Optics and Lasers in Engineering 111 (2018) 98–107

Contents lists available at ScienceDirect

Optics and Lasers in Engineering

journal homepage: www.elsevier.com/locate/optlaseng

Accelerate multi-thread path-dependent digital image correlation by minimizing thread competition for real-time deformation measurement

Jianguang Shuai[a], Jiaqing Zhao[b], Liping Lei[a,*], Pan Zeng[a], Xinxin Wu[b], Libin Sun[b]

[a] *Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China*
[b] *Institute of Nuclear and New Energy Technology of Tsinghua University, the Key Laboratory of Advanced Reactor Engineering and Safety, Ministry of Education, Collaborative Innovation Center of Advanced Nuclear Energy Technology, Beijing 100084, China*

ARTICLE INFO

Keypoints:
Digital image correlation
Real-time measurement
Multi-thread parallelization
Path-dependent
Thread competition

ABSTRACT

With the increasing demands on the real-time measurement using digital image correlation (DIC), the fast and ultra-fast computation are attracting much more attention from academic and engineering fields in last decades. Multi-thread parallelization is a recognized effective method to raise the correlation speed of traditional single-thread DIC up to higher order of magnitude. However, for the path-dependent parallel DIC based on specific initial guess propagation method, competition between different threads greatly cuts down the overall speed, because different threads visit the same queue or status matrix at the same time. Therefore, the desired linear speed-up ratio growth could hardly be achieved especially for large thread number. In this work, three novel strategies are proposed to minimize the thread competition for the existing path-dependent parallel DIC, including i) allocating private queues for private thread, ii) restarting idle thread if others are busy, and iii) collecting more uncorrelated neighboring points for subsequent correlation. In this way, the conflict between threads is obviously decreased and experiment results reveal that the proposed method could upgrade the 2D-DIC speed up to 145,000 points per second using 48 threads, which makes the real-time measurement feasible for region of interest (ROI) with less than 5000 points.

1. Introduction

Nowadays, digital image correlation (DIC) [1–4] has become a well-know, widely accepted and effective non-contact full-field deformation measurement technique, which finds its wide applications in diverse fields (e.g., experimental mechanics [5–7], biomechanics and medical measurement [8–10]) and in multi-scale (including nano-scale [11], micro-scale [12], and macro-scale). The computation burden in DIC is becoming larger than ever before which derives from the facts that, i) the much higher resolution camera [13] are used by researchers, ii) the higher density data is required to improve the spatial resolution, and iii) the larger number of images needs to be analyzed in high or ultra-high speed experiment (e.g. the blast and impact test [14]). In addition, the real-time DIC for dynamic strain measurement [15] comes up with more stringent requirements on computation speed as well. Therefore, there is an urgent need to put forward high-speed deformation computation algorithms to satisfy the rapid development of DIC.

In the last thirty years, much effort has been made by researchers to improve the computation efficiency of DIC with Newton–Raphson (NR) algorithm [16–18], which is a gold standard for sub-pixel registration. Huang et al. [19] developed a global sum-table approach to evaluate the double sums arising in the zero-normalized cross-correlation coefficient (ZNCC) and proposed a fast recursive scheme combining with the sum tables to accelerate the calculation of the cross-correlation term in the ZNCC. Due to this improvement, the computational efficiency of integer-pixel correlation searching is increased up to 10 to 50 times on the condition of keeping the measurement accuracy. Zhou and Chen [20] put forward two propagation functions to get more accurate initial estimate of desired parameters and to reduce the iterations at the same time. Subsequently, with the presence of inverse compositional Gauss–Newton (IC-GN) [21–23], which eliminates the redundant calculation of Hessian matrix during iteration, the speed of subpixel registration reached an unprecedented high level and it was widely adopted by researchers. Based on IC-GN, Wu [15] proposed an efficient integer-pixel search scheme with combination of an improved particle swarm optimization (PSO) algorithm and the block-based gradient descent search (BBGDS) algorithm for a high-accuracy real-time DIC method. In addition, Jiang et al. [24] introduced the integral image technique into DIC to handle the complex items (Hessian matrix) in the equations of DIC algorithm. Lu et al. [25] developed a fast non interpolation method for calculating displacement with subpixel precision using fast Fourier transform (FFT). Recently, Simončič and Podržaj [26] modified the classical IC-GN algorithm and put forward the parametric sum of squared differences (PSSD) correlation criterion instead of the zero-mean normalized sum of square difference (ZNSSD) correlation criterion while

* Corresponding author.
E-mail address: leilp@mail.tsinghua.edu.cn (L. Lei).

https://doi.org/10.1016/j.optlaseng.2018.07.013
Received 21 April 2018; Received in revised form 5 July 2018; Accepted 24 July 2018
0143-8166/

the computational time could be reduced by up to 30% with convergence tolerance of 0.001.

Although many methods are raised to improve the computation speed, it is difficult to calculate more than 10,000 points per second by single thread. In order to meet the real-time requirements [27], parallel computation was introduced into DIC, which improved calculation speed dramatically. In parallel computation [28], multiple threads interact with each other to correlate all points of interest (POIs) in predefined ROI. Nevertheless, the traditional DIC calculation is a point-by-point sequential process that does not make full use of computation capability of increasing powerful hardware, e.g. the multi-core central processing unit (CPU), graphic processing unit (GPU), multi-processor system. Based on GPU, the path-independent parallel computation is implemented which is much more efficient than CPU in processing images. Jiang et al. [29–31] developed a path-independent method implemented on GPU and the compute unified device architecture (CUDA) on NVIDIA compute. Yet, for the path-independent method, the fast Fourier transform (FFT) based initial guess calculation method in [29,30] is primarily designed for rigid translation, it can hardly tackle the large and complex inter-frame deformation [32]. In contrast to path-independent method, Pan and Tian [33] improved robust reliability-guided displacement tracking (RGDT) strategy and proposed a new parallel path-dependent method for 2D-DIC based on CPU. Compared with path-independent method, the path-dependent method makes the best use of information of adjacent POIs and computes the initial guess quickly, hence it reduces the iterations and upgrades the success rate of correlation, which is especially true for large deformation measurement. However, in Pan's study, the speed no longer increased when the number of threads was more than eight due to the limitation of CPU cores. More recently, Shao et al. [34,35] presented a parallel scheme for real-time 3D-DIC and improved the calculation speed further. However, SHAO's method still could not get desired linear speed-up ratio growth when the number of thread is large because of thread competition (please refer to Section 3.2 for detailed analysis of thread competition).

In this work, we propose three novel strategies to minimize the thread competition in parallel path-dependent DIC to enhance the correlation speed further. The improving strategies includes 1) allocating private queue for each thread, 2) restarting idle thread if others are busy, and 3) collecting more uncorrelated neighboring points for each thread in subsequent correlation. Benefiting from these strategies, we successfully reach the speed of 145,000 points per second in the case of 48 threads with convergence tolerance of 0.01 and subset size of 31 by 31 pixels, which makes the real-time measurement feasible for ROI with less than 5000 points.

The remainder of this paper is organized as follows. In Section 2, the basic principles of DIC are introduced. Section 3 demonstrates the existing parallelization algorithm and analyzes the source of thread competition. Then we show the proposed improving strategies in detail. The performance of resultant parallel algorithm is verified by two real experiments in the Section 4. Finally, we end the paper with some conclusions and potential optimization in the Section 5.

2. Basic principles

2.1. Basic principles of DIC

DIC is a non-contact optical metrology for the full-field deformation measurement of POIs in the predefined ROI on speckled surface (Fig. 1).

The ROI is specified in the reference image and divided into evenly spaced virtual grids. Each point in the intersection of the virtual grids is called a POI, whose displacement and strain (or displacement gradient) will be calculated by registration algorithm. A square subset of $(2M+1)\times(2M+1)$ pixels centered at one POI in reference image is denoted as $f(x, y)$ and the deformed subset in the deformed image is represented as $g(x', y')$, where (x, y) and (x', y') are the local coordinates of pixels in reference and deformed subsets respectively. The grey level of the pixels in the deformed image are compared with that of the reference image. We commonly use ZNSSD criterion to describe the similarity between two subsets. ZNSSD is defined as,

$$C_{ZNSSD}(\mathbf{p}) = \sum_{x=-M}^{M}\sum_{y=-M}^{M}\left\{\frac{f(x,y)-f_m}{\sqrt{\sum_{x=-M}^{M}\sum_{y=-M}^{M}\left[f(x,y)-f_m\right]^2}} - \frac{g(x',y')-g_m}{\sqrt{\sum_{x=-M}^{M}\sum_{y=-M}^{M}\left[g(x',y')-g_m\right]^2}}\right\}^2 \tag{1}$$

where $f_m = \frac{1}{(2M+1)^2}\sum_{x=-m}^{m}\sum_{y=-m}^{m} f(x,y)$ and $g_m = \frac{1}{(2M+1)^2}\sum_{x=-m}^{m}\sum_{y=-m}^{m} g(x', y')$ are the mean value of grey level in reference and deformed subsets respectively, $\mathbf{p}$ denotes the desired deformation vector which describes the relationship between (x,y) and(x', y'), which is determined by the selection of shape function. The widely used first-order shape function is in the form of,

$$\begin{cases} x' = x + u + u_x\Delta x + u_y\Delta y \\ y' = y + v + v_x\Delta x + v_y\Delta y \end{cases} \tag{2}$$

where $\Delta x = x - x_0$, $\Delta y = y - y_0$, u and v represent the displacement of the center point in the x and y directions respectively, u_x, u_y, v_x and v_y are the displacement gradient components of the subset, and $\mathrm{p} = [u, v, u_x, u_y, v_x, v_y]^T$ is the corresponding deformation parameter vector. Second-order shape function [36] is commonly used to correlate the complex non-linear deformation, it has the form of,

$$\begin{cases} x' = x + u + u_x\Delta x + u_y\Delta y + \frac{1}{2}u_{xx}\Delta x^2 + u_{xy}\Delta x\Delta y + \frac{1}{2}u_{yy}\Delta y^2 \\ y' = y + v + v_x\Delta x + v_y\Delta y + \frac{1}{2}v_{xx}\Delta x^2 + v_{xy}\Delta x\Delta y + \frac{1}{2}v_{yy}\Delta y^2 \end{cases} \tag{3}$$

where $u_{xx}, u_{xy}, u_{yy}, v_{xx}, v_{xy}, v_{yy}$ are the additional second-order deformation parameters.

The six deformation parameters in Eq. (1) can be resolved by minimizing $C_{\mathrm{ZNSSD}}(\mathbf{p})$ by the means of NR iteration,

$$\Delta\mathbf{p} = \frac{\nabla C(\mathrm{p}_0)}{\nabla\nabla C(\mathrm{p}_0)} \tag{4}$$

where $\mathrm{p}_0$is the initial guess [32] of the deformation, $\nabla C(\mathbf{p})$ and $\nabla\nabla C(\mathbf{p})$ are the gradient and Hessian matrix of $C(\mathbf{p})$.

2.2. IC-GN algorithm

IC-GN [23] is more efficient than classic NR. In IC-GN, the role of the reference and deformed images is switched. Wrap function $\mathbf{W}$ (see Eq. (5)) is introduced instead of Eq. (2) to depict the position and the shape between the target subset relative to the reference subset by,

$$\mathbf{W}(\xi;\ \mathrm{p}) = \begin{bmatrix} 1+u_x & u_y & u \\ v_x & 1+v_y & v \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \Delta x \\ \Delta y \\ 1 \end{bmatrix} \tag{5}$$

where $\xi = [\Delta x,\ \Delta y,\ 1]^T$ is the homogeneous coordinates of the pixels in subset.

In each iteration, (x, y) is constant, so the gradients of the reference subset $\nabla f = \left[\frac{\partial f}{\partial x},\ \frac{\partial f}{\partial y}\right]$ are constant. In addition, the Hessian matrix [23] is fixed and can be pre-computed before iteration.