第19回性能に基づく橋梁等の耐震設計に関する
シンポジウム講演論文集（2016年7月）

Image Processing Based Real-Time Dynamic Displacement Monitoring Methods Using Smart Devices

Ashish SHRESTHA[1]・Ji DANG[2]・Xin WANG[3]

[1]Student Member of JSCE, Graduate Student, Department of Civil and Environmental Engineering, Saitama University（〒338-8570, 255-Shimo Okubo, Sakura-Ku, Saitama）
[2]Regular Member of JSCE, Ph.D., Assistant Professor, Department of Civil and Environmental Engineering, Saitama University（〒338-8570, 255-Shimo Okubo, Sakura-Ku, Saitama）
[3]Non Member, Ph.D., Assistant Professor, International Research Institute of Disaster Science, Tohoku University（〒980-0845, 468-1 Aramaki-Aoba, Aoba, Sendai City）

1. INTRODUCTION

During the 2011 Great East Japan Earthquake, few rubber bearings and dampers of some bridges designed by post-1995 code were found to be broken as first time in highway bridges[1)]. Though, numerical simulation of both structural seismic response analysis and ground motion simulation have been conducted for those bridges, it was very difficult to predict the behavior of both ground shake and structural response due to lack of instrument measured data. Therefore, it is realized that without such a record, neither damage and behavior of structures during strong earthquakes can be compared to the seismic design criteria nor proper decisions concerning rational repair and reconstruction could be made. However, traditional displacement measurement system, combined with displacement transducers, data log, hard disk, computer, power supply, modem and network connection, is high cost and difficult for wide application in structures.

Meanwhile, having significant computational power, large memory resources and wide functionality, modern smart devices could be realized to be one of the possible methods to measure structural vibration. Smart devices, with their on-board computational and communication capabilities, improvements in built-in sensors and easy to offline or online programmable functionality, simplifies the collection of information about existing infrastructures and thus offer new opportunities in the field of seismic and structural response measurement with extremely low initial and maintenance cost. Built-in camera of smart devices is one of the most developed devices providing higher resolution and higher speed video features. Their powerful processors and memory allows on-board processing capabilities, eliminating the need for additional computers to perform extensive image processing. However, such advanced vision and embedded processing capabilities of smart devices have not been effectively utilized for dynamic displacement monitoring applications yet.

Indrawan et al[2)] studied the effectiveness of implementing real time image and video processing on mobile devices. In the study some of the emerging image and video-processing algorithms used on the smart devices such as face detection and augmented reality are highlighted and the challenges for implementing them are described. Min et al[3)] developed smartphone application to measure absolute dynamic displacement in real time using state-of-the-art smartphone technologies; such as high-performance graphics processing unit (GPU), embedded high-resolution camera and open source computer vision libraries. Indoor and outdoor testing of the measurement application were conducted using shaking table and the results were compared with the conventional laser displacement sensor.

Most of these past researches involved in using smart devices for displacement measurement, however, provides less information about the reliability and range of using such devices. The frequency and amplitude domain still needs to be clearly identified. Therefore, this study attempts to illustrate the effectiveness of the proposed approach in more detail by utilizing such advanced vision and embedded processing capabilities of smart devices for dynamic displacement monitoring applications, which is very unique and most advanced method in this area. Real-time structural displacement measurement methods by applying digital image processing techniques and using built-in camera of smart devices have been developed. The effectiveness of measurement using smart devices in more detail is also verified by clarifying the reliable domain for frequency and amplitude measurement by performing shaking table tests for sine wave loading using different smart devices and the advantages and limitations of the proposed methods have been summarized.

2. MEASUREMENT APPLICATION DEVELOPMENT FOR SMART DEVICES

Smart device based iOS application has been developed for real time measure of dynamic displacement using the rear camera of smart device and three most common image processing methods: Motion Detection, Corner Tracking and QR Code Tracking which enables easy and low cost monitoring of absolute dynamic displacements. The displacement measuring application program performs the task of absolute displacement measuring, recording, storing and data synchronization with the cloud server via Dropbox[4]. The object of interest detected in video sequence is tracked by implementing suitable image processing algorithm and with difference in the coordinate of target object in respective frames displacement measurement is obtained.

(1) Application Development Environment

Application programs were developed based on programming language of Objective-C and developing environment Xcode for iOS application development. An open source iOS framework library known as GPUImage library[5] was used that applies GPU-accelerated filters and other effects to images, live camera video, and movies. The combination of using these filters and processing them on GPU allowed complex image analysis algorithms to run at much higher speed.

(2) Application User Interface and Proposed Methods

Three different object tracking methods described below have been explored within the image-processing

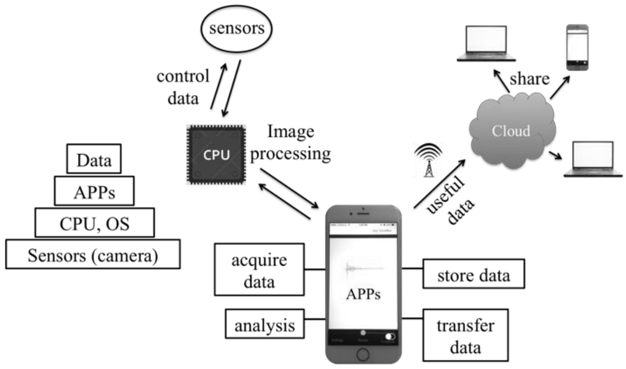

Figure 1. Measurement Application Program Development using Smart Devices

environment in order to effectively detect the target of interest and track its centroid movements at each frame to obtain displacement response of the moving target. This started with examining the filters included in the GPUImage library. Each filter in GPUImage process the incoming image and produces a resulting image that can be extracted from the GPU and presented to the user.

a) Motion Detection Method

The motion detection and tracking is actually the process of keeping tracks of the moving object in video sequence i.e. position of moving object at certain time etc. Motion tracking method try to estimate the displacement and velocity of features in a given video frame with respect to the previous one. It does frame-to-frame comparisons, based on a low-pass filter, and can identify the number of pixels that have changed between frames and the centroid of the changed area. Because it relies on pixel differences and not optical flow or feature matching, it can be prone to inaccurate tracking of desired object as they move in a frame.

b) Corner Tracking Method

The corner tracking method applied in this study is basically an improvisation to motion detection method for obtaining displacement measurement. In case of motion detection method, the movement of all the objects in a video sequence was considered due to which tracking the movement of a specific target object in the scene become difficult. However, tracking only the meaningful features of the video for example the corners allow a robust solution for accurately obtaining the movement of a desired feature target. In this study, Harris corner detection[6] and tracking algorithm have been implemented in the application program to track the motion of detected corners in real time.

c) QR Code Tracking Method

The corner-tracking algorithm applied in this study localizes only the most significant corner in the scene, and such set up to localize only the desired corner target is generally possible only in the controlled environment. However, in real field applications, using the corner tracking method to obtain displacement response is not always possible. There may be several other corner features apart from the installed target that are more significant due to which identification and localization of desired feature becomes difficult. Therefore in this study, for more practical applications, one more possibility of object tracking using QR code has been introduced.

A QR code is actually a two-dimensional (2D) barcode that consists of black modules (square dots) arranged in a square grid on a white background and contains a variety of encoded information such as URL, phone number, simple texts, etc. However, in this study any encoded information stored in the QR code is not used. Instead, the QR code is used as a target feature, which is first of all identified, by using smart devices and then tracked for its movement to obtain displacement response.

(3) Displacement Measure

Figure 2. shows the flowchart for displacement measuring application using the above proposed methods. With detection of moving objects in the scene on every incoming frame, different filters implemented under the GPUImage framework gives the centroid of movement (in normalized X, Y coordinates). From the next frame onwards the movement of centroid in terms of co-ordinate geometry is tracked, as pixels positions in an image can be treated as a 2D graph. The new centroid position is recorded. This process continues until the object disappears from image frame or if user stops the recording. After that, the total displacement of the object is obtained. However, this displacement is only in the image domain in pixel unit and after multiplying with a suitable scale factor real displacement measurement is obtained.

(4) Scaling Factor Determination

In order to obtain real structural displacements from the captured video images, relationship between the pixels coordinates and the physical coordinate was established (e.g., in units of mm/pixel). Figure 3. illustrates the actual distance calibration method using smart devices. As shown in Figure 3, when the image plane is parallel to the

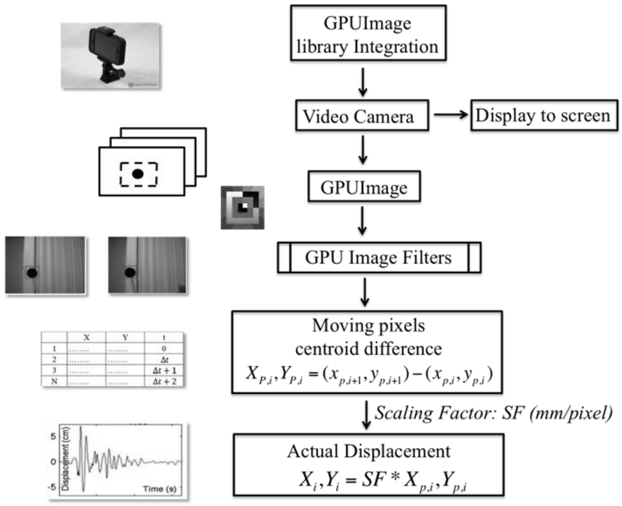

Figure 2. Displacement Measurement Application Flowchart

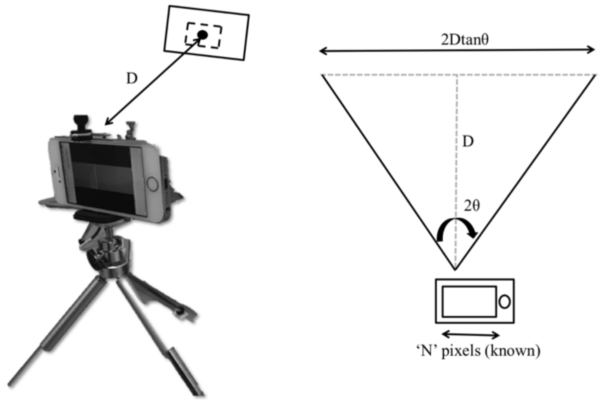

Figure 3. Actual Distance calibration using smart devices

object surface, the scaling factor in the translational direction (x-axis) can be determined by applying simple trigonometry geometry.

"N pixels" in device view corresponds to $2Dtan\theta$ in real view. Therefore, scaling factor (SF) is given by:

$$SF(1\,pixel) = \frac{2Dtan\theta}{N} \qquad (1)$$

D is the distance between target object and camera

2θ is the Field of View (FOV) of camera device (standard value for each device as shown in Table 1)

(5) Sensor System and Specification

The displacement measuring sensor system used in this study, consist of a high quality laser displacement sensor as a reference sensor and three other widely used brands and generation of iOS smart devices with different processing capabilities and camera resolution. The basic specification of each of these smart devices is shown in Table 1. Three different brands and generation of iOS smart devices were used for the study. In particular, the focus of this study is related with the smart device camera

Table 1. Hardware and camera performance of smart devices

Property	iPod Touch	iPhone 5s	iPhone 6
Release Date	10/11/2012	9/20/2013	9/19/2014
CPU	Apple A5: 800 MHz dual-core	Apple A7: 64-bit 1.3 GHz dual-core	Apple A8: 64-bit 1.4 GHz dual-core
Memory	512 MB LPDDR2 DRAM	1 GB LPDDR3 RAM	1 GB LPDDR3 RAM
Rear Camera	- 5MP iSight camera - Autofocus - f/2.4 Aperture - LED flash	- 8MP iSight camera with 1.5µ pixels - Autofocus - f/2.2 Aperture - True tone flash	- 8MP iSight camera with 1.5µ pixels - Autofocus - f/2.2 Aperture - True tone flash
HD Rear Camera Capture	-1080p@ 30fps	-720p@ 30/60/120 1080@30fps	-720p@ 30/60/120/240 -1080@30/60fps
GPU	SGX543MP2 (2-core)	Power VR G6430 (4 cluster @ 450 MHz)	Power VR Series 6 GX6450 (4 clusters)
FOV	54.40002°	58.040001°	58.080002°

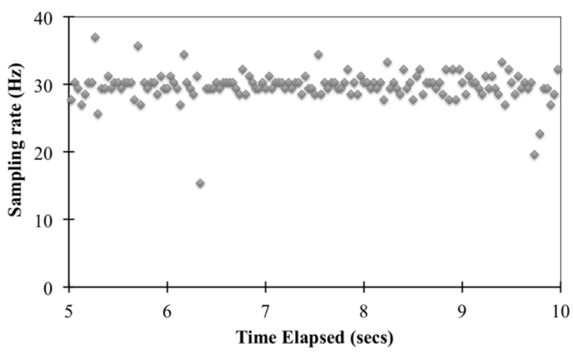

Figure 4. Variation in sampling rate (fps) for iPhone 5s smart device

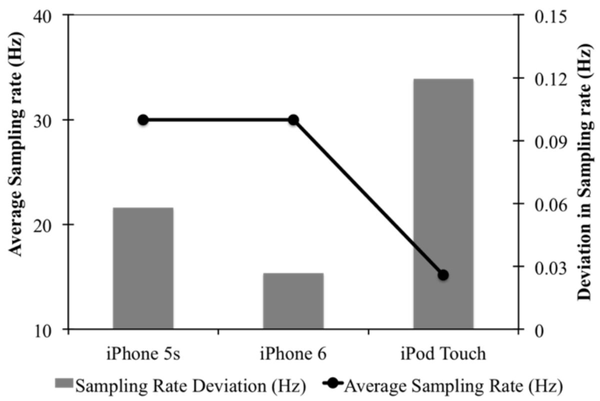

Figure 5. Sampling Time Accuracy in GPU Implementation

and on-board processing performance. For example, the older model of iOS smart device such as iPod Touch supports up to 30 fps at 1080p resolution while newer model such as iPhone 6 supports up to 60 fps at 1080p resolution. Moreover, the integrated graphic processing unit (GPU) can rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display. These smart devices can measure the displacement time history data in two orthogonal directions (in the plane of sight of camera). Using these smart devices a maximum sampling frequency of 30 fps was set in the application program, which is generally sufficient for most of the engineering applications such as ground motion displacement measurement or long period vibration measurements.

(6) Sampling Time Accuracy

Figure 4. shows the example record of the sampling rate for iPhone 5s set to 30 fps. Over the entire measurement little inconsistencies are observed, as sometimes the actual sampling rate goes higher while sometimes it goes lower than the standard set fps. This may be due to the consequence of multitasking system that image frame output data are not sampled at exactly equal intervals. Nevertheless, such inconsistencies are minimum, and when we correct the sampling frequency by taking the average value of actual sampling rate measured by the device, it does not greatly affect the measurement as can be seen from Figure 5. It is observed that iPhone 5s and iPhone 6 have a constant average sampling frequency of around 30 fps with very small deviation (in the order of less than 0.2%). However, due to low GPU capabilities of iPod Touch device, it is observed that output data are sampled at a constant average frequency of around 15 fps but with small deviation (in the order of less than 0.8%). This suggests that the sampling time accuracy offered by smart devices is quite consistent and reliable for measurement purpose.

3. EXPERIMENTAL VERIFICATION USING SHAKING TABLE TESTS

In order to identify the reliability and range of using smart devices for image process based displacement measurement, laboratory tests were carried out using shaking table. Uniaxial shaking table (APS-113) was used to simulate sinusoidal oscillation. The SVA-ST-30 amplifier amplified the input excitation of the shaking

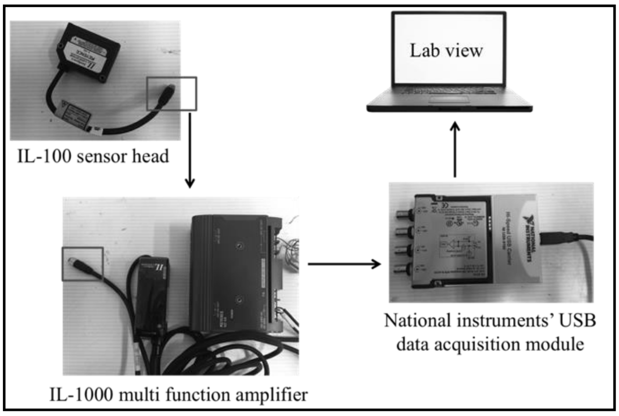

Figure 6. Instrumentations involved in using laser displacement Sensor

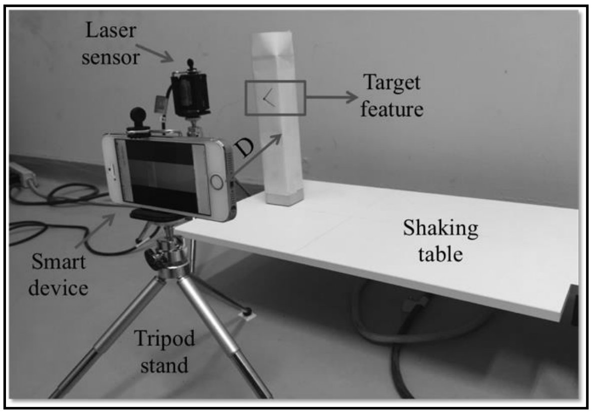

Figure 7. Shaking Table Tests Set up

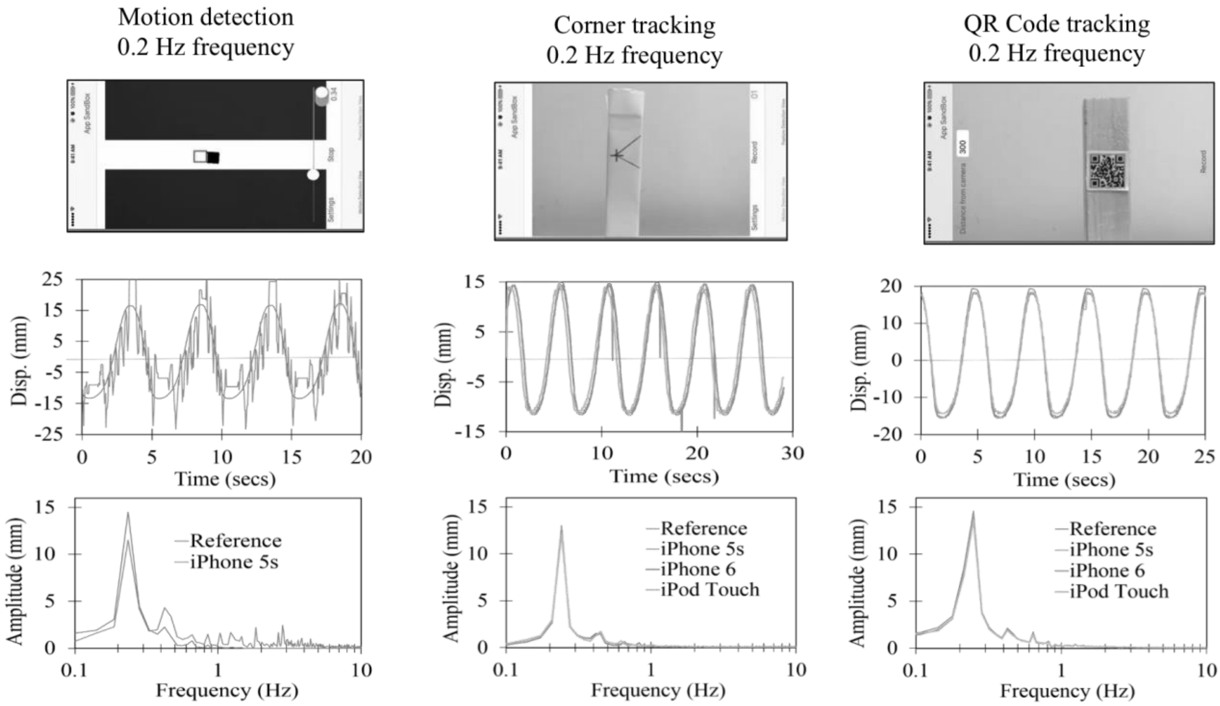

Figure 8. Comparision of time domain and frequency domain plot of sinusoidal signals measured by smart devices and laser sensor at 0.2 Hz test frequency

table and WF-1974 function generator was used to generate sinusoidal waveform with different frequencies. To compare the performance of the developed iOS app with that of a conventional displacement sensor, a laser displacement sensor head[7)] (KEYENCE, IL-100, 4μ resolution) was used as a reference as shown in Figure 6. Using IL-1000 multi-function amplifier the laser sensor head was controlled and the analog voltage outputs from the laser sensor were measured and output by the National Instruments' NI USB-9162 (24-bit ADC module) connected to a PC via lab View software. Three different models of iOS smart devices with different computational and camera capabilities i.e. iPod touch, iPhone 5s and iPhone 6 was considered for the experiment. An artificially designed target object as shown in Figure 7 was marked with corner feature in case of corner tracking method or QR code feature in case of QR code tracking method was fixed on the shaking table and driven by sinusoidal signals of different frequencies from 0.1 Hz to 5.0 Hz. The time of measurement was taken such that there were enough cycles of sinusoidal waves for each test. The sinusoidal signals recorded by the smart device were compared with that recorded by laser sensor in terms of displacement time history record and Fourier amplitude. Figure 8. shows the comparison of displacement response measurement between smart devices and reference measurement at 0.2 Hz test frequency for three different methods. It is observed that although the trajectory of motion is detected using the motion detection method, but tracking the motion seems to be less stable and is highly affected by background noise. Measurements are unstable in terms of amplitude domain, however identification of dominant frequency was possible with less error as seen from Fourier amplitude plot. In case of corner tracking method and QR code tracking method, the motion seems to be much more stable and robust to background noise. An excellent agreement in the amplitude measurement made by smart devices and reference is observed and in all the cases identification of dominant frequency was possible.

However the corner tracking method may be difficult for practical applications in real field displacement measurement applications, since the corner targets are not unique feature and hence false corners could be easily detected due to which error in measurement is bound to occur. Nevertheless, QR code tracking method can have more practical applications, as this method is highly robust to background noise and because of its unique target feature, measurement is not affected. Therefore, considering QR code tracking method as the most accurate and reliable method amongst all the three methods, to further illustrate its reliability and range of measurement, more experiments were conducted to study some factors that might affect QR code detection. The effects of variation of following three factors were taken into consideration.

1. Effect of variation of test frequency (i.e. velocity of target object in motion)
2. Effect of variation of target distance (i.e. size of target in the view of smart device)
3. Effect of variation of amplitude of vibration

For this purpose, the measurements of two different smart devices, having different CPU capabilities and camera resolution (iPhone 6 and iPod touch) were compared with the reference sensor. The shaking table was driven by sinusoidal signals of different frequencies from 0.1 Hz to 5.0 Hz and at each frequency, tests was conducted with varying amplitude and varying distance from target. This test setting is as shown in Table 2. The accuracy of displacement measurement by smart devices was evaluated in terms of cross correlation between the waveform measured by reference and smart devices. A

Table 2. Shaking table test setting to study effect of variation of frequency, amplitude and target distance

Test Frequency	Amplitude	Distance (mm)			
		100	200	300	400
0.1 Hz	±1 mm	●	●	●	●
	±5 mm	●	●	●	●
	±15 mm	●	●	●	●
0.2 Hz	±1 mm	●	●	●	●
	±5 mm	●	●	●	●
	±15 mm	●	●	●	●
0.5 Hz	±1 mm	●	●	●	●
	±5 mm	●	●	●	●
	±15 mm	●	●	●	✖
1.0 Hz	±1 mm	●	●	●	●
	±5 mm	●	●	●	✖
	±15 mm	●	✖	✖	✖
2.0 Hz	±1 mm	●	●	●	✖
	±5 mm	●	✖	✖	✖
	±15 mm	✖	✖	✖	✖
5.0 Hz	±1 mm	●	●	✖	✖
	±5 mm	✖	✖	✖	✖
	±15 mm	✖	✖	✖	✖

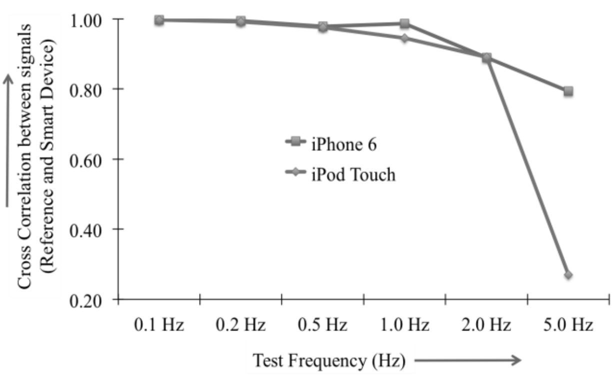

Figure 9. Effect of test frequency on waveform measurement (at amplitude 1 mm and distance from target100 mm)

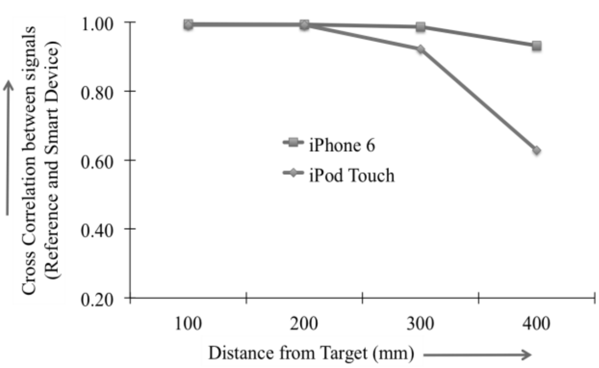

Figure 10. Effect of target distance on waveform measure (at amplitude of 1 mm and test frequency 0.2 Hz)

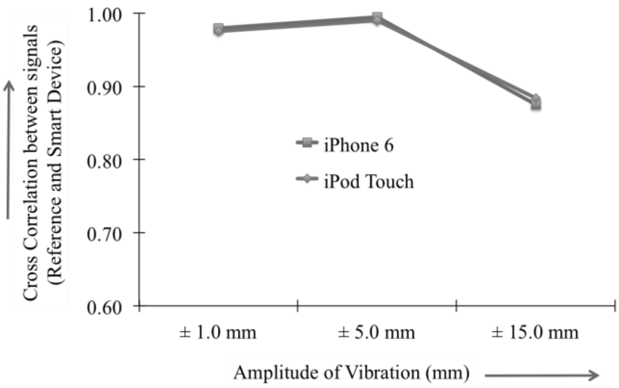

Figure 11. Effect of amplitude of vibration on waveform measurement (at constant target distance of 100 mm and test frequency 0.5 Hz)

Table 3. Effect of test frequency on QR code images captured during video sequence

Test Freq.	Images captured in video sequence			
0.5 Hz				
2.0 Hz				
5.0 Hz				

good correlation coefficient nearing ±1.0 signifies accurate and reliable measurement while a bad correlation coefficient nearing 0.0 signifies inaccurate and unreliable measurement. The circles marks shown in Table 2. signify that QR code was detected under the mentioned test settings, while cross marks signify that QR code was not detected. Therefore, it is clear that QR code recognition depends on all the above-mentioned factors.

Figure 9, 10 and 11 illustrates the effect of test frequency, target distance and amplitude of vibration on waveform measurement respectively. It is observed that with increase in test frequency, target distance and amplitude of vibration the cross correlation between the signals measured by smart device and reference decreases. This means that at lower frequency of vibration, (generally below 2 Hz) measurements from smart devices are comparable with that of reference but error increases as frequency increases. Similarly, with increase in target distance (i.e. decrease in size of QR code target in smart device's view) and amplitude of vibration, error in measurement increases. This is because whenever the frequency of vibration increases, the real time images captured by smart device's camera lose its focus and the images becomes blurred as shown in Table 3. and with a blurred image the accuracy and probability of QR code recognition decreases due to which error in measurement increases.

4. CONCLUSION

The feasibility of smart device technologies for image process based displacement measurement and monitoring has been investigated in this study. The task of measuring real time displacement response was conducted by developing measuring application for three different methods of image processing i.e. motion detection; corner tracking and QR code tracking. In order to fully utilize the GPU capabilities of smart devices, the GPUImage library was used in developing the iOS app for motion detection and corner tracking methods.

Onboard calibration of the image pixel size to a given-dimension target was implemented in the developed iOS apps. Similarly, various features for controlling camera, filters, graph settings and cloud transmission of measured data were incorporated in the app development. All the functions required for measuring the dynamic displacements of the target could successfully be operated in real-time. The performance of smart devices hardware and the iOS app developed herein were experimentally validated.

(a) The test results showed that motion detection and corner tracking methods are highly sensitive to background noise and therefore is very difficult for practical applications in real field displacement measurement, while QR code tracking method is highly robust to background noise and hence has more practical applications.

(b) From shaking table experiments, it was also confirmed that all these methods of image processing used in current study could only be applied effectively for long period displacement measurement (frequency smaller than 1.0 Hz). Nevertheless, more study still needs to be conducted in future and various other computationally efficient methods for tracking fast moving objects should be explored for more robust application in real time displacement measurement using smart devices.

REFERENCE

1) Takahashi, Y.: Damage of rubber bearings and dampers of bridges in 2011 Great East Japan Earthquake, *Proceedings of the International Symposium on Engineering Lessons Learned from the 2011 Great East Japan Earthquake*, pp. 82-101, 2012
2) Indrawan, P., Budiyatno, S., Ridho, N. M., and Sari, R.F. : Face Recognition for Social Media with Mobile Cloud Computing, *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, vol. 3. No. 1, 2013
3) Min, J., Nikolas, J. G., and Hongki, J.: Non-contact and Real-time Dynamic Displacement Monitoring using Smartphone Technologies, *Life Cycle Reliability and Safety Engineering*, 2015, vol. 4, issue 2, pp. 40-51
4) Dropbox, Dropbox Developer, Sync *API*: *http://www.dropbox.com/developers/sync*
5) Larson, B.: An open source iOS framework for GPU-based image and video processing, *GitHub*, Web. 04 Nov 2014 *http://github.com/BradLarson/GPUImage*
6) Harris, C. and Stephens , M.: A combined corner and edge detector, *Proceedings of the 4th Alvey Vision Conference*, pp. 147-151, 1988
7) Keyence IL Series CMOS Multi-Function Analog Sensor, *http://www.movetec.fi/images/pdf/Keyence-lasermittaus1.pdf*