iFringe : a fringe analysis application for mobile smart devices

Qian, Kemao; Chan, Jacob; Teo, Josias Yi Si

2015

Chan, J., Teo, J. Y. S., & Qian, K. (2015). iFringe: a fringe analysis application for mobile smart devices. Proceedings of SPIE - International Conference on Experimental Mechanics 2014, 9302, 93022B-. doi:10.1117/12.2084787

<https://hdl.handle.net/10356/88767>

<https://doi.org/10.1117/12.2084787>

© 2015 Society of Photo-optical Instrumentation Engineers (SPIE). This paper was published in Proceedings of SPIE - International Conference on Experimental Mechanics 2014 and is made available as an electronic reprint (preprint) with permission of Society of Photo-optical Instrumentation Engineers (SPIE). The published version is available at: [<http://dx.doi.org/10.1117/12.2084787>]. One print or electronic copy may be made for personal use only. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper is prohibited and is subject to penalties under law.

Downloaded on 26 Apr 2023 21:33:11 SGT

iFringe: A Fringe Analysis Application for Mobile Smart Devices

Jacob Chan, Teo Yi Si Joasis, Qian Kemao

School of Computer Engineering, Nanyang Technological University, Singapore 639798

mkmqian@ntu.edu.sg

ABSTRACT

This paper introduces iFringe, a mobile application that attempts to incorporate the resource heavy fringe analysis algorithms into the smart mobile devices platform. This first step taken towards mobility in the optical processing field aims to become a catalyst for modernization of various aspects of the field as well as to diversify developments to other applications. Predominantly, the motivation of this work stems from the vastly indifferent human interactive method of mobile devices, which enable images displayed on its touch screen to be manipulated in ways that could enhance the fringe analysis experience. Furthermore, given its hardware compatibility to the conventional fringe projection system, these mobile devices could potentially serve as a much more compact replacement. However, one imperative weakness that mobile devices pose is its limited computing ability. Therefore, to examine the feasibility of incorporating the fringe analysis algorithms into a mobile platform, we have implemented two fundamental fringe analysis techniques, namely the Fourier transform fringe analysis method and the phase-shifting technique. Formulas and processing procedures such as discrete Fourier transform (DFT) and quality-guided phase unwrapping, were included in accordance to their original algorithms to test their performance and usability on a smart mobile device. Details of the implementation and the performance results will also be presented in this paper to demonstrate the practicality of these algorithms on the smart mobile device platform.

Keywords: Mobile, fringe analysis, fringe processing, Fourier transform, phase-shifting

1. INTRODUCTION

“Going mobile” is currently one of the biggest technological transitions in the digital revolution age. Millions of applications residing in mobile app-stores take advantage of its peripherals, compactness and portability to maximize the potential of these ever-evolving mobile devices. In this work, a new application particularly for fringe analysis (coined iFringe) was developed and published in the Apple Mobile App Store. It is the first mobile application (app) of its kind that is able to process fringe images in a sequential fashion, as well as enabling new fringe analysis experiences. The motivation for the app’s development is to explore new avenues for analyzing fringe images intuitively, while preserving its accuracy. Additionally, with its peripheral compactness, mobile devices also give rise to the possibility of replacing some of the cumbersome optical and computing components in a measurement system. Besides interfacing enhancements, the app also aims serve as a starting point for mobile-related research in the field by exhibiting the ability to implement fundamental fringe processing algorithms. From here, further research can be encouraged to incorporate advanced existing algorithms, together with studying new techniques and applications with the various integrated hardware that a mobile device brings.

Being the first mobile application to graze the optics field, the pivotal unknown factor for integrating fringe-processing algorithms into a mobile device is the processing capability. Therefore, to investigate the feasibility of incorporating the fringe pattern analysis algorithms [1] into a mobile device, two fundamental techniques were implemented, namely the Fourier transform method [2, 3] and phase-shifting technique [4, 5]. Both algorithms were realized into their respective processing sequences and split up into two different modes, with the resulting wrapped phase produced by both algorithms being unwrapped by the reliable quality-guided phase unwrapping algorithm [6-8]. Apart from fringe analysis, iFringe also emphasizes on the manipulation of images. Features such as zooming, rotation and cropping were all seamlessly incorporated into the application to give users an expressive image control experience that would be difficult to achieve on a conventional personal computer (PC). Navigation within the application was also kept simple with well-designed buttons and a scrolling carousel to display thumbnails of processed images. Lastly, interfacing images can be done easily as iFringe makes full use of the photos library, which provides various ways to photo share across different platforms.

iFringe was developed in the Apple iOS mobile platform using Objective-C for the general interface, and OpenCV [9] for the image processing algorithms. The hardware peripherals used for implementation were the MacBook Air 2013 [10] and the Apple iPhone 5s [11]. Further details regarding iOS mobile development can be referred from [12, 13].

The rest of this paper is organized as follows. Section 2 introduces iFringe’s general GUI and modes. Section 3 dives into the processing sequence for the Fourier transform method. Section 4 shows the processing sequence of the phase-shift technique. Section 5 discusses some of the performance issues with regards to running the algorithms on the mobiles devices. Section 6 concludes the paper.

2. INTERFACE DESIGN

The interface design of iFringe focuses on presenting the fringe processing sequences in a simple and intuitive manner. It consists of two main views, named “control center” (Fig. 1(a)) and “image viewer” (Fig. 1(b)), respectively. The control center as the name suggests, contains all of the controls of the application. It comprises of an image carousel containing thumbnails of the processed images for users to scroll through and select, as well as seven buttons for user controls. Button functionalities are summarized in Table 1. As for the image viewer, it is fundamentally designed to allow images to be viewed in full screen, and is launched primarily during image importation or when a user taps on a carousel thumbnail. Several image manipulation features such as cropping, zooming and rotation were also integrated to enhance analysis and control over the displayed image. Detailed usage of the image viewer will be described further in Section 3.

Figure 1. The two main interfacing views of iFringe. (a) The control center shows the button controls (bottom) and the carousel (middle) that contains image thumbnails for scrolling and selection. (b) The image viewer shows the full screen image, which can be manipulated by additional sub-controls (bottom right buttons).

Table 1. Buttons and their functions in the control center.

Icon	Button Name	Button Description
Camera	Opens the device’s camera to snap an image and imports the taken image into the application
Photos	Opens the device’s photos library and imports the selected images into the application
Fourier Processing	Starts a Fourier transform processing sequence. This button can be tapped at any time to restart a new sequence

Phase-shifting Processing	Starts a new phase-shifting processing sequence. This button can be tapped at any time to restart a new sequence
Re-crop image/ROI	Tap this button to re-crop an imported image or to re-select a new ROI in the Fourier processing mode
Save Image	Saves the current centered image displayed on the carousel to the device's photos library
Help	Displays information regarding button functions

3. THE FOURIER PROCESSING MODE

The Fourier processing mode is the default mode upon application launch and can be activated whenever the “Fourier Processing” button is tapped to start a new sequence. This mode’s primary purpose is to import a single sinusoidal fringe image and sequentially guide it through the Fourier transform fringe processing sequence.

3.1 Importing a Fringe Image

A fringe image importation can be done in two ways, via the camera or the photo album. Once loaded, the image viewer displays the image in full screen, shown in Fig. 2(a), and provides interactive zooming gesture functions such as double-tap and pinch-to-zoom. Additionally, it also comes with an integrated crop mode, which can be activated upon tapping the crop button. This mode provides a cropping bounding box for users to specify the cropping shape and area, as well as enabling multi-touch gestures outside the bounding box such as twist-to-rotate and pinch-to-zoom [14]. Two cropping options are currently available, the quadrilateral crop and ellipse crop. Once satisfied, users can confirm to add the cropped image onto the control center’s carousel. A sample cropping process is shown in Figures 2(b) and 2(c). Note that because of current resource constraints, images with dimensions larger than 640x480 will be resized accordingly with its perspective preserved [15].

3.2 Fourier Transform Fringe Pattern Processing

After the imported image has been received by the control center, a gray-scale image is automatically generated in preparation for Discrete Fourier Transform (DFT). The fringe pattern with noise that can be expressed as:

$$f(x, y) = r(x, y) \sum_{m=-\infty}^{\infty} b_m e^{j[2\pi m f_0 + m\varphi(x, y)]} + n(x, y) \quad (1)$$

where $f(x, y)$ is the fringe intensity; $r(x, y)$ can be understood as a general fringe amplitude; b_m is the amplitude of the m -th harmonic; f_0 is the carrier frequency; $\varphi(x, y)$ is the phase distribution relating to a physical quantity; $n(x, y)$ represents the noise; and j is the imaginary unit. Subsequently, a two-dimensional DFT is applied to obtain the spectra of Eq. (1) and placed as a thumbnail on the carousel. Figure 2(d) shows a generated 2-D magnitude spectrum in the image viewer.

Next, users will be tasked to conduct a manual filtration to select a Region of Interest (ROI) within the spectral image to emphasize the frequencies near the carrier frequency, and eliminate the other harmonics. At this point, the application launches the image viewer’s crop mode for the selection of the desired region. Similar editing functions mentioned in Section 3.1 are applied for this ROI selection process. Upon completing the ROI selection, an image of the selected spectral area is generated for the inverse DFT process. An elliptical spectral image filtration process is shown in Figures 2(e) and 2(f) as an example. Note that the selection is up to the users’ discretion. The application will continue performing the rest of the Fourier fringe processing sequence regardless of the correctness of the selection. Finally, with the assumption that the fundamental component had been selected, the inverse DFT operation is applied to the filtered image to obtain a complex image of:

$$\tilde{f}(x, y) = b_1 r(x, y) e^{j[2\pi f_0 + \varphi(x, y)]} \quad (2)$$

From the complex image, the application then computes and displays the magnitude and the wrapped phase image, with the latter having values ranging from $-\pi$ to $+\pi$, as shown in Figure 2(g). At this point, the application will halt its process, giving users control over the preceding phase unwrapping algorithm.

3.3 Quality-Guided Phase Unwrapping

Due to the complexity of phase unwrapping, a sequence halt is incorporated to allow users to decide whether the phase should be unwrapped. The unwrapping procedure uses the quality-guided phase unwrapping algorithm [6-8], which is a reliable but time-consuming process. The current version of the application implements the “trim” algorithm from Ghiglia’s book [6]. Once the unwrapping process begins, the algorithm uses the magnitude image from the inverse DFT result as the quality map to guide the unwrapping priority. Finally, upon completion the application displays the unwrapped image result on the carousel.

At this point, a total of eight thumbnails would have been generated throughout the entire Fourier process. Users can now scroll through the carousel to review the entire process, conduct in-depth image analysis using the image viewer, choose to restart a new process, re-crop the ROI to obtain another result, or save processed images to the photos library.

Figure 2. The Fourier processing sequence. (a) The imported image from photos library shown in the image viewer. (b) Zoom, rotate and crop the imported image. (c) The cropped image to be used for processing. (d) The DFT magnitude image. (e) Selection of the ROI using elliptical crop. (f) The filtered image after ROI selection. (g) The wrapped phase image after inverse DFT. (h) The unwrapped phase by quality-guided phase unwrapping.

4. THE PHASE-SHIFTING PROCESSING MODE

Another widely used technique in fringe analysis is the phase-shifting technique, where a sinusoidal fringe pattern is observed with shifted phases on a stream of images. In this mode, the three- and four- step algorithms have been implemented due to their wide applicability, and are activated upon tapping the “Phase-shifting Processing” button.

4.1 Importing the Phase-Shifted Fringe Images

During importation, users are first brought to the photos library to choose their desired images. Currently the application automatically limits the selection to three or four images. Once loaded, the application proceeds with the phase-shifting sequence according to the number of images nominated. Likewise, images with dimensions larger than 640x480 will be resized accordingly with its perspective preserved. Figure 3(a) to 3(d) shows the imported images in preparation for the four-step phase shifting algorithm sequence.

4.2 Phase-Shifting Fringe Pattern Processing

The phase-shift process is a much more straightforward process than its Fourier processing compatriot. Once importation is complete, images are immediately displayed on the carousel and then used to compute their respective wrapped phase. The fringe images are described as:

$$f_n(x, y) = a(x, y) + b(x, y) \cos[\varphi(x, y) + n\alpha] \quad (3)$$

where $f_n(x, y)$ is the n -th imported image; $a(x, y)$ is the background intensity; $b(x, y)$ is the fringe amplitude; $\varphi(x, y)$ is the phase to be extracted; and α is the phase-shift amount. For the three-step algorithm, Eq. (3) has values of $n = 0, 1, 2$, and $\alpha = 2\pi/3$, and the phase-shift algorithm used is as follows:

$$\varphi(x, y)_3 = \arctan \left[\frac{\sqrt{3}(I_0 - I_2)}{2I_1 - I_0 - I_2} \right]. \quad (4)$$

The four-step algorithm on the other hand, would correspondingly have values of $n = 0, 1, 2, 3$, and $\alpha = \pi/2$ for Eq. (3), while implementing the following the phase-shift algorithm:

$$\varphi(x, y)_4 = \arctan \left[\frac{I_3 - I_1}{I_0 - I_2} \right] \quad (5)$$

The wrapped phase, having 2π ambiguities, will then be displayed onto the carousel and then halted; just like in the Fourier processing mode, to await the user's input to start the phase unwrapping procedure. Lastly, upon activation, the exact algorithm and display procedures explained in Section 3.3 will be executed. Figure 3 demonstrates a sequence of the four-step phase-shift processing sequence with Fig. 3(e) showing the wrapped phase image after applying Eq.(5) and Fig. 3(f) showing the unwrapped phase.

Figure 3. The four-step phase-shift processing sequence. (a)-(d) Four imported images from the photos library. (e) The wrapped phase image. (f) The unwrapped phase by quality-guided phase unwrapping.

5. PERFORMANCE EVALUATION

This section presents the performance of the algorithms that were incorporated into iFringe. However, the focus here will be on the time complexity of the algorithms to test their feasibility on current mobile devices, as the image results had already been verified against the original algorithms on the PC during integration. Furthermore, due to the vastly indifferent coding and operating system (OS) environment of a mobile application, time complexity becomes increasingly important as changes in various data structures (e.g. queues converted to mutable arrays), and underlying threads from the user interface and the OS would have substantial impact on the overall runtime.

The time measurement experiments were conducted on the Apple iPad Mini (1st generation) [16] and the Apple iPhone 5s [11], with processors A5 and A7, respectively. The average timings of twenty test images were recorded in Table 2. All tested imported images used for the evaluation were all automatically resized to 640x480 when loaded into the application.

Table 2. Average time complexity result of various image-processing operations

Mobile Device (Processor)	DFT	Inverse DFT	atan2	Quality-Guided Phase-Unwrapping
iPad (1 st Generation) (A5 Processor)	150.906ms	146.515ms	499.631ms	98.213s
iPhone 5s (A7 Processor)	21.904ms	20.314ms	112.713ms	13.429s

From the results, it can be seen that image-processing functions such as DFT, inverse DFT and atan2 execute almost instantaneously on the A7 processor. On the contrary, the quality-guided phase unwrapping turned out to be quite time consuming, taking up an average time of 98 seconds on the lower end processor. However, it can be pointed out that the A7 processor's performance is almost 8 times of the older generation A5, clocking in at 13 seconds on average. Therefore, we can safely conclude that it is very possible to run these complex algorithms on today's modern mobile devices. (Note that during testing, average timing results deviate about $\pm 30\%$ due to the queuing/sorting structure of the "trim" unwrapping algorithm.)

6. CONCLUSION

The development of iFringe, a mobile application that enables users to analyze fringe patterns, was presented in this paper. We have shown the application's efficiency in executing two established fundamental fringe processing techniques on current devices; and at the same time, providing the portability, compactness and convenience of a mobile device. Additionally, we have shown that these devices' natural human computer interfacing approach also enables users to interact, manipulate and analyze images with greater control and precision. In terms of image streaming and sharing, iFringe would also be able to take advantage of the device's photo sharing capabilities such as using images directly from the Internet, emails or cloud servers for processing, and conversely, sharing processed images via these means as well. Therefore, with all the mentioned benefits that it brings, we are confident that iFringe would provide a new platform for many avenues of research in fringe analysis.

With the successful development of iFringe, many future works could be branched out from it to further increase its value and usability. These works include the incorporation of functions for fringe denoising and demodulation [17], accelerating existing algorithms such as the quality-guided phase unwrapping [18] in the application, enabling hardware device synchronization and calibration, and lastly, to provide auxiliary functions such as 3D plots of the results.

ACKNOWLEDGEMENT

This work was partially supported by Multi-plATform Game Innovation Centre (MAGIC), funded by the Singapore National Research Foundation under its IDM Futures Funding Initiative and administered by the Interactive & Digital Media Programme Office, Media Development Authority.

REFERENCES

- [1] Gorthi, S. S. and Rastogi, P., "Fringe projection techniques: Whither we are?," *Opt. Lasers Eng.* 48(2), 133-140 (2010).
- [2] Takeda, M., Ina, H. and Kobayashi, S., "Fourier-transform method of fringe-pattern analysis for computer-based topography and interferometry," *J. Opt. Soc. Am.* 72(1), 156-160 (1982).
- [3] Su, X. and Chen, W., "Fourier transform profilometry: a review," *Opt. Lasers Eng.* 35(5), 263-284(2001).
- [4] Creath, K., "Phase-measurement interferometry techniques," *Progress in Optics XXVI*, 349-393 (1988).
- [5] Zhang, S., "Recent progresses on real-time 3D shape measurement using digital fringe projection techniques," *Opt. Lasers Eng.* 48(2), 149-158 (2010).
- [6] Ghiglia, D. C. and Pritt, M. D., [Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software], John Wiley & Sons, New York (1998).
- [7] X. Su and W. Chen, "Reliability-guided phase unwrapping algorithm: a review," *Optics and Lasers in Engineering*, vol. 42, no. 3, pp. 245-261, Sep. 2004.
- [8] Kemao, Q., Gao, W. and Wang, H., "Windowed Fourier-filtered and quality-guided phase-unwrapping algorithm," *App. Opt.* 47(29), 5420-5428 (2008).
- [9] OpenCV, [http:// http://opencv.org/](http://opencv.org/) (accessed July 2014).
- [10] MacBook Air., <http://store.apple.com/us/buy-mac/macbook-air> (accessed July 2014).
- [11] iPhone 5s., <https://www.apple.com/us/iphone-5s/specs/> (accessed July 2014).
- [12] Start Developing iOS Apps Today., <https://developer.apple.com/library/iOS/referencelibrary/GettingStarted/RoadMapiOS/index.html> (accessed July 2014)
- [13] OpenCV iOS., http://docs.opencv.org/doc/tutorials/ios/table_of_content_ios/table_of_content_ios.html (accessed July 2014)
- [14] Multi-Touch, <http://en.wikipedia.org/wiki/Multi-touch> (accessed July 2014).
- [15] Geometric Image Transformations, http://docs.opencv.org/modules/imgproc/doc/geometric_transformations.html (accessed July 2014).
- [16] iPad Mini., <https://www.apple.com/sG/ipad-mini-first-generation/specs/> (accessed July 2014).
- [17] Kemao, Q., [Windowed Fringe Pattern Analysis], SPIE Press, Bellingham (2013).
- [18] Zhao M. and Kemao, Q., "Quality-guided phase unwrapping implementation: an improved index interwoven linked list," *Appl. Opt.* 53(16), 3492-3500 (2014).