

# РЕАЛИЗАЦИЯ ОБРАТНОГО КОМПОЗИЦИОННОГО АЛГОРИТМА ГАУССА-НЬЮТОНА ДЛЯ ПОСТРОЕНИЯ ПОЛЕЙ ВЕКТОРОВ ПЕРЕМЕЩЕНИЙ В ЗАДАЧЕ ОЦЕНКИ ДЕФОРМАЦИОННОГО СОСТОЯНИЯ МАТЕРИАЛОВ

Беоснев А.П.

Доцент кафедры ВТ Болотова Ю.А., Младший научный сотрудник ИФПМ СО РАН Любутин П.С.  
Национальный исследовательский Томский политехнический университет  
snoopdogmaster@gmail.com

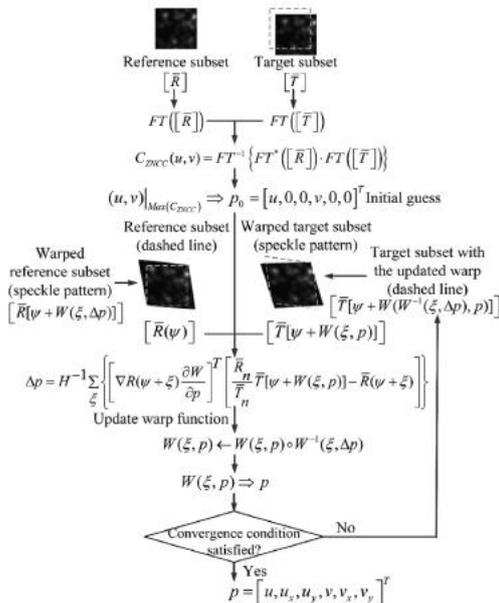
## Введение

Одно из важных направлений прогресса – повышение надежности и долговечности машин и сооружений. Для улучшения качества разрабатываемых материалов, необходима диагностика деформационного состояния элементов конструкций в процессе их изготовления и эксплуатации. Для диагностики используются как расчетные методы, так и экспериментальные, которые основаны на измерении напряжений и деформаций, которые осуществляют в процессе испытаний. При использовании экспериментальных методов, на поверхность исследуемых покрытий наносят точки, линии и полосы. Для практической оценки деформации используем обратный композиционный алгоритм Гаусса-Ньютона. Который дает относительно точный результат для сравнения с теоретическими значениями.

## Описание алгоритма

На рисунке 1[1] приведена схема алгоритма. Который функционально разбит на две части.

Для первого (целочисленного) приближения смещения используется быстрое преобразование Фурье (FFT). Далее, полученные целочисленные смещения используются для нахождения субпиксельного смещения с помощью обратного композиционного алгоритма Гаусса-Ньютона (IC-



GN).

Рис.1. Схема алгоритма вычисления смещения и использованием FFT и IC-GN

В качестве входных данных используем изображение деформируемого объекта, которое разбивается на области, для каждой из которых находится смещение.

Рассматриваем некоторую область первого изображения  $R$  и ставим ей в соответствие область  $T$  второго изображения, координаты которой относительно области  $R$  смещены на величину  $(u, v)$ , где  $u$  – смещение по координате  $x$ , а  $v$  – смещение по координате  $y$ . Данное смещение находится целочисленно, что не является достаточно точным результатом, согласно поставленной задаче.

После того, как мы получили вектор  $p = [u, 0, 0, v, 0, 0]^T$ , согласно алгоритму, необходимо вычислить инкрементирующий вектор  $\Delta p = [u, u_x, u_y, v, v_x, v_y]^T$ , уменьшающий ошибку вычислений. Данный вектор выражается из формулы:

$$C_{ZNSSD}(\Delta p) = \sum_{\xi} \left\{ \frac{\bar{R}(\psi + W(\xi, \Delta p))}{\bar{R}_n} - \frac{\bar{T}(\psi + W(\xi, p))}{\bar{T}_n} \right\}^2$$

где  $\psi = [x, y, 1]^T$  – координаты центра области на изображении,  $\xi = [\Delta x, \Delta y, 1]^T$  – локальные координаты относительно центра области,  $\bar{R}_i = R_i - R_m$ ,  $\bar{T}_i = T_i - T_m$  – отклонение от среднего значения,  $R_m = \frac{1}{N} \sum_i R_i$ ,  $T_m = \frac{1}{N} \sum_i T_i$  – среднее значение интенсивности пикселей области изображения,

$$\bar{R}_n = \sqrt{\sum_{\xi} \{ \bar{R}[\psi + W(\xi, \Delta p)] \}^2}, \quad \bar{T}_n = \sqrt{\sum_{\xi} \{ \bar{T}[\psi + W(\xi, p)] \}^2},$$

Функция аффинных преобразований:

$$W(\xi, p) = \begin{bmatrix} 1 + u_x & u_y & u \\ v_x & 1 + v_y & v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ 1 \end{bmatrix}$$

Формула для вычисления вектора  $\Delta p$ :

$$\Delta p = H^{-1} \sum_{\xi} \left\{ \left[ \nabla R(\psi + \xi) \frac{\partial W}{\partial p} \right]^T \left[ \frac{\bar{R}_n \bar{T}}{\bar{T}_n} \bar{T}[\psi + W(\xi, p)] - \bar{R}(\psi + \xi) \right] \right\}$$

где  $H^{-1}$  – обратная матрица Гессе  $H$

$$H = \sum_{\xi} \left\{ \left[ \nabla R(\psi + \xi) \frac{\partial W}{\partial p} \right]^T \left[ \nabla R(\psi + \xi) \frac{\partial W}{\partial p} \right] \right\}$$

После вычисления вектора  $\Delta p$ , он суммируется с вектором  $p$ , функция аффинных преобразований  $W(\xi, p)$ . Вычисление вектора  $\Delta p$  производится итеративно. На каждой итерации  $\Delta p$  уменьшается. Выполнение алгоритма прекращается, когда достигается необходимая точность вычислений.

После вычисления вектора  $\Delta p$  также

необходимо пересчитывать интенсивности пикселей на изображении  $T$ , с учетом новых координат для области. Поэтому используется интерполяция. Для более точного нахождения интерполируемого значения было решено использовать бикубическую интерполяцию.

### Реализация алгоритма

Для удобного и быстрого выполнения матричных операций, используемых в алгоритме, используется библиотека Eigen 3й версии.

Для того, чтобы не загромождать код программы и поддерживать принципы абстрагирования, создан отдельный класс, в котором описаны поля и функции для вычислений отдельной области.

Массив из объектов данного класса, который описывает векторное поле смещений для всего изображения используется в другом классе, в котором описаны операции уже более высокого уровня, уже относительно всего изображения.

Результаты вычислений(вектора) записываются в файл формата HDF5 (Hierarchical Data Format). Далее полученный файл возможно открыть в специальном визуализаторе, подмодулем которого и является данная программа.

Используя формат HDF5 вычисления можно производить сразу для серии изображений, таким образом, затем просмотреть динамику изменения смещений.

Результаты работы по данному алгоритму результаты предоставлены на рисунке 3 предоставлен. Для наглядности масштаб векторов увеличен.

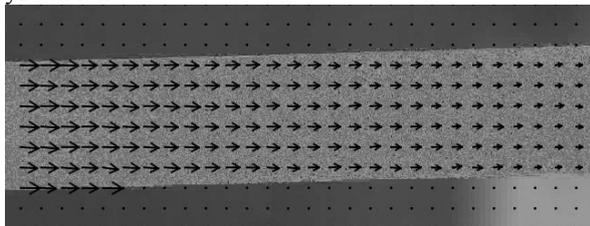


Рис.3. Пример работы программы

### Оценка точности работы программы и сравнение с другими программами

Используем результаты тестов в виде графиков для тестовых изображений, взятых из источника [2]. Тестовые изображения представлены в виде симуляции смещения изображения без шума и с шумом по Гауссу с радиусом 16. Данные тестовые изображения представляют из себя серию из 50 изображений, каждое последующее изображение сдвинуто относительно предыдущего на 0,02 пикселя, таким образом, 50е изображение будет смещено на 1 пиксель. Зная теоретическое значение смещение и вычислив результаты, используя программу, мы можем оценить случайную и систематическую ошибку смещения.

На рисунке 4, взятом из источника [2], изображены графики систематической ошибки для различных реализаций (pack 1-9). Для изображений без шумов слева и для изображений с шумами справа.

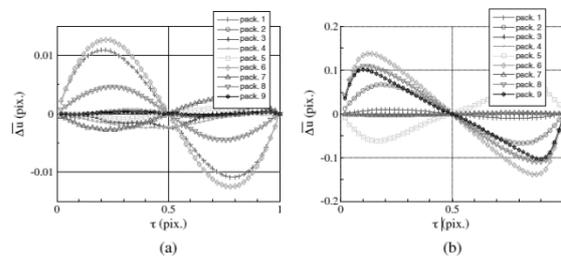


Рис.4. Графики систематической ошибки вычислений

Из результатов вычислений нашей программы, построим аналогичные графики и сравним (рис.5) результаты.

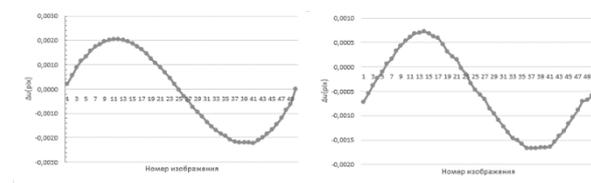


Рис.5. Графики систематической ошибки вычислений для созданной программы

Очевидно, что созданная нами программа может конкурировать с программами, представленными в источнике [2] по точности вычислений.

### Заключение

Проанализировав графики, можно сказать, что систематическая ошибка максимальна при субпиксельном смещении на 0,25 пикселя и минимальна при смещении на 0,5 и 1 пиксель.

Таким образом, делаем вывод о том, что созданная программа позволяет вычислить смещение с довольно высокой точностью.

### Список использованных источников

1. Jiang Z, Kemaq Q, Miao H, Yang J, Tang L. Path-independent digital image correlation with high accuracy, speed and robustness. Opt Laser Eng 2015;65:93–102.
2. F. Amiot, M. Bornert, P. Doumalin, J. -C. Dupré, M. Fazzini, J. -J. Orteu, C. Poilâne, L. Robert, R. Rotinat, E. Toussaint, B. Wattrisse and J. S. Wienin. Assessment of Digital Image Correlation Measurement Accuracy in the Ultimate Error Regime: Main Results of a Collaborative Benchmark. Strain (2013) 49, 483–496.