

光学学报

基于快速最小二乘法的两步随机相移算法研究

张宇^{1,2*}

¹东北电力大学理学院, 吉林 吉林 132012;

²中国科学院长春光学精密机械与物理研究所应用光学国家重点实验室, 吉林 长春 130022

摘要 在相移干涉测量中,为了在较短时间内实现较高的精度,提出了一系列基于快速最小二乘法的两步随机相移算法。以双滤波和差归一化算法、单滤波和差归一化算法和格兰-施密特正交化算法计算出来的相位作为迭代初始值,利用没有滤波的两幅相移干涉图进行最小二乘法运算以获取最终的相位,为了节省时间,只选取有限数量的像素来参与迭代运算。通过比较发现,基于单滤波和差归一化算法和快速最小二乘法的两步相移算法的综合性能最好,该算法在较短时间内能获得较高的准确度。

关键词 测量;干涉;相移算法;迭代算法;快速最小二乘法

中图分类号 O436

文献标志码 A

doi: 10.3788/AOS202141.0312003

Two-Step Random Phase Shifting Algorithms Based on Fast Least-Squares Method

Zhang Yu^{1,2*}

¹College of Science, Northeast Electric Power University, Jilin, Jilin 132012, China;

²State Key Laboratory of Applied Optics, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun, Jilin 130022, China

Abstract To achieve high accuracy with less computational time in the phase shifting interferometry, a series of two-step random phase shifting algorithms based on fast least-squares method are proposed. First, the phases extracted by the sum and difference normalization algorithm with double pre-filtering, the sum and difference normalization algorithm with single pre-filtering, and Gram-Schmidt orthonormalization method are used as the initial values for iteration. Then, the two phase shifted interferograms without filtering are used in the calculation with the least-squares method, and the final phase is obtained. In order to save time, only limited pixels are chosen to take part in the iteration. The comparison indicates that the two-step phase shifting algorithm based on the sum and difference normalization algorithm with single pre-filtering and fast least-squares method has the best comprehensive performance, and it can obtain high accuracy with less time.

Key words measurement; interferometry; phase shifting algorithm; iterative algorithm; fast least-squares algorithm

OCIS codes 120.3180; 120.2650; 120.3940; 120.5050

1 引言

随着光学制造技术的发展,光学检测技术面临

着前所未有的挑战。干涉仪作为一种简单、准确、快速的检测工具通常被用作检测标准^[1-3]。相移干涉法可以通过多幅干涉图轻松提取相位分布,在光学

收稿日期: 2020-07-30; 修回日期: 2020-08-21; 录用日期: 2020-09-10

基金项目: 国家自然科学基金(61905039)、吉林省科技发展计划项目(20190701018GH)、吉林省教育厅“十三五”科学技术项目(JJKH20190691KJ)

* E-mail: 521zhangyu2008@163.com

测量中得到了广泛的应用^[4-5]。除了干涉仪本身的误差和实验环境的影响,相移算法也会影响相移干涉仪的性能,一个优秀的相移算法应该具有精度高和省时的优点。随机相移算法对由压电陶瓷的标定误差、探测器误差、振动误差、空气扰动误差及激光器的不稳定误差等引起的相移误差不敏感,因此利用该算法可以提高精度。另外,如果随机相移算法易于实现,且只需要较少的相移干涉图,那么就可以实现省时。因此,随着光学在线检测技术的发展,快速随机相移算法的研究具有重要的意义。

近年来出现了许多随机相移算法,可分为迭代和非迭代相移算法。迭代相移算法具有较高的精度。2004年,Wang等^[6]提出了一种基于最小二乘法的先进迭代算法(AIA),该算法具有稳定的收敛性,并且能精确地提取相位。2008年,Xu等^[7]提出了一种可从随机的且空间不均匀的相移干涉图中提取相位分布的先进迭代算法。2013年,为了克服移相器产生的随机相移误差和倾斜误差,Chen等^[8]开发了基于最小二乘原理的迭代相移算法。一般来说,对于光学测量,尤其是在线检测,相移算法的即时性是非常重要的,而大多数迭代相移算法需要消耗很多时间。另外,上述迭代相移算法至少需要三幅干涉图来提取相位分布,对于迭代相移算法来说,干涉图越多,精度越高,但干涉图采集和计算的时间就越长,不适合在线检测。

近年来,科学家们对非迭代相移算法进行了广泛的研究。2016年,Liu等^[9]利用李萨如图和椭圆拟合同时提取了测试相位和两幅干涉图之间的相移值。从2011年到2017年,科学家们提出了一系列基于主成分分析(PCA)的相移算法,它将一组可能相关的变量转化为一组不相关的变量,但是该算法至少需要三幅干涉图来求解相位^[10-16]。2012年,Vargas等^[17]提出了两步格兰-施密特正交化算法(GS)。2014年,Wang等^[18]提出了一种三步格兰-施密特正交化算法(GS3)。2015年,Luo等^[19]提出了一种基于菱形对角线向量正交性的先进两步相移算法。同年,Niu等^[20]提出了一种基于相移干涉图内积商的两步相移算法。2016年,Xu等^[21]提出了三步欧几里得矩阵范数相移算法,相移的有效范围仅为 $0\sim\pi$ 。2018年,Cheng等^[22]介绍了一种快速准确的二步相移算法,但是只能得到相移值的余弦值,相移值的范围被限制在 $0\sim\pi$ 之间,所以这种算法不是真正的随机相移算法。非迭代相移算法花费很少的时间,但是精度不如迭代算法。尤其是上述

两步相移算法,一般需要通过预滤波来去除背景光强,这不仅花费了更多的时间,还会影响精度,使得它们不适用于高精度检测。

针对上述问题,本文提出了一系列基于快速最小二乘法(FLSA)的两步随机相移算法。以双滤波和差归一化算法(SD1)、单滤波和差归一化算法(SD2)和格兰-施密特正交化法(GS)计算出来的相位作为迭代初始值,利用没有滤波的两幅相移干涉图进行FLSA运算以获取高精度的相位。由于只选取有限数量的像素参与迭代运算,因此FLSA花费的时间较少。模拟和实验结果表明,基于单滤波和差归一化算法和最小二乘法(SD1&FLSA)的两步相移算法的综合性能最好,在较短的时间内获取了较高的精度,有较好的应用前景。

2 基本原理

总像素为 K 的两幅相移干涉图的光强可以表示为

$$\begin{cases} I_{1k} = a_{1k} + b_{1k} \cos(\varphi_k + \theta_1) \\ I_{2k} = a_{2k} + b_{2k} \cos(\varphi_k + \theta_2) \end{cases}, \quad (1)$$

式中: k 为像素位置, $k=1,2,\dots,K$; I_{1k} 和 I_{2k} 为两幅相移干涉图的强度; a_{1k} , a_{2k} , b_{1k} 和 b_{2k} 分别为两幅相移干涉图的背景强度和调制幅度; φ_k 为测试相位; θ_1 , θ_2 为相移值。 $\varphi_k + \theta_1$ 和 φ_k 之间只相差一个常数 θ_1 ,并不影响相位分布,因此为了简单起见,本文在下面的讨论中定义 $\theta_1=0$, $\theta_2=\theta$ 。

2.1 非迭代两步相移算法

从(1)式中可以发现,两步随机相移算法有6个未知数,分别是 a_{1k} , a_{2k} , b_{1k} , b_{2k} , φ_k 和 θ ,即使 $a_{1k}=a_{2k}$, $b_{1k}=b_{2k}$,至少还有4个未知数,因此至少需要4幅干涉图来提取相位。但是,相移干涉图越多,干涉图采集时间和相位计算时间就越多,且系统对环境越敏感。下面将设计一系列非迭代两步相移算法,并对众所周知的格兰-施密特正交化法进行简单的介绍。

2.1.1 双重滤波和差归一化算法

首先使用Hilbert-Huang滤波算法对两幅相移干涉图的背景强度进行滤波,并假设 $a_{1k}=a_{2k}=a_k$, $b_{1k}=b_{2k}=b_k$,滤波后得到

$$\begin{cases} \tilde{I}_{1k} = b_k \cos(\varphi_k) \\ \tilde{I}_{2k} = b_k \cos(\varphi_k + \theta) \end{cases}. \quad (2)$$

然后计算 \tilde{I}_{1k} 和 \tilde{I}_{2k} 的和、差为

$$\tilde{I}_k^{\text{sum}} = \tilde{I}_{1k} + \tilde{I}_{2k} = 2b_k \cos\left(\varphi_k + \frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right), \quad (3)$$

$$\tilde{\mathbf{I}}_k^{\text{dif}} = \tilde{\mathbf{I}}_{1k} - \tilde{\mathbf{I}}_{2k} = 2\mathbf{b}_k \sin\left(\varphi_k + \frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right). \quad (4)$$

由于两幅相移干涉图之间的相移并不总是等于 $\pi/2$ ，因此大部分情况下 $2\mathbf{b}_k \cos\left(\frac{\theta}{2}\right) \neq 2\mathbf{b}_k \sin\left(\frac{\theta}{2}\right)$ ，也就是说 $\tilde{\mathbf{I}}_k^{\text{sum}}$ 和 $\tilde{\mathbf{I}}_k^{\text{dif}}$ 的幅值不同。为了消除不同振幅的影响，对(3)、(4)式进行归一化处理。

向量 \mathbf{u} 的归一化一般可以表示为

$$\mathbf{u}^* = \mathbf{u} / \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle} = \mathbf{u} / \|\mathbf{u}\|, \quad (5)$$

式中： \mathbf{u}^* 表示归一化向量； $\|\cdot\|$ 和 $\langle \cdot, \cdot \rangle$ 分别表示 2 范数和内积。

通过归一化 $\tilde{\mathbf{I}}_k^{\text{sum}}$ 和 $\tilde{\mathbf{I}}_k^{\text{dif}}$ ，可以得到

$$\mathbf{I}_k^{*\text{sum}} = \frac{\tilde{\mathbf{I}}_k^{\text{sum}}}{\|\tilde{\mathbf{I}}_k^{\text{sum}}\|} = \frac{\mathbf{b}_k \cos\left(\varphi_k + \frac{\theta}{2}\right)}{\sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \cos^2\left(\varphi_k + \frac{\theta}{2}\right)}}, \quad (6)$$

$$\mathbf{I}_k^{*\text{dif}} = \frac{\tilde{\mathbf{I}}_k^{\text{dif}}}{\|\tilde{\mathbf{I}}_k^{\text{dif}}\|} = \frac{\mathbf{b}_k \sin\left(\varphi_k + \frac{\theta}{2}\right)}{\sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \sin^2\left(\varphi_k + \frac{\theta}{2}\right)}}, \quad (7)$$

式中： $\mathbf{I}_k^{*\text{sum}}$ 为 $\tilde{\mathbf{I}}_k^{\text{sum}}$ 归一化后的结果； $\mathbf{I}_k^{*\text{dif}}$ 为 $\tilde{\mathbf{I}}_k^{\text{dif}}$ 归一化后的结果。

如果干涉图中不止有一条条纹，则有

$$\sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \sin^2\left(\varphi_k + \frac{\theta}{2}\right)} \approx \sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \cos^2\left(\varphi_k + \frac{\theta}{2}\right)}, \quad (8)$$

上述归一化后的向量可以改写为

$$\mathbf{I}_k^{*\text{sum}} = \mathbf{c}_k \cos\left(\varphi_k + \frac{\theta}{2}\right), \quad (9)$$

$$\mathbf{I}_k^{*\text{dif}} = \mathbf{c}_k \sin\left(\varphi_k + \frac{\theta}{2}\right), \quad (10)$$

式中： $\mathbf{c}_k = \frac{\mathbf{b}_k}{\sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \cos^2\left(\varphi_k + \frac{\theta}{2}\right)}}$ 。

最后，根据(9)式和(10)式，可以得到

$$\varphi_k + \frac{\theta}{2} = \arctan\left(\frac{\mathbf{I}_k^{*\text{dif}}}{\mathbf{I}_k^{*\text{sum}}}\right). \quad (11)$$

φ_k 和 $\varphi_k + \frac{\theta}{2}$ 之间只有一个常数 $\frac{\theta}{2}$ ，不影响相位

分布，因此可以用 $\varphi_k + \frac{\theta}{2}$ 来表示测试相位。

2.1.2 单滤波和差归一化算法

两步相移算法的主要目标是节省时间，但是滤波过程会花费更多的时间，为了进一步节省时间，下面我们只采用一次滤波。

首先直接计算 \mathbf{I}_{1k} 和 \mathbf{I}_{2k} 的和、差为

$$\mathbf{I}_k^{\text{sum}} = \mathbf{I}_{1k} + \mathbf{I}_{2k} = 2\mathbf{a}_k + 2\mathbf{b}_k \cos\left(\varphi_k + \frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right), \quad (12)$$

$$\tilde{\mathbf{I}}_k^{\text{dif}} = \mathbf{I}_{1k} - \mathbf{I}_{2k} = 2\mathbf{b}_k \sin\left(\varphi_k + \frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right). \quad (13)$$

因为(13)式已经消除了背景强度，所以下面只对(12)式进行滤波，滤波后得到

$$\tilde{\mathbf{I}}_k^{\text{sum}} = 2\mathbf{b}_k \cos\left(\varphi_k + \frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right). \quad (14)$$

后续的相位计算过程与上面的 SD1 相同。

2.1.3 格兰-施密特正交化算法

此部分我们简单介绍 GS，它也是两步随机相移算法。

首先对两幅相移干涉图分别进行滤波，得到(2)式。

然后通过 GS 对两个向量 $\mathbf{u}_1, \mathbf{u}_2$ 进行正交归一化计算，共包含三个简单的步骤。

1) 取其中一个向量，对其进行归一化处理，即

$$\tilde{\mathbf{u}}_1 = \mathbf{u}_1 / \|\mathbf{u}_1\|. \quad (15)$$

2) 利用 $\tilde{\mathbf{u}}_1$ 对 \mathbf{u}_2 进行正交化处理，有

$$\hat{\mathbf{u}}_2 = \mathbf{u}_2 - \langle \mathbf{u}_2, \tilde{\mathbf{u}}_1 \rangle \cdot \tilde{\mathbf{u}}_1. \quad (16)$$

3) 对 $\hat{\mathbf{u}}_2$ 进行归一化处理，有

$$\tilde{\mathbf{u}}_2 = \hat{\mathbf{u}}_2 / \|\hat{\mathbf{u}}_2\|. \quad (17)$$

据 GS，对 $\tilde{\mathbf{I}}_{1k}$ 进行归一化处理，有

$$\tilde{\mathbf{I}}_{1k}^* = \frac{\tilde{\mathbf{I}}_{1k}}{\|\tilde{\mathbf{I}}_{1k}\|} = \frac{\mathbf{b}_k \cos(\varphi_k)}{\kappa_1} = \mathbf{b}_k \cos(\varphi_k) / \sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \cos^2(\varphi_k)}, \quad (18)$$

式中： $\kappa_1 = \sqrt{\sum_{k=1}^K \mathbf{b}_k^2 \cos^2(\varphi_k)}$ 。

然后对 $\tilde{\mathbf{I}}_{2k}$ 进行正交化处理，得

$$\tilde{\mathbf{I}}'_{2k} = \mathbf{b}_k \cos(\varphi_k + \theta) - \left[\sum_{k=1}^K \mathbf{b}_k^2 \cos(\varphi_k + \theta) \cos(\varphi_k) \right] \mathbf{b}_k \cos(\varphi_k) / \kappa_1^2. \quad (19)$$

如果干涉图中不止有一条条纹,则有

$$\left| \sum_{k=1}^K \cos^2(\varphi_k) \cos(\theta) \right| \gg \left| \sum_{k=1}^K \sin(\varphi_k) \cos(\varphi_k) \sin(\theta) \right|, \quad (20)$$

因此,可以将(19)式改写为

$$\tilde{I}'_{2k} = -b_k \sin(\varphi_k) \sin(\theta), \quad (21)$$

最后,对 \tilde{I}'_{2k} 进行归一化处理,有

$$\tilde{I}_{2k}^* = \frac{\tilde{I}'_{2k}}{\|\tilde{I}'_{2k}\|} = \frac{-b_k \sin(\varphi_k)}{\kappa_2} = -b_k \sin(\varphi_k) / \sqrt{\sum_{k=1}^K b_k^2 \sin^2(\varphi_k)}, \quad (22)$$

式中: $\kappa_2 = \sqrt{\sum_{k=1}^K b_k^2 \sin^2(\varphi_k)}$ 。

如果干涉图中不止有一条条纹, $\kappa_1 \approx \kappa_2$, 测试相位计算公式为

$$\varphi_k = \arctan(-\tilde{I}_{2k}^* / \tilde{I}_{1k}^*). \quad (23)$$

2.2 快速最小二乘法

对于大多数相移算法,速度和准确性是非常重要的,上面三种两步相移算法都能在较短时间内获取相位。但是,由于它们都是两步非迭代相移算法,因此很难实现高精度,并且滤波过程也会引入额外的误差。为了进一步提高精度,我们设计了快速最小二乘法,该算法能在较短的计算时间内获得较准

确的相位分布。下面详细介绍该算法。

1) 从(1)式中等间隔地选择有限数量的像素,所选像素组成的两个相移干涉信号可以表示为

$$I_{ij} = a_{ij} + b_{ij} \cos(\varphi_j + \theta_i), \quad (24)$$

式中: a_{ij} 为所选像素的背景强度; b_{ij} 为所选像素的调制幅度; φ_j 为所选像素的相位; θ_i 为相移值; i 为干涉图的序号 ($i=1, 2$); j 为所选像素的序号, $j=1, 2, \dots, N$, N 为所选像素总数。

2) 假设背景强度 a_{ij} 和调制幅度 b_{ij} 与 j 无关,只与 i 相关,则 $a_{i1} = a_{i2} = \dots = a_{iN} = a_i$, $b_{i1} = b_{i2} = \dots = b_{iN} = b_i$ 。令 $\eta_i = b_i \cos \theta_i$, $\xi_i = -b_i \sin \theta_i$, (24)式变为

$$I_{ij} = a_i + \eta_i \cos \varphi_j + \xi_i \sin \varphi_j. \quad (25)$$

相移干涉信号的理论值和实际值的差的平方和可以表示为

$$S_i = \sum_{j=1}^N (I_{ij}^t - I_{ij})^2 =$$

$$\sum_{j=1}^N (a_i + \eta_i \cos \varphi_j + \xi_i \sin \varphi_j - I_{ij})^2, \quad (26)$$

式中: I_{ij} 为通过实验得到的由所选像素构成的相移干涉信号的实际值。

根据最小二乘理论^[4-6],为了得到最小的 S_i ,对于已知的 φ_j ,有 $\partial S_i / \partial a_i = 0$, $\partial S_i / \partial \eta_i = 0$, $\partial S_i / \partial \xi_i = 0$,因此得到

$$\mathbf{X}_i = \mathbf{S}_i^{-1} \mathbf{R}_i, \quad (27)$$

$$\mathbf{S}_i = \begin{bmatrix} N & \sum_{j=1}^N \cos(\varphi_j) & \sum_{j=1}^N \sin(\varphi_j) \\ \sum_{j=1}^N \cos(\varphi_j) & \sum_{j=1}^N \cos^2(\varphi_j) & \sum_{j=1}^N \sin(\varphi_j) \cos(\varphi_j) \\ \sum_{j=1}^N \sin(\varphi_j) & \sum_{j=1}^N \sin(\varphi_j) \cos(\varphi_j) & \sum_{j=1}^N \sin^2(\varphi_j) \end{bmatrix}, \quad (28)$$

$$\mathbf{X}_i = [a_i \quad \eta_i \quad \xi_i]^T, \quad (29)$$

$$\mathbf{R}_i = \left[\sum_{j=1}^N I_{ij} \quad \sum_{j=1}^N I_{ij} \cos(\varphi_j) \quad \sum_{j=1}^N I_{ij} \sin(\varphi_j) \right]^T. \quad (30)$$

由(27)式可得 η_i 和 ξ_i , 并且可以计算出相移值:

$$\theta_i = \arctan\left(-\frac{\xi_i}{\eta_i}\right). \quad (31)$$

也可以从(27)式中提取背景强度 a_i 和调制幅度 $b_i = \sqrt{\eta_i^2 + \xi_i^2}$ 。

3) 计算相对相移 $\theta = \theta_2 - \theta_1$, 并令 $\theta_1 = 0$, $\theta_2 = \theta$, 然后利用步骤 2) 得到的背景强度 a_1 、 a_2 和调制幅度 b_1 、 b_2 来计算新的相位分布。

$$\begin{cases} \cos(\varphi_j) = \frac{I_{1j} - a_1}{b_1} \\ \sin(\varphi_j) = \frac{\left[\frac{b_2 I_{1j} \cos(\theta) - I_{2j}}{b_1} \right] - \left[\frac{b_2 a_1 \cos(\theta) - a_2}{b_1} \right]}{b_2 \sin(\theta)}, \end{cases} \quad (32)$$

$$\varphi_j = \arctan \left\{ \frac{\left[\frac{b_2 I_{1j} \cos(\theta) - I_{2j}}{b_1} \right] - \left[\frac{b_2 a_1 \cos(\theta) - a_2}{b_1} \right]}{(I_{1j} - a_1) b_2 \sin(\theta)} \right\}. \quad (33)$$

4) 重复步骤 2)、3), 直到 $E_{\text{RMS}}(\varphi^l - \varphi^{l-1}) < \epsilon$, 迭代结束, 同时得到相移计算值, 其中 φ^l 为第 l 次迭代计算出来的相位, E_{RMS} 为均方根值, ϵ 为预先设定的迭代收敛阈值, 例如, $\epsilon = 10^{-5}$ rad, l 表示迭代次数。

5) 利用提取的相移值和(1)式中的所有像素, 执行步骤 3), 可以得到准确的相位分布, 最后一步需要进行相位解包, 这是整个相位计算过程中唯一的一次解包过程。

在实际情况下, 由于光源的不稳定性, 不同干涉图之间的波动和不同像素之间的非均匀性都会影响背景强度和调制幅度的分布。使用 SD1、SD2 和 GS 提取相位时, 忽略了不同干涉图之间的波动, 但是 SD1、SD2 或 GS 提取的相位只被设置为迭代的初始值, 因此不会影响精度。而在 FLSA 中, 忽略不同像素之间的非均匀性将会影响精度。但对于所有两步相移算法, 由于未知数的数量超过方程的数量, 两步相移算法不能计算出每个像素的背景强度和调制幅度, 这也是两步相移算法的精度不是很高的原因。FLSA 采用了原始的没有经过滤波的两幅相移干涉图, 能够计算出不同干涉图的背景强度和调制幅度, 提高了普通两步相移算法的精度。在接下来的仿真中, 我们将讨论非均匀且波动的背景强度和调制幅度影响精度的规律。

虽然 FLSA 是一种迭代算法, 但由于只选取有限的像素参与迭代运算, 而且整个计算只使用一次

解包过程, 因此花费的时间较少。另外, 由于算法的迭代运算及没有滤波的相移干涉信号均参与了相位计算过程, 因此 FLSA 的精度较高。

3 模 拟

为了评价该算法的性能, 进行了一系列的数值模拟。所有计算均在 Intel(R) Core(TM) i7-6700 的中央处理器和 8GB 的内存下运行, 并使用 Matlab 软件进行运算。

首先, 在不同的情况下测试 SD1、SD2、GS、SD1&FLSA、SD2&FLSA 及 GS&FLSA 六种相移算法, 如完美的背景强度和调制幅度、波动的且非均匀的背景强度和调制幅度、波动的且非均匀的背景强度和调制幅度并同时干涉图中引入噪声。因为大部分相移算法对不同类型的干涉条纹都是有效的, 所以下面只讨论圆条纹。

图 1 为模拟相位分布图和两幅相移干涉图。假设测试相位为 $\varphi = 5\pi(x^2 + y^2)$, 其中 $x = \frac{X-200}{200}$, $y = \frac{Y-200}{200}$, (X, Y) 为像素位置, $0 \leq X \leq 400$, $0 \leq Y \leq 400$ 。图 1(a) 为理论相位分布。在情况 1 中, $|a_1| = |a_2| = 1$, $|b_1| = |b_2| = 1$ 。在情况 2 中, 背景强度和调制幅度分别设置为 $a_i(x, y) = N_a \exp[-0.02(x^2 + y^2)]$ 和 $b_i(x, y) = N_b \exp[-0.02(x^2 + y^2)]$, 其中 N_a 为背景强度系

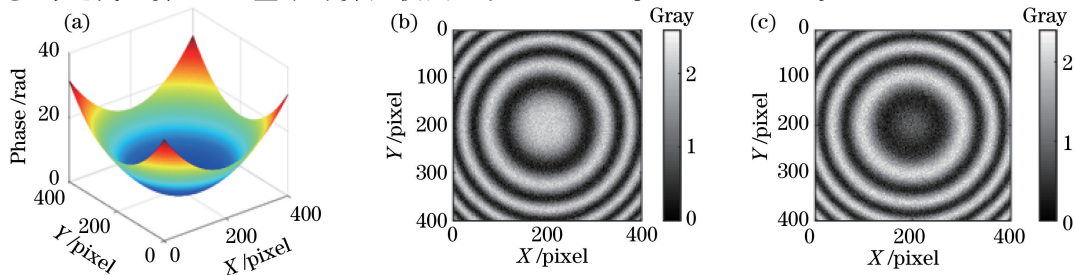


图 1 模拟相位分布图和两幅相移干涉图。(a) 理论相位分布图;(b)第一幅和(c)第二幅干涉图

Fig. 1 Simulated phase distribution and two phase shifted interferograms. (a) Theoretical phase distribution; (b) first and (c) second interferograms

数, N_b 为调制幅度系数。因为背景光强和调制幅度的波动和非均匀性同时存在, 所以第一幅和第二幅干涉图的 N_a 分别设置为 1.00 和 0.95, 第一幅和第二幅干涉图的 N_b 分别设置为 0.90 和 0.85。在情况 3 中, 将 Matlab 中的 Awgn 函数产生的信噪比为 20 dB 的噪声加入到情况 2 中。上述三种情况两幅相移干涉图之间的相移设置为 2 rad, 干涉图尺寸设置为 401 pixel \times 401 pixel。情况 3 的两幅模拟

相移干涉图如图 1(b)、(c) 所示。此外, 为了节省时间, SD1&FLSA、SD2&FLSA 和 GS&FLSA 仅均匀选择 41 \times 41 个像素参与迭代运算, 并且预定义迭代收敛阈值为 10^{-4} rad。

图 2 是三种不同情况下六种不同相移算法提取的相位分布图, 它们与图 1(a) 相似, 也就是说六种相移算法在不同情况下都是有效的。然而, 相位误差分布图是不同的, 如图 3 所示。由于滤波误差,

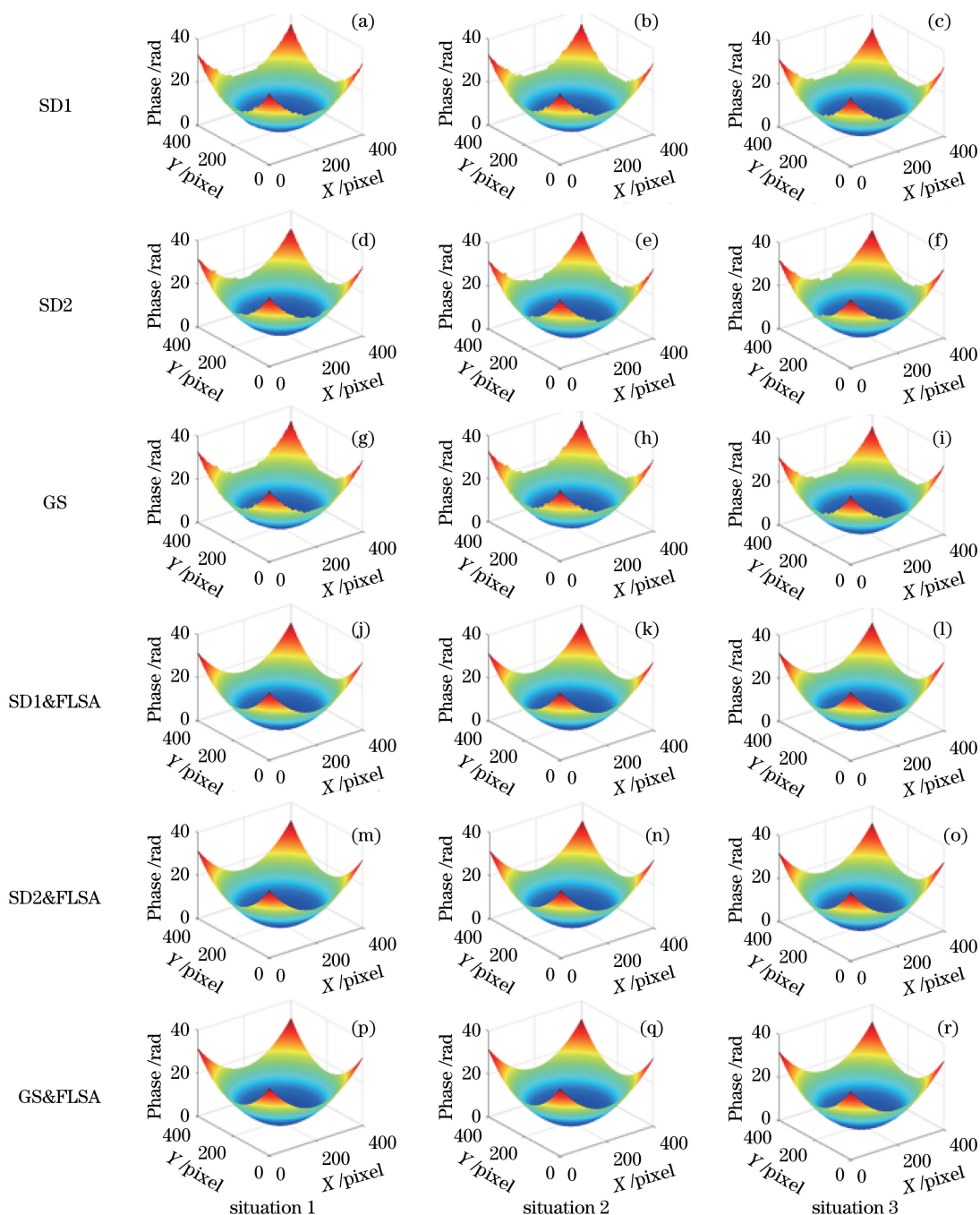


图 2 不同相移算法在三种不同情况下提取的相位分布图

Fig. 2 Phase distributions extracted by different phase shifting algorithms in three different situations

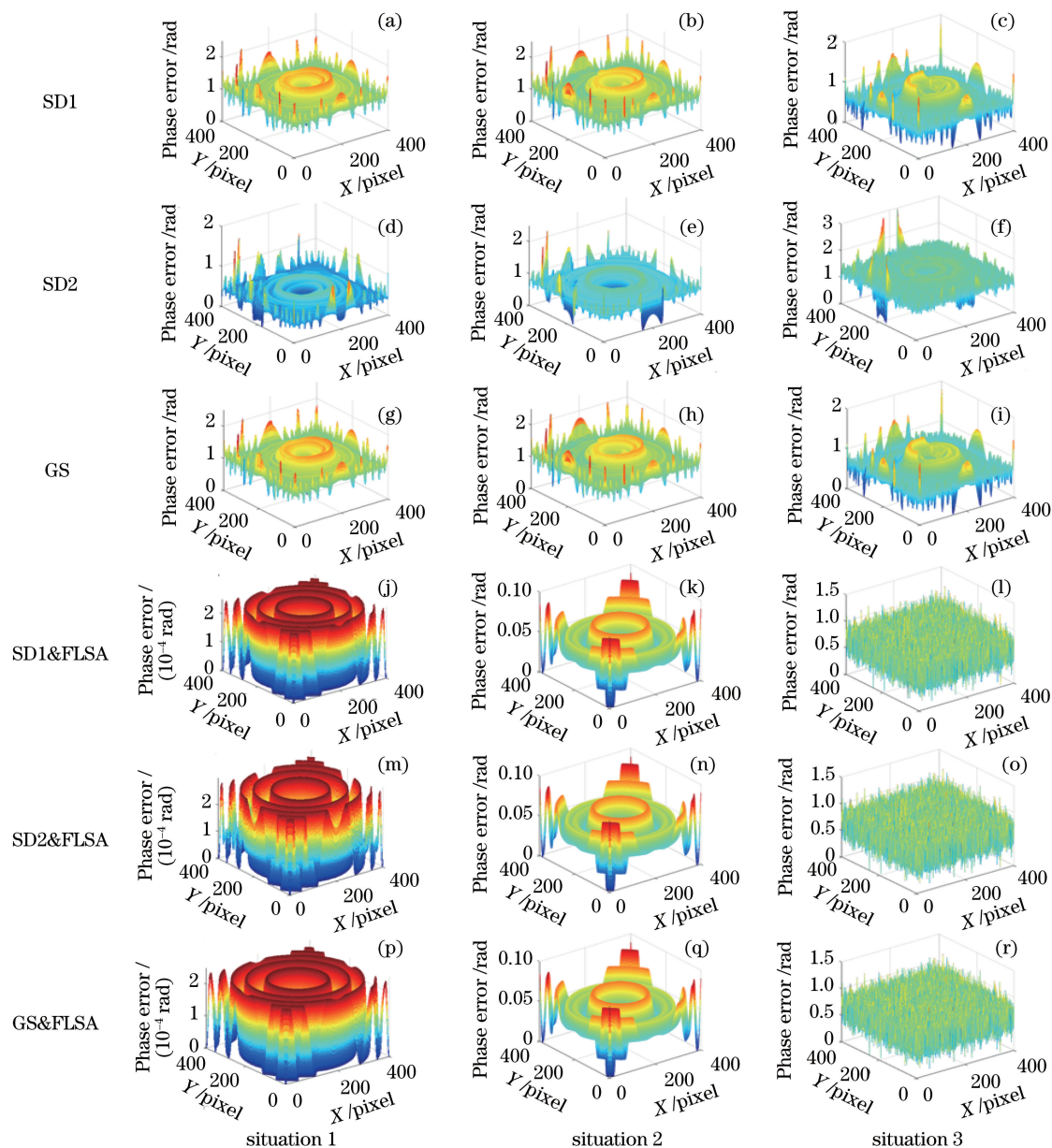


图 3 不同相移算法在三种不同情况下的相位误差分布图

Fig. 3 Phase error distributions of different phase shifting algorithms in three different situations

SD1、SD2 及 GS 的相位误差相对较大。对于 SD1&FLSA, SD2&FLSA 及 GS&FLSA, 由于 FLSA 的迭代运算及没有滤波的相移干涉图均参与到相位计算中, 这三种相移算法的相位误差相对较小。此外, SD1&FLSA、SD2&FLSA 及 GS&FLSA 在情况 3 中, 噪音是主要的误差, 因此相位误差分布类似于噪音分布。图 4 是 SD1&FLSA、SD2&FLSA 及 GS&FLSA 在三种不同情况下的迭代曲线图, 纵轴是两次迭代相位分布之差的均方根 (RMS) 值。可以看出, 对于三种情况下的第一次迭代, GS&FLSA 的两次迭代相位分布之差小于 SD1&FLSA 和 SD2&FLSA 的两次

迭代相位分布之差, 主要原因是 GS 提取的初始相位是 φ_k , SD1 和 SD2 提取的初始相位是 $\varphi_k + \frac{\theta}{2}$, 但是在执行完 FLSA 之后, 提取的相位均变为 φ_k 。虽然 SD1&FLSA 和 SD2&FLSA 的初始相位不是 φ_k , 但是迭代次数完全不受影响。从图 4 可以看出, 对于不同算法及不同情况, 迭代次数是相似的, 可以说迭代时间不受算法、初始相位及情况的影响。

图 5 展示的是六种不同算法在三种不同情况下的 RMS 相位误差和计算时间。

$$E = W' - W, \quad (34)$$

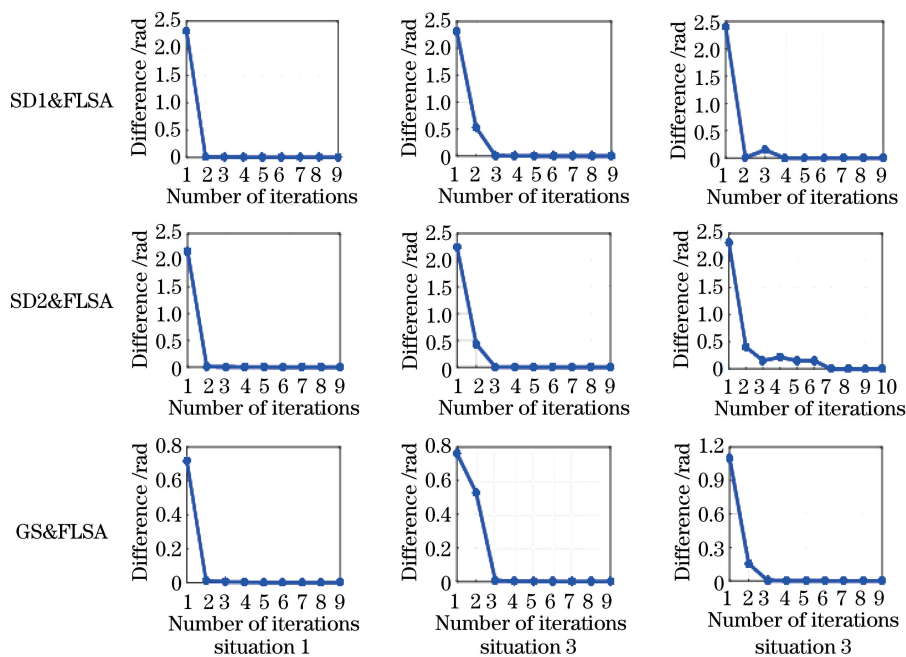


图 4 三种不同情况下不同相移算法的迭代曲线图

Fig. 4 Iterative curves of different phase shifting algorithms in three different situations

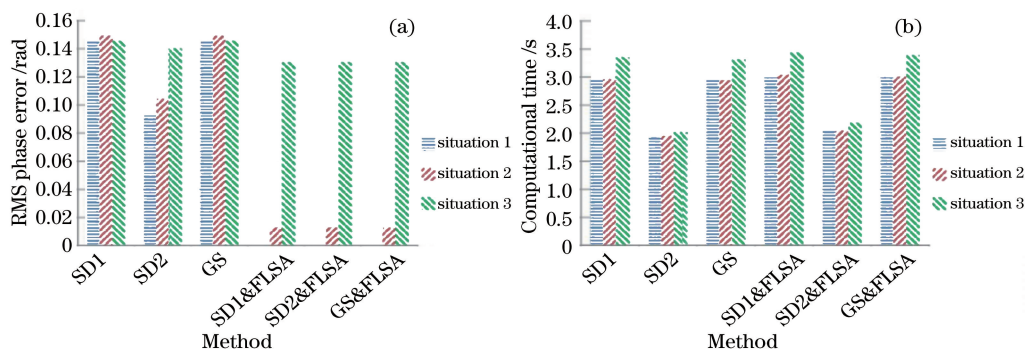


图 5 三种不同情况下不同算法的 RMS 相位误差和计算时间。(a) RMS 相位误差;(b)计算时间

Fig. 5 RMS phase errors and computational time of different algorithms in three different situations.

(a) RMS phase errors; (b) computational time

$$E_{\text{RMS}}(\Delta W) = \sqrt{\frac{1}{K}(\Delta W - \Delta \bar{W})^2}, \quad (35)$$

式中： W 为理论相位分布； W' 为计算相位分布； ΔW 为相位误差分布； $\Delta \bar{W}$ 为相位误差平均值； K 为像素总数； $E_{\text{RMS}}(\Delta W)$ 为 RMS 相位误差。

由于 SD1 和 GS 都执行过两次滤波，并且滤波误差是这两种算法的主要误差，因此在三种不同情况下这两种算法的 RMS 相位误差是类似的。由于 SD2 只使用过一次滤波，滤波误差相对较小，因此情况越复杂，RMS 相位误差越大，在任何情况下 SD2 的 RMS 相位误差都小于 SD1 和 GS 的 RMS 相位误差。另外，SD1、SD2 和 GS 提取的相位仅被当作为 SD1&FLSA、SD2&FLSA 和 GS&FLSA 的初始相位，而误差不会影响 FLSA 的精度，因此

SD1&FLSA、SD2&FLSA 和 GS&FLSA 的 RMS 相位误差小于 SD1、SD2 和 GS 的 RMS 相位误差。虽然 SD1&FLSA、SD2&FLSA、GS&FLSA 的初始相位不同，但是他们最终都使用 FLSA 提取相位，所以它们的 RMS 相位误差是相同的。在情况 1 中，SD1&FLSA、SD2&FLSA 和 GS&FLSA 可以提取绝对精确的相位分布；在情况 2 中，RMS 相位误差也不是非常大；只有在情况 3 中，当在干涉图中加入噪音后，它们的 RMS 相位误差相对较大，原因是这三种算法采用没有滤波的干涉图计算相位，因此如果需要高精度，应该抑制噪声。好的算法除了高精度外还需要省时。从图 5(b) 可以看出，在迭代运算过程中，SD1&FLSA、SD2&FLSA 和 GS&FLSA 花费的时间比 SD1、SD2 和 GS 多，但是由于仅有部分

像素参与迭代运算,FLSA 花费的时间少于 0.1 s,因此不用担忧 FLSA 会浪费时间。另外,因为 SD2 和 SD2&FLSA 只使用一次滤波,SD2 比 SD1 和 GS 花费的时间少,SD2&FLSA 也比 SD1&FLSA 和 GS&FLSA 花费的时间少。对于同样的方法,因为滤波噪声需要花费额外的时间,情况 3 比情况 1 和 2 花费的时间多。综合考虑,SD2&FLSA 是以上六种算法中最好的相移算法,因为它可以花费相对较少的时间就达到较高的精度。

通过以上比较和分析,可知 SD2&FLSA 的综合实力最强,下面采用不同的像素数运行 SD2&FLSA,从而比较情况 3 下 SD2&FLSA 的迭代次数、计算时间、RMS 相位误差和相移误差,仿真结果如图 6 所示,像素的数量分别为 6×6 、 11×11 、 21×21 、 41×41 、 81×81 、 101×101 、 201×201 和 401×401 。参与迭代运算的像素数量和迭代次数都会影响计算时间,从图 6(a)可以看出,如果选择 6×6 个像素参与迭代运算,由于参与运算的像素数太少,迭代次数相对较多,因此计算时间相对较长,如图 6(b)所示,在这种情况下很难计算出精确的相位。其他数量的像素参与迭代运算时,迭代次数是类似的。另外,图 6(b)中除了像素数量为 6×6 外,计算时间随着像素数量的增加而增加,且从 11×11

个像素增加到 101×101 个像素,计算时间相似,均相对较小,接近于 2 s。RMS 相位误差如图 6(c)所示,当参与迭代运算的像素数量从 6×6 增加到 41×41 时,相位误差随着选择像素的增加而减小,当选择像素数量大于 41×41 时,RMS 相位误差是稳定的。最后,如图 6(d)所示,当所选像素数量为 41×41 时,相移误差最小,其中相移误差为计算出的相移值与实际相移值的差。综合考虑计算时间、RMS 相位误差和相移误差,对于尺寸为 $401 \text{ pixel} \times 401 \text{ pixel}$ 的干涉图,最佳选择像素的数量为 41×41 ,此时可以获得较高的精度且花费较少的时间。为了判断其他尺寸的干涉图的最佳像素数量是否为 41×41 ,我们分析了尺寸为 $801 \text{ pixel} \times 801 \text{ pixel}$ 的干涉图,干涉图的其他模拟条件同上,模拟结果如图 7 所示。可以看出,当选择像素的数量大于 401×401 时,计算时间显著增加,当选择像素的数量大于 101×101 时,RMS 相位误差是稳定的且达到最小。综合考虑,对于尺寸为 $801 \text{ pixel} \times 801 \text{ pixel}$ 的干涉图,最佳选择像素的数量为 101×101 ,此时可以获得较高的精度且花费较少的时间。通过上面分析得知,对于不同尺寸的干涉图,最佳选择像素的数量是不同的,但是总像素数和最佳选择像素间是 10 倍左右的关系,可以以此作为实际最佳像素数量选择的依据。

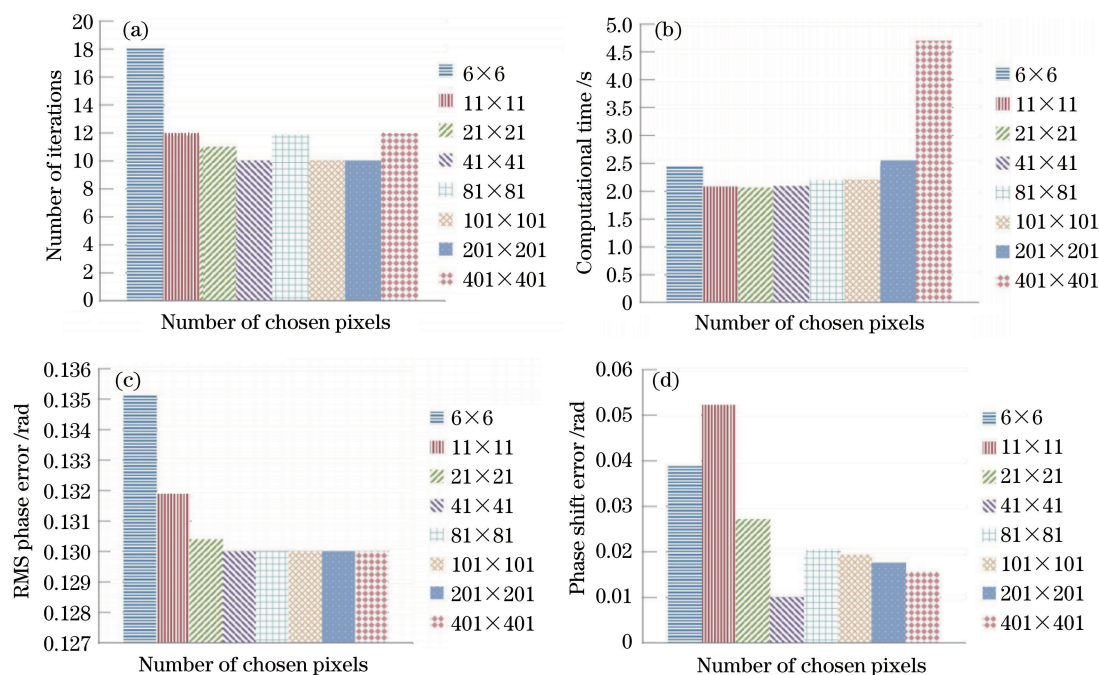


图 6 干涉图尺寸为 $401 \text{ pixel} \times 401 \text{ pixel}$ 时 SD2&FLSA 在不同选择像素数量下的仿真结果。

(a) 迭代次数; (b) 计算时间; (c) RMS 相位误差; (d) 相移误差

Fig. 6 Simulation results of SD2&FLSA with different numbers of chosen pixels when size of interferogram is $401 \text{ pixel} \times 401 \text{ pixel}$. (a) Number of iterations; (b) computational time; (c) RMS phase error; (c) phase shifting error

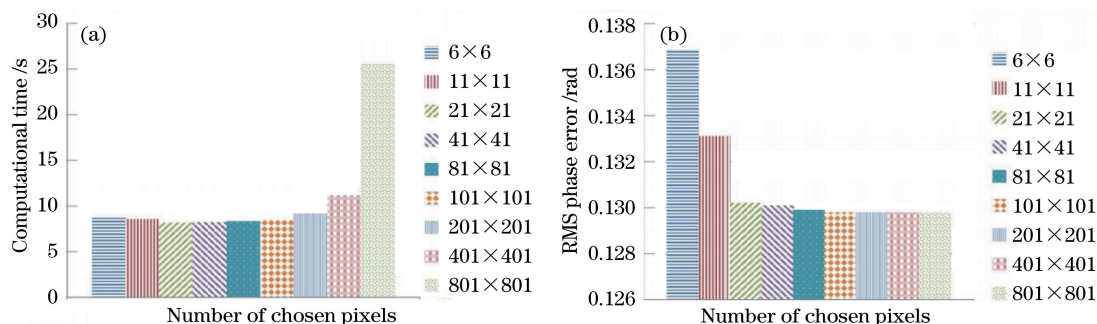


图 7 干涉图尺寸为 801 pixel×801 pixel 时 SD2&FLSA 在不同选择像素数量下的模拟结果。

(a) 计算时间; (b) RMS 相位误差

Fig. 7 Simulation results of SD2&FLSA with different numbers of chosen pixels when size of interferogram is 801 pixel×801 pixel. (a) Computational time; (b) RMS phase error

为了分析不同相移和不同噪声对 SD2&FLSA 的影响,下面计算 SD2&FLSA 在六种不同情况下的 RMS 相位误差,相移的范围为 0.1~3.1 rad,除了情况 1 和情况 2,我们还讨论了引入不同噪声的情况 3,噪声的信噪比分别是 20 dB, 30 dB, 40 dB, 50 dB,结果如图 8 所示。在情况 1 下,对于所有相移,RMS 相位误差都接近于 0,也就是说,当实验条件比较理想时,SD2&FLSA 可以得到绝对准确的结果。由于背景强度和调制幅度的波动和非均匀性,情况 2 下的 RMS 相位误差大于情况 1 下的 RMS 相位误差。对于不同噪声的情况 3 来说,由于 SD2&FLSA 使用原来没有滤波的相移干涉图来获取相位,并且 SD2&FLSA 不能抑制噪声,因此 RMS 相位误差随着噪声的增加而增大。另外,情况 3 下的 RMS 相位误差比情况 1 和 2 下的大,因此,可以得出,情况越复杂,RMS 相位误差越大。从图 8 还可以看出,情况越简单,相移值的范围越大。SD2&FLSA 对于情况 1 中的任何相移值都是可行的;对于情况 2 和加入 50 dB 和 40 dB 噪声的情况 3,当相移值为 3.1 rad 时,SD2&FLSA 是无效的;对于加入 30 dB 噪声的情况 3,当相移值为 0.1, 3.0, 3.1 rad,SD2&FLSA 是无效的;加入 20 dB 噪声的情况 3 是最复杂的情况,因此它的相移值的有效工作范围最小,为 0.5~2.6 rad。通过上面的分析可知,理论上 SD2&FLSA 对于不同的相移都是有效的,但是复杂的情况影响了相位提取尤其是当相移值接近于 0 和 π 时。最后,对于所有情况,相移值越远离 0 和 π ,RMS 相位误差越小。当相移值为 0 时, $|\mathbf{I}_k^{* \text{ dif}}|$ 等于 0,无论 $\mathbf{I}_k^{* \text{ sum}}$ 为多少,采用 SD2 计算出的相位值都是 0,而当相移值是 π 时, $|\mathbf{I}_k^{* \text{ sum}}|$ 等于 0,无论 $\mathbf{I}_k^{* \text{ dif}}$ 为多少,采用 SD2 计算出的相位值都是 $\pi/2$ 或者 $-\pi/2$,正负号取决于 $\mathbf{I}_k^{* \text{ dif}}$

的正负。这两种情况计算出来的相位值显然是不对的,由于 SD2 计算出来的完全错误的相位值被作为 FLISA 的迭代初始值,因此 FLISA 最终计算出来的相位值也是错误的。当相移值距离 0 和 π 比较近时, $|\mathbf{I}_k^{* \text{ dif}}|$ 和 $|\mathbf{I}_k^{* \text{ sum}}|$ 总有一项与 0 比较接近,这会导致较大的相位误差,只有当 $|\mathbf{I}_k^{* \text{ dif}}|$ 和 $|\mathbf{I}_k^{* \text{ sum}}|$ 都远离 0 时,才能计算出相对准确的相位值。因此,如果需要较高的精度,最好在使用 SD2&FLSA 之前抑制噪声,并且相移值最好远离 0 和 π 。在实际测量中,相移器实际移动的距离都是预先设定的,虽然存在的相移误差导致实际相移值和理论相移值不一致,但是随机相移算法可以避免相移误差对精度的影响,可以在进行相移前设置一个合理的理论随机相移值,从而实现更高的相位提取精度。

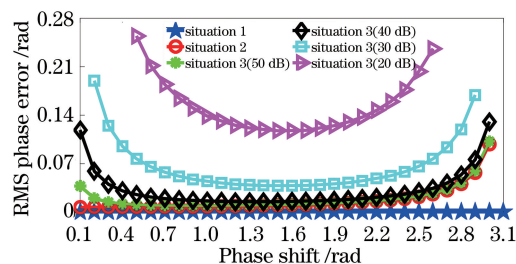


图 8 六种不同情况下具有不同相移的 SD2&FLSA 的 RMS 相位误差

Fig. 8 RMS phase errors of SD2&FLSA with different phase shifts in six different situations

4 实 验

接下来采用 SD1、SD2、GS、SD1&FLSA、SD2&FLSA 和 GS&FLSA 对实验数据进行相位提取。我们测量的是一个打磨抛光后的平滑表面,实验装置是一个同步相移干涉仪(SPSI)^[23],采用像素数量为 1208×1348、像素大小为 7.4 μm 的 12 位偏

振相机,采集四幅相移分别为 $0, \pi/2, \pi$ 和 $3\pi/2$ 的干涉图,干涉图尺寸为 $401 \text{ pixel} \times 401 \text{ pixel}$,选取 41×41 个像素参与迭代运算。由于待测相位分布未知,而当四幅相移干涉图之间的相移精确控制在 $\pi/2$ 时,标准四步相移算法^[4]的相位提取精度较高,因此设置其提取的相位分布为参考相位分布。图 9(a)、(b)为第一和第二幅相移干涉图,图 9(c)为参考相位分布图,其中参考相位峰谷(PV)值为 43.4713 rad ,参考相位 RMS 值为 8.9328 rad 。把所设计的相移算法提取的相位分布和参考相位分布之间的差异视为相位误差分布,六种不同算法的相位分布、相位误差分布及 SD1&FLSA、SD2&FLSA 和 GS&FLSA 的迭代曲线图如图 10 所示。可以看出,六种不同相移算法对实验数据都是有效的。表

1 是不同算法提取的相位分布的 PV 值和 RMS 值及算法的 RMS 相位误差和计算时间,通过相位分布的 PV 值和 RMS 值及 RMS 相位误差,可以看出,SD2 的精度比 SD1 和 GS 的精度高,SD1&FLSA、SD2&FLSA 和 GS&FLSA 的精度一样。实验数据的计算时间比模拟数据的计算时间稍长一点,虽然干涉图的大小都是 $401 \text{ pixel} \times 401 \text{ pixel}$,但是对于不同的背景强度和噪声,干涉图的滤波时间和迭代次数是不同的。此外,与模拟结果一样,SD2 花费的时间最少,SD2&FLSA 花费的时间比 SD2 稍多,FLSA 花费的时间不到 0.1 s 。通过以上实验结果,可以得到与模拟相同的结论,SD2&FLSA 在精度和计算时间方面的综合性能最好。

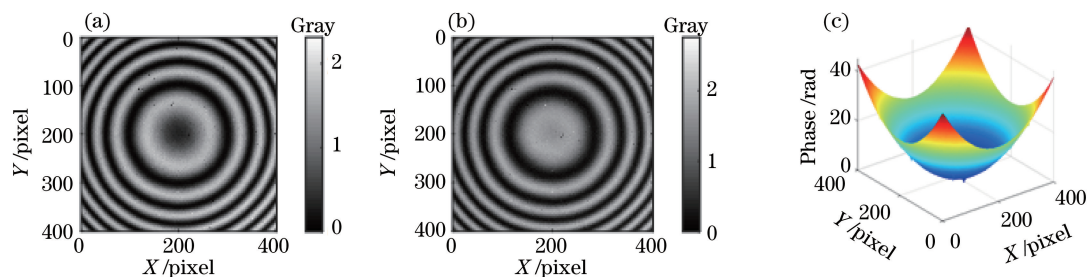


图 9 实验相移干涉图与相位分布图。(a)第一和(b)第二幅干涉图;(c)四步相移算法提取的参考相位分布图

Fig. 9 Experimental phase shifted interferograms and phase distribution. (a) First and (b) second interferograms; (c) reference phase distribution extracted by four-step phase shifting algorithm

表 1 不同算法提取的相位分布的 PV 值和 RMS 值及算法的 RMS 相位误差和计算时间

Table 1 PV and RMS values of phase distributions extracted by different algorithms as well as RMS phase errors and computational time of different algorithms

Parameter	SD1	SD2	GS	SD1&FLSA	SD2&FLSA	GS&FLSA
Phase PV value /rad	43.9181	44.0529	43.9180	43.8193	43.8193	43.8193
Phase RMS value /rad	8.8640	8.9178	8.8639	8.9339	8.9339	8.9339
RMS phase error /rad	0.1448	0.1434	0.1449	0.0810	0.0810	0.0810
Time /s	3.77	2.61	3.78	3.84	2.67	3.84

5 结 论

提出了一系列基于快速最小二乘法的两步随机相移算法。以 SD1、SD2 和 GS 计算出来的相位作为迭代初始值,利用没有滤波的两幅相移干涉图,通过 FLSA 提取了最终相位分布。由于只选取有限数量的像素参与迭代运算,因此 FLSA 花费的时间较少。通过仿真和实验数据,对 SD1、SD2、GS 与 SD1&FLSA、SD2&FLSA 和 GS&FLSA 进行了比较,发现 SD1&FLSA、SD2&FLSA 和 GS&FLSA 可以达到相同的精度,并且它们的精度比 SD1、SD2 和 GS 高。另外,由于 SD2 只使用了

一次滤波,它花费的时间最少,而 FLSA 花费的时间不到 0.1 s ,因此 SD2&FLSA 花费的时间比 SD2 稍多。综合考虑计算时间和精度,在六种算法中,SD2&FLSA 的性能最好。对于尺寸为 $401 \text{ pixel} \times 401 \text{ pixel}$ 的相移干涉图,最佳选择像素的数量为 41×41 ,此时可以获得较高的精度,同时节省计算时间。在越复杂的情况下,RMS 相位误差越大;在越简单的情况下,相移值的有效范围越大。此外,相移值越远离 0 和 π ,RMS 相位误差越小。因此,如果需要高精度,最好是在使用 SD2&FLSA 之前抑制噪声,并且相移值最好远离 0 和 π 。

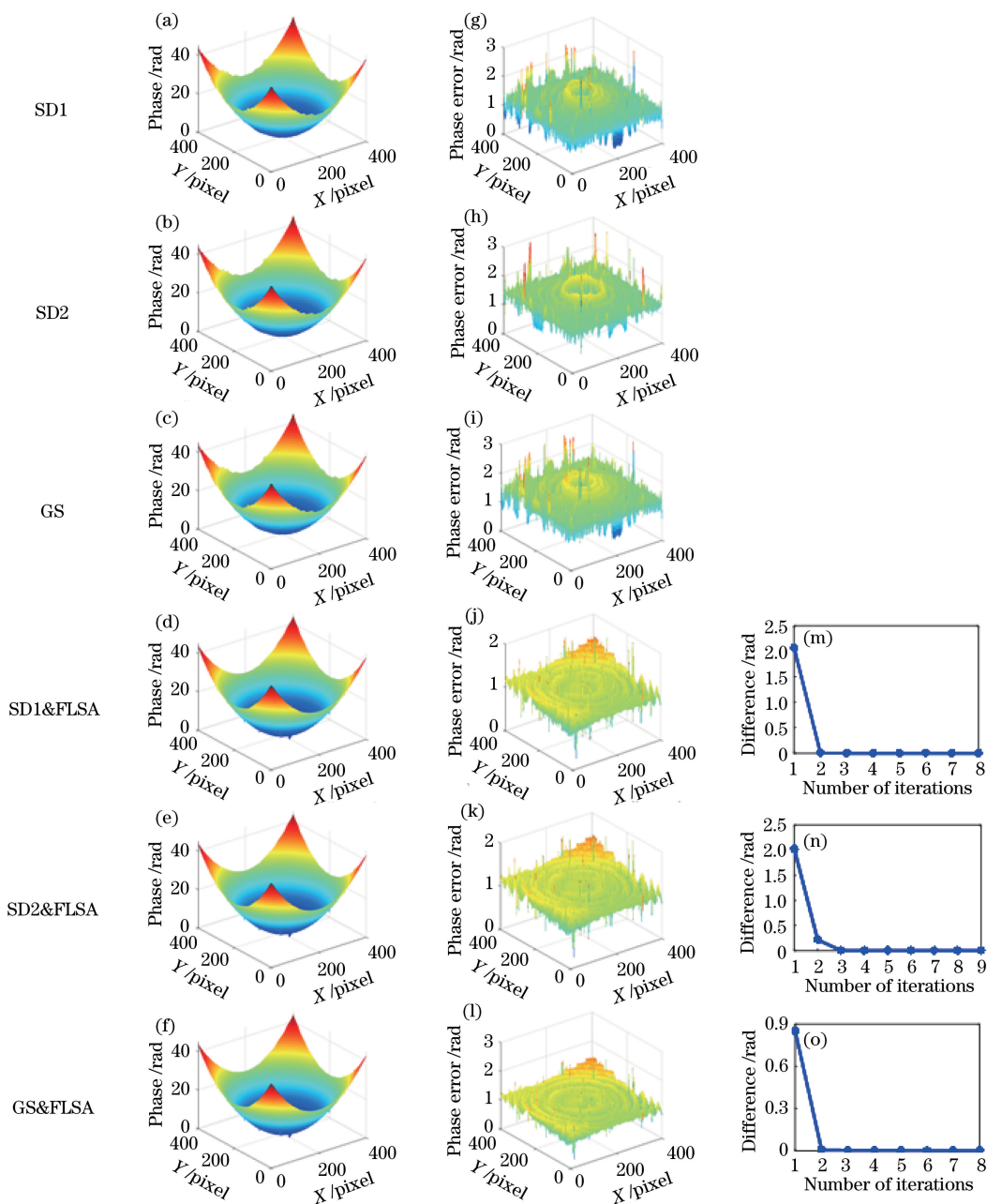


图 10 不同算法的实验结果。(a)~(f)相位分布;(g)~(l)相位误差分布;(m)~(o)迭代曲线

Fig. 10 Experimental results of different algorithms. (a)–(f) Phase distributions; (g)–(l) phase error distributions; (m)–(o) iterative curves

参 考 文 献

[1] Song M P, Yin C, Lu Y, et al. Four-channel detecting phase demodulation Φ -OTDR based on 3×3 Michelson interferometer [J]. Acta Optica Sinica, 2018, 38(8): 0806001.
宋牟平, 尹聪, 陆燕, 等. 基于 3×3 迈克尔孙干涉仪的四路检测相位解调 Φ -OTDR [J]. 光学学报, 2018, 38(8): 0806001.

[2] Li X L, Shen H, Li J, et al. Optical path difference calibration method of optical fiber array point source

generator in tilted-wave-interferometer [J]. Acta Optica Sinica, 2018, 38(5): 0512002.
李小柳, 沈华, 李嘉, 等. 倾斜波面干涉仪中光纤阵列型点源发生器的光程误差标定方法 [J]. 光学学报, 2018, 38(5): 0512002.

[3] Wu Z, Li Y, Xiangli B, et al. Full-field heterodyne long-cavity interferometry [J]. Acta Optica Sinica, 2019, 39(9): 0912003.
伍洲, 李杨, 相里斌, 等. 全视场外差长腔干涉测量技术 [J]. 光学学报, 2019, 39(9): 0912003.

[4] Malacara D. Optical shop testing [M]. Hoboken,

- NJ, USA: John Wiley & Sons, Inc., 2007.
- [5] Tian C, Liu S C. Two-frame phase-shifting interferometry for testing optical surfaces[J]. *Optics Express*, 2016, 24(16): 18695-18708.
- [6] Wang Z, Han B. Advanced iterative algorithm for phase extraction of randomly phase-shifted interferograms[J]. *Optics Letters*, 2004, 29(14): 1671-1673.
- [7] Xu J C, Xu Q, Chai L Q. Iterative algorithm for phase extraction from interferograms with random and spatially nonuniform phase shifts [J]. *Applied Optics*, 2008, 47(3): 480-485.
- [8] Chen Y C, Lin P C, Lee C M, et al. Iterative phase-shifting algorithm immune to random phase shifts and tilts[J]. *Applied Optics*, 2013, 52(14): 3381-3386.
- [9] Liu F W, Wang J, Wu Y Q, et al. Simultaneous extraction of phase and phase shift from two interferograms using Lissajous figure and ellipse fitting technology with Hilbert-Huang prefiltering [J]. *Journal of Optics*, 2016, 18(10): 105604.
- [10] Vargas J, Antonio Quiroga J, Belenguer T. Phase-shifting interferometry based on principal component analysis [J]. *Optics Letters*, 2011, 36(8): 1326-1328.
- [11] Vargas J, Quiroga J A, Belenguer T. Analysis of the principal component algorithm in phase-shifting interferometry[J]. *Optics Letters*, 2011, 36(12): 2215-2217.
- [12] Deng J, Wang K, Wu D, et al. Advanced principal component analysis method for phase reconstruction [J]. *Optics Express*, 2015, 23(9): 12222-12231.
- [13] Xu J C, Jin W M, Chai L Q, et al. Phase extraction from randomly phase-shifted interferograms by combining principal component analysis and least squares method[J]. *Optics Express*, 2011, 19(21): 20483-20492.
- [14] Yatabe K, Ishikawa K, Oikawa Y. Improving principal component analysis based phase extraction method for phase-shifting interferometry by integrating spatial information[J]. *Optics Express*, 2016, 24(20): 22881-22891.
- [15] Yatabe K, Ishikawa K, Oikawa Y. Simple, flexible, and accurate phase retrieval method for generalized phase-shifting interferometry [J]. *Journal of the Optical Society of America A*, 2017, 34(1): 87-96.
- [16] Yatabe K, Ishikawa K, Oikawa Y. Hyper ellipse fitting in subspace method for phase-shifting interferometry: practical implementation with automatic pixel selection[J]. *Optics Express*, 2017, 25(23): 29401-29416.
- [17] Vargas J, Antonio Quiroga J, Sorzano C O S, et al. Two-step demodulation based on the Gram-Schmidt orthonormalization method [J]. *Optics Letters*, 2012, 37(3): 443-445.
- [18] Wang H L, Luo C S, Zhong L Y, et al. Phase retrieval approach based on the normalized difference maps induced by three interferograms with unknown phase shifts[J]. *Optics Express*, 2014, 22(5): 5147-5154.
- [19] Luo C S, Zhong L Y, Sun P, et al. Two-step demodulation algorithm based on the orthogonality of diamond diagonal vectors [J]. *Applied Physics B*, 2015, 119(2): 387-391.
- [20] Niu W H, Zhong L Y, Sun P, et al. Two-step phase retrieval algorithm based on the quotient of inner products of phase-shifting interferograms[J]. *Journal of Optics*, 2015, 17(8): 085703.
- [21] Xu Y Y, Wang Y W, Ji Y, et al. Three-frame generalized phase-shifting interferometry by a Euclidean matrix norm algorithm [J]. *Optics and Lasers in Engineering*, 2016, 84: 89-95.
- [22] Cheng Z T, Liu D. Fast and accurate wavefront reconstruction in two-frame phase-shifting interferometry with unknown phase step[J]. *Optics Letters*, 2018, 43(13): 3033-3036.
- [23] Zhang Y, Tian X B, Liang R G. Fringe-print-through error analysis and correction in snapshot phase-shifting interference microscope [J]. *Optics Express*, 2017, 25(22): 26554-26566.