) = x(t, x_0, u), x_0 \in M_0\} \quad (3)$$

сечение пучка траекторий в момент времени t для заданного вектора u .

Будем полагать, что плотность (яркость) $\rho_0(x)$, определяющая изображение в момент времени $t = 0$, известна. Также предполагаем, что плот-

ность изменившегося за время Δt изображения тоже известна. Обозначим её за $\hat{\rho}(x)$. Зафиксируем время $T = \Delta t$.

Сформулируем задачу оптимизации. Для этого введем в рассмотрение функционал

$$J(u) = \int_{M_{t,u}} g(x_T, \rho(T, x_T)) dx_T, \quad (4)$$

здесь $M_{T,u}$ - сечение пучка траекторий в момент T , $g(x, \rho)$ - неотрицательная непрерывно дифференцируемая по x и ρ функция. Например, в качестве функции $g(x, \rho)$ можно рассматривать функцию в виде, предложенном в статье [14]:

$$g(x, \rho(T, x)) = (\rho(T, x(T)) - \hat{\rho}(x(T)))^2, \quad (5)$$

где $\hat{\rho}(x)$ - заданная плотность в R^n . Отметим, что момент T здесь зафиксирован, но он также может варьироваться.

В данной работе будут рассмотрены следующие задачи:

- Построение алгоритма определения поля скоростей как решение задачи поиска минимума функционала (4);
- Программная реализация предложенного алгоритма;
- Разработка и реализация алгоритма разбиения изображения на под-области;
- Анализ полученных результатов.

Глава 1. Определение поля скоростей

1.1. Вариация и градиент функционала

Запишем формулу вариации функционала (4) в виде, полученном в работах Овсянникова [14, 22]:

$$\delta J = - \int_0^T \int_{M_{t,u}} [\psi^*(t, x_t) \Delta_u f(t, x_t, u)] dx_t dt, \quad (6)$$

где

$$\Delta_u f(t, x_t, u) = f(t, x_t, u + \Delta u) - f(t, x_t, u),$$

вектор-функция $\psi(t, x)$ является вспомогательной функцией, которая удовлетворяет вдоль траектории системы (1) следующему уравнению:

$$\frac{d\psi}{dt} = - \left(\frac{\partial f(t, x(t), u)}{\partial x} \right)^*, \quad (7)$$

с условиями в конечный момент времени

$$\psi^*(T, x(T)) = - \frac{\partial g(x(T), \rho(T, x(T)))}{\partial x}. \quad (8)$$

Предположим, что функция f дифференцируема по u . Тогда с учетом выпуклости множества U и с использованием выражения (6) формула для градиента функционала будет записана в виде (4) [14, 22]

$$\frac{\partial J}{\partial u} = - \int_0^T \int_{M_{t,u}} \psi^* \frac{\partial f}{\partial u} dx_t dt. \quad (9)$$

Полученное выражение (9) для вычисления градиента функционала будем использовать при решении оптимизационной задачи градиентным методом.

1.2. Алгоритм определения поля скоростей

Яркость $\rho(t, x)$ в силу системы (1) удовлетворяет уравнению [14]

$$\frac{d\rho}{dt}|_{(1)} = -\rho \operatorname{div}_x f(t, x(t, x_0, u), u), \quad (10)$$

но в силу сделанного предположения о постоянстве яркости вдоль траектории, получаем, что $\operatorname{div}_x f = 0$. Таким образом, мы имеем дело со случаем

оптического потока.

Решение уравнения (10) с начальным условием

$$\rho(0, x_0) = \rho_0(x_0) \quad (11)$$

и учетом условия на постоянство яркости вдоль траектории может быть записано в следующем виде:

$$\rho(t, x(t, x_0, u)) = \rho_0(x_0) \quad (12)$$

В общем случае вид функции $f(t, x, u)$, являющейся правой частью системы (1), неизвестен. Будем рассматривать данную функцию как некоторую функцию, представленную отрезком ряда Тейлора. Тогда компоненты искомого вектора параметров u будут являться коэффициентами этого ряда. Например, на первом этапе построения поля скоростей будем рассматривать функцию $f(t, x, u)$ в виде линейной вектор-функции:

$$\dot{x} = Ax + C, \quad (13)$$

где $A = \{a_{ij}\}_{i,j}^n$ – матрица, $C = \{c_i\}_{i=1}^n$ – вектор. Тогда вектор параметров u будет состоять из элементов матрицы A и компонент вектора C , т.е.

$$u = (a_{11}, a_{12}, \dots, a_{nn}, c_1, \dots, c_n)^T.$$

Верхний индекс T обозначает транспонирование вектора. Как и раньше, мы предполагаем, что $u \in U$, где U – компакт.

Рассмотрим частный случай, когда $n = 2$ – случай двумерного изображения. Тогда система (13) будет иметь следующий вид:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & -a_{11} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}. \quad (14)$$

Матрица A имеет такие компоненты в силу равенства нулю дивергенции функции f .

В этом случае формулы для вычисления компонент градиента функционала будут иметь вид:

$$\frac{\partial J}{\partial a_{11}} = - \int_0^T \int_{M_{t,u}} (\psi_1 x_1 - \psi_2 x_2) dx_t dt, \quad (15)$$

$$\frac{\partial J}{\partial a_{ij}} = - \int_0^T \int_{M_{t,u}} (\psi_i x_j) dx_t dt, i, j = 1, 2, \quad (16)$$

$$\frac{\partial J}{\partial c_i} = - \int_0^T \int_{M_{t,u}} \psi_i dx_t dt, i = 1, 2. \quad (17)$$

Уравнение (7) переписется в виде:

$$\frac{d\psi}{dt} = -A\psi \quad (18)$$

с условием в конечный момент времени T :

$$\psi(T, x(T))^* = 2(\rho(T, x(T)) - \hat{\rho}(x(T))) \frac{\partial \hat{\rho}(x(T))}{\partial x}. \quad (19)$$

Рассмотрим итерационный алгоритм определения поля скоростей для некоторой окрестности точки изображения. Мы предполагаем, что M_0 , $\rho_0(x)$, $\hat{\rho}(x)$ заданы, а T фиксировано.

1. Зафиксируем начальное приближение вектора параметров u^0 . Положим номер итерации $k = 0$. Зафиксируем неотрицательные константы S , G , N , используемые в условии останова. Константы S и G являются достаточно малыми.
2. Найдем закон смещения пикселей в силу системы (1) и найдем пересчитанное изображение при данном векторе параметров u^k .
3. Найдем значение вспомогательной функции ψ из (18) при условии (19).
4. Вычислим градиент функционала $\frac{\partial J}{\partial u} |_{u=u^k}$ по формулам (15) – (17).
5. Если i -я компонента градиента $\frac{\partial J}{\partial u_i} |_{u=u^k}$ по модулю больше константы S , то найдем u_i^{k+1} градиентным методом по формуле:

$$u_i^{k+1} = u_i^k + \alpha \frac{\partial J}{\partial u_i} |_{u=u^k},$$

где u_i – i -я компонента вектора u , параметр градиентного спуска α выбирается как такой $\bar{\alpha}$, который минимизирует функцию $Z(\alpha)$, то есть

$$\min_{\alpha} Z(\alpha) = J(u^k - \alpha \text{grad}(J(u^k))) = Z(\bar{\alpha}).$$

6. Проверяем условия остановки.
7. Если условия остановки не выполнены, то полагаем $k=k+1$ и переходим к шагу 2.

В качестве условий остановки алгоритма используются следующие условия – каждая компонента градиента функционала по модулю меньше константы S , значение функционала J меньше константы G , номер итерации k меньше заданного числа итераций N . Остановка алгоритма происходит, если выполнено хотя бы одно из условий. Использование нескольких условий остановки позволяет своевременно останавливать работу алгоритма и экономить вычислительные ресурсы.

Проверка на удовлетворение неравенству компонент градиента функционала в пункте 5 также позволяет сокращать время работы алгоритма и экономить вычислительные мощности.

Глава 2. Разбиение изображения на подобласти

2.1. Алгоритм разбиения изображения

На изображениях могут присутствовать сложные движения разных объектов, в силу чего может быть затруднительно описать поле скоростей одной системой для всего изображения. Поэтому был разработан и реализован алгоритм разбиения изображения на подобласти на основе морфологического подхода [15]. В основе алгоритма лежат операции дилатации и замыкания. Такой подход позволяет описать каждую подобласть своей системой уравнений, в частности системой линейного приближения. Использование данного алгоритма позволяет повысить точность и скорость вычислений.

Алгоритм разбиения изображений на подобласти можно представить следующим образом:

1. перевод изображений из RGB в Grayscale;
2. применение медианного фильтра;

3. расчет абсолютной разности между изображениями;
4. применение операций замыкания и дилатации;
5. перевод изображений из Grayscale в бинарные;
6. применение функции для поиска контуров;
7. поиск координат квадратов, описывающих каждый из объектов;
8. выделение областей движений.

После того, как произошло выделение областей движений, к каждой из областей может быть применен алгоритм построения поля скоростей.

2.2. Программная реализация алгоритма разбиения

Как было сказано выше, будем рассматривать разбиение изображения на подобласти в случае, когда на изображении присутствует несколько движущихся объектов. Для каждой такой окрестности будем рассматривать линейное приближение в виде (14).

В численном эксперименте, представленном на рисунке 1, были рассмотрены последовательные изображения, на которых присутствуют несколько движущихся объектов.

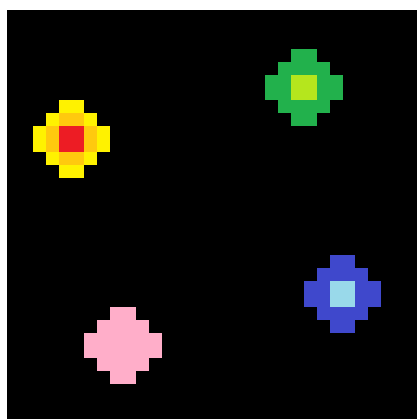


Рисунок 1.1 – Первый кадр

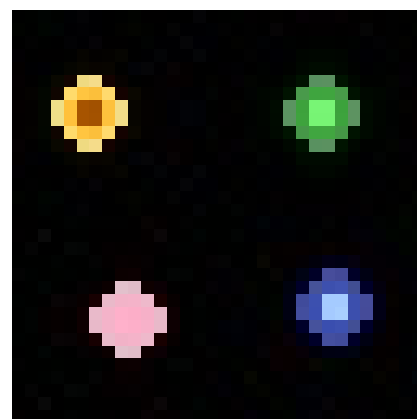


Рисунок 1.2 – Второй кадр

Сначала к исходным изображениям был применен предложенный алгоритм разбиения изображений на подобласти на основе морфологического

подхода. После чего к каждой из полученных подобластей был применен алгоритм построения поля скоростей. Разбиение изображений на подобласти представлено на рисунке 2.

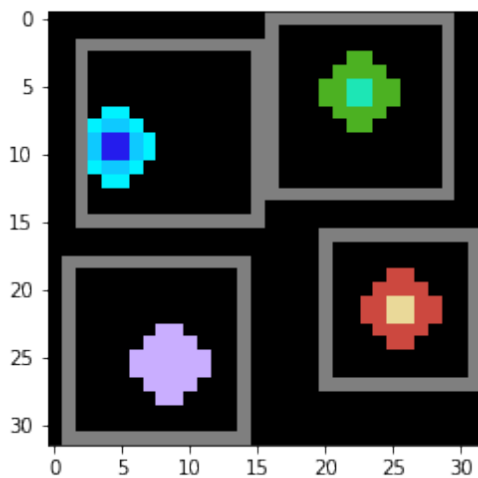


Рисунок 2.1 – Первый кадр с разбиением

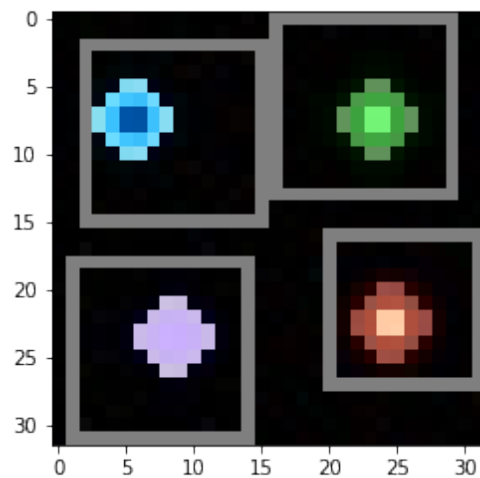


Рисунок 2.2 – Второй кадр с разбиением

Поле скоростей, найденное с использованием модернизированного алгоритма, приведено на рисунке 3.

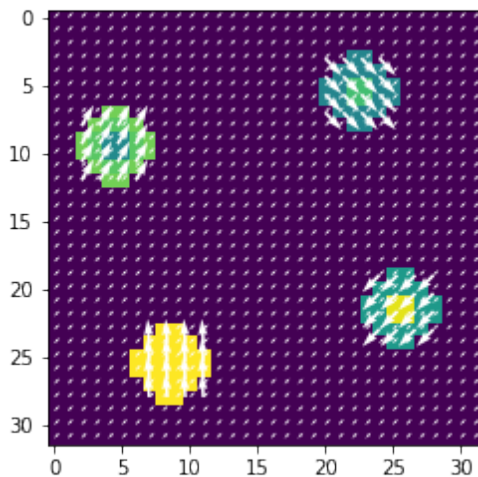


Рисунок 3.1 – Первый кадр

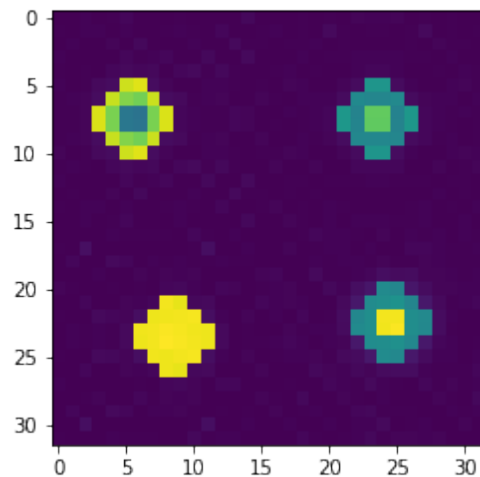


Рисунок 3.2 – Второй кадр

Не смотря на то, что все представленные на кадрах объекты движутся в разных направлениях, алгоритм правильно определил направление движения для каждого из объектов благодаря разбиению кадров на отдельные подобласти.

Глава 3. Программная реализация построения поля скоростей

Приведенный выше алгоритм был реализован на языке программирования Python. Для реализации были использованы следующие библиотеки:

1. *NumPy* - для матричных вычислений;
2. *SciPy* - для интегрирования систем (14), (18) с использованием функции *odeint*, в основе которой лежит метод Рунге-Кутты 4-5 порядка;
3. *OpenCV* - для преобразования и обработки изображений, в частности для реализации алгоритма разбиения на подобласти;
4. *Matplotlib* - для визуализации результатов, в частности метод *quiver*, позволяющий производить визуализацию найденного поля скоростей.

3.1. Случай сдвига

В ходе численных расчетов были рассмотрены последовательные изображения, на которых присутствует смещение области интереса. В случае, когда на изображениях присутствует только сдвиг, будем полагать матрицу A постоянной, а именно нулевой, и будем производить отыскание компонент вектора C .

В одном из численных экспериментов были использованы два последовательных снимка желчного пузыря. Рассмотренные изображения приведены ниже на рисунке 4.

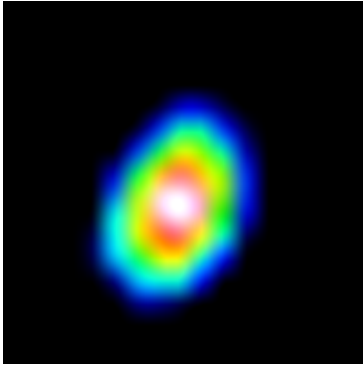


Рисунок 4.1 – Желчный пузырь на первом кадре

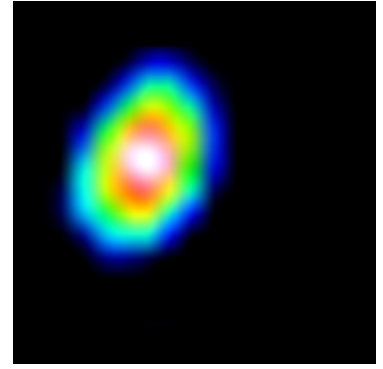


Рисунок 4.2 – Желчный пузырь на втором кадре

Для вектора C было рассмотрено нулевое начальное приближение. Для данной пары изображений было проделано 30 итераций работы алгоритма. Результаты численных расчетов для данной последовательности изображений представлены на рисунке 5.

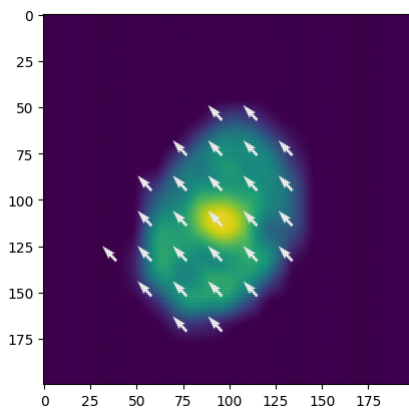


Рисунок 5.1 – Желчный пузырь на первом кадре

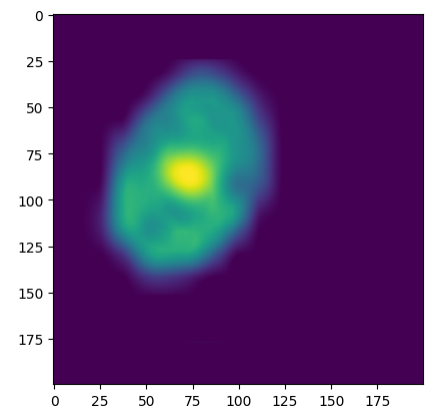


Рисунок 5.2 – Желчный пузырь на втором кадре

На рисунке 5.1 видно, что алгоритм правильно определил направление поля скоростей. Также об этом может свидетельствовать разность пересчитанного изображения в силу найденного вектора C и второго кадра, представленная на рисунке 5.3 ниже.

На рисунке 5.3 видно, что помимо направления поля скоростей, алгоритм правильно определил величину сдвига.

Также были получены графики значений компонент вектора C и компонент вектора градиента $\frac{\partial J}{\partial u_i}$ к номеру итерации. В случае сдвига вектор

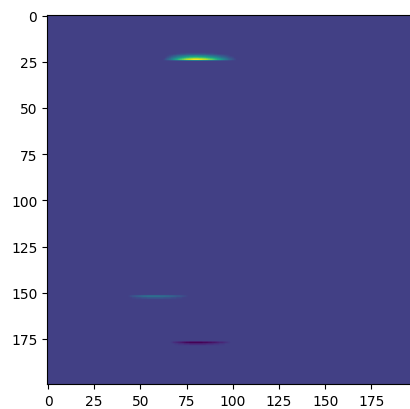


Рисунок 5.3 – Разность
пересчитанного изображения
и второго кадра

градиента имеет вид

$$\frac{\partial J}{\partial u} = \left(\frac{\partial J}{\partial c_1}, \frac{\partial J}{\partial c_2} \right)$$

в силу того, что матрица A фиксирована.

Графики предствалены на рисунке 6.

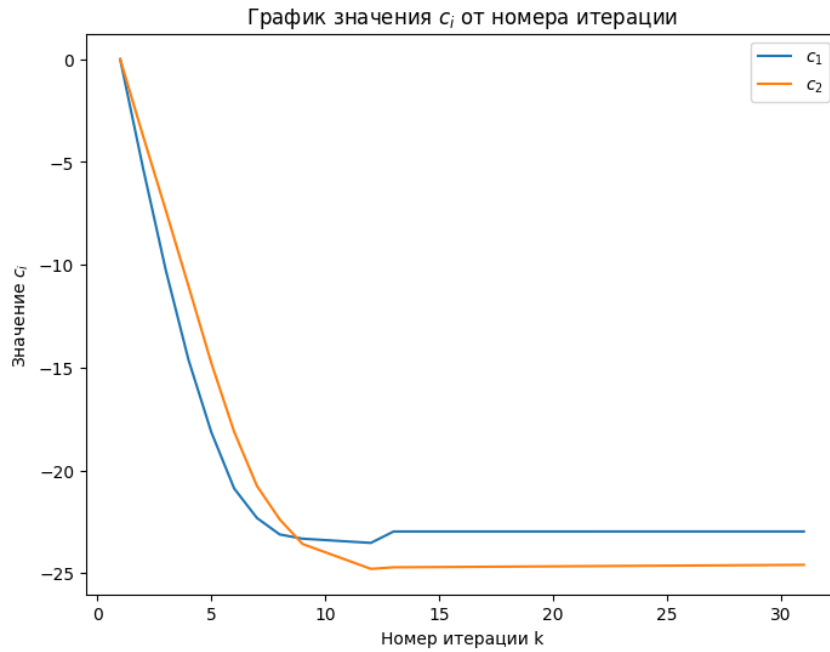


Рисунок 6.1 – График изменения значения c_i от номера итерации

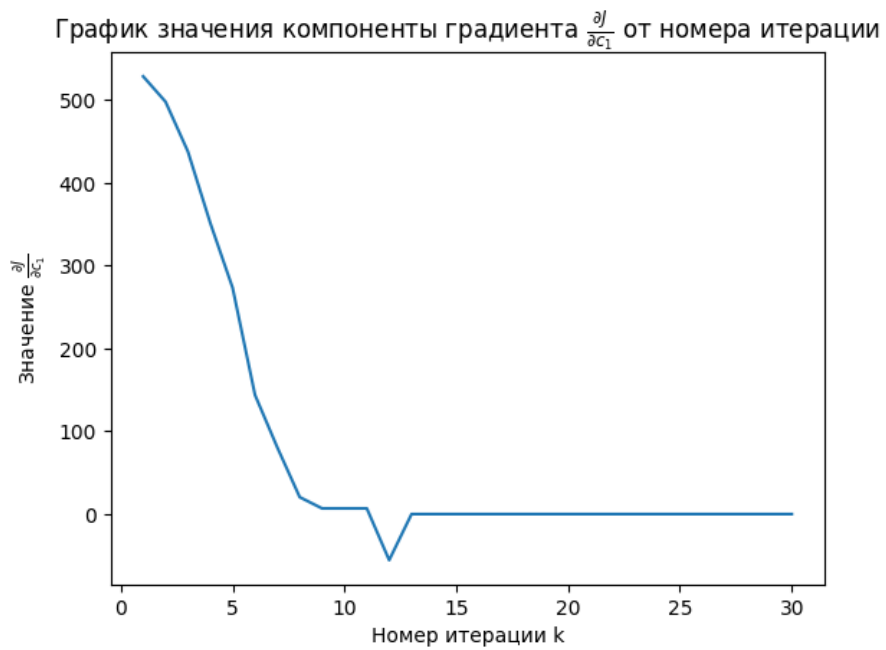


Рисунок 6.2 – График изменения значения $\frac{\partial J}{\partial c_1}$ от номера итерации

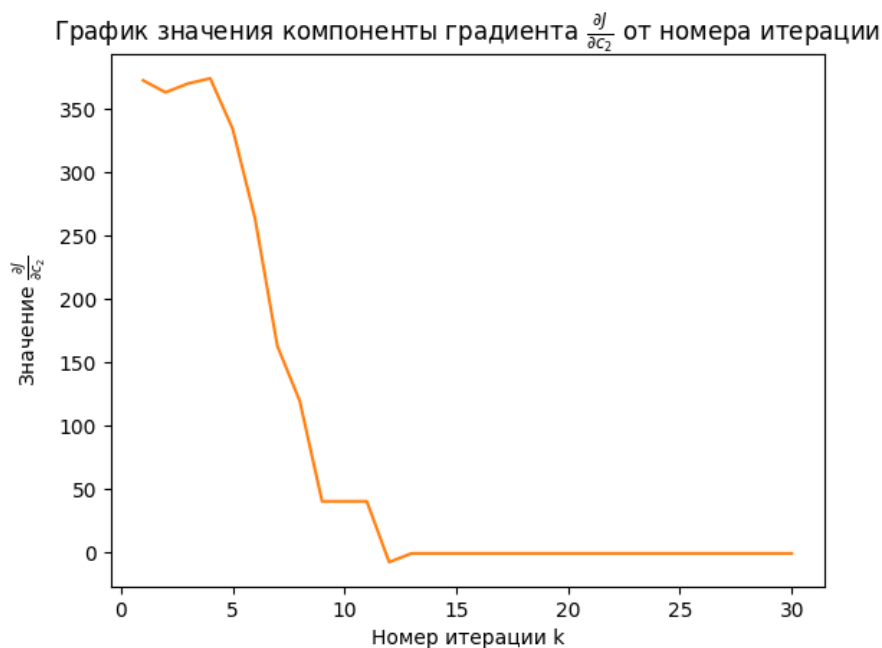


Рисунок 6.3 – График изменения значения $\frac{\partial J}{\partial c_2}$ от номера итерации

На представленных графиках видно, что для алгоритма была достигнута сходимость. При этом следует отметить, что в случае, когда на кадрах присутствует только сдвиг, сходимость алгоритма наступает достаточно быстро. Так, например, для данной пары изображений сходимость была достигнута уже на 13 итерации.

3.2. Случай поворота

Помимо сдвига, на изображениях может присутствовать поворот области интереса. В этом случае мы будем иметь дело с ненулевой матрицей A .

В ходе численного эксперимента были рассмотрены последовательные изображения, на которых присутствует поворот круга с вырезанным сектором. В силу того, что сдвиг области интереса отсутствует, зафиксируем вектор C и положим его компоненты равными нулю. Результат работы алгоритма для случая поворота представлен на рисунке 7.

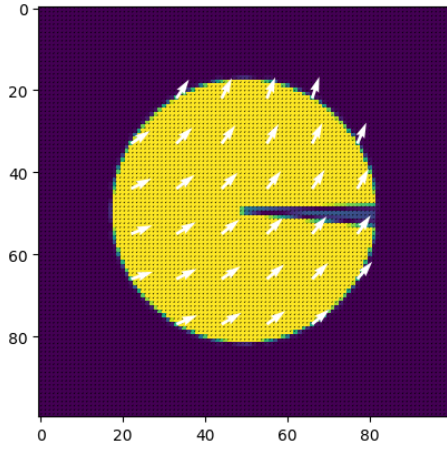


Рисунок 7.1 – Первый кадр

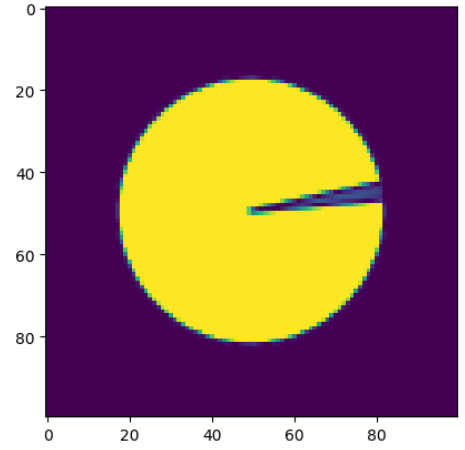


Рисунок 7.2 – Второй кадр

Видно, что для случая поворота алгоритм так же верно определил направление поля скоростей. На рисунке 7.3, как и для случая сдвига, приведено изображение разности пересчитанного в силу найденной матрицы A изображения и второго кадра.

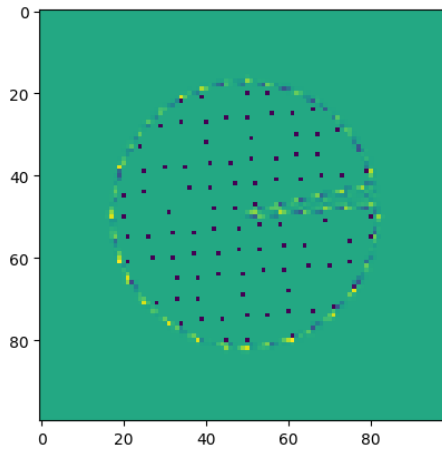


Рисунок 7.3 – Разность пересчитанного изображения и второго кадра

Таким образом, можно сделать вывод, что алгоритм правильно определил поле скоростей и угол, на который была повернута область интереса.

На рисунке 8 приведены графики компонент матрицы A и компонент градиента к номеру итерации. Для компоненты a_{22} графики не приведены в силу в того, что, как было установлено выше, $a_{22} = -a_{11}$.

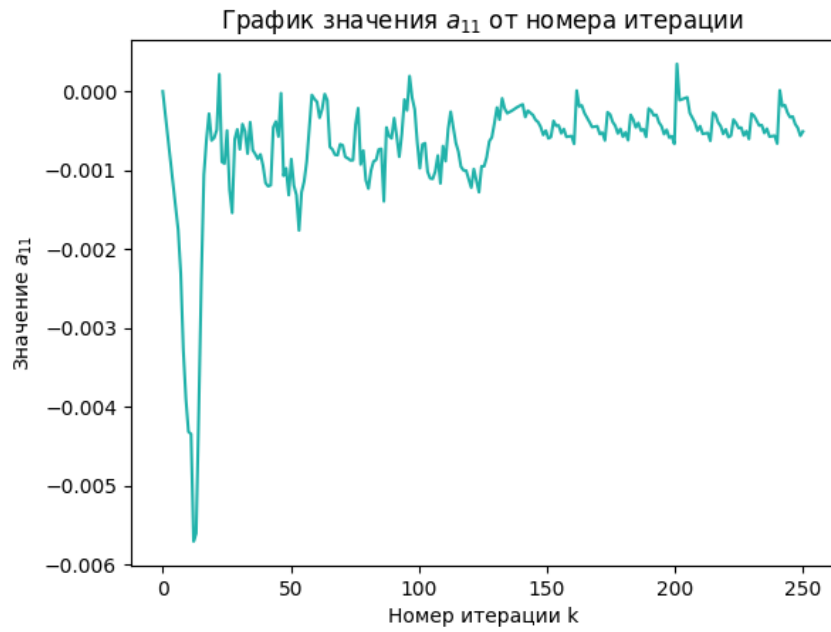


Рисунок 8.1 – График изменения значения a_{11} от номера итерации

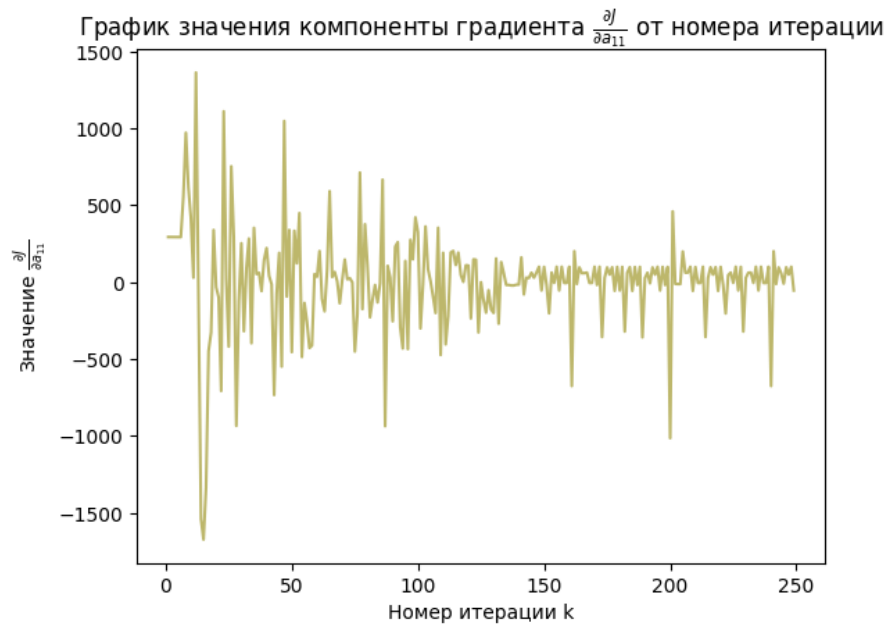


Рисунок 8.2 – График изменения значения $\frac{\partial J}{\partial a_{11}}$ от номера итерации

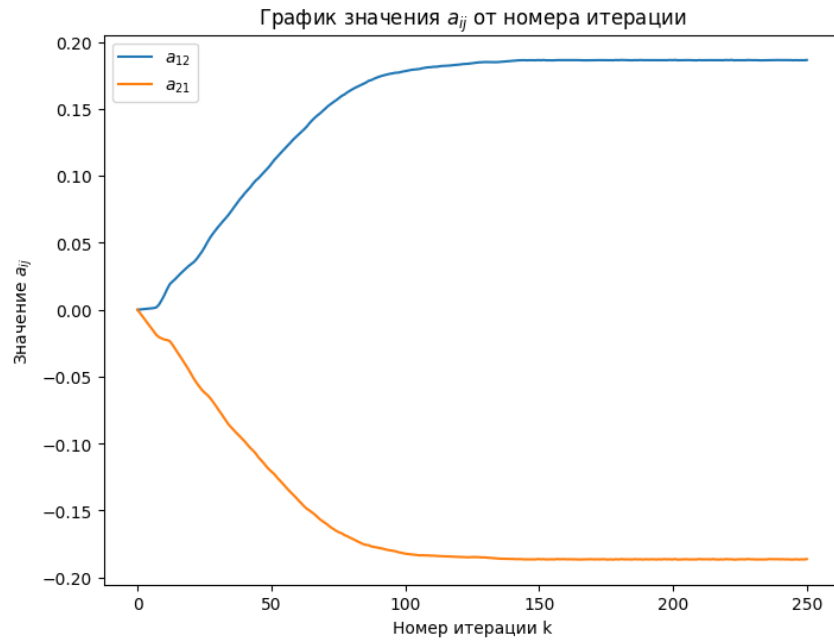


Рисунок 8.3 – График изменения значения a_{ij} от номера итерации

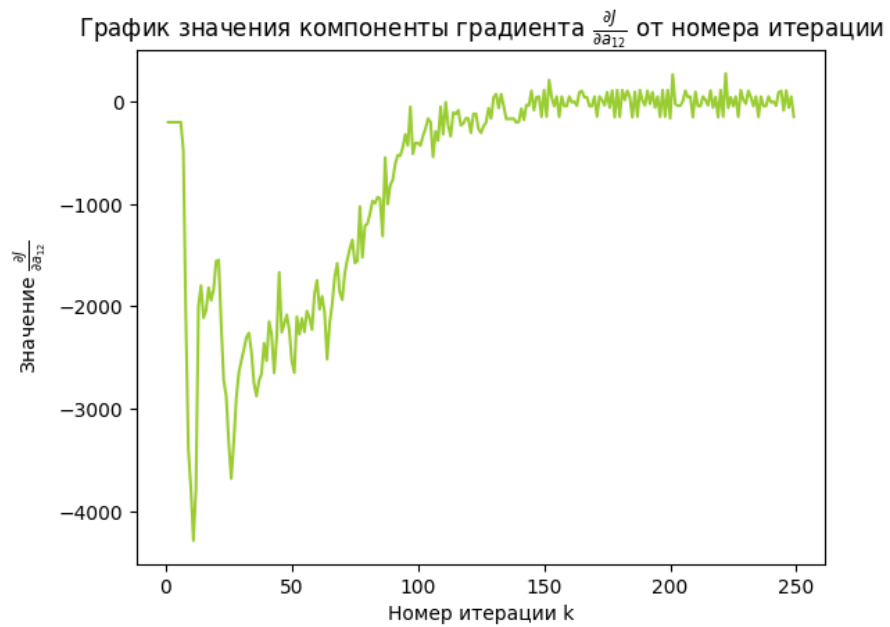


Рисунок 8.4 – График изменения значения $\frac{\partial J}{\partial a_{12}}$ от номера итерации

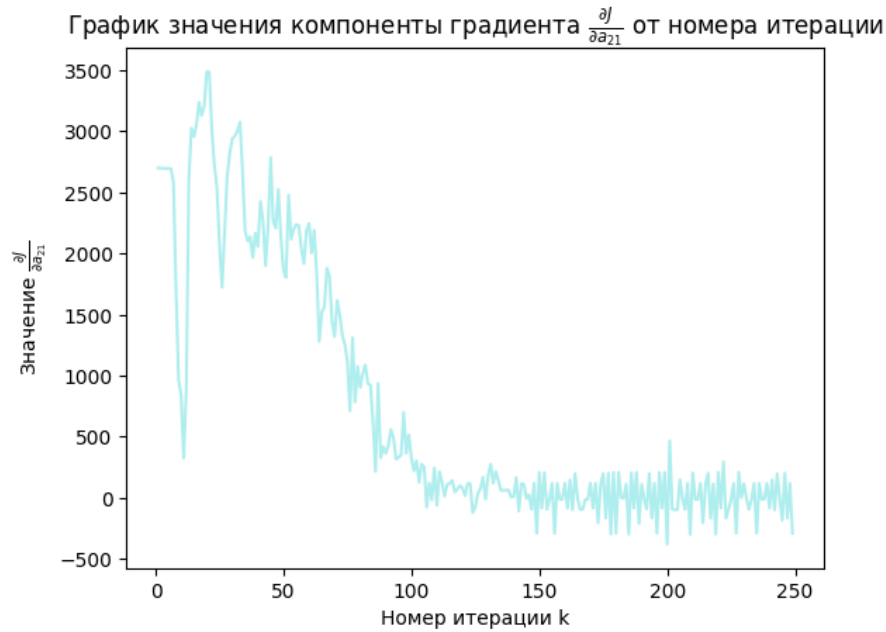


Рисунок 8.5 – График изменения значения $\frac{\partial J}{\partial a_{21}}$ от номера итерации

На графиках видно, что значения градиентов колеблются около нулевого значения. Для компонент a_{12} и a_{21} наблюдается четкая сходимость, компонента a_{11} колеблется в окрестности некоторого значения.

Таким образом, в силу приведенных выше рисунков и графиков, можно заключить, что для случая поворота алгоритм работает корректно и правильно определяет поле скоростей.

3.3. Случай сдвига и поворота

В общем случае на изображениях могут присутствовать и сдвиг и поворот одновременно. В этом случае будут изменяться и матрица A и вектор C . Для этого случая был разработан объединенный алгоритм сдвига и поворота, который параллельно вычисляет компоненты вектора параметров $u = (a_{11}, a_{12}, a_{21}, c_1, c_2)^T$.

На графиках значений градиента к номеру итераций выше видно, что для случая сдвига и случая поворота компоненты градиента имеют разные порядки, поэтому было принято решение использовать шаги градиентного спуска α_A и α_C разного порядка для вычисления компонент матрицы A и компонент вектора C соответственно.

Для оптимизации вычислений был использован следующий подход: компоненты вектора C и матрицы A пересчитываются до тех пор, пока не выполняется условие на компоненты градиента, например, для компонент вектора C :

$$\left| \frac{\partial J}{\partial c_1} \right| \leq S,$$

$$\left| \frac{\partial J}{\partial c_2} \right| \leq S,$$

где S – некоторая фиксированная достаточно малая константа, например, равная 0.1. Аналогичным образом останавливался пересчет и для компонент матрицы A . Такой подход позволяет не пересчитывать компоненты вектора параметров u , отвечающих за сдвиг или поворот, в случае если одна из частей уже пересчитана правильно и для нее выполнены условия выше.

Работа алгоритма останавливается полностью в том случае, если выполнено хотя бы одно из трех условий:

- Значение функционала на k -ой итерации $J(u^k) \leq G$;
- Значение модуля каждой компоненты вектора градиента на k -ой итерации $\left| \frac{\partial J}{\partial u_i^k} \right| \leq K, i = 1, \dots, 5$;
- Число итераций превысило заранее заданное число N , т.е. $k > N$.

Для численного эксперимента были взяты следующие последовательные изображения:

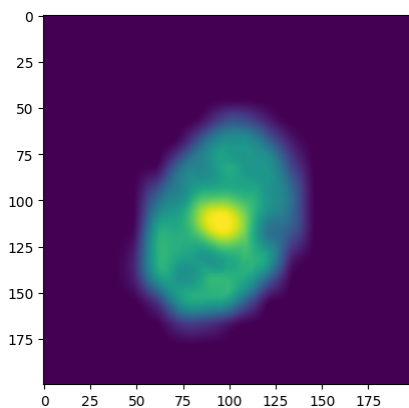


Рисунок 9.1 – Первый кадр

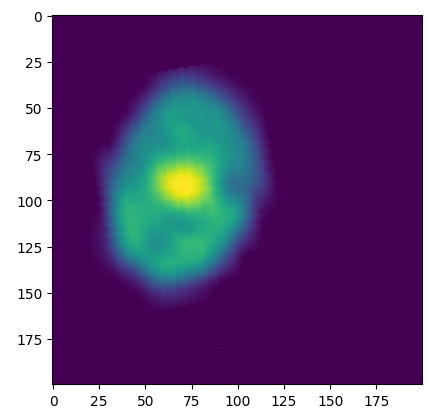


Рисунок 9.2 – Второй кадр

Для вектора параметров u были взято нулевое начальное приближение. Полученное поле скоростей приведено на рисунке 10:

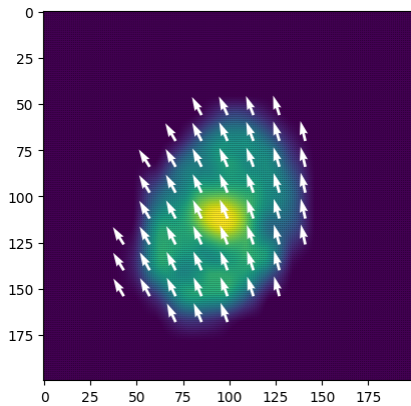


Рисунок 10.1 – Первый кадр

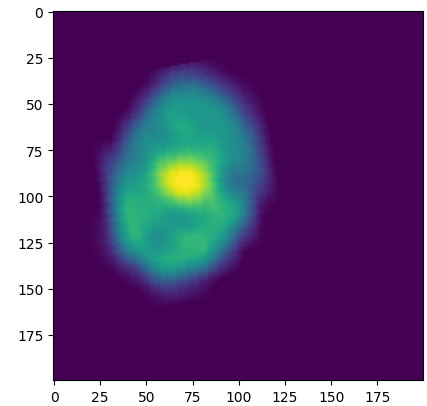


Рисунок 10.2 – Второй кадр

На рисунке ниже представлено пересчитанное в силу найденных матрицы A и вектора C изображение и, как и для случаев, когда на изображении присутствовали только сдвиг или поворот, изображение разности пересчитанного изображения и второго кадра:

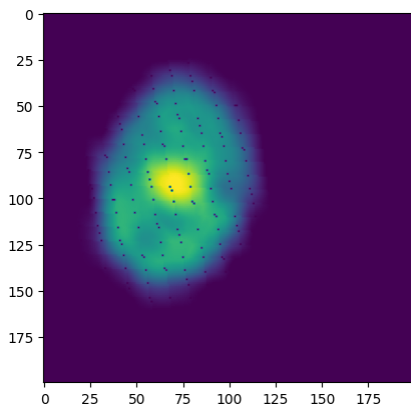


Рисунок 11.1 –
Пересчитанное изображение

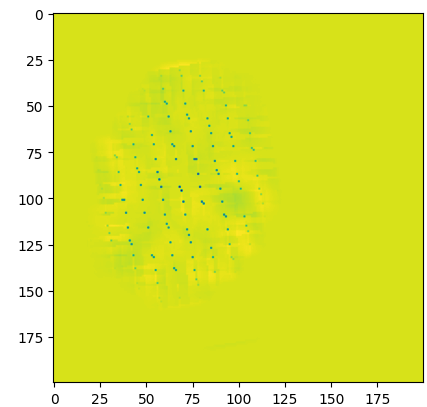


Рисунок 11.2 – Разность
изображений

Можно сделать вывод, что совмещенный алгоритм верно определил поле скоростей и компоненты вектора параметров u .

Ниже на рисунке 12 представлены графики компонент вектора u и компонент вектора градиента к номеру итерации:

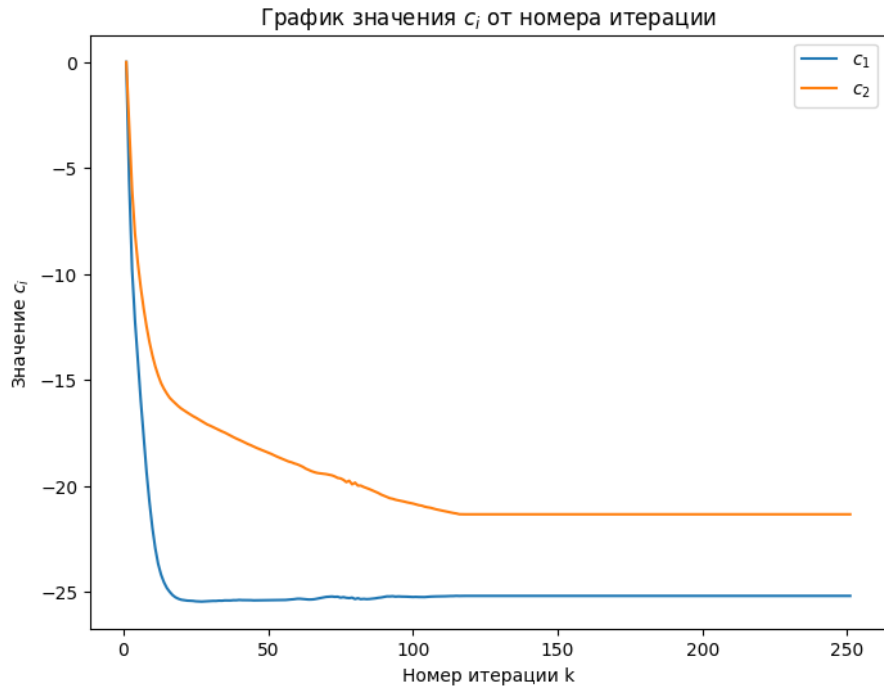


Рисунок 12.1 – График изменения значения c_i от номера итерации

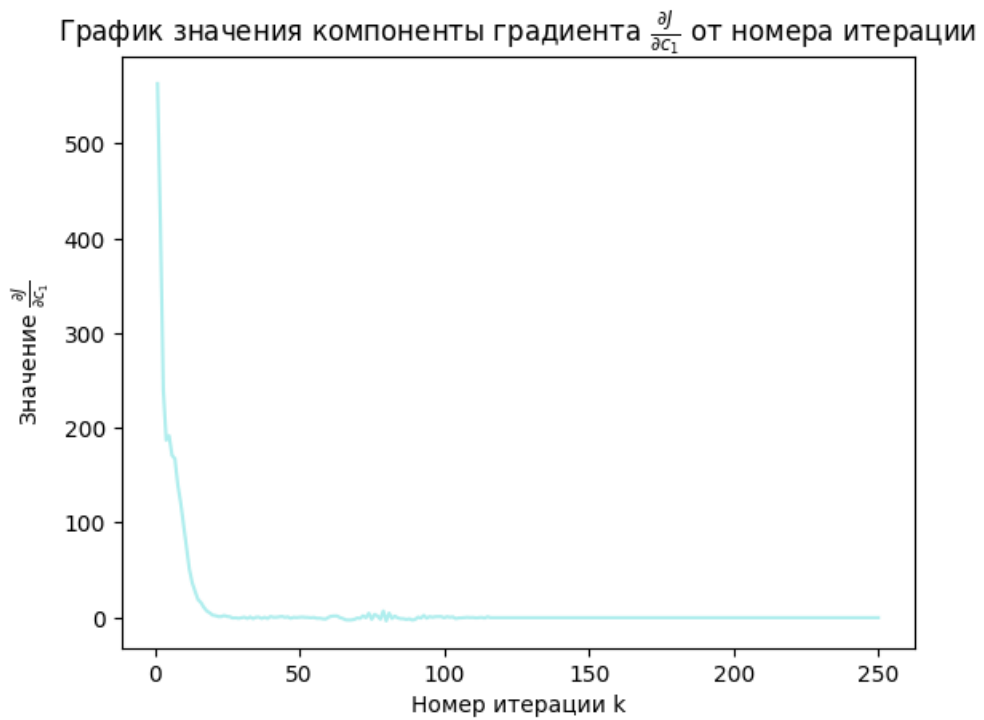


Рисунок 12.2 – График изменения значения $\frac{\partial J}{\partial c_1}$ от номера итерации

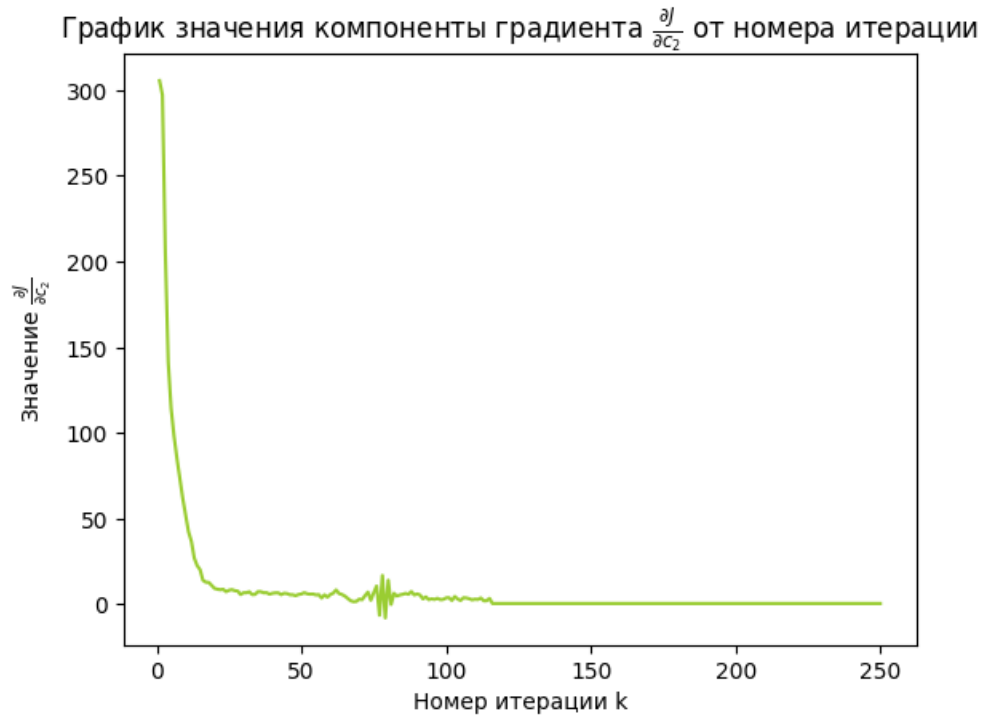


Рисунок 12.3 – График изменения значения $\frac{\partial J}{\partial c_2}$ от номера итерации

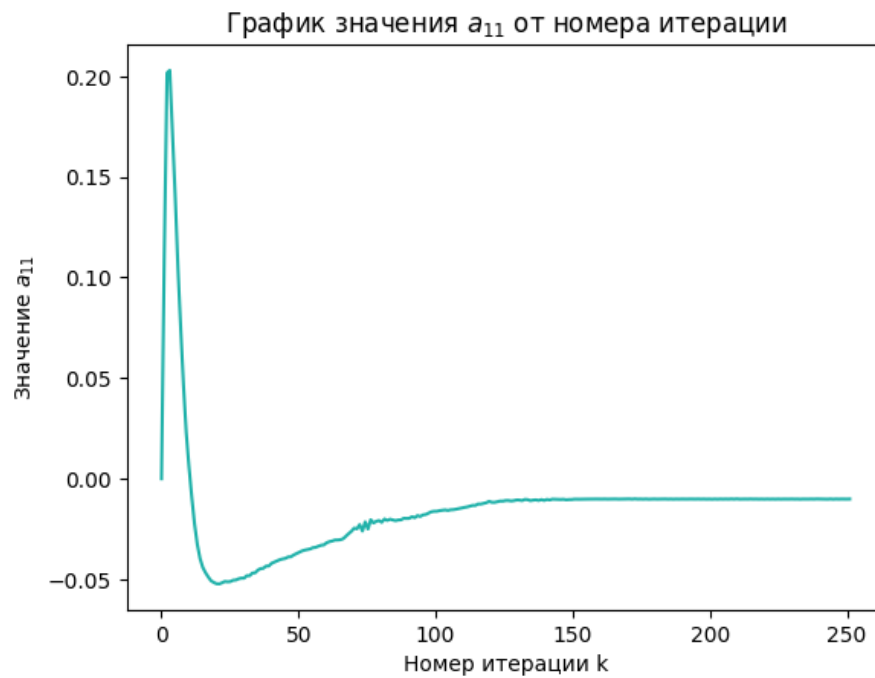


Рисунок 12.4 – График изменения значения a_{11} от номера итерации

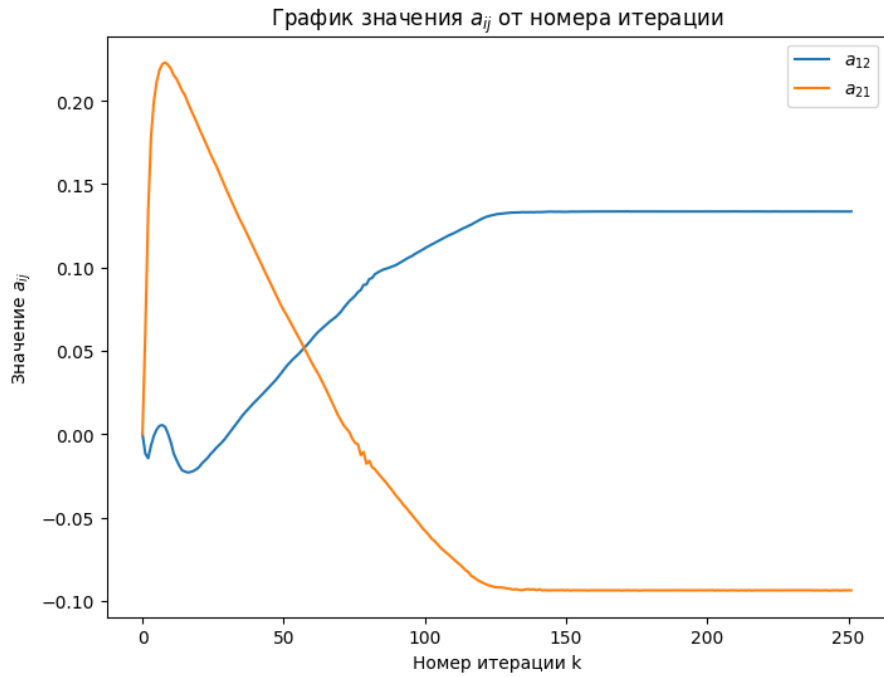


Рисунок 12.5 – График изменения значения a_{ij} от номера итерации

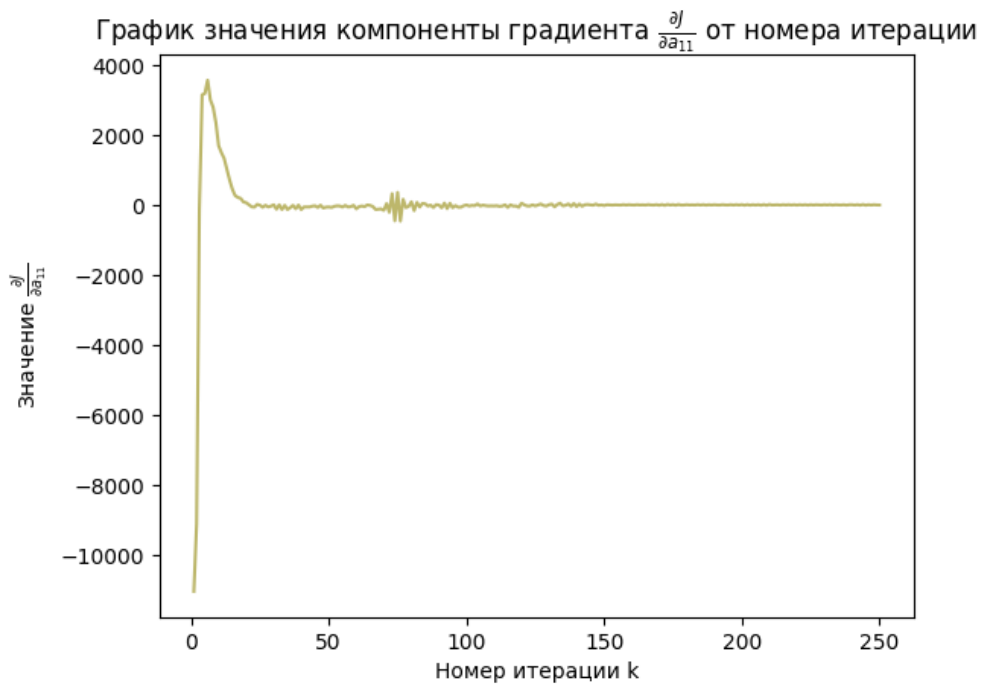


Рисунок 12.6 – График изменения значения $\frac{\partial J}{\partial a_{11}}$ от номера итерации

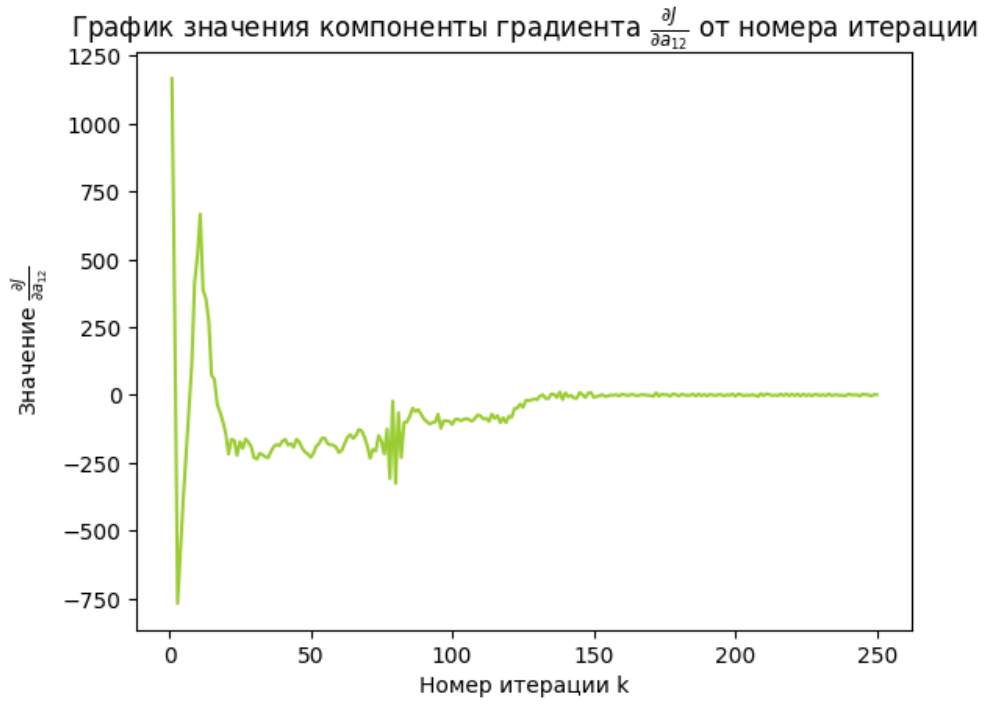


Рисунок 12.7 – График изменения значения $\frac{\partial J}{\partial a_{12}}$ от номера итерации

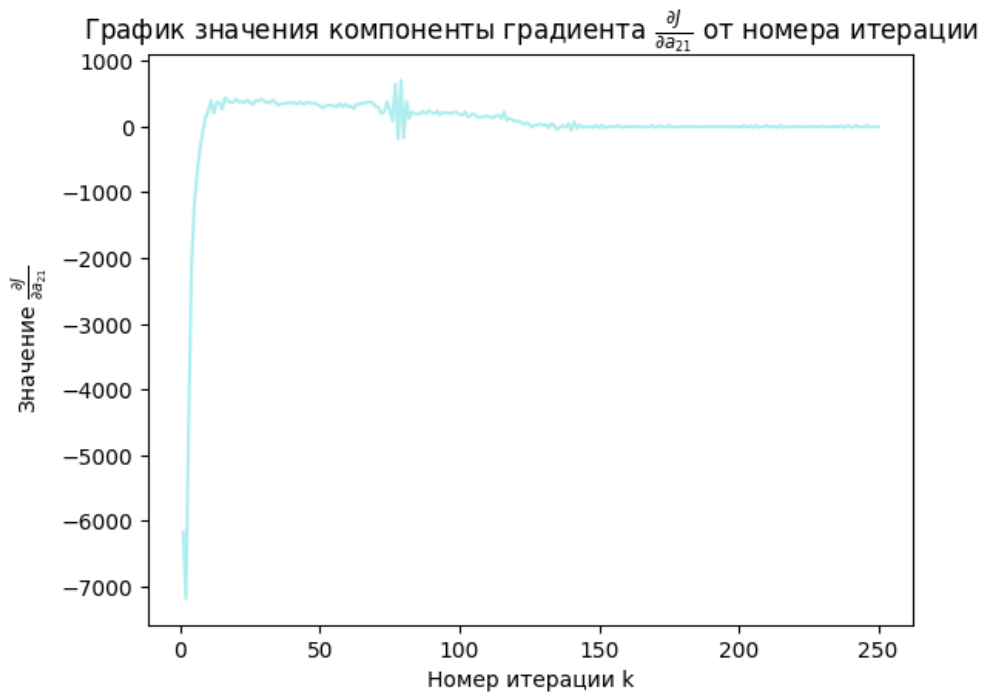


Рисунок 12.8 – График изменения значения $\frac{\partial J}{\partial a_{21}}$ от номера итерации

По графикам видно, что в данном численном эксперименте сходимость наступила практически одновременно и для компонент матрицы, и для компонент вектора.

Таким образом, совмещенный алгоритм правильно определил поле скоростей, величину сдвига и поворота при параллельном вычислении компонент вектора параметров u .

3.4. Последовательность изображений

Помимо параллельного вычисления компонент вектора параметров u , возможно последовательное вычисление компонент матрицы A и вектора C . Такой подход может применяться в тех случаях, когда известно, что между соседними кадрами происходит только сдвиг или только поворот области интереса. Преимуществом такого подхода является то, что в этом случае алгоритм будет работать быстрее в силу того, что при параллельном пересчете происходит вычисление сразу пяти компонент вектора параметров, а при последовательном пересчете не более трех.

Так в одном из численных экспериментов была рассмотрена последовательность изображений, состоящая из пяти последовательных снимков печени, где между тремя последовательными парами кадров присутствует сдвиг и между одной из пар присутствует поворот области интереса. Снимки представлены на рисунке 13.

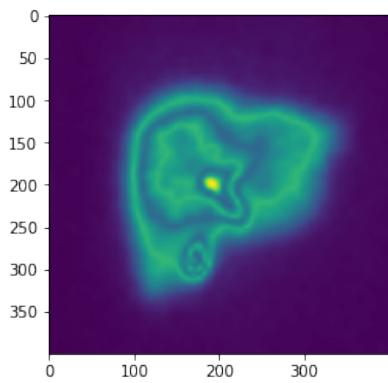


Рисунок 13.1 – Первый кадр

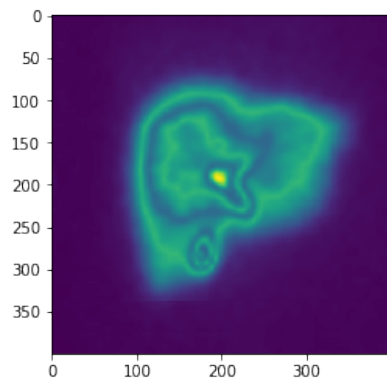


Рисунок 13.2 – Второй кадр

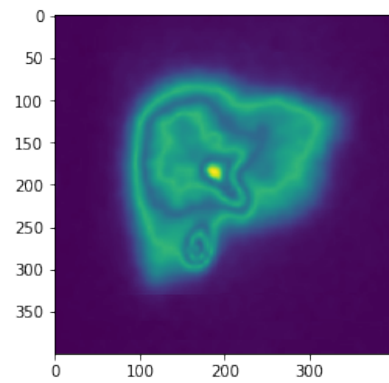


Рисунок 13.3 – Третий кадр

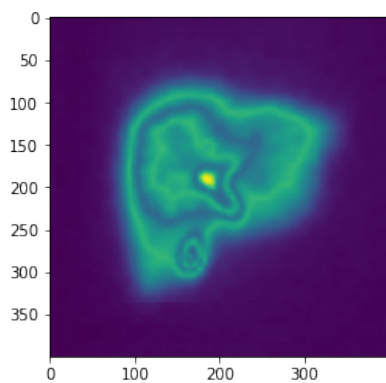


Рисунок 13.4 – Четвертый кадр

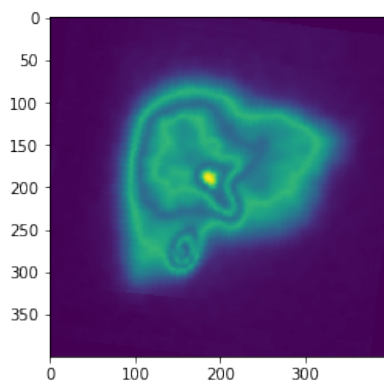


Рисунок 13.5 – Пятый кадр

Ниже на рисунке 14 приведены найденные поля скоростей между соседними кадрами:

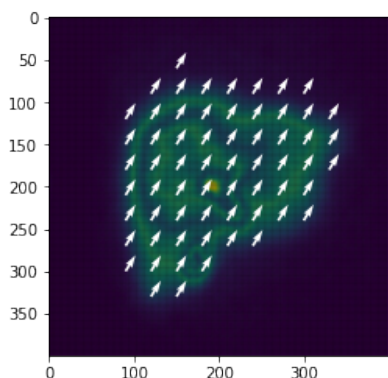


Рисунок 14.1 – Поле скоростей между кадрами 1 и 2

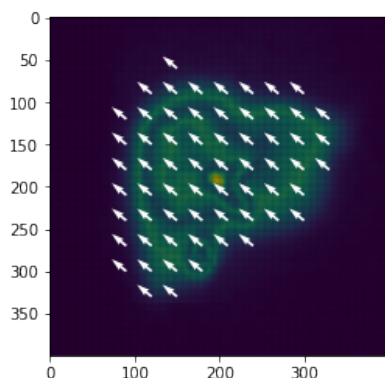


Рисунок 14.2 – Поле скоростей между кадрами 2 и 3

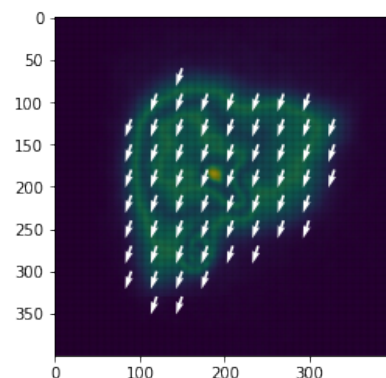


Рисунок 14.3 – Поле скоростей между кадрами 3 и 4

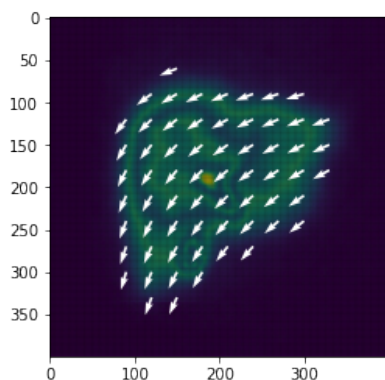


Рисунок 14.4 – Поле скоростей между кадрами 4 и 5

При обработке радионуклидных изображений построение поля скоростей может использоваться для коррекции движения и для анализа изображений, полученных при планарных динамических исследованиях.

3.5. Анализ результатов для разных моментов времени T

В ходе работы был проведен анализ влияния величины момента времени T , который фиксируется, на результат работы алгоритма. Так для случая сдвига был проведен сравнительный анализ для $T = 1$, как было во всех экспериментах выше, $T = 0.5$ и $T = 2$. Рассматривалась та же пара входных изображений, что была взята выше для случая сдвига:

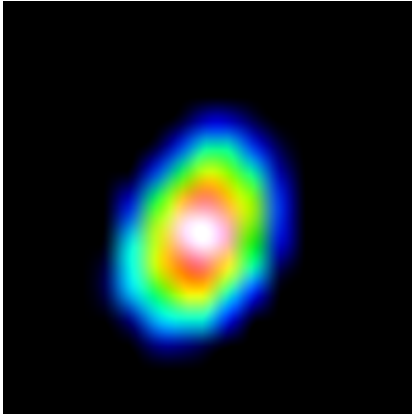


Рисунок 15.1 – Желчный пузырь на первом кадре

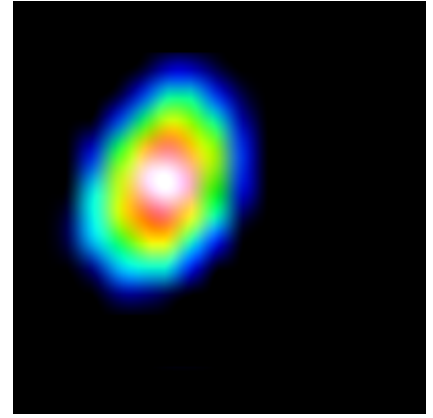


Рисунок 15.2 – Желчный пузырь на втором кадре

При фиксированном моменте времени $T = 0.5$ сходимость наступает более чем в два раза быстрее, чем для $T = 1$. Для наглядности графики для $T = 1$ и $T = 0.5$ приведены друг под другом - сверху для $T = 1$, снизу для $T = 0.5$:

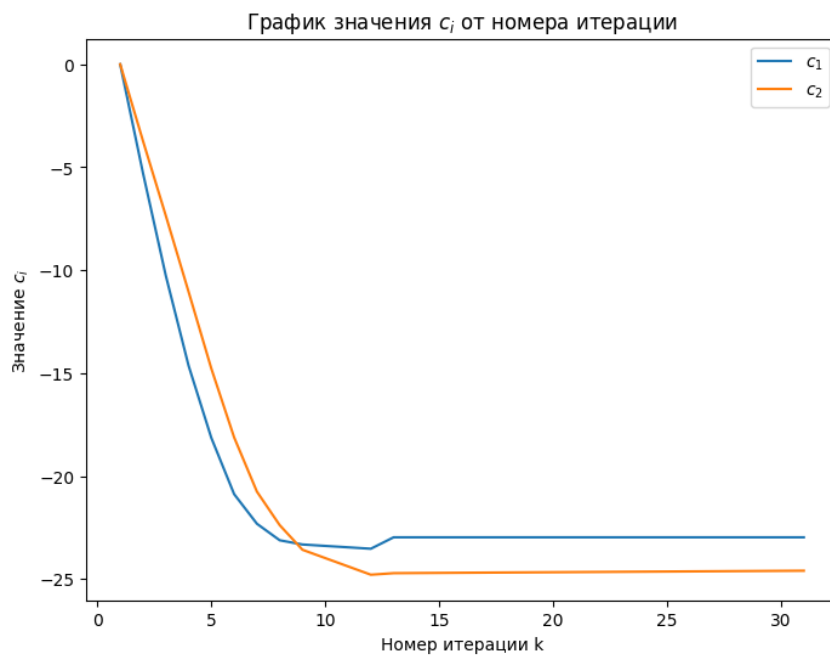


Рисунок 16.1 – Момент времени $T = 1$

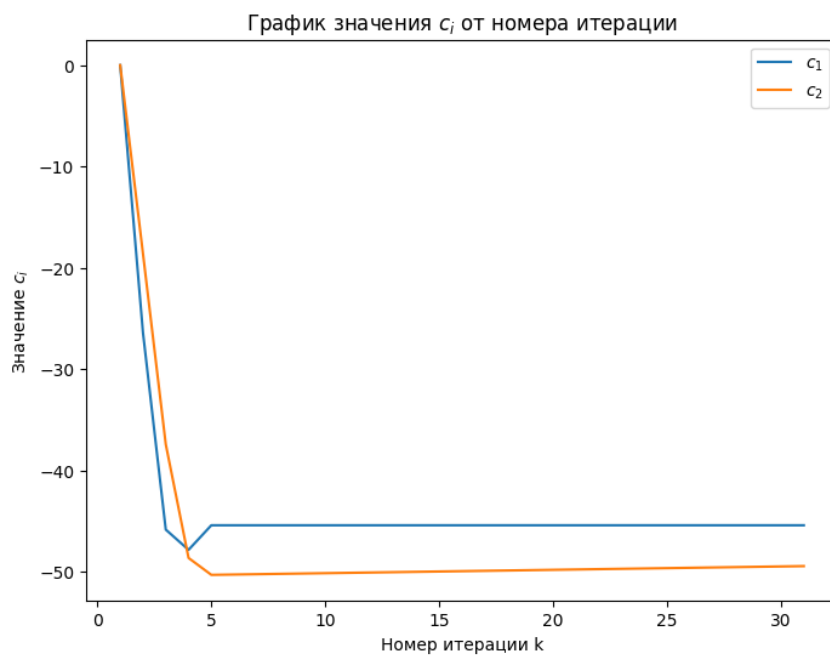


Рисунок 16.2 – Момент времени $T = 0.5$

График значения компоненты градиента $\frac{\partial J}{\partial c_1}$ от номера итерации

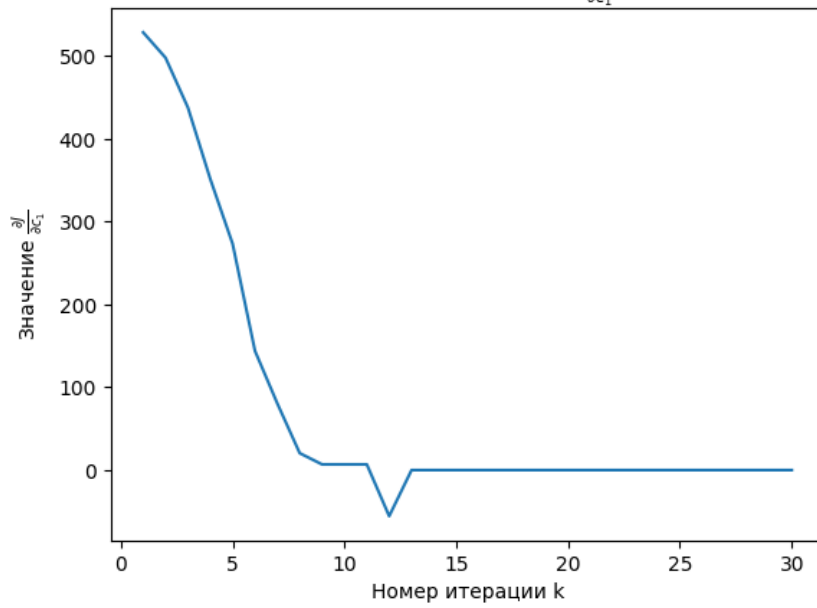


Рисунок 16.3 – Момент времени $T = 1$

График значения компоненты градиента $\frac{\partial J}{\partial c_1}$ от номера итерации

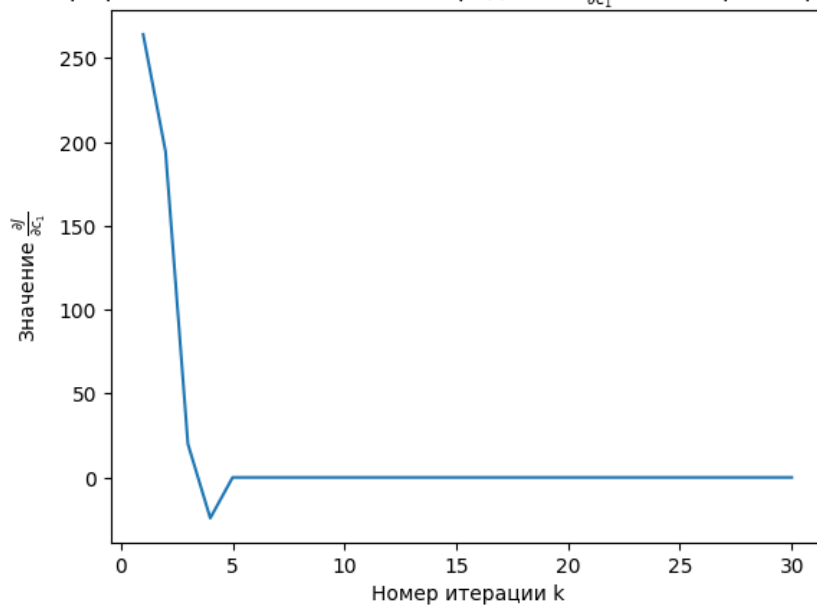


Рисунок 16.4 – Момент времени $T = 0.5$

График значения компоненты градиента $\frac{\partial J}{\partial c_2}$ от номера итерации

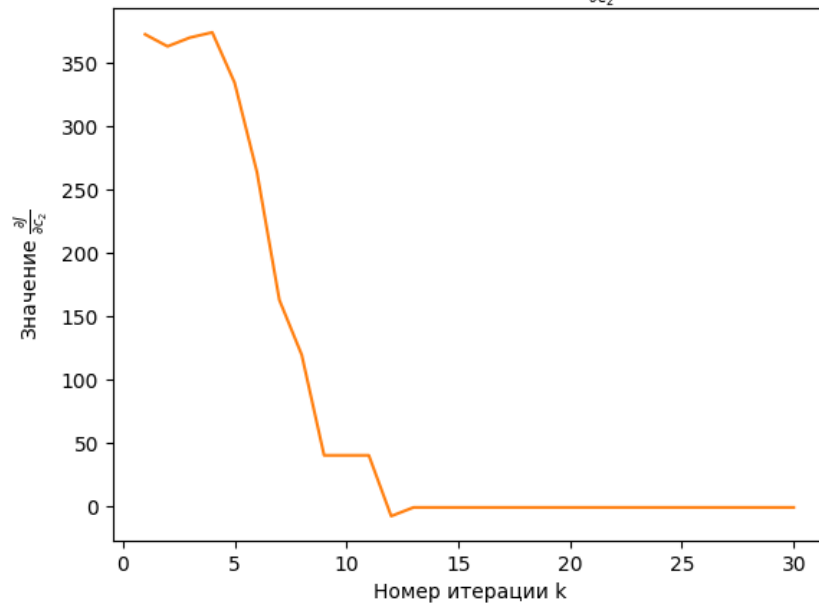


Рисунок 16.5 – Момент времени $T = 1$

График значения компоненты градиента $\frac{\partial J}{\partial c_2}$ от номера итерации

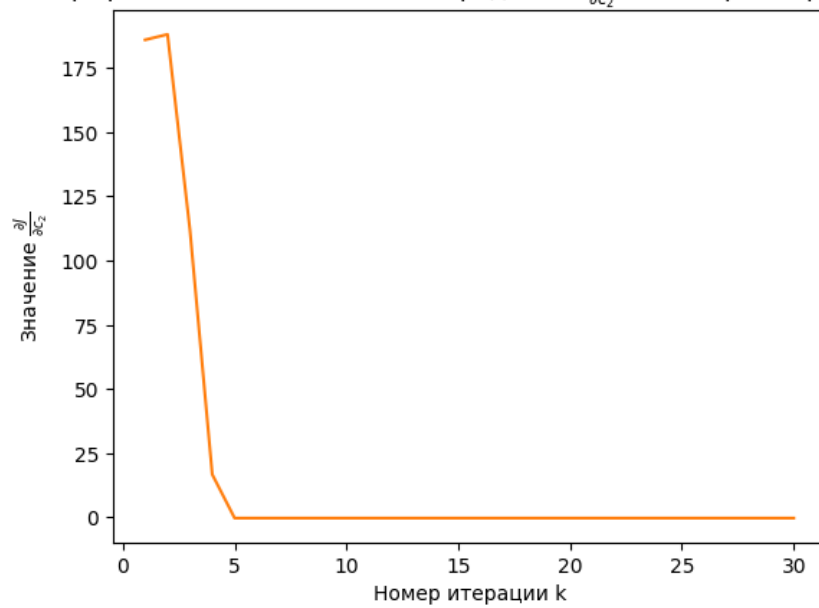


Рисунок 16.6 – Момент времени $T = 0.5$

На графиках видно, что если в случае, когда мы фиксируем момент времени $T = 1$, сходимость наступает на 13 итерации, то для той же пары изображений при фиксированном моменте времени $T = 0.5$ сходимость алгоритма достигается уже на 5 итерации.

При фиксированном моменте времени $T = 2$ наблюдается противоположная картина - в этом случае сходимость наступает позже, а именно на 28 итерации, что видно на графике:

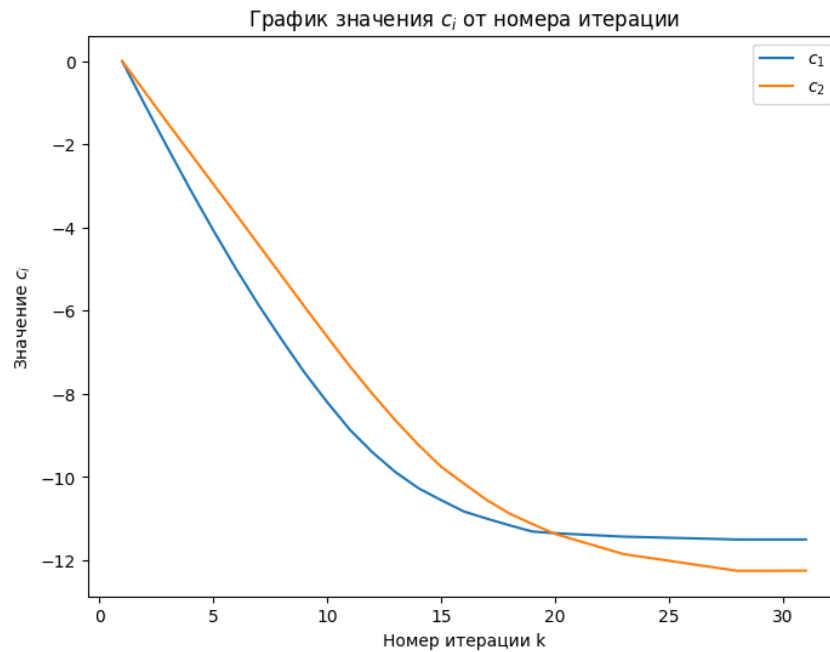


Рисунок 17 – Момент времени $T = 2$

Таким образом, при правильном выборе момента времени T возможно добиться более быстрой сходимости алгоритма. Рассмотренный алгоритм можно усовершенствовать путем добавления вариации по времени T .

Выводы

Таким образом, в данной работе был численно реализован рассмотренный подход к решению задачи построения поля скоростей. Представлены результаты работы алгоритма. В ходе численных экспериментов алгоритм показал хорошие результаты определения поля скоростей, а также величин сдвига и угла поворота. Также в работе представлена модификация алгоритма с разбиением изображения на подобласти, которая позволяет строить поле скоростей даже в тех случаях, когда на изображениях присутствует несколько движущихся объектов, а также позволяет значительно ускорить время работы алгоритма. Проведен анализ зависимости результатов от фиксируемого момента времени T , который показал целесообразность в дальнейшем заложить в алгоритм варирование данного параметра [21].

Заключение

В работе рассмотрена проблема определения поля скоростей в задачах цифровой обработки изображений. Задача построения поля скоростей решается как задача минимизации функционала с использованием градиентного метода. Был разработан и реализован алгоритм на основе предложенного подхода. Представлены результаты работы алгоритма в двух вариантах - параллельное и последовательное вычисление компонент искомого вектора параметров. Разработана и реализована модификация алгоритма с разбиением рассматриваемых изображений на подобласти на основе морфологического подхода. Реализовано построение поля скоростей для последовательностей медицинских изображений на примере радионуклидных изображений. Результаты данной работы могут применяться для обработки и анализа различных изображений, в частности медицинских – как для коррекции движения, так и для анализа последовательностей изображений.

Список литературы

- [1] Котина Е.Д. Математическое моделирование в радионуклидной диагностике // Диссертация на соискание ученой степени доктора физико-математических наук. Санкт-Петербургский государственный университет. Санкт-Петербург. 2010.
- [2] Ovsyannikov D. A., Kotina E. D. Reconstruction of velocity field // Proceedings of ICAP2012, Rostock-Warnemünde, Germany. 2012. pp. 256-258.
- [3] Kotina E.D., Ovsyannikov D.A. Velocity field based method for data processing in radionuclide studies // Problems of Atomic Science and Technology. 2018. T. 115. № 3. С. 128-131.
- [4] Lucas B. D. , Kanade T. An iterative image registration technique with an application to stereo vision // Proceedings of Imaging Understanding Workshop. 1981. С. 121 – 130.
- [5] Horn B.K.P., Schunck B.G. Determining optical flow // Artificial Intelligence. 1981. №17. С. 185 – 203.
- [6] Xiaoliang G., Bansmer S. Horn–Schunck optical flow applied to deformation measurement of a birdlike airfoil // Chinese Journal of Aeronautics. 2015. T. 28. №5. С. 1305 – 1315.
- [7] Farneback G. Two-Frame Motion Estimation Based on Polynomial Expansion // Lecture Notes in Computer Science. 2003. T. 2749. С. 363 – 370.
- [8] Alba A., Arce-Santana E., Rivera M. Optical flow estimation with prior models obtained from phase correlation // Lecture Notes in Computer Science. 2010.
- [9] Bai M., Luo W., Kundu K., Urtasun R. Exploiting semantic information and deep matching for optical flow // Proceedings of the 14th European

Conference on Computer Vision, Lecture Notes in Computer Science. 2016. №9908. С. 154 — 170.

- [10] Bailer C., Varanasi K., Stricker D. CNN-based patch matching for optical flow with thresholded hinge embedding loss // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2017. С. 3250 — 3259.
- [11] Gadot D., Wolf L. PatchBatch: A batch augmented loss for optical flow // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2016. С. 4236 – 4245.
- [12] Ranjan A., Black M.J. Optical flow estimation using a spatial pyramid network // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2017. С. 2720 – 2729.
- [13] Sun D., Yang X., Liu M.Y., Kautz J. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2018. С. 8934 – 8943.
- [14] Bazhanov P., Kotina E., Ovsyannikov D., Ploskikh V. Optimization algorithm of the velocity field determining in image processing // Cybernetics and Physics. 2018. №7. С. 174 – 181.
- [15] Гонсалес Р., Вуд Р. Цифровая обработка изображений. М.: ТЕХНОСФЕРА. – 2005.
- [16] Бажанов П.В., Котина Е.Д. Об оптимизационном подходе при построении поля скоростей в задачах обработки изображений // «Известия Иркутского государственного университета». Серия «Математика». 2018. Т. 24. С. 3 – 11.
- [17] Котина Е.Д., Пасечная Г.А. Определение поля скоростей в задачах обработки изображений // «Известия Иркутского государственного университета». Серия «Математика». 2013. Т. 6. №3. С. 48 – 59.

- [18] Kotina E., Pasechnaya G. Optical flow-based approach for the contour deflection in radionuclide images processing // Cybernetics and Physics. 2014. №2. С. 62 – 65.
- [19] Котина Е. Д., Леонова Е. Б., Плоских В. А. Обработка радионуклидных изображений с использованием дискретных систем // Вестник СПбГУ. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. Вып. 4. С. 544 – 554.
- [20] Kotina E. D., Leonova E.B., Ploskikh V.A. Displacement Field Construction Based on a Discrete Model in Image Processing Problems // «Известия Иркутского государственного университета». Серия «Математика». 2022. Т. 39. С. 3 – 16.
- [21] Kotina E.D., Ovsyannikov D.A., Elizarova M.V. Optimization approach to the velocity field determining problem // Cybernetics and Physics. 2022. Т. 11. №3. С. 131 – 135.
- [22] Овсянников Д.А. Моделирование и оптимизация динамики пучков заряженных частиц // Л: Издательство Ленинградского университета. 1990. С. 312.