

**A COST-EFFECTIVE, MOBILE PLATFORM-BASED, PHOTOGRAMMETRIC
APPROACH FOR CONTINUOUS STRUCTURAL DEFORMATION MONITORING**



CHIH-HENG WANG

BSc. Civil Engineering

MSc. Environmental and Spatial Information Science

Thesis submitted for the degree of Doctor of philosophy

School of Civil Engineering and Geosciences
Newcastle University

JUNE 2013

ABSTRACT

With the evolution of construction techniques and materials technology, the design of modern civil engineering infrastructure has become increasingly advanced and complex. In parallel to this, the development and application of appropriate and efficient monitoring technologies has become essential. Improvement in the performance of structural monitoring systems, reduction of labour and total implementation costs have therefore become important issues that scientists and engineers are committed to solving.

In this research, a non-intrusive structural monitoring system was developed based on close-range photogrammetric principles. This research aimed to combine the merits of photogrammetry and latest mobile phone technology to propose a cost-effective, compact (portable) and precise solution for structural monitoring applications. By combining the use of low-cost imaging devices (two or more mobile phone handsets) with in-house control software, a monitoring project can be undertaken within a relatively low budget when compared to conventional methods. The system uses programmable smart phones (Google Android v.2.2 OS) to replace conventional in-situ photogrammetric imaging stations. The developed software suite is able to control multiple handsets to continuously capture high-quality, synchronized image sequences for short or long-term structural monitoring purposes. The operations are fully automatic and the system can be remotely controlled, exempting the operator from having to attend the site, and thus saving considerable labour expense in long-term monitoring tasks. In order to prevent the system from crashing during a long-term monitoring scheme, an automatic system state monitoring program and a system recovery module were developed to enhance the stability. In considering that the image resolution for current mobile phone cameras is relatively low (in comparison to contemporary digital SLR cameras), a target detection algorithm was developed for the mobile platform that, when combined with dedicated target patterns, was found to improve the quality of photogrammetric target measurement. Comparing the photogrammetric results with physical measurements, which were measured using a Zeiss P3 analytical plotter, the returned accuracy achieved was 1/67,000.

The feasibility of the system has been proven through the implementation of an indoor simulation test and an outdoor experiment. In terms of using this system for actual structural monitoring applications, the optimal relative accuracy of distance measurement was determined to be approximately 1/28,000 under laboratory conditions, and the outdoor experiment returned a relative accuracy of approximately 1/16,400.

TABLE OF CONTENTS

Chapter 1.	Introduction.....	1
1.1.	Overview.....	1
1.2.	Background to structural health monitoring	2
1.1.1.	Definition and potential problems	2
1.1.2.	Motivation for study	3
1.3.	Aim and objectives	4
1.4.	Research scope.....	5
1.5.	Thesis structure	6
Chapter 2.	Techniques for structural measurement and monitoring.....	7
2.1	Introduction.....	7
2.2	Conventional structural measurement and monitoring	7
2.2.1	Direct measurements.....	8
2.2.1.1	Strain gauge	9
2.2.1.2	Optical fibre gratings (Fibre Bragg Grating-FBG).....	11
2.2.2	Geodetic measurements	15
2.2.2.1	Total stations	16
2.2.2.2	Global navigation satellite system - GNSS.....	19
2.2.2.3	Terrestrial laser scanning-TLS	22
2.2.3	Close range photogrammetric measurements	27
2.3	Summary	41
Chapter 3.	A photogrammetric methodology for monitoring flexible structures..	42
3.1	Introduction.....	42
3.2	A photogrammetric solution for structural health monitoring	43
3.2.1	Imaging devices details.....	44
3.2.2	Camera calibration	47
3.2.2.1	Field calibration	47
3.2.2.2	PhotoModeler dedicated calibration method	50
3.2.3	Photogrammetric processing software and optimal target size....	52
3.2.4	Additional instruments.....	55
3.2.5	The OS, programming languages and photogrammetric processing software.....	57
3.2.6	System design and integration	60
3.3	Mobile platform based structural monitoring system	65
3.3.1	System architecture and design principles.....	65
3.3.2	Principles of system health monitoring and automated system recovery	74
3.3.3	Automatic data sorting.....	78
3.3.4	Summary of the mobile platform based structural monitoring module	78
3.4	Advanced photogrammetric processing function	79
3.4.1	System layout.....	79
3.4.2	Conversion to greyscale.....	81
3.4.3	Greyscale to binary conversion.....	87
3.4.4	Developed target detection algorithm.....	88
3.4.5	Summary of the onboard photogrammetric processing software	

suite	92
3.5	Summary 92
Chapter 4.	Performance evaluation 94
4.1	Introduction..... 94
4.2	Photogrammetric network configuration 94
4.2.1	Experimental configuration 94
4.2.2	Test results 96
4.3	Inspection of system synchronisation 98
4.3.1.	Experimental configuration 99
4.3.2.	Test results 101
4.4	Performance evaluation of developed target detection algorithm 105
4.4.1	Optimal target size determination for developed target detection algorithm 105
4.4.1.1	Experimental configuration 105
4.4.1.2	Test results 106
4.4.2	Comparison of target detection algorithms 108
4.4.2.1	Experimental configuration 109
4.4.2.2	Test results 112
4.5	Summary 116
Chapter 5.	System implementation..... 118
5.1.	Introduction..... 118
5.2.	Laboratory experiment – shaking table test..... 118
5.2.1	Experimental configuration and workflow 118
5.2.2	Photogrammetric targets 125
5.2.3	Physical measurement..... 127
5.2.4	Simulation of structural damage detection 128
5.2.5	Test results 129
5.2.5.1	Evaluation of photogrammetric precision..... 130
5.2.5.2	Assessment of photogrammetric accuracy in distance measurements..... 133
5.2.5.3	Demonstration of structural damage detection 135
5.3.	Outdoor implementation 140
5.3.1	Experimental configuration 140
5.3.2	Photogrammetric targets 141
5.3.3	Operational processes 142
5.3.4	Test results 143
5.3.4.1	Inspection of system synchronisation 143
5.3.4.2	Evaluation of photogrammetric precision..... 144
5.3.4.3	Assessment of photogrammetric accuracy in distance measurements..... 145
5.3.4.4	3D modelling and deformation tendency..... 146
5.4.	Summary 149
Chapter 6.	Discussion of results 150
6.1	Introduction..... 150
6.2	System inspection 150
6.3	Assessment of photogrammetric accuracy..... 156
6.3.1	The influence of the quantity of imaging stations on

	photogrammetric precision	156
6.3.2	The influence of photogrammetric network configuration and target sizes on photogrammetric precision	157
6.3.3	Performance assessment	163
6.3	Potential problems in operation	166
6.3.1	Deficiency in exposure frequency	166
6.3.2	The degree of automation of system operation.....	166
6.4	Summary	168
Chapter 7.	Conclusions and future work	169
7.1	Introduction.....	169
7.2	Summary of work	169
7.3	Research contributions.....	171
7.4	Suggestions for future work.....	174
7.4.1	Considerations of photogrammetric accuracy	175
7.4.2	OS considerations	177
7.4.2	The development of a stable power supply for long-term monitoring purposes	180
7.4.3	Performance improvement of the low-cost monitoring system.....	182
7.5	Concluding remarks	184
Appendix i	System operation of multiple-station module	186
A1.1.	Environment settings and system operation	186
A1.2.	System synchronisation and time correction	191
A1.2.1	Time adjustment on the terminal computer	192
A1.2.2	Time adjustment on the mobile phone platform	195
Appendix ii	The Generalised Hough transform (GHT).....	204
A2.1.	GHT for target detection	204
Appendix iii	Source code.....	208
A 3.1.	Main program	208

LIST OF FIGURES

Figure 2-1: Relationship between object size and accuracy for different measurements	8
Figure 2-2: Illustration of resistance strain gage.....	10
Figure 2-3: Illustration of roof monitoring through the use of strain gauge measurement	11
Figure 2-4: Illustration of the FBG	11
Figure 2-5: Illustration of different optical fibres.....	13
Figure 2-6: Reinforced concrete beams with FBG sensors	14
Figure 2-9: Illustration of the use of a total station for bridge structure monitoring ...	18
Figure 2-10: Configuration of observation points (Karbhari and Ansari, 2009)	19
Figure 2-11: Network configuration of GPS and photogrammetric stations for a dam monitoring project.....	21
Figure 2-12: Network thumbnail of GPS stations for a dam monitoring project	21
Figure 2-13: 3D laser scanning for tunnel structure monitoring.....	23
Figure 2-14: Illustration of using terrestrial laser scanning technology for tunnel 3D structural modelling	24
Figure 2-15: Configuration of the geodetic network reporting laser scanner stand-points for a dam monitoring project.....	25
Figure 2-16: Laser scanned model of the membrane structure.....	26
Figure 2-18: Stereo image configurations.....	30
Figure 2-19: Stereo image configurations.....	31
Figure 2-20: Relationship between network geometry and related photogrammetric precision.....	33
Figure 2-21: Configuration of control points and 3D modelling.....	35
Figure 2-22: Slope monitoring using photogrammetric measurements.....	36
Besides improving photogrammetric results by changing the	37
Figure 2-23: The use of photogrammetric technique for a bridge monitoring	38
Figure 2-24: Configuration of a large wind turbine monitoring project.....	39
Figure 2-25: The Layout of photogrammetric targets on the turbine	40
Figure 3-1: The illustration of test field configuration	47
Figure 3-2: The illustration of target area	48
Figure 3-3: The configuration of camera calibration.....	51
Figure 3-4: Geometrical construction for a typical thin lens system.....	55
Figure 3-5: Illustration of system operation.....	60
Figure 3-6: Example of a conventional long term in-situ photogrammetric imaging station.....	61
Figure 3-7: Workflow diagram for the low-cost photogrammetric monitoring system	64
Figure 3-8: Illustration of the multiple-station module's functions.....	66
Figure 3-9: Illustration of the star network configuration	67
Figure 3-10: Illustration of the grouping function operation interface	69
Figure 3-11: The coding procedures of Vector Quantisation	71
Figure 3-12: Illustration of mobile phone memory usage	73
Figure 3-13: The comparison of circuit structure between the CCD (left) and CMOS (right) sensors	75

Figure 3-14: The screen capture form the system health monitoring program.....	76
Figure 3-15: Illustration of the single-mobile phone module functions	80
Figure 3-16: Illustration of real-time greyscale conversion.....	84
Figure 3-17: Comparison for luminosity histograms.....	86
Figure 3-18: Illustration of thresholding.....	87
Figure 3-19: The three-quarter-circle target pattern.....	88
Figure 3-20: Manual selection of the targets	89
Figure 3-21: Estimation of curves and straight lines	90
Figure 3-22: Illustration of linear regression	90
Figure 3-23: The targets' binary images	92
Figure 4-1: Experimental configuration and target arrangement.....	95
Figure 4-2: Illustration of eight different imaging positions	96
Figure 4-3: Relationship between the precision and the quantity of imaging stations	97
Figure 4-4: The system synchronisation experiment	100
Figure 4-5: The comparison of exposure epochs	103
Figure 4-6: The comparison of the absolute time of capture between the two mobile phone stations	104
Figure 4-7: The field configuration for optimal target size estimation.....	106
Figure 4-8: Flowchart for comparison of target detection algorithms	109
Figure 4-9: Configuration of the test field	110
Figure 4-10: Glass plates with two different target patterns.....	110
Figure 4-11: Illustration of targets	111
Figure 4-12: The chosen distances for comparison	112
Figure 4-13: Comparison in standard deviations of two different target patterns	115
Figure 4-14: Comparison in root-mean-square error of two different algorithms.....	115
Figure 5-1: (a) Experimental configuration. (b) A close-up photograph of an individual image station.....	119
Figure 5-2: (a) Central computer and control component. (b) and (c) show the switch box and the camera relay panel.....	120
Figure 5-3: Illustration of test objects for shaking table test	121
Figure 5-4: Configuration of the shaking table experiment.....	123
Figure 5-5: Shaking table test	124
Figure 5-6: Customised target stickers with different target sizes.....	127
Figure 5-7: Unscrewing some mounting bolts of the three-story stainless steel frame	129
Figure 5-8: Coordinate standard deviations under the different configurations (DSLR)	131
Figure 5-9: Coordinate standard deviations under the different configurations (smart phone)	131
Figure 5-10: Standard deviations of 9 mm targets based on different motion states.	133
Figure 5-11: Selected measurements for validation comparison	134
Figure 5-12: RMSE of photogrammetric measurements (smart phone camera stations)	134
Figure 5-13: Configuration of coded targets for the three-story stainless steel frame	136
Figure 5-14: Comparisons of displacements under two motion status (from Point_1 to Point_6 on X_axis)	138

Figure 5-15: Comparisons of displacements under two motion status (from Point_7 to Point_12 on X_axis)	139
Figure 5-16: Schematic illustration of test site	141
Figure 5-17: The chosen distances for comparison	143
Figure 5-18: the exposure timing of each mobile phone camera.....	144
Figure 5-19: Configuration of photogrammetric targets for the rolling shutter test ..	145
Figure 5-20: Displacement of a rolling shutter in Z direction (PT_13).....	146
Figure 5-21: Discrepancy map derived from 3D conformal transformation (at the 92nd exposure epoch)	147
Figure 5-22: Textured 3D model and Deformation sequence of rolling shutter.....	148
Figure 6-1: The arrangement of test field	151
Figure 6-2: Experimental setup for the cushion test	152
Figure 6-3: Cushion rise against pressure.....	153
Figure 6-4: Mis-identified target points	155
Figure 6-5: Configuration of imaging stations for shaking table test	159
Figure 6-6: Configuration of imaging stations for outdoor test.....	161
Figure 6-7: Flowchart of using the mobile platform based photogrammetric system for structural monitoring applications	167
Figure 7-1: Principle of ellipse operator (the Zhou ellipse operator)	176
Figure 7-2: The relationship between different point measurement operators and the measurement accuracy of target centre detection	177
Figure 7-3: Illustration of error detection module	179
Figure 7-4: Illustration of the external battery pack	180
Figure 7-1: Illustration of the developed solar energy generating system.....	181
Figure A-1: Illustration of start screen	186
Figure A-2: The main control panel of the multiple-station module	187
Figure A-3: Illustration of the command script	188
Figure A-4: Illustration of the status report	189
Figure A-5: A source code fragment of the interrupter program	191
Figure A-6: Setting the computer clock by the use of the self-programmed time adjustment module.....	193
Figure A-6: The confirmation window for time zone setting	194
Figure A-7: NTP server connection	194
Figure A-8: The mobile phone-based time adjustment program	197
Figure A-9: Illustration of A-GPS.....	197
Figure A-10: Two methods to adjust the internal clock of each mobile phone	199
Figure A-11: Two way message exchange between two nodes	200
Figure A-12: The confirmation of the standard for synchronisation	201
Figure A-13: To confirm the mobile phones are going to use the computer's time information as the standard for time adjustments	202
Figure A-14: Illustration of a generalised shape	205
Figure A-15: R-table	207

LIST OF TABLES

Table 2-1 Categorisation of photogrammetry	29
Table 2-2: Comparison of different measurements in measurement characteristics ...	41
Table 3-1: HTC Incredible S technical specifications	45
Table 3-2: Canon EOS 50D technical specifications	46
Table 3-3: Calibration results of camera calibration parameters	49
Table 3-4: Calibration results of main parameters	51
Table 3-5: The comparison of photogrammetric results in distance measurement between two different photogrammetric processing software packages	54
Table 3-6: HP Compaq Presario V3000 technical specifications	56
Table 3-7: TP-LINK TL-WA901ND technical specifications	56
Table 4-1: The comparison of curvature changes	107
Table 4-2: Photogrammetric results of full circle target datasets.....	113
Table 4-3: Photogrammetric results of 3/4 circle target datasets	114
Table 4-4: Brief summary of experimental findings.....	116
Table 5-1: Experimental configurations for the shaking table test	125
Table 5-2: The total number of pixels in diameter of each target under different experimental configurations.	126
Table 5-3: Comparison in standard deviations under the different configurations....	130
Table 5-4: Coordinate standard deviations under different motion states	132
Table 5-5: Comparisons in standard deviation of control points	137
Table 5-6: Comparisons of displacements under two different experimental configurations	138
Table 5-7: Coordinate standard deviations of control points	145
Table 5-8: Comparison between the photogrammetric results and physical measurements.....	146
Table 6-1: Photogrammetric measurements of cushion pressure and rise	153
Table 6-2: Relative accuracy of different measurement methods.....	164

LIST OF ABBREVIATIONS

2D	Two-dimensional
3D	Three-dimensional
3G	3rd-Generation
API	Application programming interface
APP	Application software
CCD	Charge-coupled device
CCP	Clock correction program
CMOS	Complementary metal–oxide–semiconductor
CPU	Central processing unit
DV	Digital video
DLT	Direct linear transformation
DSLR camera	Digital single-lens reflex camera
EOP	Exterior orientation parameter
ETFE	Ethyl Tetra Fluoro Ethylene
EXIF	Exchangeable image file format
FBG	Fibre Bragg grating
FOV	Field of view
GHT	Generalised Hough transform
GNSS	Global navigation satellite system
GPS	Global positioning system
IGS	International GNSS service
ISP	Internet service provider
IOP	Interior orientation parameter
LED	Light emitting diode

MB	Megabyte
NAP	Network access point
NTP	Network time protocol
OS	Operating system
PC	Personal computer
PIT	Programmable interval timer
RAM	Random-access memory
RMSE	Root-mean-square error
SD	Standard deviation
SDK	Software development kit
SEGS	Solar energy generating system
SHM	Structure health monitoring
SLR	Single-lens reflex
TCP/IP	Transmission control protocol / Internet protocol
TLE	Two-line orbital element
UDP	User datagram protocol
LAN	Local area network
VQ	Vector quantisation
WLAN	Wireless local area network

CHAPTER 1. INTRODUCTION

1.1. Overview

With the advancement of construction engineering, the design of modern buildings has become more and more complex (Chintalapudi et al., 2006). Considering civil structures as examples, due to the extensive use of composite materials, the strength of the structure is enhanced significantly which has led to increased sizes of modern structures (Yang, 2010). The widening of measuring areas (Baofeng and Gang, 2011) and the use of diverse building materials (Elaldi, 2005) have increased the difficulties associated with structural monitoring. In addition, because of the wider distribution of the population, artificial structures are no longer only located in densely populated areas. For example, dam constructions (Erfeng and Yongqiang, 2008) and large electric towers (e.g. large high-voltage pylons) (Ling et al., 2011) may be located in mountain areas or suburbs, but they also require long-term monitoring to ensure on-going structural safety.

In terms of using conventional systems for structural monitoring, the following problems are usually encountered:

1. Costly and heavy equipment (Chintalapudi et al., 2006);
2. The susceptibility and performance of devices can easily be interfered with by environmental factors (Rodrigues et al., 2010);
3. A system's ability to provide reliable measurements over a wide spatial area (Ansari, 2005);

A cost-effective, photogrammetric-based solution to structural monitoring is presented in this research. Combined with the use of mobile phone technologies, the photogrammetric monitoring system can significantly reduce equipment costs (for a four-mobile-phone configuration, the total expenditure is under £1,500). One of the advantages of programmable mobile phones is that the programmer can develop dedicated application software (apps) on a mobile phone platform to perform specific functions. The development of a multi-function structural monitoring software suite on a mobile phone platform (in this case Android OS v. 2.2) and its subsequent application to civil engineering structures is the focus of this thesis.

The research focuses on unlocking the potential of smart phone technologies for structural deformation monitoring applications. The study involved the development of photogrammetric functions onboard the smart phone platform. By means of computer programming, the smart phones were adapted to meet the requirements of long-term, structural monitoring of flexible structures (such as the capabilities of long period of synchronised image acquisition, automated operation, etc.).

1.2. Background to structural health monitoring

1.1.1. Definition and potential problems

The term structural monitoring covers a broad category of measurements, which includes contact and non-contact measurements, long-distance and close-range measurements, manual operation and automatic measurements, etc. (Ming-Chih et al., 2010; Qilin and Dalin, 2010; Konstantinidis et al., 2007; Mesch, 2007). Each monitoring system may appear in more than one category. For example, strain gauges are frequently used to measure small strains of an object in traditional monitoring projects. The strain gauges need to be attached onto the structural surface to detect the deformation of the small area of the target structure (Beug et al., 2012). Therefore, it can be classified as one of the contact measurements. Also, the real-time deformation can be sent to a database directly (Noppe and Decréton, 1993). For this reason, the use of strain gauges for structural deformation monitoring can be regarded as a real-time, automatic measurement. Numerous methods are available for structural monitoring applications. Operators can make an exhaustive study before the operation

begins and choose the most appropriate measurement for a specific monitoring project.

In recent decades, with the increasing number of large civil structures, there is growing attention on structural health monitoring (SHM) (Li and Liu, 2010). The concept of SHM is to establish a motion model for target structure (in a normal state) via a series of long period regular monitoring data (Dan et al., 2010). The supervisor can detect the abnormalities instantly by comparing the real-time state with the long-term observation data (Chongchong et al., 2010).

SHM demands instruments with the capability for long-running operation (Jindal and Liu, 2012). Through the continuous operations of monitoring systems, surveyors can fully assess the status of the target structures at all times (Dong-Hyun et al., 2011). For this reason, the systemic stability for the long period operation becomes a very important issue. However, the long period operation and the need for in-person monitoring results in higher operating and maintenance costs (Seunghee et al., 2006). As mentioned previously, many important modern constructions are not only located in urban areas. If they suffer structural damage, it is likely to directly endanger the lives of the general public. Therefore, the implementation of regular structural inspection is necessary. In such circumstances, the costs for transportation and maintenance may be considerable. Therefore, SHM has been considered as a labour-intensive and costly process. Moreover, some specific structures (such as nuclear power plants) may not be suitable for utilising manual measurements due to safety reasons (Noppe and Decréton, 1993). To design a fully automatic operation to reduce human intervention has become one of the key projects in the development of a SHM system.

1.1.2. Motivation for study

A structure can be likened to a human body since the strength of the structure will gradually age over time (Kyung-Tae et al., 2010). In addition, it will also speed up the deterioration of structural strength if the structure suffers from additional loads or sudden external forces which have exceeded the permissible limits of the structures' original design. SHM can be regarded as the regular medical check-ups to target

structures and each part of the structure can be likened to the organs of the human body. If any part of the structure suffers damage and/or lost structural functions, the damage may continue and jeopardise the overall safety of the structure. Through the use of SHM systems for long-term data collection, the structural response data can be used to identify the existence, location, and severity of structural damage (Loh et al., 2007). The implementation of SHM has become an essential part of modern constructions. How to improve the performance of SHM systems and to reduce labour and total implement costs have become the priority that developers are committed to solving.

This research attempts to develop an inexpensive, reliable, photogrammetric-based SHM system, which allows the operator to perform long-term monitoring of structures whilst appending basic functions for photogrammetric processing to a mobile phone platform.

1.3. Aim and objectives

In order to allow civil engineers to determine whether or not structures are maintaining a stable state, this geomatic engineering research aims to develop a low cost, in-situ, autonomous photogrammetric solution for monitoring the displacement and deformation of as-built structures. Specific research objectives to achieve this aim are:

1. To understand the most commonly used structural monitoring measurements in civil engineering and to identify the requirements for a photogrammetric system which could be deployed as a long-term structural monitoring solution.
2. Based on the findings of objective 1, to propose a low cost, in-situ photogrammetric monitoring system based on off-the-shelf mobile technology to meet the requirements of a variety of structural monitoring applications.
3. To comprehensively implement and test the necessary requirements and functions which were proposed in objective 2, on a mobile platform.
4. To prove the capability of the mobile photogrammetric solution to ensure the

performance of the system is capable of long-term structural monitoring proposes.

1.4. Research scope

The priority of this research is to develop a precise SHM system with relatively low budgeted spending. This research is intended to utilise the analytical power of the surveying technique of photogrammetry to address practical structural monitoring projects and to improve the practice of automatic operation. Combining with the cooperation of mobile phone techniques to achieve the aim which is presented in § 1.3 (the development of an unattended operation, remote control and affordable monitoring system).

Photogrammetry has been considered as a mature measurement technology (Luhmann et al., 2007). One of the superiorities of photogrammetric measurement is the characteristic of wide-area coverage (Jui-Teng and Chi-Kuei, 2011). The operator can use few imaging stations to perform large-area monitoring and the monitoring costs can therefore be reduced significantly. To this end, the research does not consider the use of high-end, expensive cameras but employs Android-based smart phones with built-in cameras, which surveyors can easily arrange in a test site for monitoring purposes. The operator also can take images conveniently on site. A dedicated software suite was developed to exploit this modern mobile phone technology for photogrammetric purposes. With the use of the developed software, the operator can remotely control two or more handset devices simultaneously via a 3G or wireless network. Once these mobile phone-based imaging stations has been arranged in the field, the operator can send commands, monitor the system operation and collect data in the office rather than have to visit the site in person. An automatic system state monitoring program and a system recovery module are developed in this research to enhance the stability of this low-cost system for long-term monitoring tasks.

The developed software also accesses the settings of the mobile phone's internal sensors (such as the GPS sensor, accelerometers and gyroscope). These sensors are now able to be used to record real-time information of position and orientation whilst the mobile phone is performing the image acquisition process. The fully automatic operation can reduce labour costs significantly.

1.5. Thesis structure

Chapter 1 has presented a general introduction to the thesis and the research aim and objectives.

Chapter 2 reviews conventional structural monitoring measurements, which are the most frequently used in modern civil engineering and also examines photogrammetric approaches.

Chapter 3 outlines the main system components of the low-cost, photogrammetric monitoring system, which was developed in this research. The contents include the introduction to system functions, the operational processes, the use of hardware and software, and the choice of operating platform. The design logic and principles of each function are introduced in this chapter

Chapter 4 presents a series of experiments which were performed to inspect the individual functions of the developed system.

Chapter 5 presents an indoor experiment and a practical outdoor implementation to inspect the overall performance of the developed system.

Chapter 6 evaluates the appropriateness of this system for long-term structural monitoring based on the experimental outcomes that are presented in Chapter 4 and Chapter 5.

Chapter 7 concludes the research in terms of the contributions. Also, possible future extensions are recommended in this chapter.

CHAPTER 2. TECHNIQUES FOR STRUCTURAL MEASUREMENT AND MONITORING

2.1 Introduction

Structural monitoring has been considered to be a complex issue because the optimal measuring instrument varies according to the differences in test site conditions and target objects (Sun et al., 2010). A review of some of the most frequently used measurements in modern civil engineering is presented in this chapter.

2.2 Conventional structural measurement and monitoring

In terms of structural monitoring in civil engineering, selecting a proper measurement for a specific project relies on the professional judgment of the planning team (Connor and Faraji, 2012). In general civil engineering projects, the size of target objects can be as small as a single storey house or as large as a dam (Wood, 2009). According to the demands of required accuracy and the difference of target sizes, surveyors can choose the appropriate measurement method to meet the engineering requirements. Figure 2-1 provides a general description of the relationship between the size of target objects, required measurement accuracy and corresponding appropriate technology.

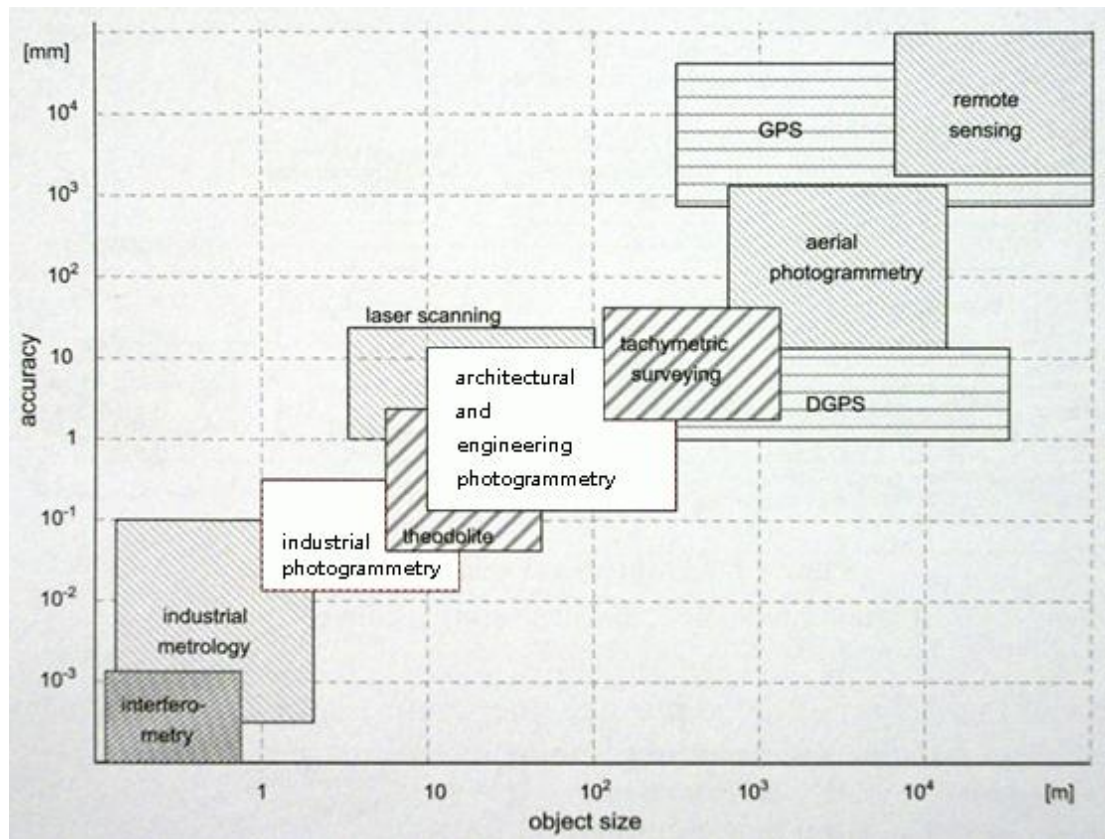


Figure 2-1: Relationship between object size and accuracy for different measurements (Luhmann et al., 2007).

As shown in Figure 2-1, due to the requirement of the acceptable range of error being not as strict as the industrial measurements (often micron levels or higher), there are many different methods that can be used by surveyors (Takewaki et al., 2007). In the following sections, some of the most commonly used monitoring measurement methods in civil engineering are presented that address both the principles and the practical applications for structure monitoring.

2.2.1 Direct measurements

Direct measurements (also known as contact measurements) are one of the most straightforward approaches for obtaining information from the measured objects (Ansari, 2005). These measurement methods are normally used to measure the changes in stress of a single point or small area (Li et al., 2009). The sensors are normally buried in the target objects and these sensors are wired to a receiver, which

is connected to a database (some systems use wireless transmission) in order to transfer the real-time deformation data for further analyses (Choi et al., 2012). Real time data transmission is one of the biggest advantages of direct measurements, but the rather small measuring range and difficulty of maintenance (because the sensors are normally not exposed on the structural surface) are two major disadvantages.

2.2.1.1 Strain gauge

A strain gauge is a low-cost device used to measure the change in strain on an object when the object is placed under stress. The strain gauge can convert the mechanical force into an electrical signal (Yongdae et al., 2010). There are many types of strain gauges and each one has different operating principles and applications. Among them, resistance strain gauges are the most commonly used in structural monitoring applications (Haksoo et al., 2008). As the name implies, a resistance strain gauge uses changes in resistance to detect changes of strain. The most common type of resistance strain gauge has a structure with a grid-shaped thin metallic resistive foil attached on a base of thin plastic film and is laminated with a thin film. Figure 2-2 illustrates the resistance strain gauge. When the strain gauge is attached on the surface of a target object, the strain gauge will be stretched or compressed with the target object and the metallic resistive foil will be elongated or shortened with the strain gauge. Thus, as the object is deformed, the metallic resistive foil will follow the deformation accordingly and cause its electrical resistance to change. Engineers can measure the change in resistance to determine the principal strains.

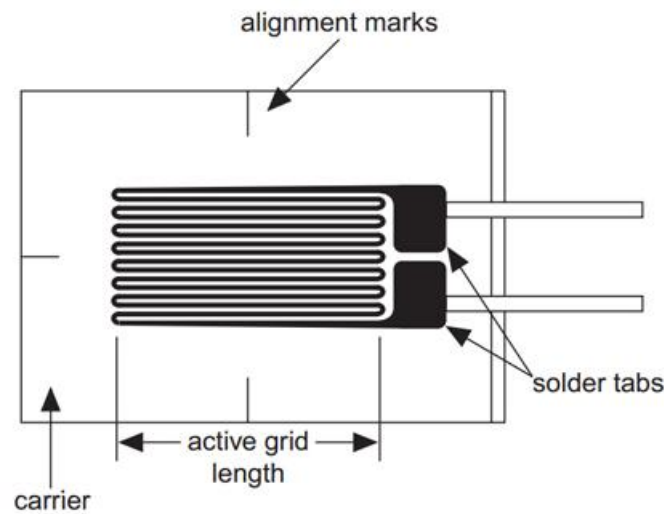


Figure 2-2: Illustration of resistance strain gage (National_Instruments, 1998).

Strain gauges can not only be used to measure the changes in strain of a single point, but can also expand the measuring range to a small area (Stefanescu, 2011).

However, strain gauges are easily interfered with by temperature variations and the rather small measuring range. This results in some restriction in applications. However, strain gauges are not expensive (current retail price is less than £10 each). Therefore, this measurement approach is still widely used in modern engineering to detect the real-time strain changes of small areas. However, it is usually used in conjunction with other measurement methods (Ramana and Sarma, 1981).

Figure 2-3 shows a practical application of the strain gauge measurement for a roof deformation monitoring application. In this case, a total of 5,943 strain gauges were attached onto the specific positions of the steel space frame roof structure to provide real-time information of structural deformation (Yuan, 1991). This study also highlighted another disadvantage of the strain gauge, which is the difficulty in installing multiple sensors. When monitoring a large structure using strain gauge observations, engineers need to properly deal with complex wiring problems to ensure that the receivers can receive the information from every sensor. In order to overcome this difficulty, a series of wireless sensors have been developed. However, at present wireless sensors are more expensive than conventional strain gauges.



Figure 2-3: Illustration of roof monitoring through the use of strain gauge measurement (Yuan, 1991).

2.2.1.2 *Optical fibre gratings (Fibre Bragg Grating-FBG)*

In order to improve the issue of a small measuring range, whilst also wanting to retain the capability of real-time monitoring, strain gauges have gradually been replaced with optical fibre gratings in wide-area measurement applications (Choi et al., 2012).

An optical fibre is a flexible, transparent fibre which is made of silica or plastic (Lee et al., 2009). Optical fibres are normally used to transmit light signals and audio signals between the two ends of the fibre. Due to the low impedance of the inner wall, signals travel along optical fibres with less loss (Hogari et al., 2010).

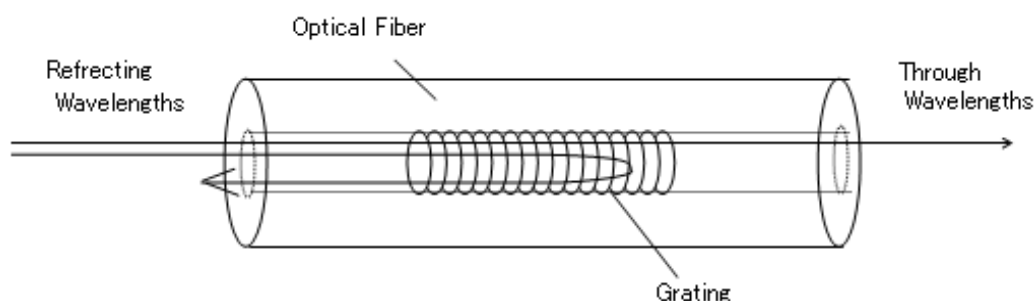


Figure 2-4: Illustration of the FBG (Zhu et al., 2011).

For some good quality optical fibres, the attenuation rate of the signal strength is approximately 4.5% a kilometre (Tsujikawa et al., 2005). Optical fibres are classified into two categories which are multi-mode fibre and single-mode fibre, respectively. In addition, multi-mode fibres are also divided into two types – step-index fibres and graded-index fibres (Figure 2-5). The step-index fibre has a wider core diameter than the other two types (approximately 200 μm) and the attenuation rate of signal strength is largest among these three types (Senior, 2008). The core diameter of a graded-index fibre is narrower than a step-index fibre (50 to 100 μm) and the attenuation rate has approximately 45% more improvement than a step-index fibre (Babchenko and Maryles, 2007). Single-mode fibres have the finest core diameter (less than 10 μm) and the modal dispersion is the lowest among these three types of optical fibres. As shown in Figure 2-5, a single-mode fibre is designed to carry only a single ray of light (mode) and the ray travels parallel to the length of the fibre. The signal processing of single-mode fibres is easier than other two types. When these optical fibres are buried inside the target objects, the signal analysis is more difficult (in comparison to normal situations). The use of simpler signal for transmission, the less error will be caused.

Compared with traditional electronic sensor technology, optical fibres have the following advantages:

1. Optical fibre sensors are small in size; the diameter of optical fibre sensors is approximately 250 μm (some are even smaller) (Rivera et al., 2002).
2. The main component of optical fibre is silicon (Jones et al., 1988). Therefore, optical fibre sensors can be buried inside the target objects for a long period and will not corrode like metal sensors (Bulur and Sahin, 2012).
3. Optical fibre sensors are able to transmit signals over long distances and are immune to electromagnetic interference (Bock et al., 1992).
4. Engineers can perform multi-point or distributed monitoring using a single optical fibre (Wild et al., 2010).

When using optical fibre for structural monitoring, engineers normally use

single-mode optical fibres rather than multi-mode fibres (in comparison with multi-mode fibres, single-mode optical fibres have the advantage of thinner fibre cores, wider bandwidth with high transmission and longer transmission distance) (Rivera et al., 2002). However, because of the need for a laser light source, the implementation cost of the single-mode optical fibre is relatively higher than the multi-mode optical fibre (Jeunhomme and Thompson, 1989).

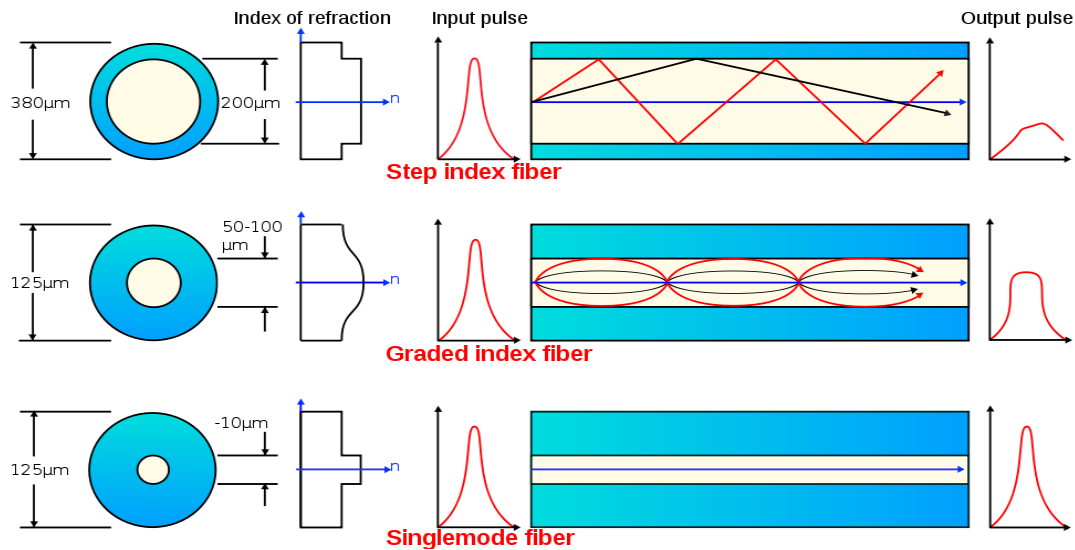


Figure 2-5: Illustration of different optical fibres. The graphs shown on the right hand side present the attenuation of signal strength to each type of optical fibre.

(http://www.enotes.com/topic/Optical_fiber).

In terms of using optical fibres for structure monitoring, the optical fibre needs to be processed into an optical fibre grating (also known as Fibre Bragg Grating - FBG). The fabrication of an optical fibre grating typically involves the illumination of the core material with a high-energy laser beam to induce permanent periodical change of refractivity of a section of the optical fibre.

When the sensor is influenced by external stress or temperature variations the wavelength of reflected light from the FBG changes accordingly. Engineers can utilise this information to analyse the local deformation of target structures (Ma and Yang, 2010).

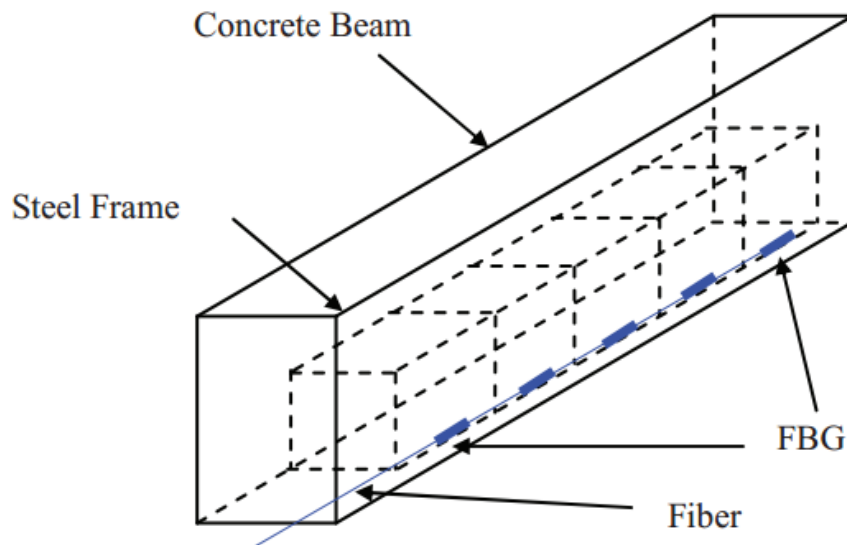


Figure 2-6: Reinforced concrete beams with FBG sensors (Junjie et al., 2011).

Figure 2-6 illustrates the configuration of using FBG for civil structural monitoring. FBG sensors were evenly distributed along the concrete beam to provide real-time monitoring. When using the FBG measurement for structural monitoring, the FBG sensors need to reduce the sensitivity to avoid receiving excessive noise. Using these sensors to monitor a 20mm concrete crack gives a return of accuracy of approximately 0.002 mm (Rodrigues et al., 2010).

When embedding the FBG sensors into the target structures, there is an emphasis that cannot be ignored – the operator should avoid excessive bending of fibre-optic cables as it causes energy loss of the light source (Li et al., 2011c). As mentioned above, this measurement method normally requires the burying of FBG sensors into the target objects. During the covering process, the operators cannot precisely control whether those sensors have been excessively warped or not. This uncertainty factor tends to take over as the major source of inaccuracy (Schukar et al., 2012).

Figure 2-7 shows another application of the use of FBG measurement to monitor the deflections of a concrete bridge. FBG sensors and other transducers are evenly embedded into concrete (in the bottom and top slab of the box girder) for long-term monitoring. Engineers can use real-time information such as the direct mid-span

vertical displacement measurement, the strain measurement and the bending measurement to evaluate the structure global deformed shape. The acquisition rate can be increased up to 500 Hz. However, these sensors and transducers are permanently buried in the concrete bridge. It is very difficult to maintain without undermining the structure and this is the most common problem in terms of using the FBG measurement for structural monitoring.

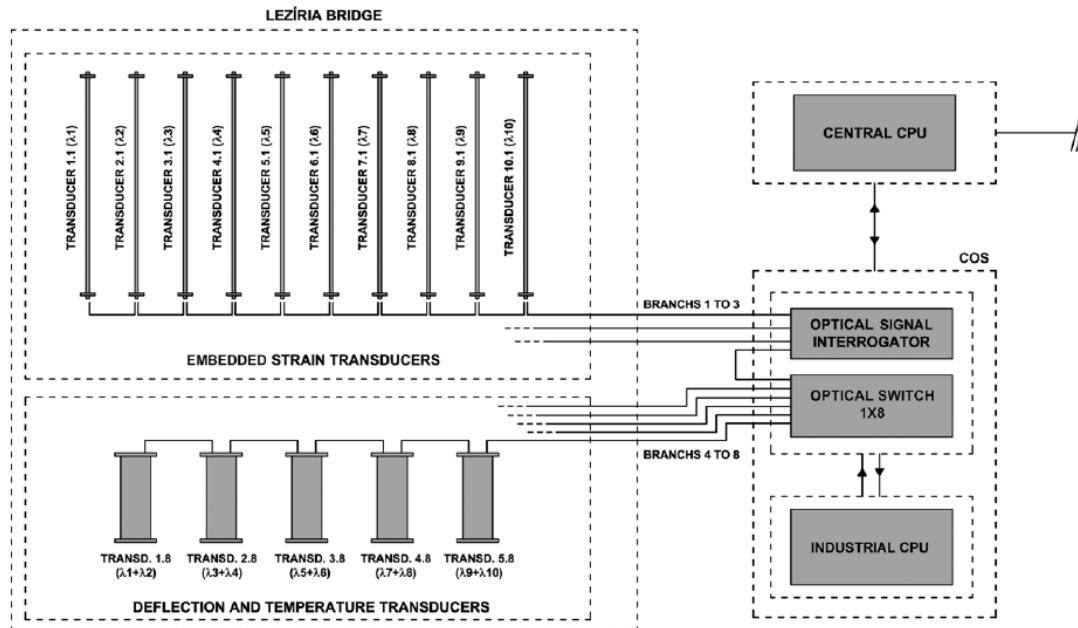


Figure 2-7: Fibre optic based sensor network architecture for a concrete bridge monitoring (Rodrigues et al., 2010).

2.2.2 Geodetic measurements

Compared with direct measurement techniques, geodetic measurements tend to be adopted when measuring large objects or wide areas (Min, 2008) and this is one of the major advantages in terms of using these measurement methods for structural health monitoring. The instruments of this measurement method are normally larger and heavier than in the methods that have been described above. However, geodetic measurements are the most frequently used methods in civil engineering because the required instruments are the most commonly used equipment in general construction and civil engineering (Karbhari and Ansari, 2009). Total stations are one of the most

familiar measuring instruments in civil engineering because this instrument is very common in general construction sites (Uren and Price, 2010). For this reason, this measuring instrument has become the most frequently used tool for construction site monitoring (Irvine and Maclennan, 2005).

2.2.2.1 Total stations

When construction work is in progress, large amounts of underground water will be pulled out of the ground and overexploiting underground water is a major cause of land subsidence and causes buildings to shift (or tilt). In addition to this, when the load is gradually increased onto the ground, the engineer needs to check buildings regularly to ensure these buildings and the foundation of the construction site can withstand the extra weight and has not deformed (Cooke and Williams, 2009). The total station can be used to perform three-dimensional coordinate measurements. Some of the latest models of total stations have been able to perform automated operation and can be remotely controlled. In general, surveyors need to choose stable locations to establish permanent observation points where they can still allow the operators to use a total station to observe the target objects (as shown in Figure 2-10). The operator regularly mounts the total station onto each observation point to measure the angle data from each target and to calculate the coordinates of targets using space intersection. By comparing the coordinate data of each target, which were measured in different periods, the deformation of the target structure can be obtained. Some of the latest total stations are capable of automatic target recognition and the manufacturers also provide different software packages that allow users to perform the necessary coordinate calculation automatically. These improvements can significantly ameliorate efficiency and reduce manpower costs.

Leica Geosystems (2010) states that, through the use of a remote control, the accuracy for bridge and large structural surveying applications is better than $\pm(2mm+2ppm)$. Lutes and et al. (2001) used several robotic total stations, GPS antennas and other sensors to automatically monitor the displacement of two dams in the Diamond Valley Lake. In order to perform long-term observation, eight Leica TCA1800 robotic total stations were permanently mounted on dedicated shelters and the configuration of

instruments is shown in Figure 2-8. The experimental results showed that these robotic total stations are capable of detecting displacements of targets on the downstream faces of the dams with an accuracy of 10mm at the 95% confidence level.

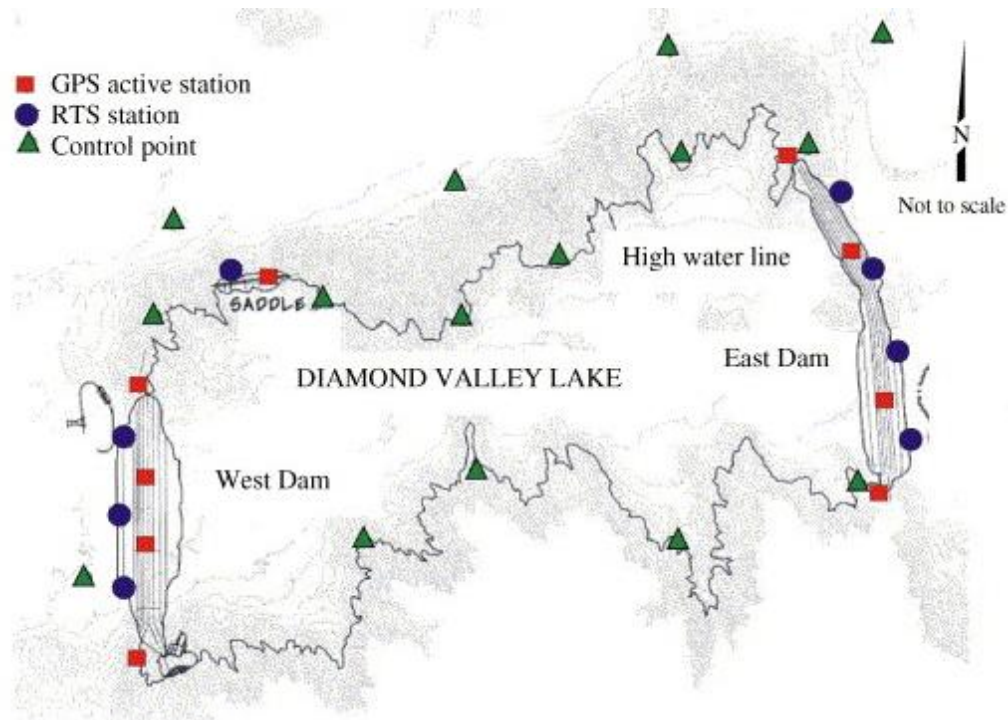


Figure 2-8: Geodetic monitoring scheme at Diamond Valley Lake (Lutes J. et al., 2001)

Xu and Hu (2009) used a robotic total station (Leica TCA2003) to perform dynamic structural monitoring of a long span cabled-stayed bridge. As shown in Figure 2-9, a fixed prism was permanently set on the box girder. One observation pier was located on the pylon of the same bridge and the distance between the observation pier and the prism was approximately 540m. In terms of the performance for moving object monitoring, the robotic total station was capable of automatic target recognition with a sampling frequency of around 3Hz data rate. Moreover, the system can detect abnormal deformation by means of analysing the hopping phenomenon of the slope distance measurements and vertical angles. However, it was difficult to have a clear line of sight between the total station and the prisms in harsh environmental condition and this disadvantage may decrease the sampling rate and significantly affect the

observation accuracy. The atmospheric refraction problem is another factor that affects the accuracy when using a total station for long range observation (Anna et al., 2005).



Figure 2-9: Illustration of the use of a total station for bridge structure monitoring (Xu and Hu, 2009).

Though robotic total stations can be used to perform automatic measurements, they are more expensive than conventional total stations. In addition, in order to drive these instruments for different automatic applications, specific software suites are needed. Apart from the disadvantages that have been mentioned above, both of these experiments point out that a total station lacks the ability of simultaneous multi-target observation. If the target object is solid, the surveyor can sequentially perform observation on each point. When the target structure is rapidly deformed, each total station can only focus on one single target to ensure the observation station will not lose any details of the target's movements. Thus, the practicability of using this measurement for flexible structural monitoring will be undermined.

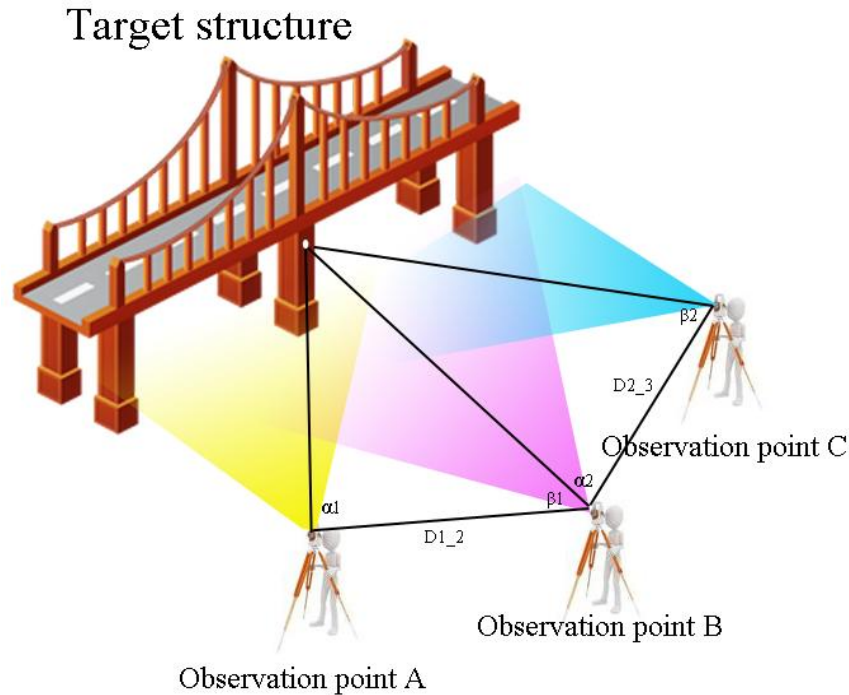


Figure 2-10: Configuration of observation points (Karbhari and Ansari, 2009).

2.2.2.2 Global navigation satellite system - GNSS

In comparison to measurements made through the use of total stations, the data collection and data transmission processes of GNSS measurements are fully automatic. Therefore, GNSS measurements can be seen as a type of in-situ, automated system. The collected data can be remotely transferred back to the remote database and most of the receiving antennas are weatherproof. Therefore, the GNSS measurement method can be used in those test sites that are in remote locations (Sariff et al., 2011).

When using GNSS for long-term structural monitoring, most manufacturers have developed their own monitoring packages (including GPS antennas, receivers, controllers and software packages) and the measurement methods are slightly different as they depend on what kind of system the operator has chosen. However, if surveyors do not use any commercial package Deng (2003) proposed a solution that allows surveyors to detect the vertical movements of each remote target using basic GPS data. In terms of the calculation of the orthometric height, the data that was sent

from each GPS antenna has to be analysed first. The height information of GPS is ellipsoidal height which based on the reference ellipsoid. However, in engineering surveying, the surveyors are normally interested in the orthometric height rather than ellipsoidal height (Wu et al., 2010). Theoretically, for the same target point, the difference between the ellipsoidal height and the orthometric height is identical; assuming the operator has received the GPS data from the same measuring station twice. Equation (2.1) shows the relationship between the ellipsoidal height and the normal height (orthometric height) in these two periods:

$$\Delta H = \Delta h + \xi \quad (2.1)$$

Where:

ΔH is the difference of ellipsoidal height during these two periods;

Δh is the difference of orthometric height;

and ξ is the height anomaly (to the same measuring station, the ξ can be deemed to be zero).

The equation (2.1) outlines that the ΔH is essentially proportional to Δh . From a monitoring point of view, the surveyors can use the changes of ellipsoidal height to estimate the movements of each target in direction that is perpendicular to the ground (i.e., the surveyors do not need to convert the ellipsoidal height into the orthometric height). Currently, the two-line orbital element (TLE), which is provided by The International GNSS Service (IGS), has been able to ensure the quality of ellipsoidal heights (Diggelen, 2009). Thus, through the use of this method, the accuracy of vertical displacement measurement can achieve 1 mm (the longest baseline is approximately one kilometre) (Deng, 2003). This measurement has been applied to the actual engineering environment to monitor objects such as dams (Hu and Liu, 2011; Li and Wang, 2011; Xinzhong and Chuanxi, 2011), landslides (Baldi et al., 2008; Gili et al., 2000), etc.

In terms of using GNSS measurements to monitor civil engineering infrastructure, GPS antennas can be permanently fixed onto a structures' surface to collect real-time

data (Figure 2-11). These data can be processed to draw a variety of real-time curves a, such as a dam displacement curve, animation.

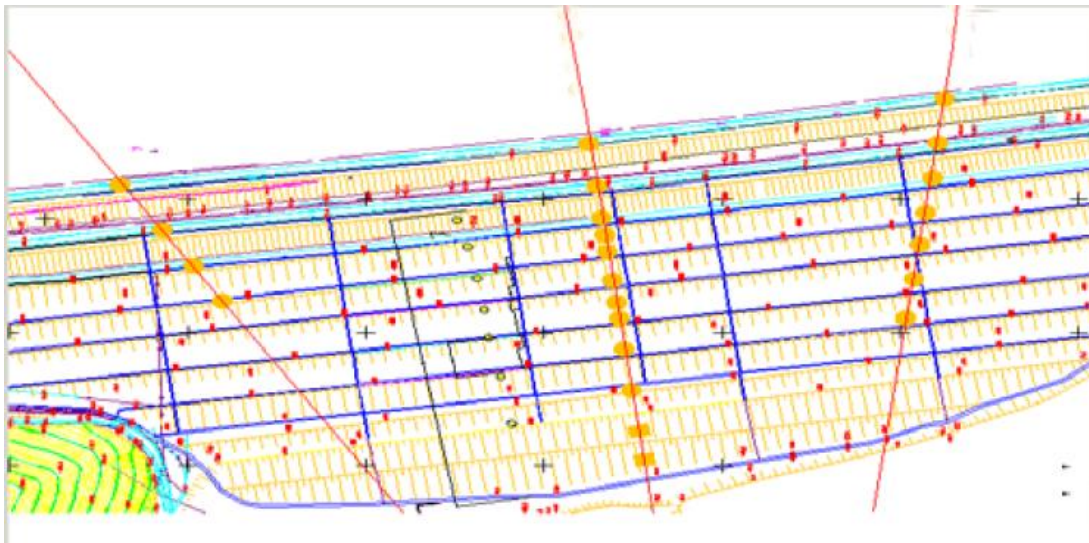


Figure 2-11: Network configuration of GPS and photogrammetric stations for a dam monitoring project. A total amount of 478 GPS antennas (red dots) and 23 imaging stations (orange dots) were evenly distributed on the test area (Li and Wang, 2011).

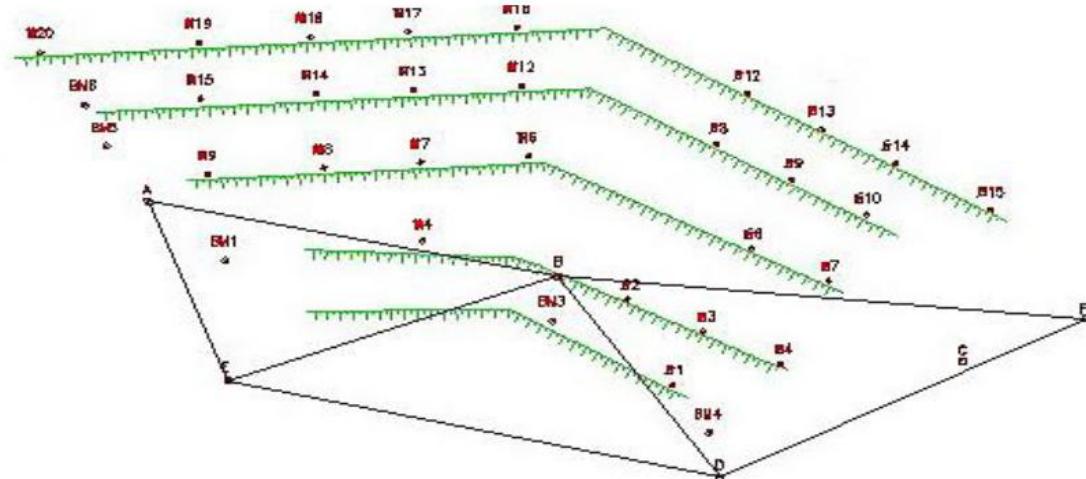


Figure 2-12: Network thumbnail of GPS stations for a dam monitoring project. A total of 32 GPS antennas were arranged on the side wall to record the reservoir water level effects of a dam (Hu and Liu, 2011).

In terms of using GNSS measurements for solid structural monitoring applications,

the application practice shows that the horizontal and vertical precision is better than 1.5 mm for broadcasting ephemeris in 1-2 hours, and vertical precision is approximately 1.5 mm (Li and Wang, 2011). However, when using GNSS measurements to monitor flexible structures (such as long span cabled-stayed bridges), the horizontal and vertical precision will reduce to approximately 10 mm and the vertical precision to about 20 mm (Axelsson, 1999). In addition, GPS signals are easily blocked by some outside forces, such as trees, caves, buildings, etc. Also, GPS data need to be processed and analysed by professionals (Akkose et al., 2008). Engineers need to be aware of these restrictions before application.

2.2.2.3 Terrestrial laser scanning-TLS

The application of terrestrial laser scanning technology for structural monitoring provides an efficient solution for obtaining a target objects' 3D information. The use of terrestrial laser scanning technology for structural monitoring has the following advantages:

1. It is a non-contact 3D measurement (Syahmi et al., 2011; Kaasalainen et al., 2008);
2. Distance measurements do not require prisms (Xinlian et al., 2012) and operators do not need to attach any target or reflector on a target object's surfaces;
3. This measurement can generate a huge number of 3D point clouds in a short period (i.e., high data capture rates) (Yinghui et al., 2007).

Unlike previous point-based approaches, terrestrial laser scanning captures a continuous surface of target structures. Engineers can compare these point cloud models, which are captured at different periods, to detect slight changes of target structures. Therefore, terrestrial laser scanning measurements have been widely used to monitor large, solid target objects such as a dam (Alba, 2006), landslide sites (Syahmi et al., 2011; Chen et al., 2010), tunnels (Xiongyao and Chenchao, 2012)). In comparison to other optical measuring systems, terrestrial laser scanners can be operated in low-light environments due to the active illumination of the laser light

(Kaasalainen et al., 2008). A terrestrial laser scanner is a illumination-independent measuring apparatus (Mora et al., 2003). Therefore, it can be performed during the night time, in tunnels (Figure 2-11), etc. This is an advantage for long-term monitoring applications (especially for those projects that need to be performed for more than 24 hours).



Figure 2-13: 3D laser scanning for tunnel structure monitoring (Xiongyao and Chenchao, 2012).

Terrestrial laser scanning technology is an efficient solution for a large range of monitoring projects. As shown in Figure 2-14, a 60 m (L) x 8 m (W) tunnel 3D model can be assembled from 11 aligned scans. The scanning process in each station can be completed within just thirty minutes (density acquisition at 100,000 points per second). The overall positional accuracy was 6 mm whilst the distance accuracy was less than 4 mm at distances up to 25 m (Fekete et al., 2010).

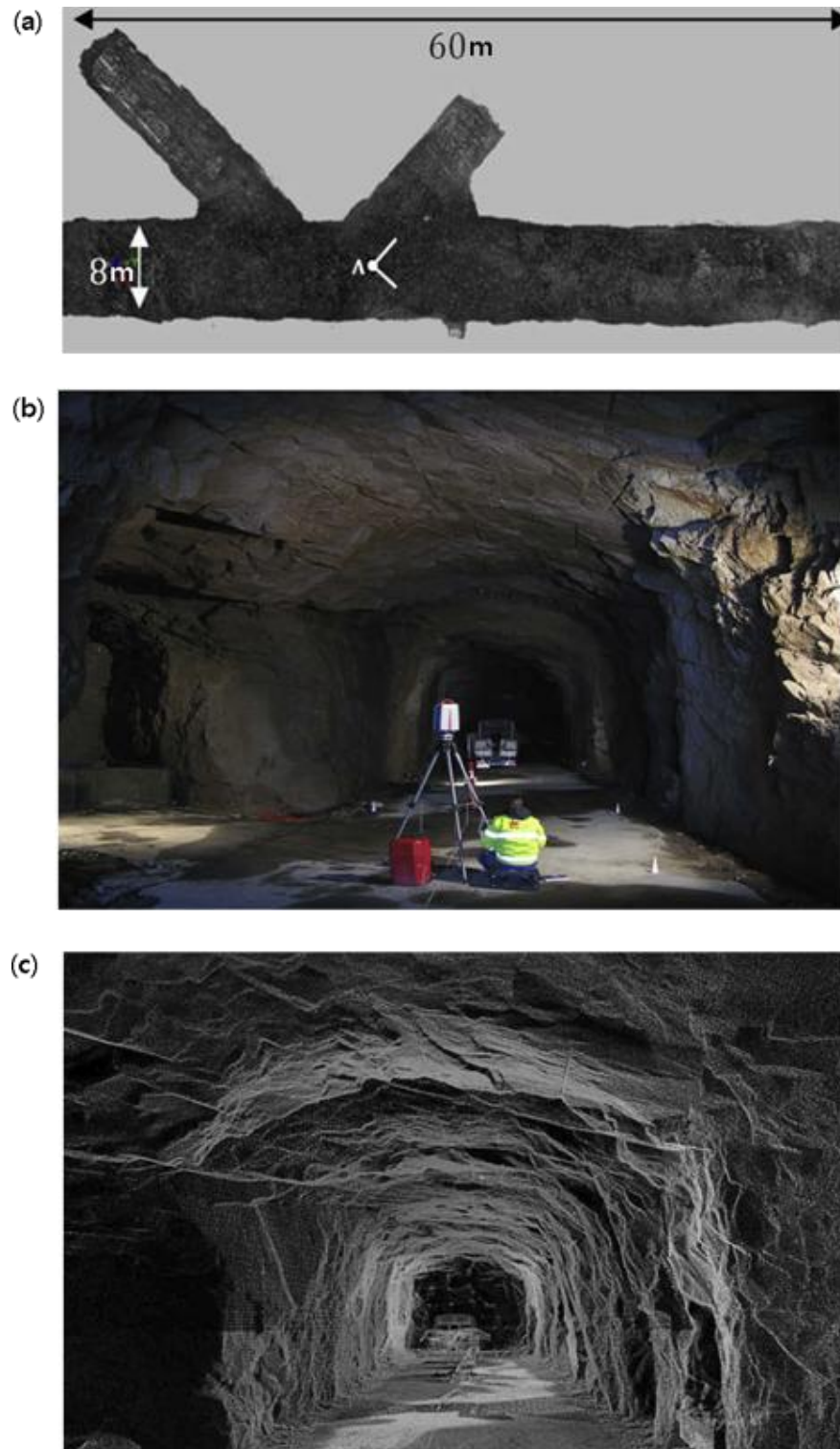


Figure 2-14: Illustration of using terrestrial laser scanning technology for tunnel 3D structural modelling: (a) shows the structure of the test tunnel; (b) shows the configuration of a laser scanner and (c) is the scanning sample in viewpoint A (Fekete et al., 2010).

Figure 2-15 shows another example of using terrestrial laser scanning measurements to conduct SHM for a large civil structure. In comparison with previous point-based measurements, the use of terrestrial laser scanning measurements shows the advantage of a wide-area coverage sampling ability. The 3D model of the whole surface of the dam (frontage) and the downstream can be generated using only seven-station aligned scans. Comparing the reference points, which were measured by a total station, the accuracy resulted in ± 4 mm in the X and Y-axes whilst the accuracy in the Z-axis was ± 8 mm (Alba, 2006).

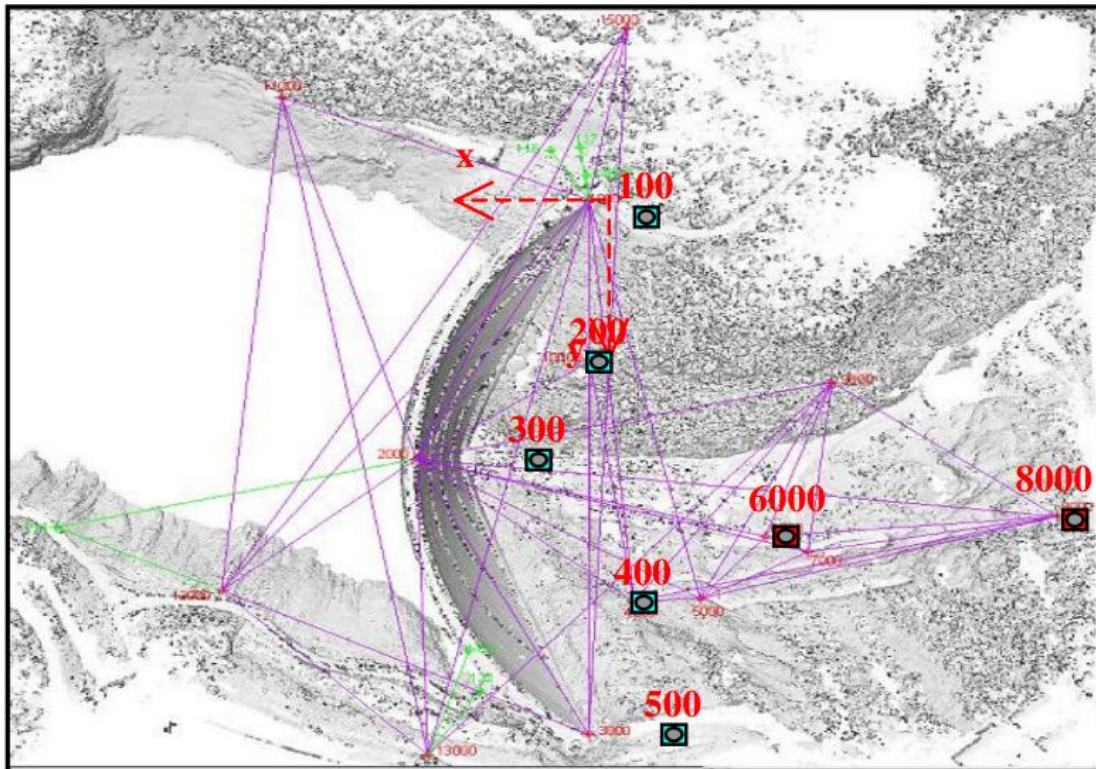


Figure 2-15: Configuration of the geodetic network reporting laser scanner stand-points for a dam monitoring project (Alba, 2006).

According to the differences of required resolutions, a complete scanning process typically takes a few minutes (Ghazali et al., 2011). Although a huge number of 3D points can be captured from target objects during this period, the longer sampling time has become a major disadvantage for flexible structure monitoring (i.e., the deformation of target structures cannot be instantaneously captured by a terrestrial laser scanner) (Soloviev and De Haag, 2010). In order to resolve this problem, Lin et

al. (2008) combined terrestrial laser scanning and videogrammetry to increase the sampling rate. In this experiment, terrestrial laser scanning was used to produce an as-built model of a membrane roof structure (Figure 2-16) and this as-built model was statistically compared with a mathematical model using a surface matching approach to inspect the difference between the original design and the actual structural behaviour. The amalgamated model had a registration error of 6 mm. In terms of the implementation of membrane structure monitoring, three consumer-grade digital video (DV) camcorders (JVC GRD30U, Sony DCR-PC101 and Sharp VL-Z1) were used to capture continuous deformation of the flexible roof structure. After photogrammetric processing, the overall measurement precision was approximately ± 7 mm.

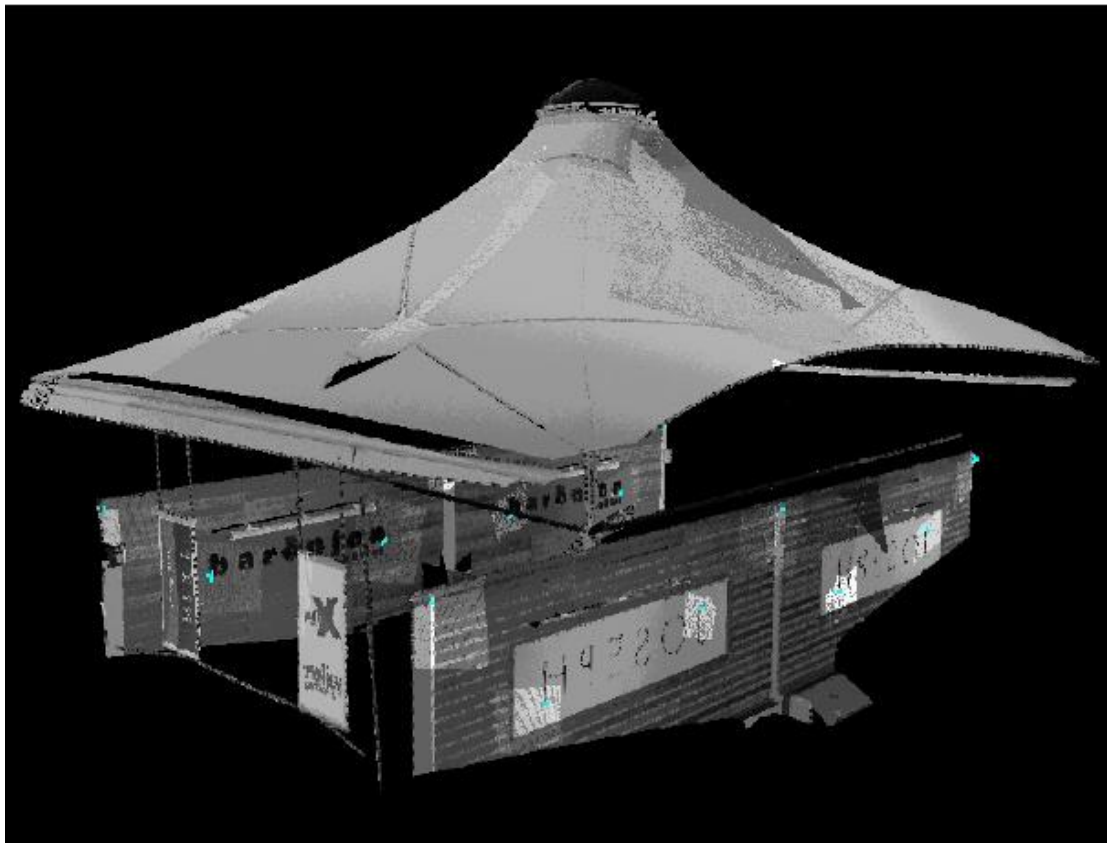


Figure 2-16: Laser scanned model of the membrane structure.

Terrestrial laser scanning measurements have shown excellent monitoring capability for large, solid civil structures. However, the price of a terrestrial laser scanner is

much higher than previous point-based instruments; also, the weight of the equipment may be too heavy for carrying to remote locations in the field. When performing a monitoring project in a rugged terrain, the surveyors have to move the scanner between multiple observation sites (Figure 2-15) and the oversized (or heavy) instruments will increase operational difficulties (Cooke and Williams, 2009). Furthermore, because these scanners are expensive, the operators cannot leave them in the observation sites for long-term observations, although some of the models can be remotely controlled. In these circumstances, the use of terrestrial laser scanning technology for long period structural monitoring is not as popular as the use of total stations and GNSS in practical engineering applications (Ansari, 2005). In terms of structural monitoring, sometimes surveyors need to focus on specific parts of the building for more in-depth analysis (Irvine and MacLennan, 2005). However, laser scanning has the difficulties with identifying features within the cloud of points (Axelsson, 1999). Therefore, it is sometimes difficult to use the results of terrestrial laser scanning measurements to precisely analyse the localised deformation of a structure (Alba, 2006).

2.2.3 Close range photogrammetric measurements

The implementation of photogrammetry presents the process of transforming images into spatial three-dimensional information (Feng, 1991). Through photogrammetric procedures, users can reconstruct the 3D geometry of objects from images (Li et al., 2011a). Following the work flow shown in Figure 2-17, the 3D coordinates of target objects in space can be calculated from corresponding 2D information extracted from recorded photographs through photogrammetric processing. The benefits of the implementation of photogrammetric measurements are listed below:

- The use of multiple imaging stations are suitable for being used to perform large area measuring projects (Baldi et al., 2008);
- The use of photogrammetric measurement is a relatively inexpensive monitoring solution (Westoby et al., 2012);
- Photogrammetric measurements are non-contact observations, the spatial and

geometric measurements can be accomplished without contacting the measured objects (Feng, 1991).

- The ability for rapid data acquisition (Baltsavias, 1999).

Like terrestrial laser scanning measurements, photogrammetry measurements do not need to contact the target object to implement measurement, which means these two measurements are especially useful in the conditions where the object to be measured is inaccessible. Moreover, both of these measurements can use a few sampling stations to cover a large test area (Mora et al., 2003). In addition, photogrammetric image acquisition devices are normally much cheaper than terrestrial laser scanners or other optical measuring equipment (such as a total station). Baltsavias (1999) made a comprehensive comparison of these two measurements and stated that the implementation costs of laser scanning measurements are normally seven to ten times more than photogrammetric measurements in terms of hardware spending. Therefore, photogrammetric measurements can be regarded as a cost-effective approach. Diego (2009) stated that photogrammetric measurement has the feature of rapid data acquisition and this advantage is especially useful when applying the photogrammetric measurement to monitor rapid deformation objects.

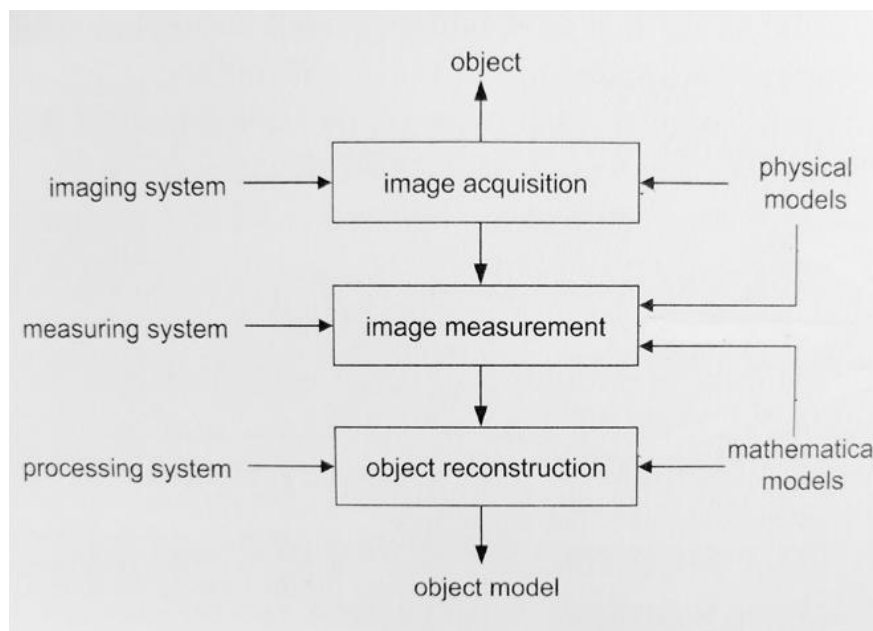


Figure 2-17: Photogrammetric process: form object to model (Luhmann et al., 2007).

By camera position and object distance	
-Satellite photogrammetry	Processing of satellite image, $h > ca. 200$ km
-Aerial photogrammetry	Processing of aerial photographs, $h > ca. 300$ m
-Terrestrial photogrammetry	Measurements from a fixed terrestrial location
-Close range photogrammetry	Image distance $h < ca. 300$ m
-Macro photogrammetry	Microscope imaging, image scale > 1
By number of measurement images	
-Single image photogrammetry	Single image processing
-Stereo photogrammetry	Dual image processing
-Multi-image photogrammetry	N image where $N > 2$
By method of recording and processing	
-Plane table photogrammetry	Graphical evaluation (until <i>ca.</i> 1930)
-Analogue photogrammetry	Analogue cameras, opto-mechanical measurement systems (until <i>ca.</i> 1980)
-Analytical photogrammetry	Analogue images, computer-controlled measurement
-Digital photogrammetry	Digital images, computer-controlled measurement
-Videogrammetry	Digital image acquisition and measurement
-Panorama photogrammetry	Panoramic imaging and processing
-Line photogrammetry	Analytical methods based on straight lines and polynomials.
By availability of measurement results	
-Real-time photogrammetry	Recording and measurement completed within a specified time period particular to the application
-Off-line photogrammetry	Sequential, digital image recording, separated in time or location from measurement
-On-line photogrammetry	Simultaneous, multiple, digital image recording, immediate measurement

Table 2-1 Categorisation of photogrammetry (Egels and Kasser, 2002).

Photogrammetry can be categorised in a multitude of ways. Table 2-1 shows a categorisation of general photogrammetry. According to different requirements, engineers can choose different photogrammetric techniques to solve a variety of engineering problems. In the field of civil engineering, close-range photogrammetric measurements are widely applied for structural modelling, architecture reconstruction, structural profile measurement and deformation monitoring, etc. (Egels and Kasser, 2002). Focusing on structural monitoring applications, each imaging station of a photogrammetric system can capture instantaneous photograph (or video stream).

Combined with long-lasting power supply and stable operating performance, photogrammetric measurements are suitable to perform routine structural health monitoring and inspection (Poudel et al., 2005).

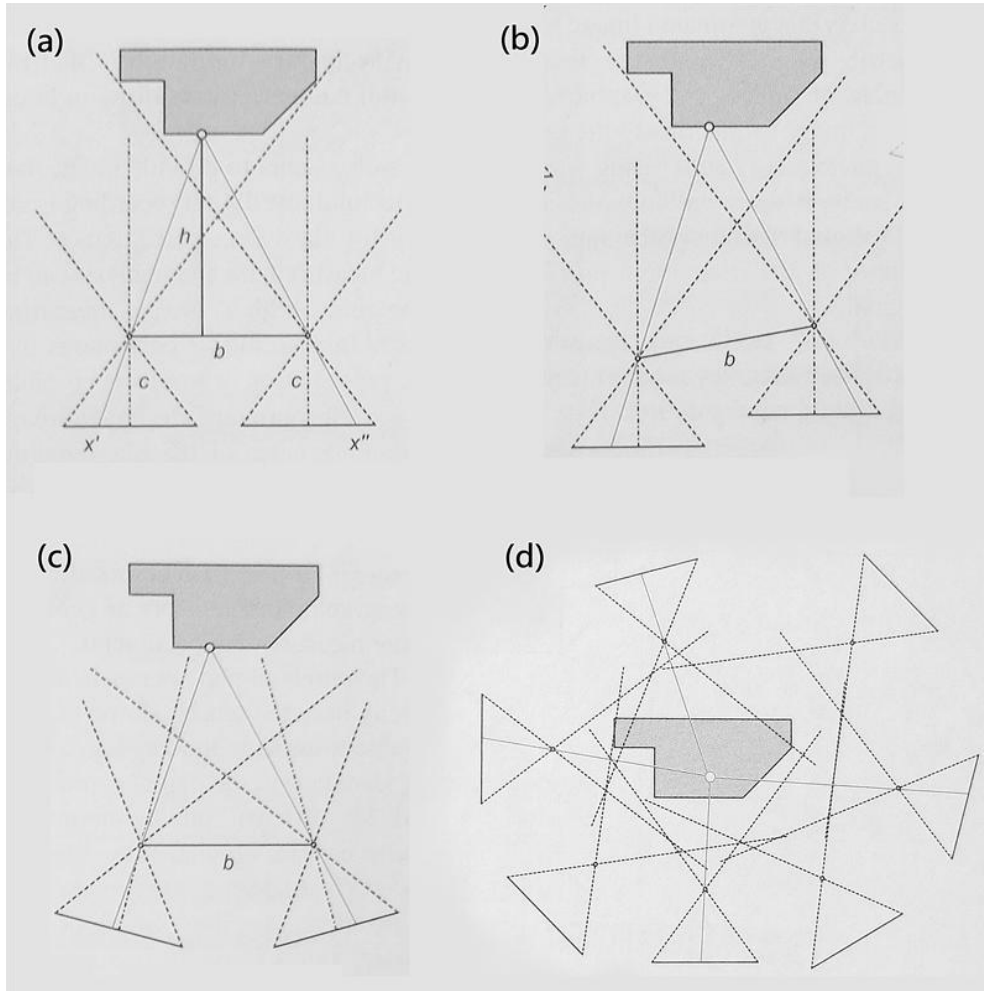


Figure 2-18: Stereo image configurations. (a) Parallel configuration; (b) shifted configuration; (c) convergent and (d) all-round configuration (multiple-image acquisition) (Luhmann et al., 2007).

In terms of the use of multiple imaging stations for structural monitoring applications (Figure 2-18 (d)), network design plays an important role in the optimisation of measurement performance (Ethrog, 1987). Figure 2-18 illustrates some of the most frequently used photogrammetric network configurations. As shown in Figure 2-18 (a), imaging stations are arranged in parallel in front of the target object and the collimation axis of each imaging device is also parallel. Like parallel configuration,

the collimation axis of each imaging device of shifted configuration is parallel. However, the image scale of each imaging station is different. In order to seek more overlap within image pairs, surveyors normally apply convergent configuration (as shown in Figure 2-18 (c)) for photogrammetric network configuration. The all-round configuration (Figure 2-18 (d)) is derived from the convergent configuration by applying more than two imaging stations to surround a target object. In close-range photogrammetry, the convergent configuration and the all-round configuration are the most commonly used configurations for structural monitoring and modelling purposes (Arias et al., 2007; Luhmann et al., 2007).

To further investigate the relationship between the photogrammetric measurement precision and imaging station configurations. Figure 2-19 shows the stereo image configuration for a close range photogrammetric measurement.

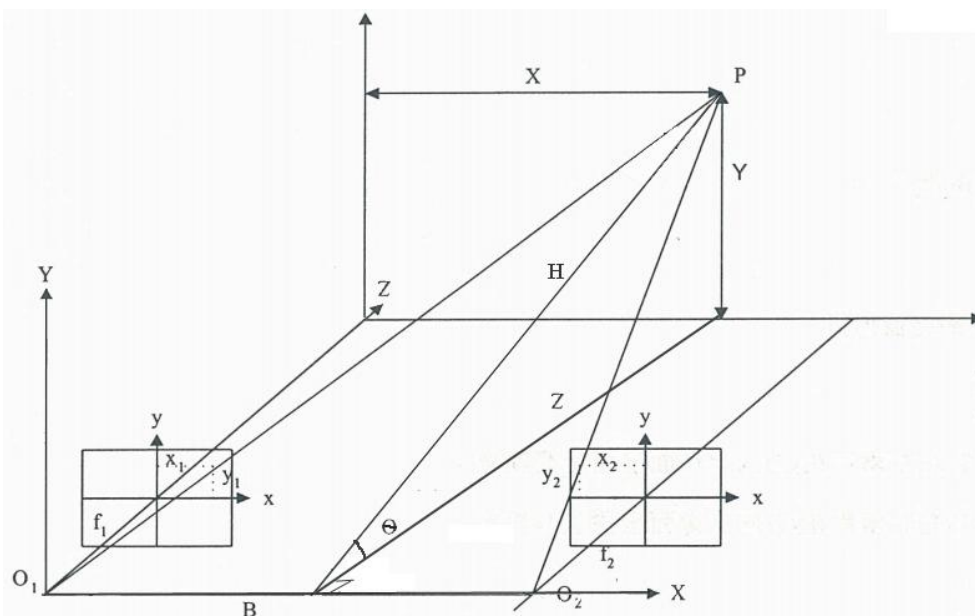


Figure 2-19: Stereo image configurations. (Marshall, 1989)

Where O_1 and O_2 are the locations of two imaging stations, f_1 and f_2 are the focal length of two cameras, B is the distance of two image stations (i.e., the baseline), P is the object point and (x_1, y_1) and (x_2, y_2) are the 2D coordinates of point P on two images, the 3D coordinate of point P is (X, Y, Z) and H is the object distance.

Assume both of these two cameras have an identical focal length (i.e., $f_1=f_2=f$); then, the measurement error in the object system are represented by M_X, M_Y and M_Z .

$$M_X = \pm \frac{H \times \cos \theta}{B \times f} \times \sqrt{X^2 + (X - B)^2} \times m_x \quad (2.2)$$

$$M_Y = \pm \frac{H \times \cos \theta}{f} \times \sqrt{0.5 + 2\left(\frac{Y}{B}\right)^2} \times m_y \quad (2.3)$$

$$M_Z = \pm \frac{\sqrt{2} \times (H \times \cos \theta)^2}{B \times f} \times m_x \quad (2.4)$$

(Marshall, 1989)

The equations (2.2), (2.3) and (2.4) outlined the object distance causing different effects to M_X, M_Y and M_Z . If the distance between cameras (baseline) is fixed, the measurement errors in X and Y axes (i.e., M_X and M_Y) are proportional to the object distance. However, the measurement error in Z-axis is proportional to the square of the object distance. Meanwhile, the length of baseline is inversely proportional to the M_X, M_Y and M_Z . Therefore, to improve the photogrammetric results by adjusting the relative positions between the target objects and each imaging station can be another effective method.

When planning a photogrammetric network, the quantity of imaging stations has become an important issue in terms of both accuracy (network strength) and economy (Mason, 1995). In considering the budget control, surveyors need to use the most economical configuration to create a strong photogrammetric network. Theoretically, network geometry is established with various numbers of cameras and their positions employed. Performing the photogrammetric measurement in strong network geometry results in higher triangulation precision (Fraser and Edmundson, 2000). Figure 2-20 shows the relationship between photogrammetric network geometry and related triangulation precision. This test presented the results of using three different photogrammetric networks to measure the same target array and the error ellipses of the measured points are illustrated alongside. Among these three types of

configurations, the configuration of two-station has relatively low precision. In comparison to the two-station configuration, the measurement precision significantly improved when the quantity of imaging stations were increased from two to four. The most optimal result is shown in the third network (four-station network with strong convergence), which is considered as “strong” network geometry. The results outline that the quantity of imaging stations and the photogrammetric network configuration are two key factors related to the improvement of overall photogrammetric measuring precision.

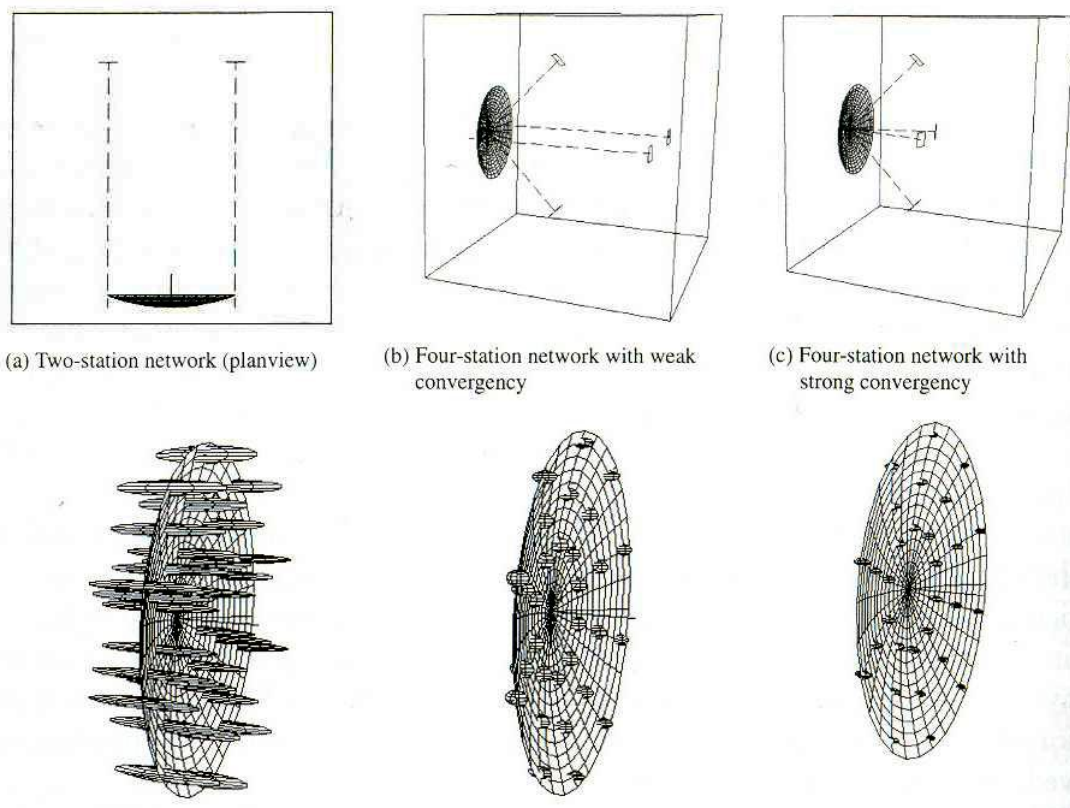


Figure 2-20: Relationship between network geometry and related photogrammetric precision (Atkinson, 2001).

Apart from the quantity of imaging stations and photogrammetric network geometry, equation (2.5) lists some indicators that may also affect the photogrammetric precision (Fraser, 1981).

$$\sigma_c = \frac{q}{\sqrt{k}} \frac{d}{c} \sigma \quad (2.5)$$

Where σ_c is the RMS (root-mean-square) value of XYZ object point coordinate; d is the mean object distance; c is the focal length of lens (thus $\frac{d}{c}$ is the given scale ratio); σ is the image coordinate standard error; q is design factor expressing the strength of the camera station configuration (stronger network geometry is represented by smaller q value); and k is the average number of exposures at each station.

Following the equation, increasing the number of exposures at each station and using smaller scale ratios can improve the photogrammetric precision as well.

In addition to these factors mentioned above, how to use a limited number of stations to create a highly convergent multi-image network is an important issue to improve the photogrammetric precision (it is also important for budget control) (Marshall, 1989). For this reason, the operators have to calculate the field of view (abbreviated FOV) of each imaging station. Thus, the range of coverage of each camera station can be estimated. The FOV computational formula is shown below:

$$FOV = 2 \times \tan^{-1} \left(\frac{L}{2 \times f} \right) \quad (2.6)$$

Where:

L is the maximum distance between two corners of imaging sensor (i.e., the length of the diagonal); f is the focal length of lens (Luhmann et al., 2007).

In terms of using close-range photogrammetric measurements for actual engineering applications, Yilmaz et al. (2007) applied this measurement method to reconstruct an earthen archaeological site and the generated precision was $\pm 2\text{mm}$. As shown in Figure 2-21 (a), a large amount of control points were arranged onto the structural surface. A camera took photographs around the target building to create a sequence of overlapping images.

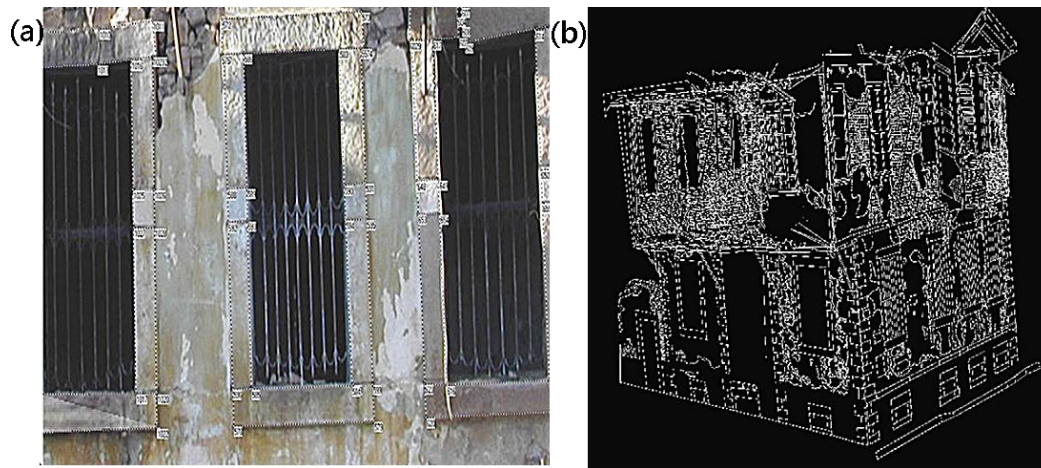


Figure 2-21: Configuration of control points and 3D modelling (Yilmaz et al., 2007).

After photogrammetric processing, the 3D coordinate of each control point was computed. Engineers used these data to reconstruct the building contour and compute the orthophoto images for surface patching purposes. This research highlights a common problem of close-range photogrammetric measurements (or optical measurements) – photogrammetry cannot create the objects that do not appear in the original images. The 3D model lacked information about the housetop because the researcher did not provide any image that contained the roof for photogrammetric processing. Moreover, if an object is obscured in the original image data, this object will not be displayed in the final photogrammetric result. Therefore, the characteristic of intervisibility between stations and object is essential to the photogrammetry measurements. In terms of the limitation of optical measuring instruments, the photogrammetric imaging stations cannot be operated under low-light or dark environments (Atkinson, 2001).

In addition to using photogrammetric measurements for structural 3D modelling (or reconstruction), structural monitoring is another important application.

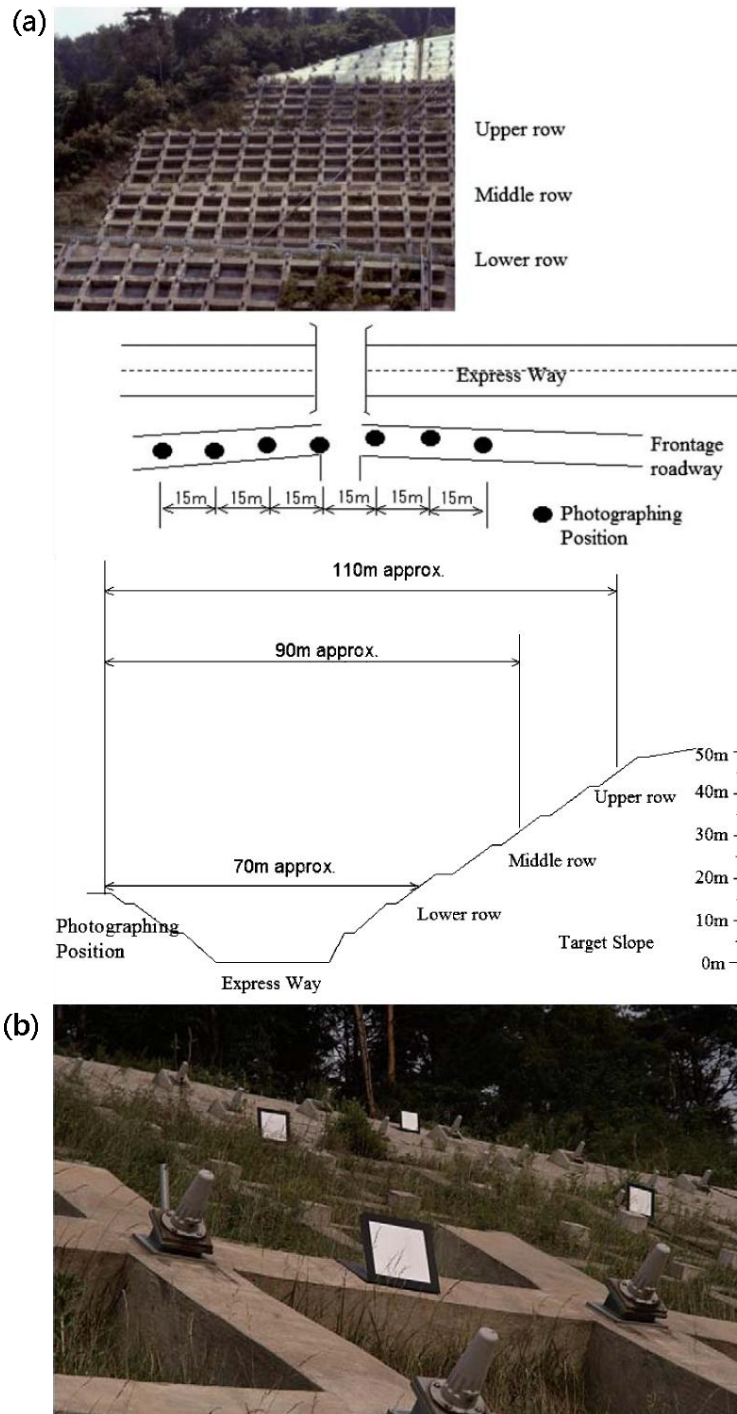


Figure 2-22: Slope monitoring using photogrammetric measurements. (a) Location of monitoring points and target slope; (b) photographic view of measurement points (Ohnishi et al., 2006).

Figure 2-22 shows an application of using a close-range photogrammetric approach for long-term slope monitoring. As shown in Figure 2-22 (b), a total of 432

photogrammetric targets were permanently fixed onto a concrete brace wall whilst eight imaging stations were arranged in the foot of the slope (Figure 2-22 (a)). During the experimental period, researchers periodically used one camera to capture photographs at eight different imaging stations. In this test, the object distances changed in the range from 70 m to 110 m. In order to determine the relationship between photogrammetric measurement precision and scale ratio (i.e., the $\frac{d}{c}$ value in equation (2.5)), the research group used two types of lenses with different focal lengths (35 and 50 mm) to create different combinations of scale ratio. This solid structural monitoring project delivered the overall measurement precision in horizontal, vertical and depth directions; 2.19 mm, 1.44 mm and 2.40 mm, respectively whilst the best relative accuracy of distance measurement can reach the level of 1/12,000. The experimental results also confirmed that the precision was reduced in proportion to the increasing scale ratio.

Besides improving photogrammetric results by changing the scale ratio, intensifying the photogrammetric network configuration is another effective approach (i.e., to minimise the factor q in equation (2.5)). Jiang and Jauregui (2010) used a close-range photogrammetric technique to develop a bridge deflection measurement system. During the experimental period, the research group had strictly inspected the changes of photogrammetric precision by using different camera network configurations. As shown in Figure 2-23 (b), three camera placement options were used in the field test. Option_1 arranged one row of five camera stations on the ground. Option_2 added another row of camera stations at the same elevation as Option_1, but on the other side of the bridge; and Option_3 added one row on the same side with Option_1, but at different height.

By comparing the photogrammetric results with the measurements, which were measured by dial gages, the absolute average difference of Option_1 is 1.18 mm, the absolute average difference of Option_2 is 0.29 mm and the difference of Option_3 is 0.54 mm (the average object distance is about 5.2 m). Cameras, which were arranged at Option_1 and Option_3 captured the same group of targets. The results of Option_3 are approximately two times better than Option_1, which means the use of stronger

network geometry can significantly improve the photogrammetric result.

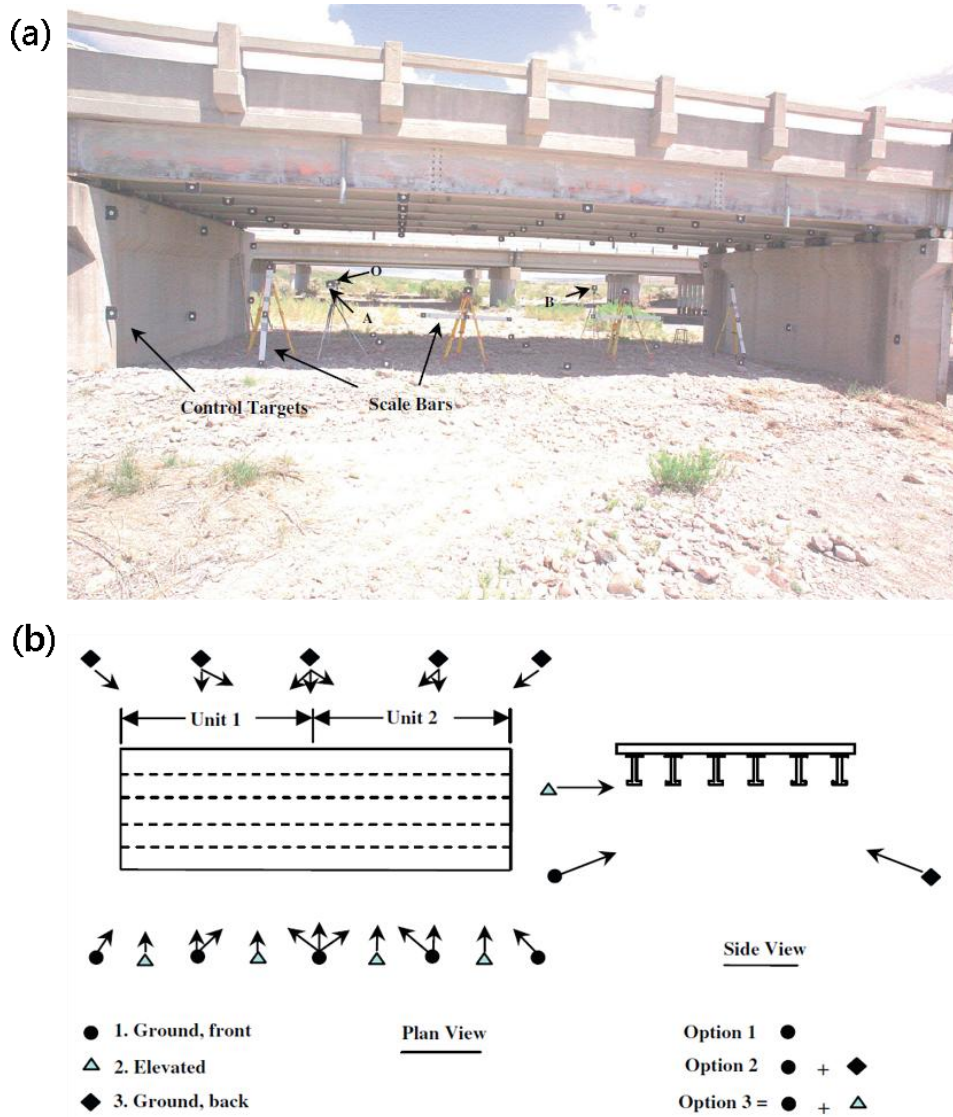


Figure 2-23: The use of photogrammetric technique for a bridge monitoring (Jiang and Jauregui, 2010).

When using close-range photogrammetric measurements for flexible structural monitoring (or dynamic object monitoring), it is essential to ensure all cameras are synchronised. A highly-synchronised photogrammetric image sequence shows that each imaging station can simultaneously capture the dynamic behaviour of target structures (Madeira et al., 2010). In order to coordinate multiple camera stations for the synchronised image acquisition process, Raguse and Heipke (2006) divided the

methods into three categories. The first method simply relies on using hardware components (for example, hardware trigger boxes or frame grabbers) and the second method is using a combination of hard- and software. The last solution only uses software to control each imaging station; and using this approach, cameras do not have to be physically connected to any master system and no special hardware devices are required. If the camera synchronisation is performed via hardware instruments, engineers also have to construct wires or transmission cables between devices. Therefore, the imaging stations are not easily arranged on a complex terrain or a large test area. In terms of the actual application, Ozbek and et al. (2010) utilised close-range photogrammetric measurements to monitor large wind turbines. Figure 2-24 illustrates the configuration of the measurement setup and camera locations.

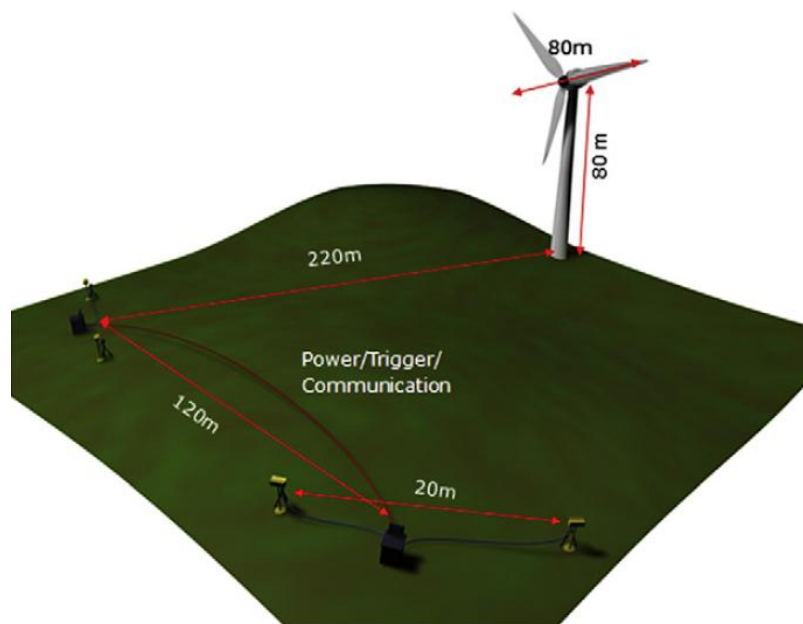


Figure 2-24: Configuration of a large wind turbine monitoring project (Ozbek et al., 2010).

It is difficult to perform conventional close-range photogrammetric measurements in low light environments because cameras cannot capture clear images (Egels and Kasser, 2002). In order to ensure the photogrammetric targets can be detected by each camera throughout the entire day, a total amount of 55 retro-reflective targets were

attached onto the surface of each wind turbine (Figure 2-25 (a)). Using high intensity LED (light emitting diode) lamps as light source to project these targets, the reflection of each target is obvious in the dark (Figure 2-25 (b)).

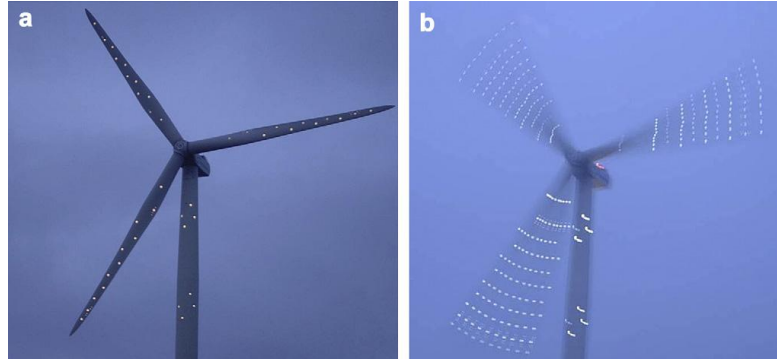


Figure 2-25: The Layout of photogrammetric targets on the turbine (Ozbek et al., 2010).

In this experiment, all imaging stations and LED lamps were controlled by a developed software suite. By using the software, engineers can set up the exposure frequency and shutter speed in advance. Then, the automatic imaging acquisition process can be performed. Although the communication between computer and each imaging station still relies on wires and cables, the system has achieved semi-automatic operation in terms of using close-range photogrammetric measurements for flexible structural monitoring. The experimental results show that the coordinate measurement error is in the range of ± 5 mm (1/16,000) of field of view and the deformations on the turbine can be measured with an average accuracy of ± 25 mm (the measurement distance is 220 m).

When performing close-range photogrammetric measurements for flexible structural monitoring in civil engineering, because the target objects are normally large, the camera photogrammetric network is also widely expended in order to provide high quality convergent image pairs (the optimal convergent angle of a two-camera configuration is approximately 90 degree (Ethrog, 1987)). Under such circumstances, the use of hardware synchronisation approaches has become more difficult. Recently, most related researches have focused on dealing with the camera synchronisation issues through the use of software solutions. In comparison with the use of hardware

approaches, it is easier to combine multiple electronic devices with the software solutions for automatic operation and this is an obvious advantage for long-term monitoring applications.

2.3 Summary

This chapter has outlined a number of structural monitoring techniques that are commonly used in modern civil engineering. The strengths and weaknesses of each measurement in structural monitoring aspects have been presented. Through the description of actual applications, the instrument configuration, experimental restrictions and achievable precision, etc. of each measurement were introduced. In this chapter, the requirements that are related to flexible structural monitoring and long-term monitoring were highlighted. Specific characteristics of each measurement were summarised in Table 2-2. Close-range photogrammetric techniques allow engineers using only a few imaging stations to perform non-contact measurements with wide-area coverage. Based on an understanding of these requirements and restrictions, a mobile phone-based, photogrammetric system is proposed in this research which is developed to meet a variety of structural monitoring demands in civil engineering. The details and design logic of this system are introduced in the following chapter.

	Strain gauge	FBG	Total station	GNSS	TLS	Close-range photogrammetry
measurement range (single/multiple-point measurement)	S	M	S	S	M	M
contact/non-contact measurement	C	C	N	C	N	N
ability for flexible structural monitoring	good	good	difficult	good	difficult	good
instrument cost ^{*1}	low	Median	Median	High	High	Median
<p>*1: The price has varied according to the category of products, the country, and the period of time.</p> <p>*2: The approximately operating accuracy for structural health monitoring of each measurement method are introduced in the article.</p>						

Table 2-2: Comparison of different measurements in measurement characteristics.

CHAPTER 3. A PHOTOGRAMMETRIC METHODOLOGY FOR MONITORING FLEXIBLE STRUCTURES

3.1 Introduction

A generic photogrammetric structural monitoring system capable of long-term inspection of the static and dynamic behaviour of civil engineering infrastructure is proposed. After comprehensively reviewing the different possible approaches of structural monitoring in the previous chapter, the close-range photogrammetric and mobile phone technologies are introduced in relation to the proposed system. Controlled by an in-house software suite, a fully automated long-term monitoring operation is achieved. The main differences between the developed smart phone-based, close-range photogrammetric system and commercial mobile phone photogrammetric applications are the use of modified (and self-developed) image processing algorithms and the capability for long-term, multiple-device operations. In this research, conventional image processing algorithms were modified to comply with the restrictions of smart phone hardware specifications. Meanwhile, a dedicated target detection algorithm was developed to improve the precision when performing target identification on low resolution images (for example, images that were captured by a mobile phone camera). This smart phone platform-dedicated image processing software suite not only provides high quality photogrammetric measurements, but also significantly reduced the system load. Combining with the improvement of internal memory allocation of mobile phone devices, the stability for long-term

operations can be effectively ameliorated. In this chapter, the details of the proposed methodologies are presented.

3.2 A photogrammetric solution for structural health monitoring

The process of implementing a system for assessing the state of health of civil engineering structures demands instruments with special features for continuous monitoring (Zhang et al., 2010). Such equipment is also required to provide high accuracy, good resolution, fast processing speed and possibly portability, etc. (Armer, 2001). Amongst varieties of structural monitoring projects, monitoring of flexible structures is one of the most complex issues. It is of paramount importance that any monitoring equipment employed has the function of rapid sampling. Equipment with the capability of high sampling speed can be applied to record detailed deformation on rapidly moving objects (Abt et al., 2011). Photogrammetry, a non-contact measurement technique which utilises images to make precise 3D measurements of complex objects (Luhmann et al., 2007), is therefore likely to become one of the most practical methods for structural health monitoring of flexible structures (Maas and Hampel, 2006).

In accordance with the advantages of photogrammetric measurements which were outlined in the previous chapter, a close-range photogrammetric approach is proposed in this research. To develop a close-range photogrammetric system for long-term structural monitoring, some critical issues need to be considered (Feng, 1991):

- Photogrammetric processing software;
- Optimal photogrammetric targets;
- Network configuration;
- The details of imaging devices/sensors;
- System integration;

Furthermore, when using photogrammetric measurements for structural health

monitoring of flexible structures, operators also need to deal with the system synchronisation issue during the image acquisition process.

With regard to these concerns, a series of corresponding methodologies are introduced in this chapter to ensure that this developed system can meet the purpose of the objectives proposed in Chapter 1.

3.2.1 Imaging devices details

In order to further reduce the total cost, programmable mobile phones (also known as smart phones) are adopted in place of conventional imaging instruments which have been extensively used in general photogrammetric applications. With the rapid development of mobile phone technology, mobile phones have become common electronic devices in modern daily life (Wakefield et al., 2007). At present, modern mobile phones are not simply used as communication tools (Goggin, 2006). High-end smart phones feature large internal storage, a powerful CPU and abundant memory. These features make modern mobile phones suitable for exploitation for high speed computing, with the performance almost equal to an entry-level personal computer (Allen et al., 2010). In order to meet a variety of requirements (such as the implementation of multimedia, navigation and entertainment applications), mobile phones are continually upgraded in terms of hardware specifications (Asif, 2011). Most modern handset devices have been equipped with high-resolution cameras, high-performance processors and sensors (such as GPS sensor, accelerometers and gyroscope) (Levine, 1999). Meanwhile, smart phones provide programmable platforms that allow the development of software packages for specific applications. Through the act of combining mobile phone technology with integration software, these mobile phones can be used to perform high quality photogrammetric applications (Akca and Gruen, 2009).

In this research, the HTC Incredible S (which features an 8-megapixel built-in camera with a built-in 4.57 mm f2.4 lens) was adopted for use in experimentation. This CMOS sensor produces a JPEG file with an image size of 1.9 Megabyte (MB) (the resolution is 3264x2448 pixels). The HTC Incredible S features an electronic shutter

that has a shutter speed range from 1 / 500 s to 1 s. Control of the electronic shutter can be achieved via third-party software. Hence, the shutter speed and other camera settings can be controlled from a remote computer by the use of software which was developed in this research to realise the concept of creating remotely operated photogrammetric imaging stations. In terms of flexible structural monitoring tasks, the higher the image sampling rate the camera can achieve, the more information on structural movements can be obtained (Syahmi et al., 2011). The image sampling performance of a mobile phone camera is dependent on the size of the temporary memory. The HTC Incredible S features 768 MB random-access memory (RAM) (the actual measured memory amount is 615 MB). Under normal working conditions, there are only 225 MB of free RAM that can be used by third-party software. The function for continual image acquisition is therefore developed based on this kind of memory usage condition. The technical specification of the mobile phone is outlined in Table 3-1.

In comparison, Canon EOS 50D digital single-lens reflex (SLR) cameras were used to create photogrammetric benchmarks in each laboratory test. The lens used in this research was a Canon EF 28mm f/2.8 Lens with just 5 elements in 5 groups, including one aspherical element. The minimum focal distance is 0.3 m whilst the maximum object magnification is about 1:8. The technical specification of the DSLR camera and the 28 mm lens are outlined in Table 3-2.

HTC Incredible S	
OS	Android OS v. 2.2
Processor	Single core, 1000 MHz, Snapdragon
RAM	768 MB
Image format	3264 x 2448; 8-megapixel
Sensor size	4.6 x 3.4 mm
Pixel size	1.4 microns
Shutter speed (sec)	1/500 ~ 1
Manual focus	Yes
Output format	JPEG

Table 3-1: HTC Incredible S technical specifications.

Canon 50D digital SLR camera	
Image format	4752 x 3168; 15-megapixel
Sensor size	22.3 x 14.9 mm
Pixel size	4.7 microns
Shutter speed (sec)	1/8000 ~ 30
Manual focus	Yes
Output format	JPEG, TIFF and RAW
Canon EF 28mm lens	
Focal length prime	28 mm (APS-C 35mm equiv = 44.8 mm)
Aperture Range Prime	f2.8 - f22
Motor	AFD
Field of View (degrees) Horizontal, Vertical, Diagonal Full Frame - 36x24mm APS-C - 22.5x15mm	Full Frame : H - 65.5 : V - 46.4 : D - 75.4 APS-C : H - 43.8 : V - 30 : D - 51.6

Table 3-2: Canon EOS 50D technical specifications.

Because mobile phone cameras and DSLR cameras are not especially designed for photogrammetric purposes (i.e., these cameras are non-metric), the essential camera interior orientation parameters that are used for photogrammetric processing are unknown. Therefore, it is important to have a comprehensive understanding of these camera interior orientation parameters via camera calibration procedures.

Through camera calibration procedures, the elements of interior orientation of the camera (including focal length, principal point location, and radial and decentring lens distortion) are determined and can be used for refining the image measurement. (Akca and Gruen, 2009). In this research, both the mobile phone cameras and DSLR cameras were calibrated using PhotoModeler. The PhotoModeler camera calibration program is based on the method of space resection. Following the principles of the collinearity condition equations and using image point coordinates as observations, the internal and external orientation elements of the camera are calculated (Weizheng et al., 2010).

3.2.2 Camera calibration

In this research, the camera calibration process was divided into two stages. The first stage was to inspect the differences in camera calibration parameters between those derived from the PhotoModeler software and those from another academic photogrammetric software package: The Vision Metrology System (VMS). Computations were performed using the same experimental configuration, the test being undertaken in order to independently validate the PhotoModeler results. The second stage was to re-calibrate the imaging devices using the dedicated calibration method that is adopted by PhotoModeler.

3.2.2.1 Field calibration

In terms of the implementation of the field calibration, the camera calibration parameters were obtained based on a precise and sufficient number of object space control points. By incorporating the additional parameters into the collinearity condition equations, the interior orientation elements to the camera can be calculated (Fraser, 1977). Figure 3-1 shows the test field configuration and Figure 3-2 illustrates the detail of the target area.

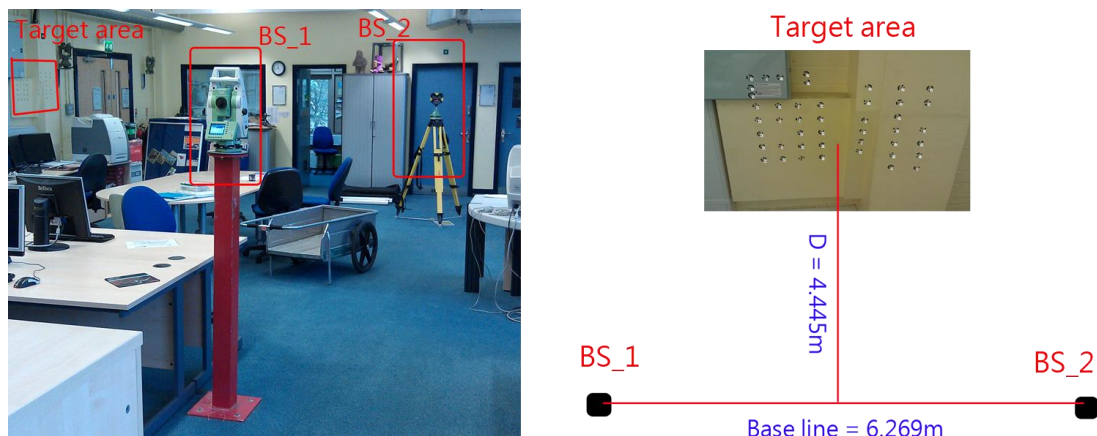


Figure 3-1: The illustration of test field configuration.

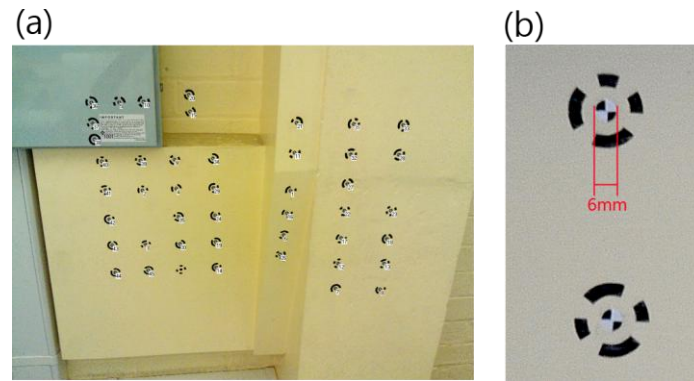


Figure 3-2: Illustration of target area; (a) shows the arrangement of photogrammetric targets; (b) shows the size of photogrammetric targets.

As shown in Figure 3-b (a), 45 targets were attached to a solid brick wall at different depths and positions. The target patterns were modified from the PhotoModeler target patterns using AutoCAD software. These patterns were sent to a vinyl cutter and cut to the required size.

Different photogrammetric processing software suites use different target patterns for processing. The required target patterns for the PhotoModeler and VMS were also different. In this experiment, the black and white quartered patterns were designed to be measured by three different methods (PhotoModeler, VMS and a total station). Theoretically, the PhotoModeler software can automatically identify these targets. The intersections of these black and white quartered patterns were designed to assist the surveyor in aligning target centres when performing physical measurement using a total station. The VMS software cannot automatically identify these targets; therefore, manually selection of the centre of each target was necessary when performing the VMS processing.

The coordinates of these stationary targets, which were attached onto the brick wall, were measured in three dimensions by a Leica total station (TCRP 1201) using spatial intersection. These coordinates could then be incorporated into the collinearity condition equations for further camera calibration.

In terms of the image acquisition process, two cameras (one HTC Incredible S mobile phone camera and one Canon 50D DSLR camera with a 28 mm lens) were used to

capture convergent images from six different angles to complete a sampling cycle. Each type of camera was used to collect ten sampling cycles in order to obtain an adequate number of image data for precision analysis. The calibration results for each determined parameter of these two cameras are shown in Table 3-3.

PhotoModeler				
	Canon 50D		HTC Incredible S	
	Value	σ	Value	σ
Focal Length (mm)	28.146	0.023	4.577	0.034
Xp - principal point x (mm)	11.415	0.019	2.262	0.036
Yp - principal point y (mm)	7.797	0.017	1.637	0.025
Fw - format width (mm)	22.827	0.020	4.461	0.023
Fh - format height (mm)	15.281	0.019	3.272	0.026
K1 - radial distortion 1	4.57×10^{-04}	2.01×10^{-06}	-6.420×10^{-03}	2.69×10^{-05}
K2 - radial distortion 2	-5.59×10^{-07}	7.85×10^{-09}	1.112×10^{-03}	3.18×10^{-05}
K3 - radial distortion 3	8.66×10^{-08}	8.56×10^{-10}	1.658×10^{-07}	2.40×10^{-09}
P1 - decentering distortion 1	-3.22×10^{-06}	2.35×10^{-08}	-1.509×10^{-04}	3.16×10^{-06}
P2 - decentering distortion 2	2.32×10^{-06}	2.14×10^{-08}	1.526×10^{-05}	2.49×10^{-07}
VMS				
	Canon 50D		HTC Incredible S	
	Value	σ	Value	σ
Focal Length (mm)	28.331	0.028	4.392	0.029
Xp - principal point x (mm)	11.445	0.023	2.232	0.032
Yp - principal point y (mm)	7.733	0.016	1.701	0.026
Fw - format width (mm)	22.891	0.017	4.397	0.026
Fh - format height (mm)	15.254	0.020	3.299	0.025
K1 - radial distortion 1	2.38×10^{-05}	2.47×10^{-07}	-6.20×10^{-04}	1.81×10^{-06}
K2 - radial distortion 2	2.35×10^{-07}	3.33×10^{-09}	1.11×10^{-04}	2.22×10^{-06}
K3 - radial distortion 3			2.52×10^{-07}	1.334×10^{-09}
P1 - decentering distortion 1	4.58×10^{-06}	3.27×10^{-08}	1.58×10^{-04}	2.221×10^{-06}
P2 - decentering distortion 2	-2.14×10^{-06}	1.98×10^{-08}	-1.97×10^{-05}	1.874×10^{-07}

Table 3-3: Camera calibration results for Canon 50D and HTC Incredible S using PhotoModeler and VMS software.

It can be seen that the experimental results for each parameter shown in Table 3-3

differ depending on what software was used to process the bundle adjustment. This can be attributed to a number of factors. For example, it is important to note that although the black and white quartered patterns were designed to be applied for multiple measurements, the centre of each target was not obvious enough to be identified by both photogrammetric software suites (even when performing the automatic target identification process using PhotoModeler). The operator therefore needed to manually select target centres during the target identification process for each process. For this reason, these black and white quartered patterns were not applied in the subsequent experiments and alternative target designs were instead adopted (as introduced in § 3.4.4). Nevertheless, the different bundle adjustments were based on different observations. Moreover, VMS adopts a slightly different camera model, incorporating affinity and orthogonality parameters into the image coordinate system transformation. This (together with weightings used) will affect, for example, the determination of the image frame size, the calibrated focal length and subsequently the determined image parameters. However, although the calibration parameters were not exactly the same, the RMSE values for each parameter between these photogrammetric software suites were similar orders of magnitude. As expected, the RMSE values for the smart phone camera were slightly higher than the DSLR camera in both PhotoModeler and VMS calculations. This gives confidence in the parameters determined by PhotoModeler.

3.2.2.2 PhotoModeler dedicated calibration method

In terms of the configuration of camera calibration using the dedicated PhotoModeler calibration method, the calibration grid board was produced following a specified dimension that was included in the PhotoModeler instructions. The un-calibrated camera captured images from 1.5 metres high. A complete camera calibration procedure contains twelve convergent images that were captured from different angles (an illustration of this is presented in Figure 3-3). What is more is that the operator needs to ensure that each image is covered by the calibration grid board. These calibration procedures were repeated twelve times.

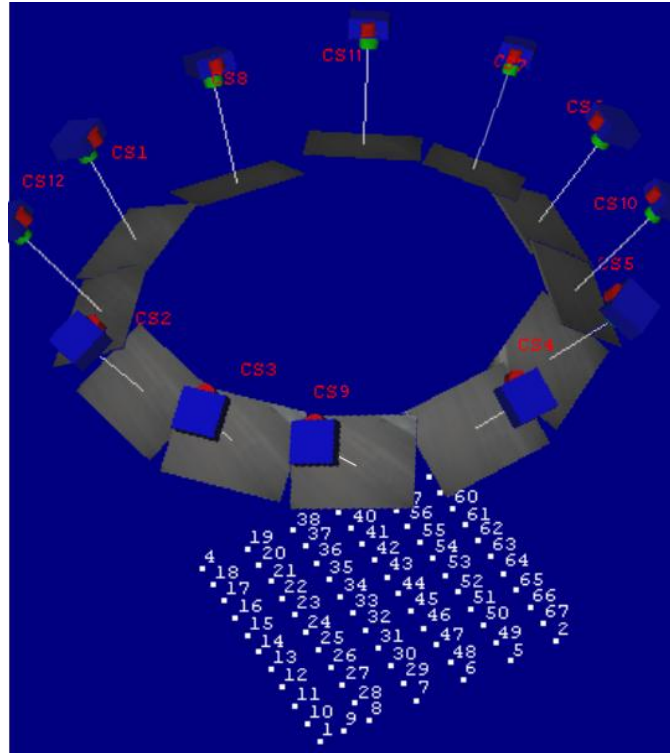


Figure 3-3: The configuration of camera calibration.

The calibration results and the allowance range of measurement error (3σ) for these two types of cameras are shown in Table 3-4. These statistics were calculated based on twelve repeated calibrations.

	Canon 50D			HTC Incredible S		
	The most probable value	σ	3σ	The most probable value	σ	3σ
Focal Length (mm)	28.157	0.012	0.037	4.566	0.020	0.060
X_p (mm)	11.418	0.010	0.030	2.259	0.016	0.049
Y_p (mm)	7.779	0.008	0.025	1.655	0.010	0.029
Fw (mm)	22.726	0.010	0.029	4.562	0.012	0.036
Fh (mm)	15.164	0.010	0.030	3.389	0.012	0.037
K1	1.183×10^{-04}	1.11×10^{-06}	3.34×10^{-06}	-6.081×10^{-03}	9.27×10^{-06}	2.78×10^{-05}
K2	-1.432×10^{-07}	9.53×10^{-10}	2.86×10^{-09}	1.112×10^{-03}	1.06×10^{-05}	3.18×10^{-05}
K3	3.913×10^{-08}	9.60×10^{-11}	2.88×10^{-10}	2.133×10^{-07}	1.05×10^{-09}	3.16×10^{-09}
P1	-8.963×10^{-06}	9.23×10^{-09}	2.77×10^{-08}	-1.452×10^{-04}	1.06×10^{-06}	3.18×10^{-06}
P2	1.445×10^{-06}	8.03×10^{-09}	2.41×10^{-08}	1.614×10^{-05}	8.77×10^{-08}	2.63×10^{-07}

Table 3-4: Calibration results of main parameters.

Theoretically, the initial radial distortion coefficients should include K1, K2 and K3. Good quality non-distorting lenses may not need K2 and K3, and some telephone lenses may not require P1 and P1 (Green, 2002). The lenses, which were applied in this research, are not fisheye or wide angle lenses; the K3 values are close to zero and can be ignored. Focusing on the K1 and K2 values, obviously the value of K2 in the DSLR camera group was significant smaller than K1, whilst the value of K1 and K2 in the smart phone camera group were more similar. The results show that the effect of lens distortion is less significant for the DSLR camera with a 28 mm lens, in comparison to the smart phone camera.

In comparison to the results shown in Table 3-3, the RMSE values in Table 3-4 are lower. The use of inappropriate target patterns may have led to higher RMSE values, shown in Table 3-3. Based on these experimental results, the camera calibration parameters shown in Table 3-4 were used as references to compare with the subsequent calibration parameters, which were recalibrated prior to each practical experiment. If the subsequent calibration parameters were within the allowable range (the residuals of new calibration parameters need to be smaller than 3σ), these camera parameters were to qualify for further photogrammetric calculations. Otherwise, the operator should re-execute the camera calibration procedures to clarify the interior orientation elements of the camera.

3.2.3 Photogrammetric processing software and optimal target size

The system developed in this research aims to provide a low-cost, photogrammetric solution for structural monitoring applications. For this reason, this system uses instruments (smart phones, laptop and personal computer) that are commonly available for everyday use. The application software is following the same principle to control the cost effectively. The uses of software packages in this system are not only common but also inexpensive. The photogrammetric processing software used in this research was PhotoModeler Scanner Version 6 (herein subsequently referred to as PhotoModeler). This is commercial, inexpensive, off-the-shelf software. The commercial software is relatively easy to be obtained when compared to academic software. According to the user manual, PhotoModeler (2010) claims that when using

a high quality digital camera, accuracies of 1:30,000 can be achieved. Given its relatively low cost and user-friendly graphical user interface (GUI), PhotoModeler has become a popular choice for reverse engineering and many other engineering measurement applications (Young and Garde, 2007). In order to estimate the performance of this commercial software, Wang and et al. (2010) compared the differences of photogrammetric results in distance measurements between the PhotoModeler software and the VMS software suite. Two different types of cameras (an F-810C video camera and a Nikon D300 DSLR camera) were used to generate two sets of photogrammetric results for comparison. During the test, the experimental settings (the test field configuration and camera settings) for both cameras were exactly the same. A test board was setup 650 mm away from the camera stations and a total of six distances on the target board were selected for comparison. The endpoints of these distances were measured by a P3 analytical plotter and these physical measurement results were used as standards to compare against photogrammetric measurements. The results of these comparisons were summarised in Table 3-5.

In terms of the differences in standard deviation values between these two photogrammetric software suites, results for images captured by the D300 DSLR camera indicated that the maximum difference was 3 μm (the numbers of these comparisons were rounded to the micrometre level) and two of these six distances showed no difference in standard deviation results. Results for images captured by the F810 video camera showed were slightly better as the maximum difference was 1 μm and three of these six distances showed no difference in standard deviation results. As for the comparison of root mean squared errors, the maximum difference of D300 data sets was 2 μm and three of these six distances showed no difference. Moreover, the maximum difference of F810 video camera data was the same as D300 (2 μm) and two of these six distances showed no difference in the comparison of root mean squared errors. Though the precisions were not particularly well due to the experimental settings had not been adjusted to the optimum configuration (the following chapters will elaborate the relationship between the photogrammetric precision and the related experimental configurations), two photogrammetric processing software suites generated similar results.

		Distance	Pt.1~ Pt2	Pt.2~ Pt3	Pt.4~ Pt5	Pt.5~ Pt6	Pt.7~ Pt8	Pt.8~ Pt9	Max	Min
		Physical measurements (mm)	40.012	40.164	39.975	40.112	40.254	40.011	40.254	39.975
D300	Arithmetical average value (mm)	PhotoModeler	39.976	40.139	39.965	40.132	40.226	39.969	40.226	39.965
		VMS	40.072	40,214	39.987	40.097	40.301	40.063	40214	39.987
	Standard deviation (µm)	PhotoModeler	21	17	16	16	17	20	21	16
		VMS	19	20	14	15	17	22	21	15
	Root mean squared error (µm)	Difference (absolute values)	2	3	2	1	0	2	0	1
		PhotoModeler	20	17	14	14	13	16	20	13
		VMS	18	17	13	14	15	17	18	13
	Differences (absolute values)	PhotoModeler	2	0	1	0	2	1	2	0
		VMS	2	0	1	0	2	1	2	0
Difference (absolute values)		2	0	1	0	2	1	2	0	
F810	Arithmetical average value (mm)	PhotoModeler	40.076	39.997	40.017	40.132	40.181	40.067	40.181	39.997
		VMS	40.172	40.074	39.954	40.097	40.098	40.128	40.172	39.954
	Standard deviation (µm)	PhotoModeler	18	16	13	14	16	17	18	0.043
		VMS	17	16	14	14	15	16	17	14
		Difference (absolute values)	1	0	1	0	1	1	1	0
	Root mean squared error (µm)	PhotoModeler	17	15	14	12	13	12	17	14
		VMS	18	16	14	13	15	14	18	13
		Difference (absolute values)	1	1	0	1	2	2	2	0
	Differences (absolute values)	PhotoModeler	1	1	0	1	2	2	2	0
VMS		1	1	0	1	2	2	2	0	
Difference (absolute values)		1	1	0	1	2	2	2	0	

Table 3-5: The comparison of photogrammetric results in distance measurement between two different photogrammetric processing software packages.

Each photogrammetric processing software suite has its own requirements and restrictions in terms of the applicable range of photogrammetric target size. Normally, each software suite uses dedicated target patterns in concert with their target detection algorithm. According to the instructions, PhotoModeler recommends that centre target widths in the photos be at least 8-10 pixels wide. Figure 3-4 shows the geometrical construction for a typical thin lens system. The use of different focal lengths of lenses and object distances will affect the targets' size in images (Atkinson, 2001). Therefore, it is important to clarify the experimental configuration (including the required object distance, the use of camera and lenses and optimal target sizes etc.) in advance when planning a high-quality photogrammetric project.

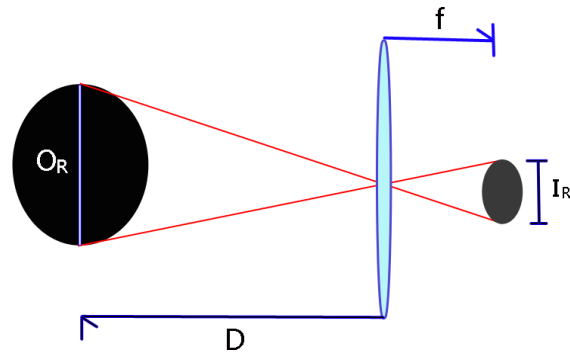


Figure 3-4: Geometrical construction for a typical thin lens system.

The relationship between the physical target size and the target size in the photograph is outlined below. The operator can use these equations to determine the optimal target sizes.

Assuming the actual target size (the diameter of the target is O_R), object distance (D) and focal length of lens (f) are known; then the target size in diameter in the photograph (I_R) can be calculated using equation (3.1).

$$I_R = \frac{O_R \times f}{D} \quad (3.1)$$

Once the I_R has been calculated, the operator can use equation (3.2) to estimate how many pixels the target occupies in the sensor.

$$P = \frac{F_p \times I_R}{F_w} \quad (3.2)$$

Where P is the number of pixels in the width of the image; F_p means the number of pixels along the image width and F_w is the width of the imaging sensor (Nakamura, 2005).

3.2.4 Additional instruments

In addition to the use of mobile phones, the monitoring system was developed using common electrical devices including a standard consumer laptop (or personal computer) and a wireless router. In this research, the terminal computer is an

entry-level 14-inch laptop - the HP Compaq Presario V3000 (the operating system is Microsoft Windows XP). The mobile phone-based photogrammetric monitoring system was developed using the Java programming language. The system load and memory usage have been strictly controlled and kept to the minimum possible level. Therefore, the terminal computer does not have to be high-specification to control the system.

HP Compaq Presario V3000	
OS	Microsoft Windows XP Professional SP3
Processor	Intel Core Duo T2400 1.83
Memory	1 GB DDR II SDRAM
Hard Drive	60.0 GB / 5400.0 rpm
Graphics Processor	Intel 945GM

Table 3-6: HP Compaq Presario V3000 technical specifications.


	Data transfer rate (max)	300 Mbit/s
	Antenna type	Omni Directional x 3
	Antenna gain level	4 dBi
	Bandwidth	2.4 - 2.4835 GHz
	Ethernet LAN (RJ-45) ports quantity	1

Table 3-7: TP-LINK TL-WA901ND technical specifications.

The connection between the terminal computer and multiple mobile phone stations relies on a wireless network. If the test field is covered by the Internet or 3G mobile telecommunications network, the devices can connect with each other using these local network services. When the system is applied in a test field that has no network service, the terminal computer can still connect to each remote mobile phone station wirelessly by the use of a wireless router to create a virtual local network (Figure 3-5). In this research, a TP-LINK TL-WA901ND 300Mbps Wireless N Access Point has been used, allowing the creation of a wireless LAN (WLAN). The closed network allows Wi-Fi devices to connect to each other under the coverage of the wireless router's signal. More information about the system operation and the details of the devices settings are introduced in Chapter 4.

3.2.5 The OS, programming languages and photogrammetric processing software

The aforementioned electrical devices are controlled by a cross-platform integration software package. It not only focuses on development on the mobile phone platform but also on the PC operating system (Microsoft Windows OS).

In terms of mobile phone operating systems (OS), there are a number of handset operating systems currently on the mobile phone market. The most common mobile operating systems include Apple's iOS, Google's Android, Microsoft's Windows Phone, Nokia's Symbian & MeeGo, Samsung's Bada, and BlackBerry OS. The mobile phone manufacturers have developed these operating systems on their own smart phone devices to provide more advanced computing performance and connectivity than conventional mobile phones (Ahonen, 2008; Rich, 2004). Although each OS has its own characteristics and is supported by a large number of applications (apps), the OS which was chosen as the development platform for this research had to meet the following conditions:

- Allow users to implement multitasking;
- The source code and software development kit of the mobile phone OS was required to be free of charge and open source.

Nokia's Symbian and Apple's iOS were excluded from consideration because of the inability to handle multitasking (Brannan and Ward, 2011; Jipping, 2007). Apple's iOS is one of the most popular smart phone OS, however, the multitasking capability for iOS is not complete as operations cannot be performed in the background. Instead, the iOS suspends the operation of programs, hiding them in the background and leaving an index file for each program. The other operating systems such as MeeGo and Bada were not chosen as the development platform because they were not in the mainstream mobile phone market when the research commenced in 2008. The manufacturers did not update the product lines by launching new successors afterwards.

System development began in December 2008; the prototype was designed on the

Microsoft Windows Phone V6.5. In September 2008, when Google announced the first Google Android phone – the HTC VIVA, the software development kit (SDK) and the programming environment were still incomplete (Gozalvez, 2008). Google’s Android OS was a new and unfamiliar interface for programmers (Deitel et al., 2011). On the contrary, the Windows Mobile (WM) OS had existed in the market since 2000 (Wigley et al., 2007). After eight years of continuous evolution, by 2008 Windows Mobile phones had become one of the most popular and mature smart phone operating systems on the market. Many manufacturers had announced a variety of WM phones with the latest technology and functionality which included the highest-resolution built-in cameras. The Windows Mobile OS was therefore the obvious choice for system development. However, Microsoft Corporation introduced the new WM 7 OS in September 2011 which was unable to support the previous WM OS (such as Windows Mobile 6.x, Windows Mobile 5 and Windows Mobile 2003 etc.) (Petzold, 2010). WM 7 OS and the follow-up WM OS (WM 8 OS will go into mass production in December 2012) use a new architecture to meet hardware requirements and the platform is incompatible with the previous WM operating systems.

Considering the future development of this research, the system needed to be transferred to a new platform which would be continuously updated. One of the biggest disadvantages was that the previous generations of WM 7 software development kits did not allow programmers to develop apps relating to the application of the mobile phone camera (Petzold, 2011). However, after only several years of development, handsets operating with the Google Android OS have become some of the most popular mobile phones on the market. This Linux-based mobile phone OS supports multiple languages and programmers primarily write in a customized version of Java (Paul and Kundu, 2010). Google has also released the source code for Android under free and open source software licenses (Cheng-Min et al., 2011). Moreover, the Android OS supports full multitasking of applications. Therefore, the Google Android OS V2.2 platform was chosen as a new OS for the low-cost, mobile phone-based monitoring system developed herein. Unlike the previous versions, Google Android OS V2.2 supports the software installation to the extended memory and enhances the speed of just-in-time compilation (JIT; also

known as dynamic translation). These changes significantly improve the runtime performance of application programs. Meanwhile, Google Android OS V2.2 allows administrators to remotely reset the device to factory defaults (Shabtai et al., 2010) and this characteristic was applied in the research to develop a real-time system recovery module. Other improvements of Android OS V2.2, such as full support for Adobe Flash V10.1 and approval for the user to use a Web browser to upload files (Bogle and Sankaranarayanan, 2011) have proven beneficial for programmers to develop a system with wireless data transmission functionality. The functions of wireless data transmission and remote control are two important themes for this low-cost, photogrammetric monitoring system.

In this research, self-programmed software was developed in order to control the mobile phone devices remotely. This software was programmed using the Java Platform, Enterprise Edition (Java EE) 6. The use of Java language in this research had the following benefits:

1. Java works on most platforms. Unlike Visual Basic (VB), which is exclusively a Windows platform, Java code can be run on most Macintoshes, Windows and Linux platforms. In this research, the system contains two different platforms. The terminal computer uses the Windows OS and the Android phones use the Linux OS. Developing software using the Java language was expected to reduce conflicts between two different operating systems.
2. Java is multi-threaded. The programmer can write programs where several sections run simultaneously in different execution threads (Goneid et al., 1997).
3. Java is object-oriented. The programmer can create modular programs and reusable code individually and combine them into a larger program package (Mills, 1995). These individual programs can be operated and maintained independently.

The software is divided into three parts; a control platform which can be run from a PC or laptop interface; the other two parts are designed for installation onto a smart phone OS to be responsible for dealing with commands which were sent from a

terminal controller (a laptop or a PC). One of these two parts focuses on performing the photogrammetric processing. The other part is designed to monitor the operation status of the mobile phone and deal with the system recovery process when a system failure occurs. By installing the software onboard smart phones, these smart phones can be treated as an automated, in-situ photogrammetric workstations; multiple tasks can be performed at the same time, such as receiving commands from a terminal computer, adjusting the photographic parameters, capturing a sequence of images, and saving previous image data, etc. The module, which uses multiple smart phones for long term, in-situ structural monitoring, is referred to as multiple-station module. Meanwhile, the software provides portable measurement functions, allowing an operator to implement basic photogrammetric measurement in the test field by using a single mobile phone.

3.2.6 System design and integration

The requirements and restrictions of photogrammetric measurements for structural monitoring have been outlined in Chapter 2. Based on this information, a mobile platform based structural monitoring system is proposed. This system is expected to discover the potential in terms of the development of photogrammetric applications on an Android OS-based smart phone platform. By means of computer programming, a multifunctional mobile platform based photogrammetric system can be achieved.

This system aims to transform off-the-shelf smart phones into in-situ photogrammetric imaging stations for long-term structural monitoring purposes (Figure 3-5).

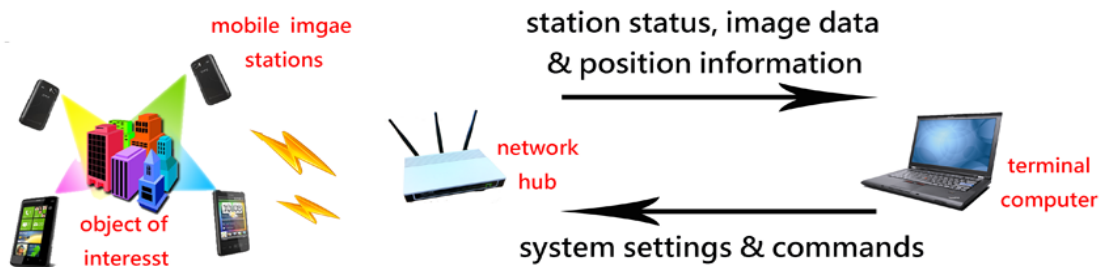


Figure 3-5: Illustration of system operation.

As shown in Figure 3-6, a conventional in-situ photogrammetric imaging station contains one or more cameras, storage devices, transmission equipment (the above two instruments are illustrated in (Figure 3-6 (b)), receiving apparatus ((Figure 3-6 (c)), power supply and processing devices (the above two instruments are illustrated in (Figure 3-6 (a)) (Plankis et al., 2007). Such systems are normally large, expensive and require high power consumption. By the integration of the software suite developed in this research, the functions of this complex combination can be converted into a portable mobile phone monitoring system. Meanwhile, compared with the conventional measurements (such as the use of laser scanners, total stations, GPS network etc.) the mobile phone-based system has a significant cost advantage. For a four-mobile-phone configuration, the total expenditure is under £1,500.

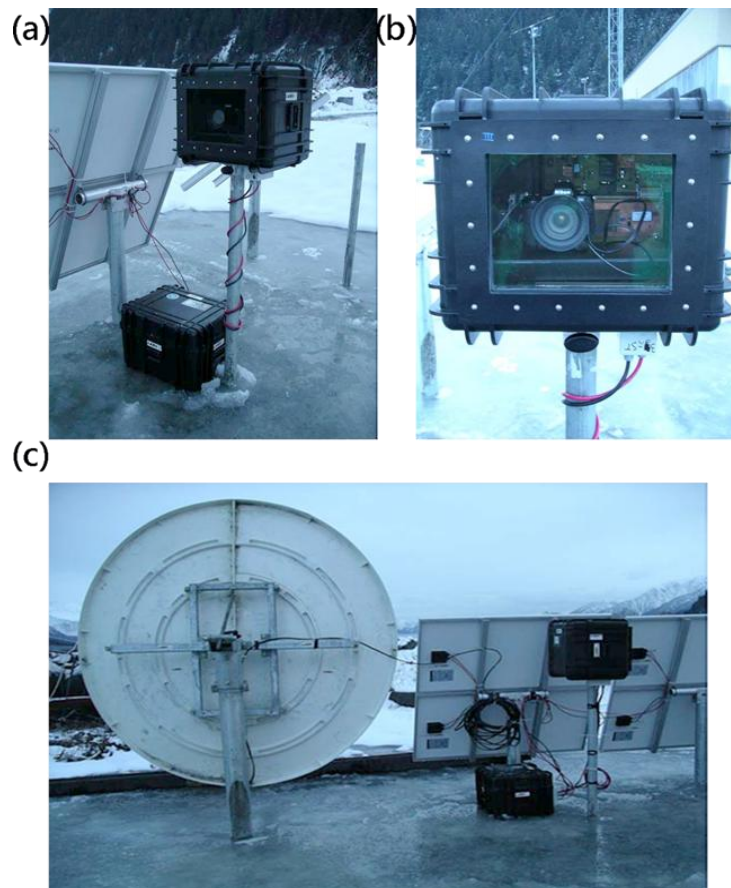


Figure 3-6: Example of a conventional long term in-situ photogrammetric imaging station (Plankis et al., 2007).

The use of digital SLR cameras can also achieve the same effect (such as similar

stereoscopic photogrammetric network layout, equivalent or even better image quality); however, the DSLR camera-based imaging station cannot operate on its own as it can only be operated by a computer through a USB cable. As a result, the system will lose the ability to operate remotely. Also, a high quality DSLR camera is more expensive than a high-end mobile phone. In addition, the mobile phones are more conveniently carried than conventional structural monitoring instruments. The combination of four mobile phones, a wireless router and one laptop can be easily packed into a suitcase or a backpack. The total weight of this compact system can be less than three kilograms. Also, the system operation is fully automatic. All of these automatic functions can be achieved because Google's Android operating system is an open-source platform that allows users to install third-party software to extend the mobile phones functionality. According to different requirements, the programmers can design specific software to perform on the mobile phone platform.

Following the illustration shown in Figure 3-5, the operator can use one command computer to remotely control multiple mobile phone-based imaging stations to perform photogrammetric monitoring schemes via a 3G or a Wi-Fi network. With the functionality listed below, the mobile phone-based monitoring system is qualified for long-term flexible structural monitoring tasks.

1. Remotely controlled by terminal computer;
2. Precise device synchronisation;
3. High sampling rate;
4. Automated, continuous sampling;
5. Full-time system status monitoring;
6. Real-time system recovery.

The monitoring operation is fully automatic. Once the system synchronisation has been performed and the operator has inserted the initial parameters, the system is launched and continues to run in a fully automatic state. The initial parameters include the number of image stations, the IP address of each mobile phone, imaging frequency etc. For advance settings, the operator can change the storage location,

shutter speed, camera ISO and some image processing functions if the operator wishes to run the processing onboard the mobile phone, for example target detection and coordinate calculation. The terminal computer can follow orders to edit the script and to assign the mission to each slave imaging station. During the processing, the computer concurrently performs real-time system health monitoring to check on the system operation. This background program can automatically detect any system failure and immediately relay a series of procedures to the remote devices to recover the system from malfunction. Compared with conventional structural monitoring measurement methods, this fully automatic photogrammetric system can massively reduce the investment budget on equipment and labour.

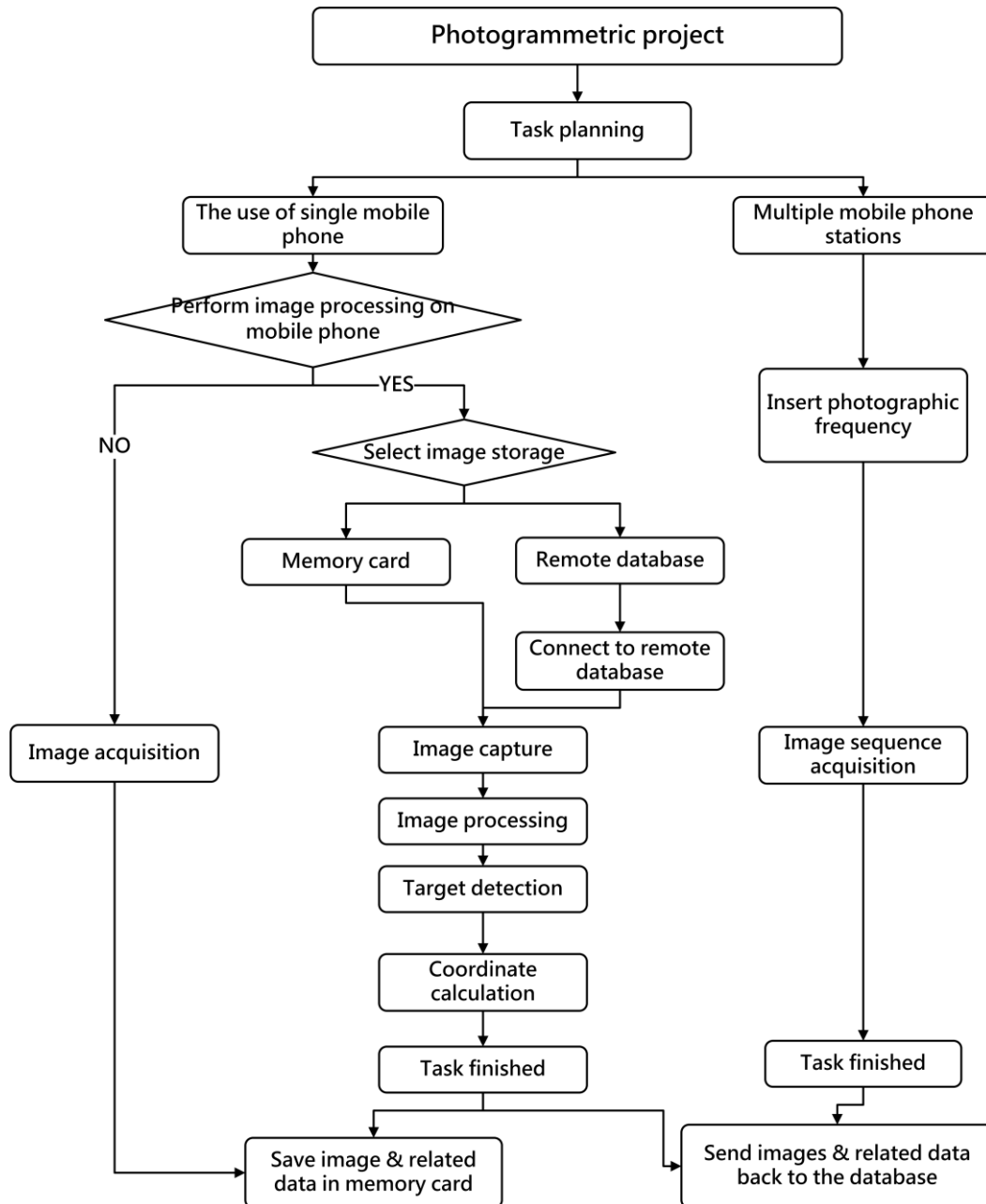


Figure 3-7: Workflow diagram for the low-cost photogrammetric monitoring system.

As previously mentioned, the electronic devices utilised in this developed system were not originally designed for photogrammetric purposes. The integration of these instruments relies on the use of a multifunctional software suite that was developed in this research. Unlike conventional photogrammetric imaging station configurations, this system does not arrange in-situ data storage. Instead, a remote database is designed to reduce the bulk of imaging stations and power consumption. In order to

avoid losing data during wireless transmission, the mobile phone's built-in memory is utilised as temporary image storage space. When the mobile phone receives an automatic confirmation from the terminal computer to confirm the data transmission has been completed, the image data in the mobile phone's built-in memory will be deleted.

This software package integrates the components of the developed system to perform photogrammetric operations. The overall process is illustrated in Figure 3-7. According to different requirements, operators can use one or multiple mobile phone devices to perform different photogrammetric tasks. In the following sections, a comprehensive introduction of the system and the design principles are presented. Meanwhile, in order to inspect the performance of this mobile phone photogrammetric system, a series of experiments were undertaken and the experimental design and results are presented in subsequent chapters.

3.3 Mobile platform based structural monitoring system

The development of this mobile platform based photogrammetric system is divided into two parts that are designed to be implemented on the computer platform and each smart phone respectively. The program on the PC platform is responsible for commanding all imaging stations, checking the system status, data collection and data sorting. In addition to this, the app on the mobile phone platform is designed to perform the commands that are sent from a terminal computer.

In order to meet the requirements for long-term structural monitoring applications, an integrated software suite has been developed to automatically perform system synchronisation, image acquisition, data transmission, data sorting, system health monitoring and recovery on a mobile phone platform. The details of these functions are described in the sections that follow.

3.3.1 *System architecture and design principles*

The multiple-station module contains six main functions (including the grouping function, system synchronisation, image acquisition, data transmission, system health

monitoring and recovery function) that are especially designed for long-term, structure monitoring purposes. Figure 3-8 illustrates the connection between these features.

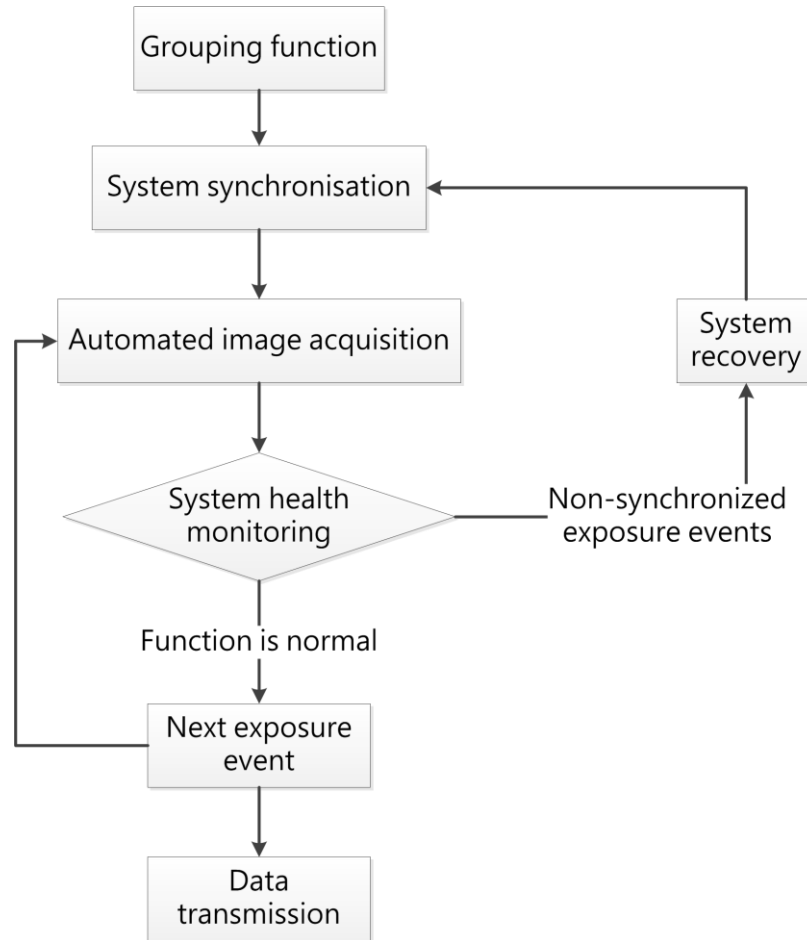


Figure 3-8: Illustration of the multiple-station module's functions.

In terms of using close-range photogrammetric measurements for structural monitoring of flexible structures, the synchronisation between each camera station is an essential condition to ensure these imaging stations can precisely capture the structural movement at the same time. In this research, each camera station is synchronised to the terminal computer using the “star network” configuration (Figure 3-9).

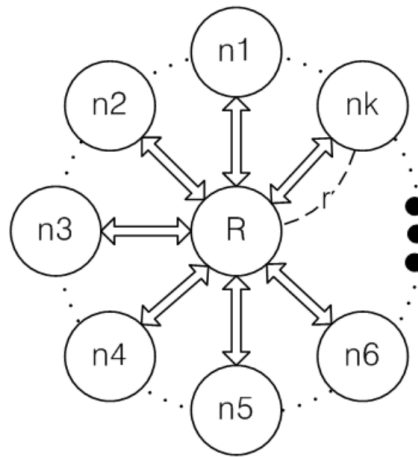


Figure 3-9: Illustration of the star network configuration (Hyun-Kwan and Seong-Lyun, 2009).

In the star network configuration (Figure 3-9), the terminal computer is defined as a central node that is responsible for organising all the commands of the six major functions shown in Figure 3-8. The “leaf nodes” represent the in-situ smart phones and these leaf nodes were designed to perform the automatic image acquisition. In order to ensure all imaging stations can take pictures synchronously, the terminal computer calculates the transmission delay to each of the leaf nodes and eliminates those offsets using the “two-way message exchange” mechanism (the detail of two-way message exchange mechanism is described in Appendix I) in order to provide high quality synchronised image sequences for further photogrammetric processing. The system synchronisation between the terminal computer and smart phone stations is established using the Internet protocol suite. The terminal computer (client) uses the Transmission Control Protocol (TCP) and Internet Protocol (which is commonly known as TCP/IP) models to design the packet transmission software (the packet is a formatted unit of data carried by a packet mode computer network (Haddad et al., 2010)) for real-time data transmission via a 3G or wireless network.

The Application Programming Interface (also known as API) of the TCP is more complicated than another popular Internet protocol suite - the User Datagram Protocol (UDP) (Partridge and Pink, 1993). However, the UDP is a non-connection oriented, one-way transmission protocol, which means that the client is unable to realize

whether the previous data transmission was successful or not because the receiving end cannot send results or reports back to the terminal computer (client end) (Bruno et al., 2008). In this self-developed system, the system health monitoring module is not able to function properly if the platform is lacking a message return mechanism. For this reason, the TCP/IP protocol suite was chosen for developing the point to point, star network. By using this type of transmission configuration, each individual imaging station can independently communicate with the terminal computer (Gen et al., 1996). In order to ensure the synchronisation mechanisms can meet the requirements in various environments, a series of policies were designed. The operators can use the appropriate settings to obtain high-quality image sequences under any circumstance. The details of operations of system synchronisation and decision making are introduced in Appendix I.

The grouping function is designed for multi-target monitoring purposes. In actual applications, a monitoring project may need to monitor more than one target object. This function can be employed to perform multi-target monitoring tasks. A terminal computer can apply this module to remotely control multiple imaging stations that are placed in different test fields. In addition, because the system currently cannot capture synchronised image sequences with too fast exposure frequency due to the lack of mobile phone temporary memory space, operators can use several imaging station groups with different exposure frequencies to cover the insufficient sampling rate. For example, assume there is a monitoring project which requires an exposure frequency of five seconds. The operator can use a two-group setting and both of these two groups have the same exposure frequency of 10 seconds, one of the two groups commences image capture five seconds ahead of time than the other group. Then, this configuration can continuously record the target object's movements every five seconds.

Through the use of grouping functions, the operator can use one terminal computer to remotely control multiple camera stations that are located in different test fields.

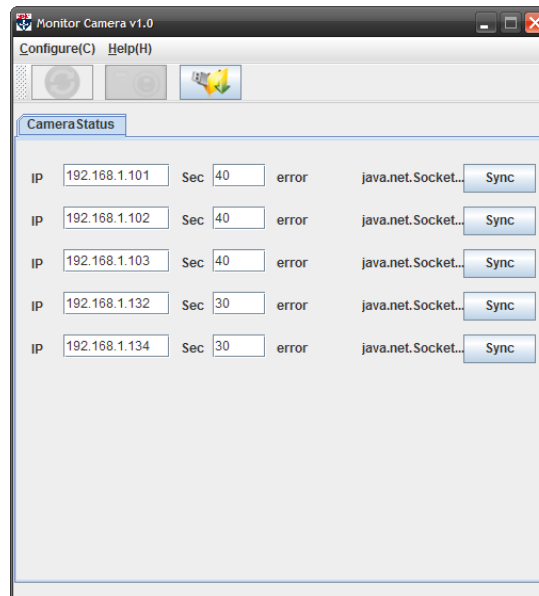


Figure 3-10: Illustration of the grouping function operation interface.

Figure 3-10 shows the operation interface for the grouping function. The columns behind each IP address can define different exposure frequencies whilst the imaging stations with the same exposure interval are regarded as the same group (Figure 3-10). In considering the transmission rate in 3G and wireless networks, one terminal computer can control up to ten imaging stations.

Though the wireless facilities that were adopted for this system are compatible with 802.11g standard (which operates at a maximum physical layer bit rate of 54 Mbit/s (Hawa et al., 2012)), the actual network transmission rates for 3G or wireless networks are below the standard. In terms of the transmission rate of the 3G network¹, according to an investigation undertaken by Ofcom (2011), the average mobile broadband speeds in good 3G coverage areas are up to 2.1 Mbit/s; however, most of the time the speed is drifting around 64 Kbit/s to 2 Mbit/s. The mobile phone that was applied in this system features a built-in 8 megapixel digital camera. In full 8-megapixel (3264 x 2448 pixels) resolution, the file size of each RGB photograph (in RGB full colour mode) is around 1.9 MB (JPEG file format). Ten camera stations

¹ Generally speaking, the transmission rates of 3G network is slower than broadband. When dealing with the data transmission issue via wireless facilities, the system should at least meets the requirement of the slowest transmission option (i.e. the 3G network).

generate 19 MB image data after each synchronised-exposure and the data transmission for this 19 MB data from each station to the terminal computer via the 3G network takes at least nine seconds (assuming the transmission rate of the 3G network can be maintained at 2.1Mbit/s at all times).

In the transmission process, if the digital image is transferred at full resolution without any compression or segmentation, this large format takes a longer time to transmit. When the network quality is unstable (this often occurs in wireless network transmission), image information can easily be lost and distorted during the communication process (Wu and Chen, 2003). In addition to transferring photographs back to the remote database, the mobile phones also need to arrange tasks that include: sending the real-time status reports back to the terminal computer; receiving the latest commands from the command centre; and updating the internal clock from the NTP server. These routine operations consume another 1.4 MB of internet usage. For this reason, these processes require very intense use of the internet. The instability of the network quality will delay the operating sequences and affect the process of synchronised-imaging. Instead of sending the full-resolution photograph back to the remote database immediately after each exposure, the system therefore adopts another solution to reduce the network load.

This alternative is to partition a digital image into multiple segments and these segments are stored in each phone's temporary memory. The image segmentation technique was modified from a traditional image compression technique –vector quantisation (VQ). The idea of traditional VQ in digital image processing is to partition the original image into several non-overlapping vectors. In other words, the photograph is divided into several non-overlapping blocks with the size of $m \times n$ pixels (Linde et al., 1980). As Figure 3-11 (a) and (b) show below, each block contains a number of different pixel values and these pixel values compose a unique vector to represent the imagery block itself. The encoding process is undertaken by matching each sample vector, which was registered in the codebook, with the image-block vector; seeing which vector it is most similar to. The image will have a little distortion when the image is restored (decoded) on the computer and that is why the VQ process is classified as a kind of lossy data compression algorithm

(Ping and Venetsanopoulos, 1993).

Though the VQ algorithm can split the image up into smaller packets for network transmission, there will still be a certain level of detail missing. Also, the VQ algorithm is too computationally intensive to run on the current smart phone platform. The generation method of the codebook often uses the Linde–Buzo–Gray (LBG) algorithm (which was introduced by Yoseph Linde, Andrés Buzo and Robert M. Gray in 1980 (Chen et al., 1995)). By means of the iterative computations, the LBG algorithm can significantly improve the pixel distortion value. However, it is still difficult to have reliable convergence results as this method can only optimize the result in partial areas (Kim, 1992). In order to obtain a reliable codebook, the program needs to keep repeating the computation and these actions cause a massive system load for a mobile phone OS.

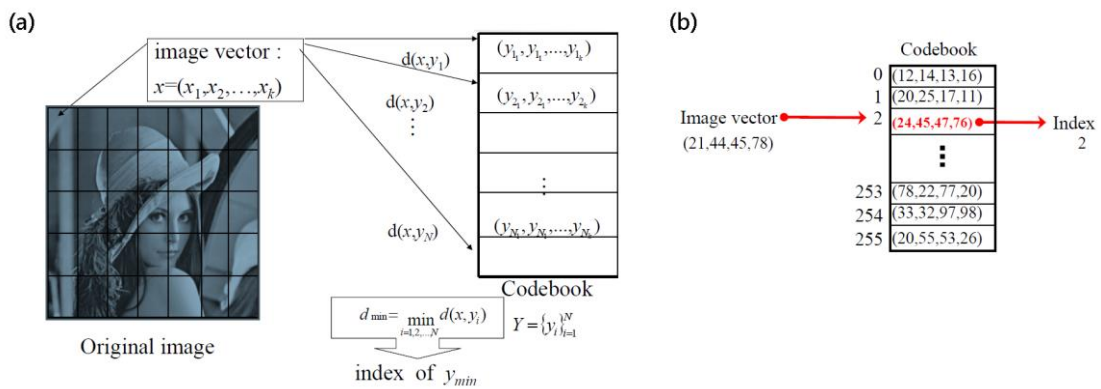


Figure 3-11: The coding procedures of Vector Quantisation (Gersho and Gray., 1992).

In this research, a new solution for image segmentation is developed by modifying the VQ algorithm. The method of segmentation for image-blocks is the same as the VQ algorithm. Each photograph was divided into several 272×272 pixel blocks in the order from top left to top right and from top row to the bottom row. The size of each image block is about 178 KB. The vectors are generated according to the pixel values from each pixel unit. The pixel 'value' is actually a vector of three numbers and the three different components are stored as three separate grey-scale images known as colour planes (one for each of red, green and blue). In order to compress a large amount of image information, the VQ algorithm simplifies these vectors. The

vectors of pixels are replaced by the index of their closest centroid (Chen et al., 1995); however, some image information is lost in this process. In this research, these vectors are sent back to the terminal computer directly without compression and this is the biggest difference to the VQ algorithm. An image reconstruction program recombines these vectors following the coding sequence and the exposure epochs. This method does not compress the image information at any point through the whole process. This developed transmission algorithm uses the same method as the VQ algorithm to perform the image segmentation process, but does not compress these image segments (i.e., the vectors of pixels are not simplified) in transit. Therefore, the performance in image compression is not as good as the VQ algorithm and the image transfer speed is also slower than the VQ algorithm. However, the image quality and image information are retained which is important from a photogrammetric standpoint.

When these files are sent back to the computer, the decoding program follows the segmentation logic and combines with the time information to reconstitute a series of images. These segments are transferred back to the remote storage device following the sequence in accordance with the transmission speed of 10 Kbit/s (default). Under such circumstances, the network traffic load can be reduced significantly. The size of segments and the transmission speed are adjustable. The settings can be changed following different camera resolutions and network quality. Despite the data transmission using the technique of small packet transmission to reduce the network load, each station still sends about 600 KB data back to the computer every minute. The data that is waiting to be sent back to the computer is stored in the smart phones' temporary memory space. If there is too much data accumulated in the temporary memory space², the imaging program will be temporarily stopped. Once the accumulated data is cleared, the computer will edit a new script and send it to each station to start a new imaging task.

² The default threshold of the chosen mobile phone in this research (the HTC Incredible S) is 16 MB. The minimum temporary memory is designed to ensure the system has enough memory buffers to maintain normal operation. Different handsets have different settings for minimum temporary memory space and consumers cannot discretionarily change this setting.

Another important issue in this operation is to determine a suitable exposure frequency for continuous imaging tasks. The settings for the exposure frequency also depend on the available end-to-end network bandwidth, quality of service capabilities and the processing speed of the mobile phone. Using the HTC Incredible S as an example, this 8-megapixel camera-phone features 768 MB RAM (the actual memory amount is 615 MB; as shown in Figure 3-12) for data processing. Under normal working conditions, there are only 225MB of free RAM that can be used for third-party software. This self-developed software package requires about 170 MB to maintain the operation and the image segmentation. Packet coding and packet decoding programs take up 23 MB of this 170 MB. The system reports show that the HTC Incredible S requires around 7 seconds to perform the process from image capture to packet transmission. If the exposure frequency is more intensive than the processing time, the additional workload will increase the memory usage significantly. And under the circumstances, the system may be forced to cease operations due to the shortage of memory space. In other words, when the remaining memory is lower than the threshold (16 MB), the system will stop capturing images until the temporary space has been cleared.

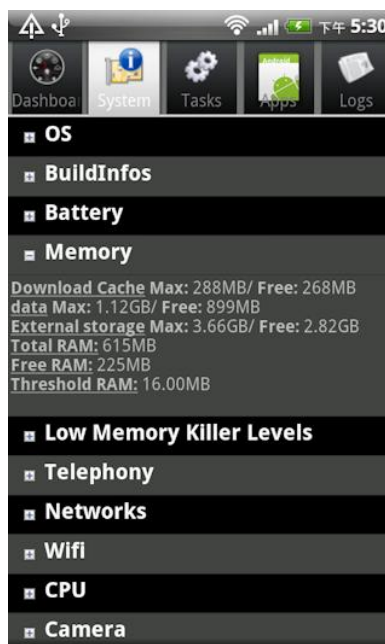


Figure 3-12: Illustration of mobile phone memory usage.

3.3.2 Principles of system health monitoring and automated system recovery

When performing a long-term photogrammetric monitoring task using the multiple-station module, the operator needs to remotely control multiple mobile platform based image stations simultaneously. In order to produce high quality simultaneous images, the synchronisation between each camera becomes a critical issue. The integrated software which was developed in this research provides a real-time monitoring function allowing the operator to maintain the system health from the terminal computer. This health monitoring module can be switched into automatic operation and provide an automatic recovery function. The automatic recovery function is able to re-start the image capture process with only minor interruption.

In photography, CCD (charge coupled device) and CMOS (complementary metal oxide semiconductor) image sensors have replaced traditional film to record image information that is passing through the lens (Nakamura, 2005). Both types of imaging devices transform natural light into electric charge and convert it into electronic signals. In terms of the CCD sensor, when the light from the object passes through the lens and reaches the surface of the image sensor, the silicon wafers are activated and electric charges are released. The amount of electric charge is in direct proportion with the amount of light received on each pixel unit (Ay, 2008). The generated electric charges are then sent to a QV (electron to voltage converter) for transforming into voltage and sending the voltage to an ADC (Analogue to Digital Converter) for converting to analogue signal via a voltage amplifier. In contrast to the CCD sensor, the CMOS device does not need to convert the electric charges to voltage via a QV. In a CMOS sensor, every pixel has its own charge-to-voltage conversion and a voltage amplifier (Figure 3-13); each pixel can directly convert the electric charge to the voltage using its own ADC without relying on the QV as an intermediate converter (Brooker, 2008). Figure 3-13 indicates the difference of the circuit structure between these two types of image sensors.

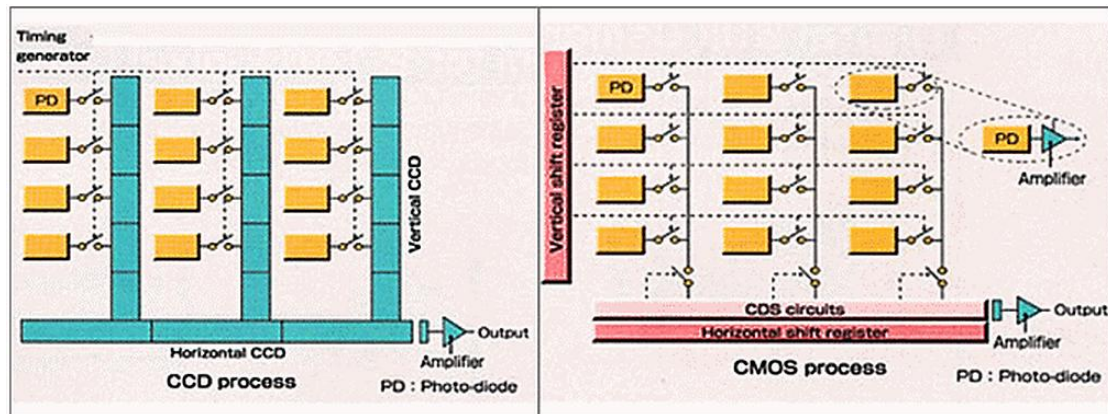


Figure 3-13: The comparison of circuit structure between the CCD (left) and CMOS (right) sensors (Holst and Lomheim, 2011).

Despite the different designs of these two image sensors, when the image sensor captures the light information (i.e. the process of transforming the natural light into electric charge) an instantaneous increase in voltage of each pixel unit is essential to excite every pixel's charge for exposure (Holst and Lomheim, 2011). The developed mobile phone application will immediately record the voltage change along with the time of occurrence of the exposure. The information is then sent back to the terminal computer automatically via a 3G or a Wi-Fi network. The system health monitoring program collects the data from each mobile phone, and a series of automatic sorting and data comparison routines will be implemented to ensure the remote camera stations are firing simultaneously. Figure 3-14 shows the screen capture from the system health monitoring program as it operates in the background. The graph shows three broken lines. From top to bottom, this illustrates the voltage changes, camera actuations, the current changes. The computer monitors the instantaneous changes of each camera and each individual mobile phone producing a chart like Figure 3-14.

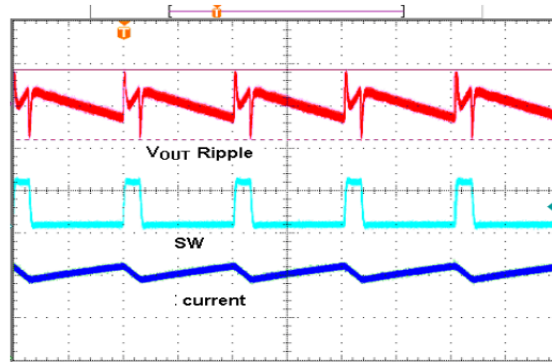


Figure 3-14: The screen capture from the system health monitoring program. The voltage changes, camera exposure and the current changes are recorded.

According to the measurements, the normal working voltage of the HTC Incredible S mobile phone which was applied in this research is 3.3 volts. This will jump to 5 volts when the camera's image sensor is exposing. Following the upper broken line (named the " V_{out} Ripple") illustrated in Figure 3-14, each peak (touching the upper red horizontal line) corresponds to an exposure trigger of the mobile phone camera. When receiving the real-time information from each mobile phone, the terminal computer can automatically match the timeline from each source and compare time of occurrence of the peaks to check the camera synchronisation.

If the monitoring program detects an uncoordinated exposure happening within any one of multiple camera stations, an automated system recovery policy will be triggered. The operating principle of the system recovery program is to recalibrate the internal clock of each electronic device. When each image device has returned to the same timeline, the system can continue the unfinished tasks. In this research, the network time protocol (NTP) was chosen as the standard for every electronic device to follow. The Network Time Protocol (NTP) is a TCP/IP protocol which is designed to synchronise the clocks of computers and various electronic devices over a network. The NTP applies Marzullo's algorithm and is designed to resist the effects of variable latency (Mills, 1999). The use of Marullo's algorithm can precisely calculate the transmission delay between devices to compensate for a difference in rate of flow of data, or time of occurrence of events, when transferring data from one device to another (Abdel-Ghaffar, 2002). In order to synchronise the internal clocks of different

electronic devices with a remote server, the NTP background program has to calculate the round-trip delay time and the offset and integrate this information into the final display (Corin et al., 2007). The downstream user can request the current time from a network server, and use the up-to-date information from NTP to adjust its own clock (Mills, 1991). The overall precisions of NTP can be kept at the level of 10 milliseconds over the public internet and 0.1 milliseconds precision in local area network, making it suitable for measurement, control systems and the standard of time adjustment (Mills, 2011). In practical applications, most workstations, routers, or servers have been equipped with NTP client software to synchronise their clock to the network time server. In most cases the client software is resident in the operating system of each device to perform the time synchronisation process automatically (Levine, 1999). The operating procedures for the system recovery program are introduced below.

Firstly, the computer refreshes its system clock from a chosen NTP server (if the computer can connect to the Internet). Also, every smart phone is forced to resynchronise its internal clock with the chosen NTP server (if the mobile phone is able to connect to the public network or the terminal computer). Then, the computer will monitor the next two exposure reports. If the synchronisation problem still exists, the recovery program will try to fix the malfunction by rebooting the mobile imaging app. The program will re-edit a new script of exposure sequences. The script includes the reboot command of the mobile imaging app, the updated exposure frequency (the system will use the original exposure frequency if the operator does not intervene to reset a new frequency), the starting time for the new imaging sequence and the command for resynchronising with the NTP server. The script will be sent back to each mobile phone to rearrange the new actions via the wireless network. The process of script editing occurs in real-time. However, considering the transmission delay and the time for the mobile reboot, the starting time for next imaging sequence is set to fifteen seconds later when the system health monitoring program has detected the failure of exposure synchronisation.

When the mobile phone receives the script, the mobile imaging app which is responsible for performing the computer's orders will reboot immediately. According

to the output report, the imaging app is rebooted and re-commences operation within around 50 seconds. However, if the handsets' imaging programs do not need to be rebooted, each smart phone only needs 7-10 seconds to resynchronise the internal clock with the NTP server. Considering the network quality and different receiving speed of each individual mobile phone, the readjustment of devices' internal clocks may not be completed at the same time. However, once the devices have resynchronised the clocks to the same standard, every smart phone can synchronously perform a new imaging task by the use of the new script.

3.3.3 Automatic data sorting

When performing a long-term, close-range photogrammetric monitoring task, a large number of image data will be captured. How to efficiently and systematically organise these data then becomes a critical issue. The final step of the whole operation is sorting the image resources on a remote database. In relation to the data sorting process, when the computer edits the imaging script, the data sorting program will copy the imaging timing and the exposure sequence of the script. Following this information, the program generates a series of index files that correspond to each script. When the images are sent back to the computer, the data sorting program will identify the serial number on each image and recognize which script they belong to. Furthermore, the program classifies these images and sends them into specific folders according to different exposure epochs. This whole process is fully automatic. Then, the operator can use this sorted image data for further photogrammetric processing.

3.3.4 Summary of the mobile platform based structural monitoring module

In terms of adopting photogrammetry for the purpose of structural monitoring, the imaging stations are generally arranged around the target objects. A stationary structural monitoring project requires two or more still photographs of the structure from convergent view points for photogrammetric processing (Luhmann et al., 2007). To monitor a vibrating object is more difficult to achieve because it requires highly-synchronised image sequences from multiple camera stations to freeze the structural movements simultaneously (Pappa et al., 2002a). The developed system is

designed to provide an economical solution for structural monitoring of both static and dynamic structures. The system puts emphasis on high quality synchronisation between devices to generate image sequences for photogrammetric applications. According to the differences of network configurations in different test fields, the operators can select appropriate settings to maintain the system clock synchronisation between devices. The operations (which include image acquisition, data transmission and system recovery) are fully automatic. The system exempts operators from attending the site for operation, and thus saves considerable labour in long-term monitoring tasks.

3.4 Advanced photogrammetric processing function

In addition to investigating the use of multiple smart phones to capture imagery for long-term structural monitoring purposes, the potential of a mobile phone platform for the development of photogrammetric processing functions was studied in this research. Some specific photogrammetric functions were developed on an existing Android-based mobile phone platform and these functions were integrated into a multifunctional photogrammetric software suite. In considering the smart phone platforms are not as powerful as personal computers, the conventional image processing algorithms need to be moderately modified to obtain optimal performance. The development restrictions of current Android OS-based smart phone platforms and the details of the modifications are introduced in the following sections.

3.4.1 System layout

The smart phone application features six functions, as shown in Figure 3-15. The ultimate goal is to develop a complete software package for photogrammetric processing on a smart phone platform. The ideal package should include the functions of image capture, image processing, photogrammetric processing using bundle adjustment and 3D modelling. However, development was constrained by the limitations of the mobile phone hardware specifications. For example, the lack of a graphics card means the current mobile phones cannot handle large amount of graphics processing. The configuration of the RAM in the best high-end smart phones

currently available on the consumer market (as at May 2012) is no more than 1 GB³. The memory capacity of a smart phone is much smaller than a computer and the lack of sufficient temporary memory space is not conducive to large amounts of repeated calculations. Therefore, the modules of bundle adjustment calculation and 3D modelling are not included in the current version.

When the surveyor takes photographs around target objects using a smart phone camera, the smart phone also is capable of performing basic photogrammetric image measurement immediately. The operator can convert the colour photograph into greyscale and produce the binary image on a mobile phone. Through the use of this binary image, the mobile phone application identifies the position of targets and calculates the 2D coordinates of targets. In addition to this, the position and orientation of the mobile phone can be recorded at the moment when the mobile phone captures an image using other sensors on the phone. This raw data can either be saved on the smart phone's memory card or can be transferred back to a remote database via a 3G or wireless network. The operator can then continue the mobile phone's unfinished work using the laboratory computer. Through the use of this raw data, the target's 3D coordinates can be reconstructed.

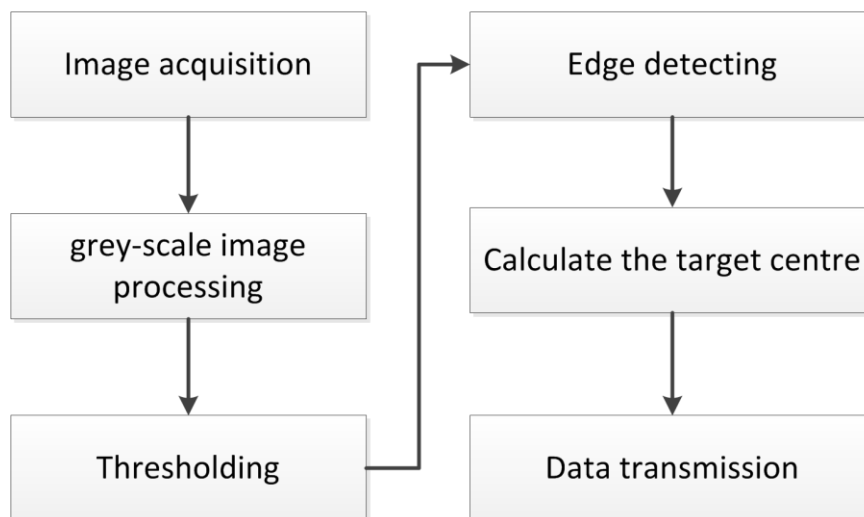


Figure 3-15: Illustration of the single-mobile phone module functions.

³ The Apple iPhone 4S features 512 MB RAM; the SAMSUNG GALAXY S III contains 1 GB RAM; and the HTC One X also features 1 GB RAM.

3.4.2 Conversion to greyscale

The implementation of target detection and coordinate calculation in digital image processing contain repeated iterative computation. The hardware specifications of smart phones are not as powerful as personal computers. In order to smoothly perform a series of image processing procedures on a smart phone platform, the conventional image processing algorithms therefore need to be modified. In terms of the process of automated target detection, the precision of target identification is affected by many factors such as the contrast of the image and the interference that was generated by complex light sources (Song et al., 2008). The preliminary work of image processing is to reduce the complexity of light sources (Petrou and Petrou, 2010). The process of converting colour images to greyscale is one of the methods that can be used to simplify the image information. Unlike the 24-bit colour image, the greyscale image is normally stored with 8 bits per pixel unit. The RGB colour model is one of the most common methods for describing the colour elements in multimedia computer technology (Umbaugh, 2011). According to the trichromatic principle, the RGB colour model uses different ratios of the three components to form a variety of colours (Bailey and Cunningham, 2009). The greyscale image uses the black tone to describe the objects appearance. The greyscale images carry only intensity information. Therefore, the YUV model is more suitable for use in expressing the change of greyscale intensity rather than the RGB colour model (Braga-Neto and Goutsias, 2004). YUV is a colour space typically used as part of a colour image pipeline, where “Y” indicates the brightness (luminance) component and “U” and “V” represent the chrominance (colour) components (Gordon et al., 1994). The conversion to YUV from RGB is calculated by the equations given below (see equation (3.3), (3.4) and (3.5))

$$Y = W_R \times R + W_G \times G + W_B \times B \quad (3.3)$$

$$U = U_{MAX} \frac{B - Y}{1 - W_B} \quad (3.4)$$

$$V = V_{MAX} \frac{R - Y}{1 - W_R} \quad (3.5)$$

Where:

W_R , W_G and W_B are weighting factors of R, G, and B (Braga-Neto and Goutsias, 2004).

In terms of the conversion of a colour image to greyscale, the conversion equation evolves from equation (3.3)

$$Grey = Y = 0.299 \times R + 0.587 \times G + 0.114 \times B \quad (3.6)$$

Where:

$$W_R = 0.299, W_G = 0.587 \text{ and } W_B = 0.114$$

The above recommended weighting factors ($W_R = 0.299$, $W_G = 0.587$ and $W_B = 0.114$) are proposed by the ITU radio communication assembly (ITU, 2012). The settings of the weighting factors are variable, different weights setting of RGB weighting factors resulting in different intensities of greyscale image. However, after repeated tests, the ITU claims that this set of weighting factors settings can provide the best visual effects to the human eye. Therefore, equation (3.6) has become the most commonly used greyscale conversion formula (Yun-shan et al., 2010).

In the course of programming, the calculation of equation (3.6) needs to use floating-point arithmetic. However, the computing performance of floating-point arithmetic on a mobile phone platform is too low due to the lack of built-in temporary memory space. In order to avoid the use of floating-point numbers in arithmetic operations, equation (3.6) was modified into equation (3.7) to improve the computing performance and reduce the system load.

$$Grey = (R \times 299 + G \times 587 + B \times 114 + 500) / 1000 \quad (3.7)$$

Equation (3.7) needs to add 500 to get a rounded result.

The accuracy of coefficient for equation (3.6) is three digits. The equation (3.7) scales the equation (3.6) 1,000 times larger to achieve the integer arithmetic algorithm. By eliminating decimal points from an equation it can significantly improve the computing performance. However, the use of a divider in equation (3.7) still restricts the computing speed. Using a mobile phone (HTC Incredible S) to perform this program, the average processing time to convert an 8-megapixel digital image to greyscale was 5.769 seconds.

In this research, the shift algorithm was used to improve the computing speed of equation (3.7) by removing the division from the greyscale conversion equations. This lightweight method can be easily operated in current Android phones. In considering the balance between the processing speed and image distortion, customarily programmers use 16-bit precision to modify the original greyscale conversion formula (the equation (3.6)) (Burger and Burge, 2007). Two to the power of sixteen is 65,536, therefore, the weighting factors can be converted as:

$$W_R' = 0.299 \times 65,536 = 19,595.264 \cong 19,595 \quad (3.8)$$

$$W_G' = 0.587 \times 65,536 = 38,469.632 \cong 38,469 \quad (3.9)$$

$$W_B' = 0.114 \times 65,536 = 7,471.104 \cong 7,471 \quad (3.10)$$

By adopting the new weighting factors of R, G, and B, equation (3.7) can be converted into:

$$Grey = (R \times 19,595 + G \times 38,469 + B \times 7,471) \quad (3.11)$$

In comparison with the equation (3.6) and (3.7), using equation (3.11) to perform greyscale conversion can improve the computing efficiency greatly. After repeated tests, it could be seen that using the lightweight module can shorten the processing time from 5.769 seconds to 42 milliseconds, and this improvement leads to the achievement of real-time greyscale conversion. Because the processing time has been shortened to 42 milliseconds, the greyscale imagery can be seen on the phone screen in real-time (Figure 4-20). The conversion delay is difficult to detect with the naked

eye.

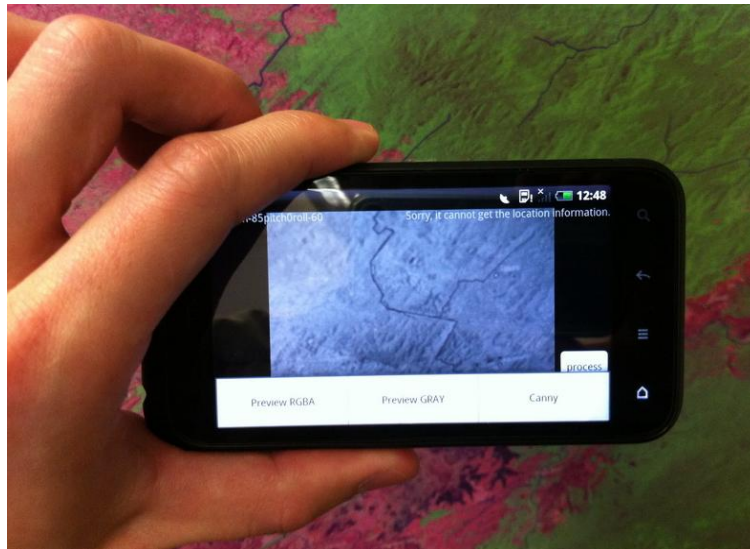


Figure 3-16: Illustration of real-time greyscale conversion.

As shown in Figure 3-16, the converted greyscale image can directly display on the mobile phone screen, and the operator can preview the result to check the quality. If the resolution of greyscale image is not good, the operator can modify those weighting factors of R, G, and B (as shown in equation (3.8), (3.9), (3.10)) using higher precision coefficients.

Although the processing time of greyscale conversion has been significantly improved, there is still the concern of the loss of image information. In the conversion process of the shift algorithm, the weighting factors (W_R' , W_G' and W_B') have been rounded. The rounding causes a minor loss of image quality. Both equation (3.7) and equation (3.11) can be used for greyscale conversion, it is necessary to inspect the difference between these two output results. As mentioned above, the greyscale images carry intensity information only. The differences between different greyscale images can be evaluated by comparing the differences of their luminosity histograms (Li et al., 2011b).

A comparison has been undertaken to evaluate the difference of luminosity histograms between these greyscale conversion methods. Figure 3-17 (a) shows an original colour photograph and its corresponding luminosity histogram. Figure

3-17 (b) is the converted image, which was converted by the program based on equation (3.7), and the Figure 3-17 (c) was converted by the program based on the equation (3.11). Each of these three luminosity histograms show similar luminosity waveforms and the similarities indicate that these three histograms were analogous in the quality of the luminance distribution (i.e. the relative strength of all luminance values of these three histograms were similar). The mean value represents the average brightness of an image (Jafar and Hao, 2007). In Figure 3-17, the average luminosity values of these three images are very similar. The average brightness of the original photograph is 156.98 nits. After conversion, the mean values were slightly decreased to 156.25 nits (Figure 3-17 (b)) and 156.29 nits (Figure 3-17 (c)) respectively. The overall brightness of the Figure 3-17 (c) is higher than that in Figure 3-17 (b). The standard deviation of each histogram describes the deviation between the brightness values of image pixels and the average. The smaller standard deviation shows lower imagery contrast (Cavalin et al., 2006). Through the use of the equation (3.11), the image contrast was slightly enhanced after conversion as Figure 3-17 (c) shows; the highest standard deviation among these histograms. The evidence shows that although the overall brightness reduced after conversion through the use of equation (3.11), the image contrast is slightly strengthened.

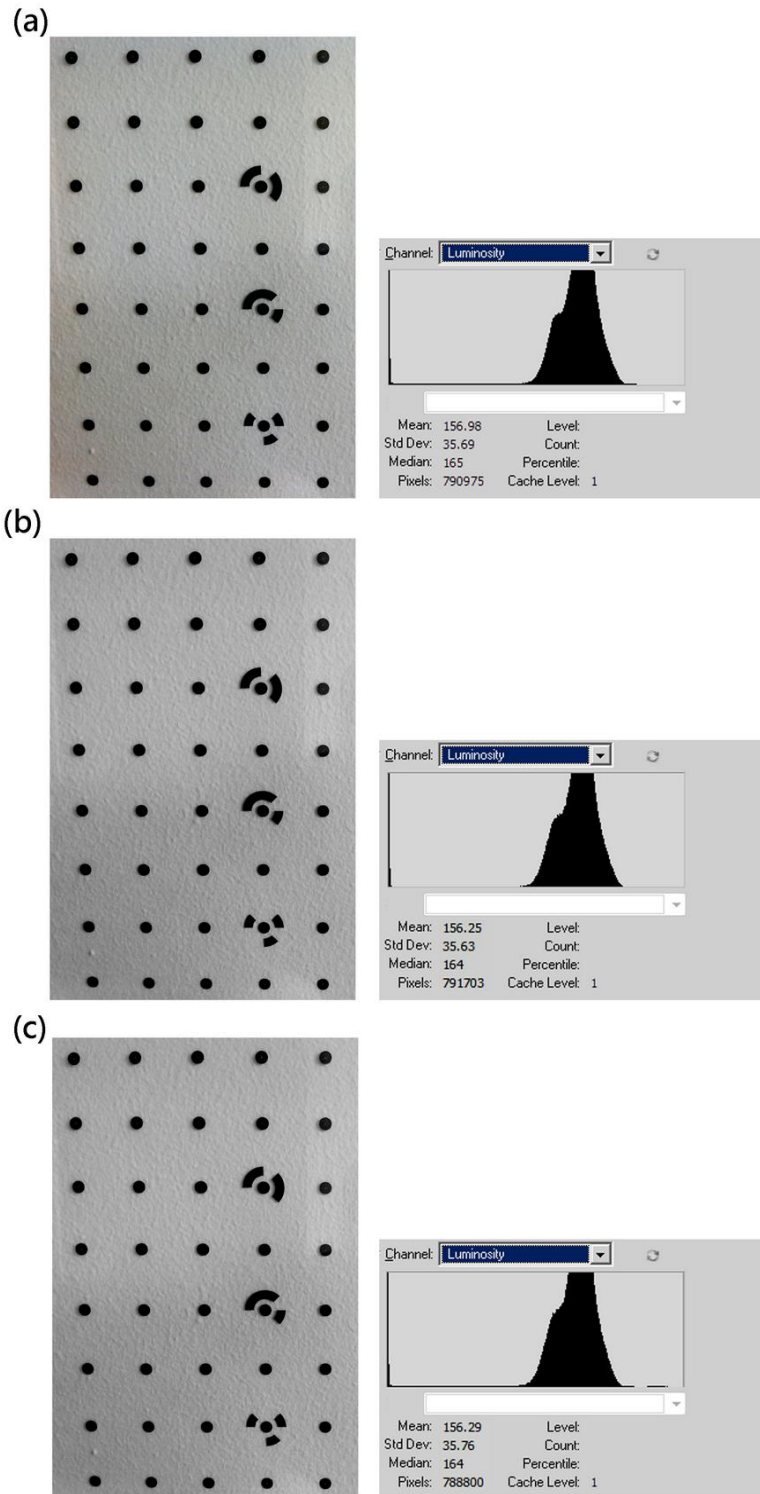


Figure 3-17: Comparison for luminosity histograms. (a) is the original colour photograph; (b) is the result which was processed using equation (3.7) and (c) is the result processed by equation (3.11).

3.4.3 Greyscale to binary conversion

The greyscale conversion process simplifies the colour information of the image (Braga-Neto and Goutsias, 2004). Thresholding can further highlight the edge features of objects in the image. Thresholding is a method of image segmentation. The operation of an image binarizing process is to use an appropriate threshold value to divide the image information (greyscale image) into two parts (bright regions and dark regions) (Pratt, 2012). If the grey intensity is higher than the threshold value, the pixel unit is defined as a ‘bright spot’. On the contrary, if the grey intensity is below the threshold, the pixel will be defined as a ‘dark point’ (as shown in Figure 3-18).

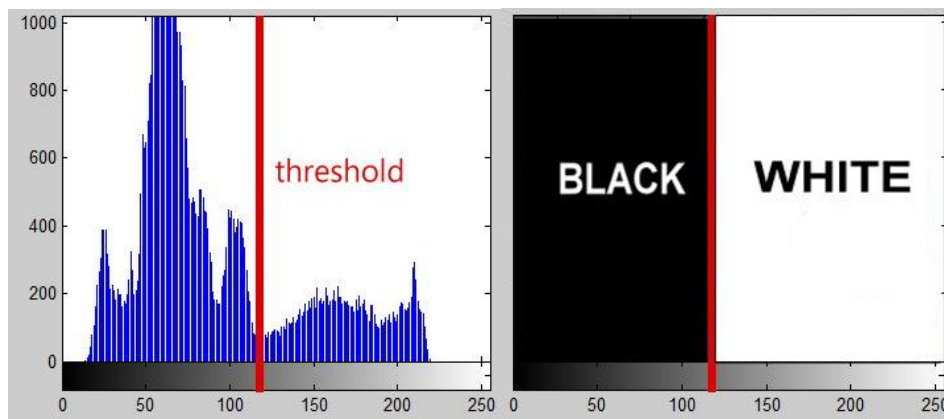


Figure 3-18: Illustration of thresholding (Marchand-Maillet, 1999).

The threshold value controls the judgment of bright and dark regions (Chun-Ming and Hsi-Jian, 2002). Otsu's method is one of the most commonly used algorithms to automatically determine the optimal threshold (Chen et al., 2012). The Otsu algorithm classifies the image into two groups (foreground and background). In order to determine these two groups, a program was developed based on the Otsu algorithm. The program calculates the optimum threshold for the image segmentation whilst the between-class variance between these two groups is maximised (i.e. equation (4-11) has the maximum result). When the variance σ_b^2 reaches its maximum, the foreground receives maximum contrast with the background and an optimal threshold can, therefore, be obtained.

$$\sigma_b^2(t) = P_1(t)P_2(t)[\mu_1(t) - \mu_2(t)]^2 \quad (3.12)$$

Where σ_b^2 is the variances between these two groups, t is the threshold, P_1 is the probability of which of the grey values of pixels is less than, or equal to, the threshold t and P_2 repres

ents the probability of grey values with pixels greater than t . μ_i is the means of groups 1 and group 2.

In this research, the mobile phone-based thresholding program was developed using the Java programming language. The use of the Otsu algorithm can automatically determine a proper threshold value (Jun and Jinglu, 2008). The system uses customized targets which were designed as a three-quarter circular black pattern on a white background (Figure 3-19). This high contrast design helps the program to process good quality binary images even though the camera quality of mobile phones is not as good as professional digital SLR cameras.

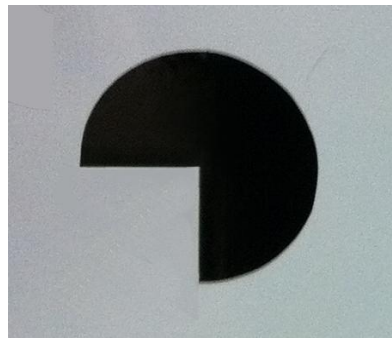


Figure 3-19: The three-quarter-circle target pattern.

3.4.4 Developed target detection algorithm

The final process of this single mobile phone module is to determine the targets' 2D coordinates relative to the image sensor frame. The procedure includes the target identification and 2D coordinate calculation. At this stage, the priority is to identify the targets in the screen and then calculate the 2D coordinates of these targets. The three-quarter-circle target pattern was designed using the AutoCAD software. The pattern was sent to a vinyl cutter and cut at the required size. In order to enhance the contrast, each black target sticker was attached onto a square white vinyl sheet. After the thresholding process, excess image information can be removed from the

photograph. The program will then automatically perform the edge detection by searching for inflection points in an image.

In terms of the most frequently used target detection algorithm, the Hough transform has been widely used for image processing to deal with feature extraction. In terms of the use of the generalised Hough transform for target detection, the computing process involves repeated calculations in order to identify and remove the invalid boundary points. Therefore, a large number of invalid calculations significantly reduce the computing speed and cause a heavy load on system memory (McManigle et al., 2012). Due to the current smart phone hardware limitations, the platform cannot smoothly execute those software packages which utilise the Hough transform algorithm for image processing. For this reason, a new algorithm was developed in this research that is used to detect the centre of each target pattern.

The developed automatic, mobile phone-based target detection app is programmed using an in-house algorithm. The first step of this target detection app is to manually select the region of interest on the mobile phone screen (Figure 3-20) and the program narrows the identification area to the size of the fingertip. Using manual selection to narrow down the identification area can significantly reduce the load on the mobile phone system.

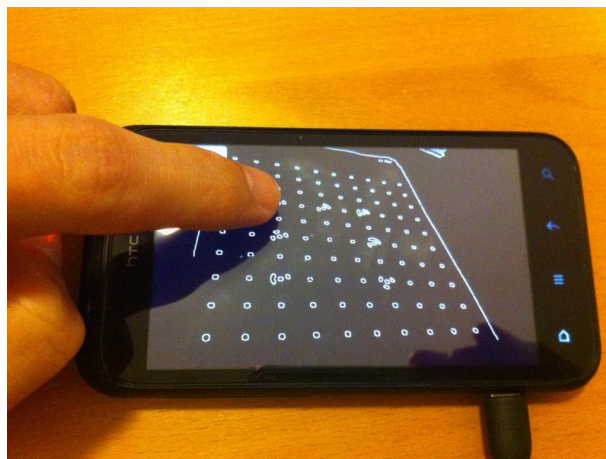


Figure 3-20: Manual selection of the targets.

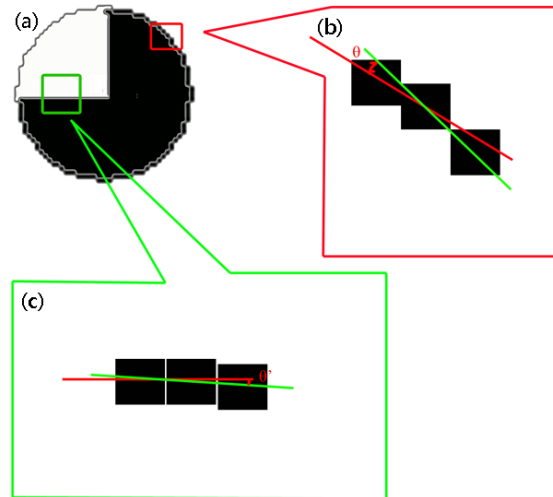


Figure 3-21: Estimation of curves and straight lines.

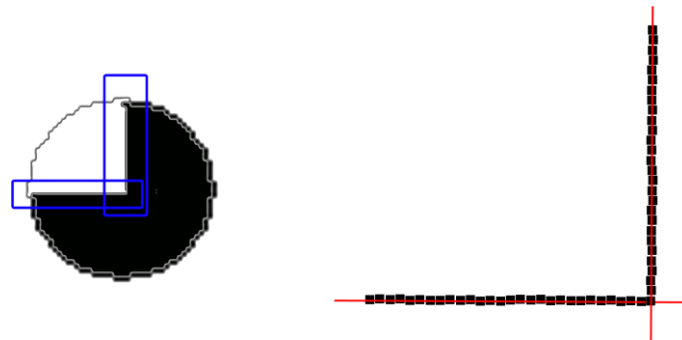


Figure 3-22: Illustration of linear regression.

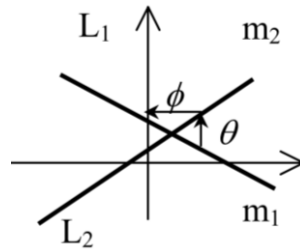
After converting colour images to binary images, the high-contrast targets can be explicitly displayed. Then, the targets' outline can be extracted using the Canny algorithm. The use of the Canny algorithm has following advantages:

1. Single response – a given edge in the image can only be marked once (Syahmi et al., 2011);
2. Efficient non-maximum suppression – the algorithm only retains the highest intensity to the borders whilst edge thinning can be achieved (Jun and Jinglu, 2008).

Since the dedicated target pattern can be clearly displayed in binary imagery, the outline of each target is easier to identify than other objects in the environment. The

Canny edge detection operator identifies these inflection points and defines them as the edge of objects (targets). The pixel units on the outline of a target are classified into two types; one group is located on the circumference and the other group is located on the two sides of the gap (i.e. located on the radii) (Figure 3-21 (a)).

In this research, a threshold value is used to distinguish the edge points that are located on the circumference or on the radii. The mobile phone program identifies these pixel units and determines the linear equations between each set of two adjacent pixels. The program then calculates the angle of each of the two adjacent straight lines using equation (3.13).



$$\tan \theta = \frac{m_1 - m_2}{1 + m_1 m_2} \quad (3.13)$$

$$\tan \phi = \frac{m_2 - m_1}{1 + m_1 m_2} \quad (3.14)$$

Where m_1 is the gradient of L_1 , and m_2 is the gradient of L_2 (Wylie, 2008).

Four pixel units that are located before and after the inflection points (the A, B and C areas in Figure 3-23) are not added to the calculation because the orientation of these tangents have irregular variations. An error will arise if the program includes these points in the calculation. If the angle is greater than the threshold value, these three points will be defined as points on the circumference (Figure 3-21 (b)). Only when four consecutive angles are all less than the threshold value can these pixel units be defined as points on the radii (Figure 3-21 (c)).

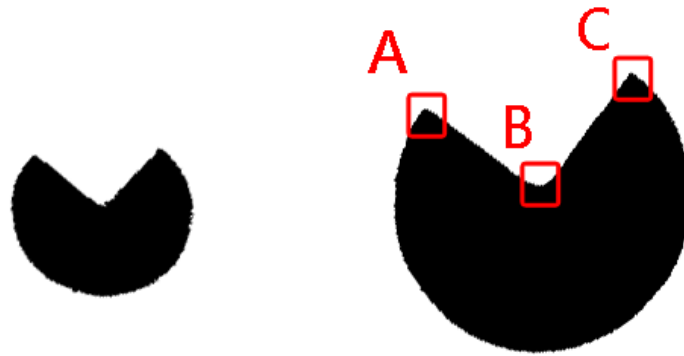


Figure 3-23: The targets' binary images.

Once the points on two sides of the gap have been found, the program computes the equations of the best fitting lines to represent these radii using the linear regression calculator (Figure 3-22). These two best fitting lines represent two radii of the three-quarters circular pattern. Therefore, the intersection point of these two line equations can be defined as the centre of the target and the coordinate of this intersection point can be used to represent the targets coordinates.

3.4.5 Summary of the onboard photogrammetric processing software suite

In considering the smart phone devices have lower computational efficiency (in comparison with the personal computers) and lower image quality (in comparison with the digital SLR cameras) a mobile platform based photogrammetric software suite was developed in this research. The software suite integrated a series of modified image processing functions and a dedicated target detection algorithm. The onboard target detection application can assist engineers to precisely identify the location of each target on low resolution images when performing a photogrammetric processing.

3.5 Summary

Focusing on the demands of using photogrammetric measurements for long-term structural monitoring, a low-cost, mobile phone-based photogrammetric monitoring system has been proposed. The system is designed based on photogrammetric principles. Controlled by the developed software, the mobile system can capture

synchronised imagery with wide-area coverage and automated operation. These characteristics allow the operator to apply this system in a wide variety of structural monitoring tasks (even for long-term structural monitoring of flexible structures). Because of the advantage of cost, the operator can arrange more imaging stations in the test field to strengthen the photogrammetric network. Meanwhile, the mobile platform based imaging station is programmable which allows the development of specific programs to meet different measurement requirements. A dedicated target detection program has been developed on an Android OS-based smart phone platform which allows surveyors to directly perform precise target coordinate calculation on a smart phone during the image acquisition process. This developed target detection program is especially designed to process those image data which are captured by low-resolution and non-metric camera sensors. A series of experiments were undertaken to examine these two modules and evaluate the performance of this system for flexible structural monitoring applications. The experiments are presented in the next chapter.

CHAPTER 4. PERFORMANCE EVALUATION

4.1 Introduction

The system which was developed in this research aims to provide a cost effective and precise solution for long-term structural monitoring applications. To attain this goal, a multifunctional integrated software suite was designed to control mobile phones for automatic photogrammetric applications. In order to inspect the operational performance of the system, a series of experiment were undertaken. The experimental layout of each test is introduced below.

4.2 Photogrammetric network configuration

Chapter 2 introduced the most frequently used photogrammetric network configurations. Meanwhile, the relationship between the measurement precision and the strength of a photogrammetric network (the scale ratio and the quantity of imaging stations) is outlined in § 2.2.3. In order to further investigate the relationship between configuration of image stations and the accuracy of photogrammetric results, an indoor experiment was undertaken.

4.2.1 Experimental configuration

Both a mobile phone camera (the HTC Incredible S) and a DSLR camera (Canon 50D) were mounted on tripods (Figure 4-1) and were setup 100 cm away from a test table to converge on the test board under laboratory conditions (the specifications of these two cameras are detailed in § 3.2.2).

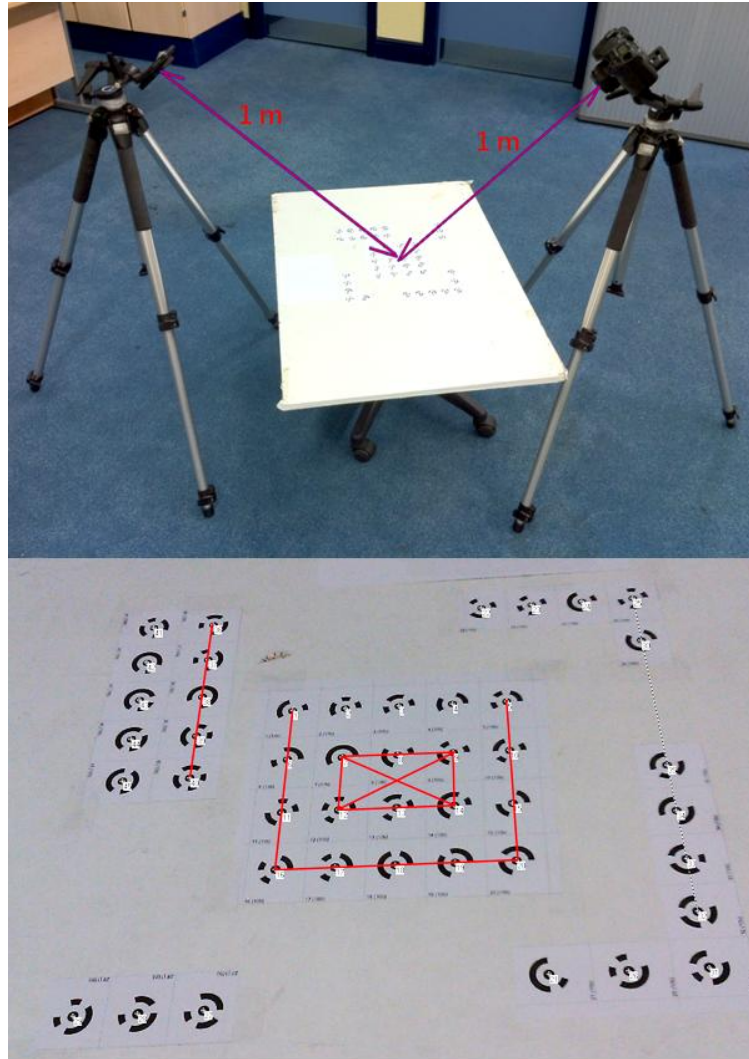


Figure 4-1: Experimental configuration and target arrangement.

The rigid target board was placed on an office chair which allowed smooth rotation to any angle for simulating multiple image acquisition stations. In this test, eight rotation angles were used. The size of the target field was 320 mm in width and 450 mm in length and the diameter of each target's central circle was 4 mm. A total of ten distances on the target board were selected for comparison. In terms of the image acquisition process, both of the cameras took fifteen rounds and every single round included 8 images which were taken from 8 different angles (as shown in Figure 4-2).

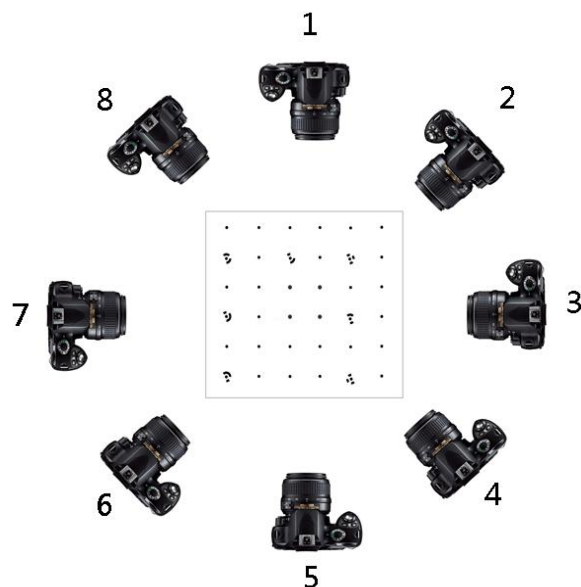


Figure 4-2: Illustration of eight different imaging positions.

In order to simulate the use of three cameras for image acquisition, the comparison used the image data which were captured from Position_1, Position_4 and Position_7. The simulation of four imaging stations used those image data from Position_2, Position_4, Position_6 and Position_8. The simulation of six imaging stations used the images from Position_2, Position_3, Position_4, Position_6, Position_7 and Position_8. Finally, the image data from all imaging positions were used for comparison to represent the use of eight imaging stations. By comparing the standard deviation of each combination, the improvement of photogrammetric precision by increasing the quantity of imaging stations was estimated.

4.2.2 Test results

Both the DSLR camera (nominal focal length 28 mm) and the mobile phone camera (nominal focal length 4.57 mm) were setup 100 cm away from the test table. The chosen target size was 4 mm in diameter. Under this configuration, the target widths in each photograph can be calculated as:

$$I_{R(\text{mobile})} = \frac{O_R \times f}{D} = \frac{4 \times 4.57}{1000} = 0.018(\text{mm}) \quad (4.1)$$

$$I_{R(DSLR)} = \frac{O_R \times f}{D} = \frac{4 \times 28}{1000} = 0.112(mm) \quad (4.2)$$

Also, the total number of pixels in diameter contained in a target in the sensor's image can be calculated as:

$$P_{(mobile)} = \frac{F_p \times I_R}{F_w} = \frac{3264 \times 0.018}{4.6} = 13(pixels) \quad (4.3)$$

$$P_{(DSLR)} = \frac{F_p \times I_R}{F_w} = \frac{4752 \times 0.112}{22.3} = 24(pixels) \quad (4.4)$$

The calculated results of equation (4.3) and equation (4.4) indicate that the chosen target size for both cameras can meet the restrictions of minimum target size of PhotoModeler.

A total amount of ten distances on the rigid test object were selected for comparison. The photogrammetric results were calculated using three, four, six and eight images for processing, respectively. The averages of standard deviations of both cameras are presented in Figure 4-3.

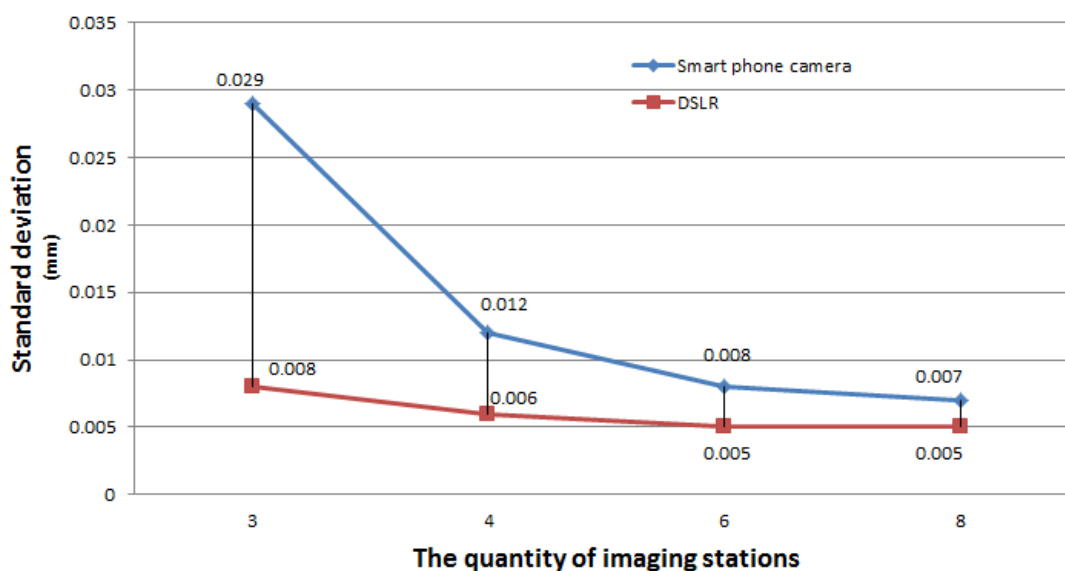


Figure 4-3: Relationship between the precision and the quantity of imaging stations.

The standard deviation shows how much variation (or dispersion) exists from the average. Among these four different configurations, as expected, the three-camera configuration results in the largest standard deviation in both camera datasets. When the quantity of camera stations was increased from three to four, the standard deviation value of smart phone camera dataset improves 47 % from 0.019 mm to 0.010 mm. The result can be improved by a further 20 % (from 0.010 mm to 0.008 mm) when the camera stations were increased from four to six. The use of eight-camera configuration results in the best performance in precision (the standard deviation is 0.007 mm). In comparison with the smart phone datasets, the improvement in standard deviations of DSLR camera datasets is relatively moderate. When the quantity of camera stations was increased from three to four, the standard deviation value of DSLR camera dataset improves 25 % from 0.008 mm to 0.006 mm. When the camera stations were increased from four to six, the result improved 17 % further (from 0.006 mm to 0.005 mm). However, the results (rounded down) shown no difference when the camera stations were increased from six to eight. Though the use of more image stations can significantly improve the photogrammetric result, the improvement will gradually reach a certain constant as per the trend presented in Figure 4-3.

Along with the increase of imaging stations comes a decrease in standard deviation (the discreteness of photogrammetric results). Clearly the more camera stations that can be utilised, the better the photogrammetric result can be obtained. When the use of image stations cannot generate high quality images for photogrammetric processing, the improvements become even more obvious. However, the increase in image stations only significantly improves the photogrammetric processing results to a certain number of stations. Constrained by development budget, this research used four mobile phone cameras to continue the subsequent experiments.

4.3 Inspection of system synchronisation

In terms of structure monitoring of flexible structures, there are several key elements and requirements for those images that are acquired and used for photogrammetric processing. High resolution is a priority since the use of high-quality imagery (clear

and high resolution) facilitates target recognition during image processing (Adamos and Faig, 1992). The capture of synchronised image sequences from multiple camera stations is another essential requirement for dynamic object monitoring (Pappa et al., 2002b).

The system which was developed in this research uses a software program to control two or more handset devices to perform structural monitoring. The system emphasises the performance of high quality system synchronisation between devices. The operator is able to capture synchronised image sequences from multiple mobile platform based imaging stations and then apply these images for photogrammetric processing to analyse the deformation or movement of target objects during the imaging sequences. In order to estimate the performance of the system for long term synchronisation, a continuous experiment was undertaken for a period of three days.

4.3.1. Experimental configuration

In the experiment, two cameras provided convergent imaging of a digital stopwatch displaying on a 15-inch LCD screen. The digital stopwatch provides a resolution of 1/1000 second and displayed on a screen with a refresh rate of 22 milliseconds. Two smart phones were mounted on mobile phone holders as shown in Figure 4-4 (a) and these two handsets were controlled by a remote computer via a Wi-Fi network. Both the computer and mobile phones were set to update their system clocks from a local NTP in Newcastle University, UK (the time servers at Newcastle are: ntp2.ncl.ac.uk and the backup servers are: ntp0.ncl.ac.uk & ntp3.ncl.ac.uk). In order to carry out a long-term experiment to collect enough image data for analysis, both of these mobile phones were connected to general-purpose alternating current (AC) electric power supplies to ensure mobile phones would not shut down because of loss of battery power. The experiment was conducted over three days for a total period of 62 hours. During this period, the two cameras continuously captured images synchronously with an exposure frequency of two frames per minute (30 seconds intervals).

Checking the time-sequenced image pairs afterwards, the images show almost no differences between the two camera stations at the same exposure epoch. For example,

Figure 4-4 (b) show the image pairs which were captured from two different cameras positions and the time information on the two photographs at the same exposure epoch were identical; the difference cannot be distinguished by human eye. In order to facilitate scientific analysis however, it is essential to quantify differences. The following comparisons aimed to estimate the camera synchronisation and the performance of system failure recovery by comparing the shutter speed and exposure timing of each exposure sequence.



Figure 4-4: The system synchronisation experiment, showing (a) the handset and monitor configuration, with examples of captured images from the two handsets at the same epoch (b).

The EXIF (exchangeable image file format) information of each photograph was extracted and compared for each exposure epoch. Each digital image contains an EXIF. The EXIF is a standard that can be applied to record the attribute information of digital imagery and associated data including information essential for image processing such as image format, image size, shutter speed, aperture etc. (Kee et al., 2011). Although the camera shutter speed of each smart phone can be fixed by the

integrated software which was developed in this research, the actual shutter speed may have slightly changed if the internal voltage of the smart phone device is not stable. For this reason, it is important to ensure the exposure of each camera is synchronised and that the shutter speed is always constant when dealing with a photogrammetric sampling process. Because the shutter release delay of any individual camera station leads to a longer exposure time, this may delay the next exposure task on that particular image station. Also, the image station will therefore capture extra imagery information than other cameras because of the longer exposure time. In order to inspect whether the shutter speed remains constant, a thorough examination was conducted for every exposure epoch between the two cameras. The images were sorted and classified into different folders following the exposure sequence. The image number (photo ID), shutter speed, and the exposure timing from the EXIF of each photograph were extracted and listed in a spreadsheet, then, the differences between the two imaging stations were compared to evaluate the performance of system synchronisation.

4.3.2. Test results

Both of the mobile phone cameras were continually capturing still photograph sequences for 62 hours and each mobile phone was therefore expected to generate 7,440 photographs. If the system recovery program sends commands to the mobile phone to reboot the handset's imaging app and the processing time is longer than the default exposure frequency, this camera station will not be able to capture images at the given time. Thus the cameras' shutter speed values at that moment drop down to zero because the camera shutter is not working and the total number of images to each camera was less than 7,440 frames.

A comparison of shutter speed values between the two cameras were undertaken to pinpoint the exact moment of system reboots. The previous tests signified that the system refresh process (the resynchronise of the devices' internal clocks with the NTP server) normally takes about 7 to 10 seconds to get the system ready for the next image acquisition task. If the handsets' imaging app needs to be rebooted, the processing would take around 50 seconds to return to normal working conditions. In

this experiment, the exposure frequency is thirty seconds (i.e. each camera took two images per minute); therefore, the exposure interval (30 seconds) is long enough for each electronic device to refresh their time information from NTP when the system error occurred. As a result, the numbers of system reboot can be calculated by counting the number of times the shutter speed is recorded as zero over 14,880 image exposures (both Camera_1 and Camera_2 produced 7,440 sets of data). In this experiment, the shutter speed of both cameras was set to five hundredths of a second (0.002 seconds). The shutter speed of each exposure of Camera_1 and Camera_2 were plotted in Figure 4-5 (a) and Figure 4-5 (b) and the comparison in shutter speed of Camera_1 and Camera_2 shown in Figure 4-5 (c). If the cameras worked properly, the shutter speed will be five hundredths of a second (0.002 seconds). However, if any one of these two cameras did not work at any moment, then the camera will not be able to send an exposure report back to the computer for analysis. As a result, the cell represents the shutter speed of the moment in the spreadsheet will be recorded as zero. To plot the shutter speed of each exposure in Excel charts, the sudden drop of the curve will help the operator to pin-point the exact timing of camera failure. If the shutter speed of both cameras shown zero at the same time, this exposure epoch can be regarded as the moment when imaging app on both cameras are being rebooted.

Figure 4-5 (a) indicates that there is one sudden vertical plunge at the 5015th set of data, which means the computer did not detect any shutter actuation from Camera_1 at that moment. Checking Figure 4-5 (b), Camera_2 shows the same trend. Figure 4-5 (c) shows the overlap of the shutter speed data of Camera_1 (Figure 4-5 (a)) and Camera_2 (Figure 4-5 (b)). The only plunge for both cameras happened at the 5015th epoch, while the other parts of the exposure data all worked correctly, consistently maintaining a shutter speed of five hundredths of a second. Examining the status reports of the 5015th exposure of Camera_1 and Camera_2, the reports highlight that both cameras did not record an exposure at the scheduled time due to the software rebooting. By checking the comparison charts, the number of the software reboots can be calculated. However, the comparison charts in Figure 4-5 cannot explicitly point out the timing of system reboot. Also, the total number and exact timing of system clock refreshes are unknown. To determine this missing information, the exposure

timings from Camera_1 and Camera_2 were listed side by side following the exposure sequence and comprehensively compared.

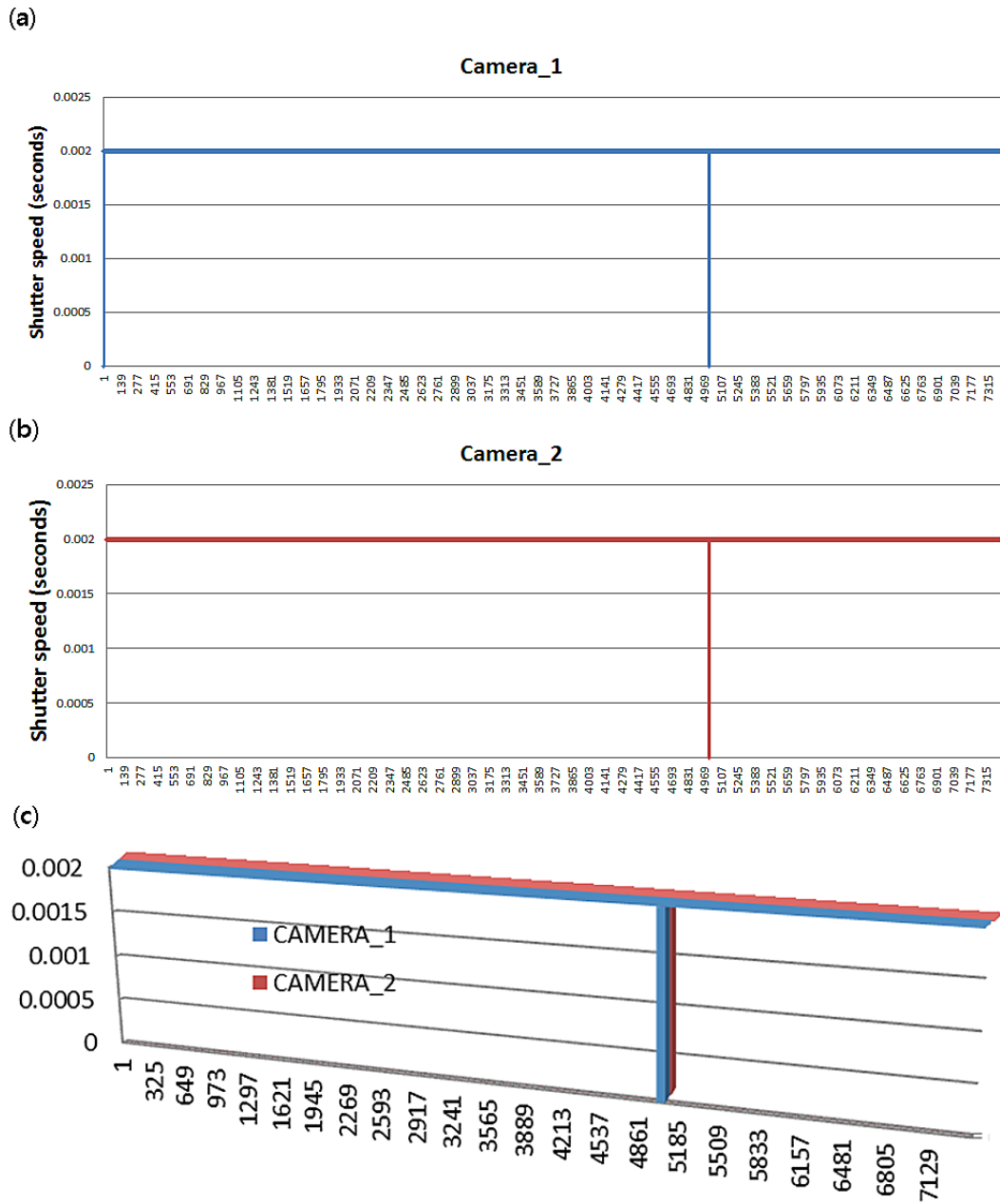


Figure 4-5: The comparison of exposure epochs, showing (a) the shutter speed of each epoch of Camera_1 whilst (b) is the shutter speed of Camera_2, and (c) is the overlapping of (a) and (b).

Figure 4-6 shows the comparison of the absolute time of capture between Camera_1

and Camera_2. The information of each absolute time of capture (timing of exposure) was obtained from each EXIF file. To subtract each absolute time of exposure of Camera_2 from the data of Camera_1, if the exposure timings are exactly identical, then the difference between these two cameras is zero (the results of exposure timing were displayed up to millisecond level). Otherwise, the differences in absolute time of exposure are shown as a sudden leap or drop in Figure 4-6. The differences were calculated using the absolute time data of Camera_1 minus Camera_2. Therefore, a positive result represents the exposure timing of Camera_1 being later than Camera_2. On the contrary, a negative result indicates the camera shutter of Camera_1 was triggered earlier than Camera_2.

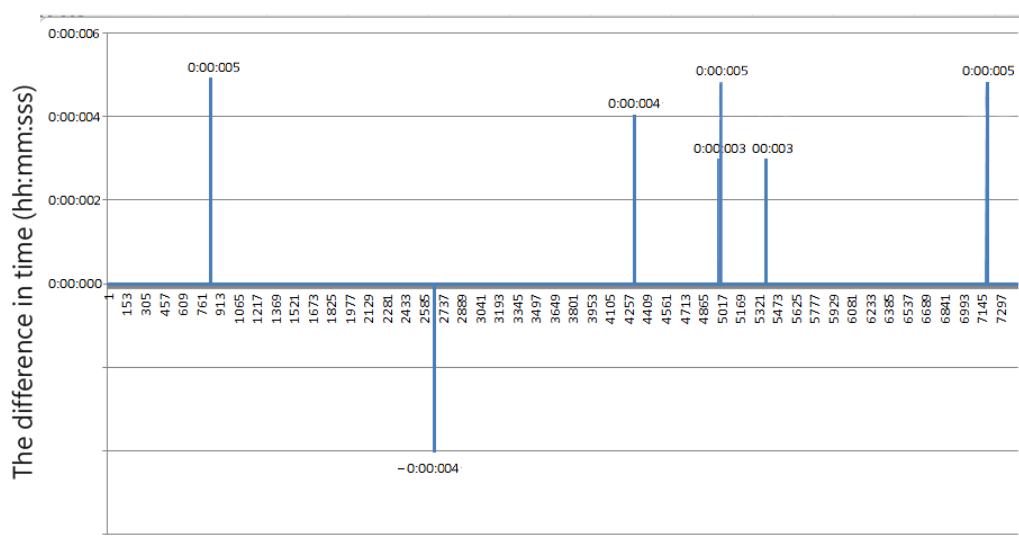


Figure 4-6: The comparison of the absolute time of capture between the two mobile phone stations.

Among 7,439 exposure epochs, the results indicate that in total there were seven non-synchronised exposure epochs that appeared during the experiment. Among this uncoordinated exposure data, six of them were positive and only one was negative. Only one successive non-synchronised exposure events happened at the 5013th and 5014th epochs. In such circumstances, the mechanisms of system health monitoring and recovery modules forced all devices to perform a system reboot process and the activating time is consistent with the conclusion of Figure 4-5.

4.4 Performance evaluation of developed target detection algorithm

In this section, a series of experiments were performed to provide an assessment of how to determine the optimal target size and the optimal threshold value for the developed mobile phone target detection program, and a comprehensive comparison between the self-developed target detection algorithm and conventional algorithms was conducted.

4.4.1 Optimal target size determination for developed target detection algorithm

The optimal target size for this developed target detection program is variable. It changes the scale of test fields (the object distances are different). Because this is a self-developed algorithm, the optimal target size setting is different from other commercial photogrammetric software packages. The target size which is mentioned in this section refers to the targets image size (in pixels) rather than physical size. As a result, the reference standard for a target size's optimal setting can be applied in different scales of test fields and different mobile phone cameras with different imaging sensors.

4.4.1.1 Experimental configuration

In order to assess the correlative relationship between the target size and the precision of target identification, six different pixel sizes of targets were applied in the test. From the biggest to the smallest, the targets were approximately 18,600 pixels (the radius of the target (r) in the image is 77 pixels), 5,000 pixels ($r = 40$ pixels), 2,800 pixels ($r = 30$ pixels), 900 pixels ($r = 17$ pixels), 400 pixels ($r = 11$ pixels), and 225 pixels ($r = 9$ pixels), respectively. These targets were designed using AutoCAD software. The patterns were printed out on a transparency film (A4 size) and attached onto a flat board. The mobile phone was mounted on a handset holder and was fixed on a solid tripod. The test board was placed onto a swivel chair which allowed smooth rotation to any angle for imaging. This configuration allowed the researcher to use one single camera to simulate the use of multiple image acquisition stations whilst keeping the object distance constant.

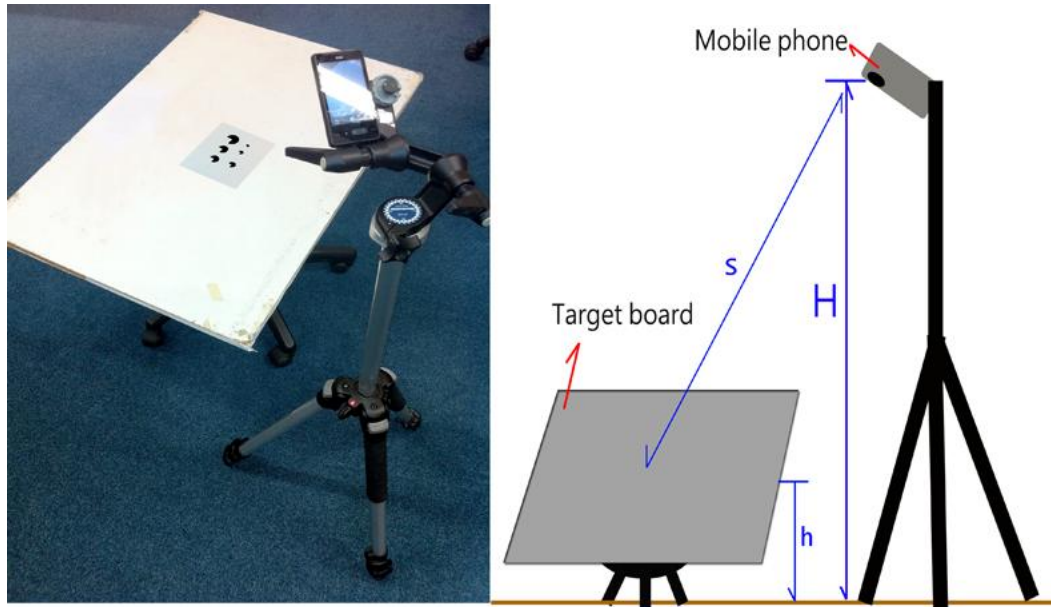


Figure 4-7: The field configuration for optimal target size estimation.

In this test, the object distance was approximately 1.1 m. By rotating the swivel chair, the camera took pictures at 12 o'clock, 2 o'clock, 4 o'clock, 6 o'clock, 8 o'clock and 10 o'clock positions. The image acquisition process was repeated five times. A total of 30 images (five images in each position) were captured for further photogrammetric processing. These images were converted to binary images so the edges of targets could be extracted. The smart phone program picked out all the pixel units on the target's edge and divided these edge points into two groups according to their location (on the circumference or on radii). The linear equations for any two adjacent pixel units on the edges were calculated. Then, the angles between two adjacent linear equations were computed. According to the calculated results, the operators can evaluate the optimal target size and the corresponding experimental configuration (object distance, camera sensor format and focal length of lens) for this target detection program.

4.4.1.2 Test results

The linear equations for any two adjacent pixel units that were located on each target's edge were calculated; also, the angles between two adjacent linear equations were computed. These data were divided into two groups according to the location of

these sample points (on the radii or on the circumference). The averages of these angles for both groups were calculated and the results are listed in Table 4-1.

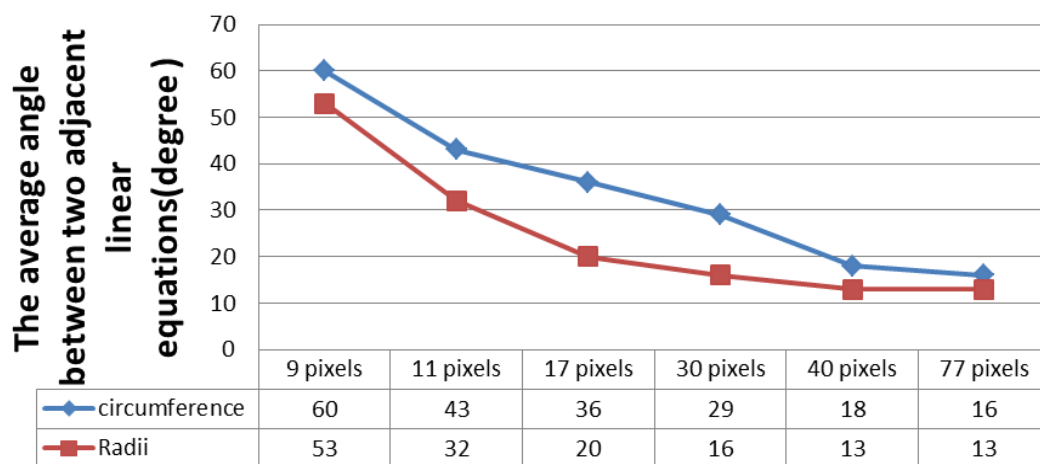


Table 4-1: The comparison of curvature changes. The ordinate represents the mean of the calculated angle, and the abscissa represents the targets' radii.

Theoretically, the calculated angles on the circumference (Figure 3-21 (b)) are bigger than the results that were located on radii (Figure 3-21 (c)) because the curvature on circumference is greater than on the radii. The general trend of Table 4-1 indicated that the calculated results on the circumference were all greater than those results located on the radii. The results were in line with the inference. According to the results shown in Table 4-1, the use of 9-pixel-radius targets results in the biggest calculated angles in both groups. The average angle on the circumference is 60 degrees whilst the calculated result on the radii is also up to 53 degrees. The average angles on both locations gradually decreased with increasing target sizes. The results of these two groups began to have a clear distinction when the target size was bigger than 11 pixels. At 40 pixels radius, the results of these two groups become close again. Unlike the 9-pixel-radius results, the average angles to the 40-pixel targets and the 77-pixel targets were two sets of results which have the minimum average angles. This phenomenon means that the graphics resolution can be improved significantly with the use of larger target sizes. However, the use of oversize targets result in the curvature of the circumference being relatively flat, the curvature of small distances on the circumference is similar to that of straight lines. For this reason, the differences

between these two groups using the 40-pixel target and 77-pixel target were getting closer than the use of smaller target sizes. When using the 40-pixel target and 77-pixel target for the test, the mobile phone-based target detection program has found it difficult to distinguish the differences between these two groups. In summary, the optimal target size for the mobile phone-based photogrammetric software (in conjunction with the HTC Incredible S) can be adjudged to be around 11 to 30 pixels radius. The average calculated angle using the 11-pixel targets was 43 degrees on the circumference and 32 degrees on the radii, whilst the results with the use of the 30-pixel target were 29 degrees and 16 degrees, respectively. Under this experimental configuration (object distance: 1.1m; imaging sensor format: 4.6 x 3.4 mm; focal length of lens: 4.57 mm), the experimental result shows that, when the chosen target sizes are in the range of 11 pixels to 30 pixels, the developed target detection program has the best performance to distinguish the location of pixel units on the target's edge.

4.4.2 Comparison of target detection algorithms

The following experiments were undertaken to evaluate the performance of the developed target detection algorithm by comparing it with one of the most common algorithms to calculate the centre of an ellipse – the Hough transform (the detail of using Hough transform for target detection is outlined in Appendix II). Despite the fact that the mobile phones' hardware specifications cannot smoothly implement the Hough transform-based program, a dedicated target centre detection program is developed for comparison purposes. These two programs (one is the mobile platform based program, and the other one is a design based on the GHT that is performed on the PC using an Android emulator) processed the same image data. Both programs computed the target's 2D coordinates and the two sets of coordinate data were provided to photogrammetric software (PhotoModeler) to assess the final results. The results were compared with each other to evaluate the performance of the developed algorithm. Figure 4-8 illustrates the process of this experiment.

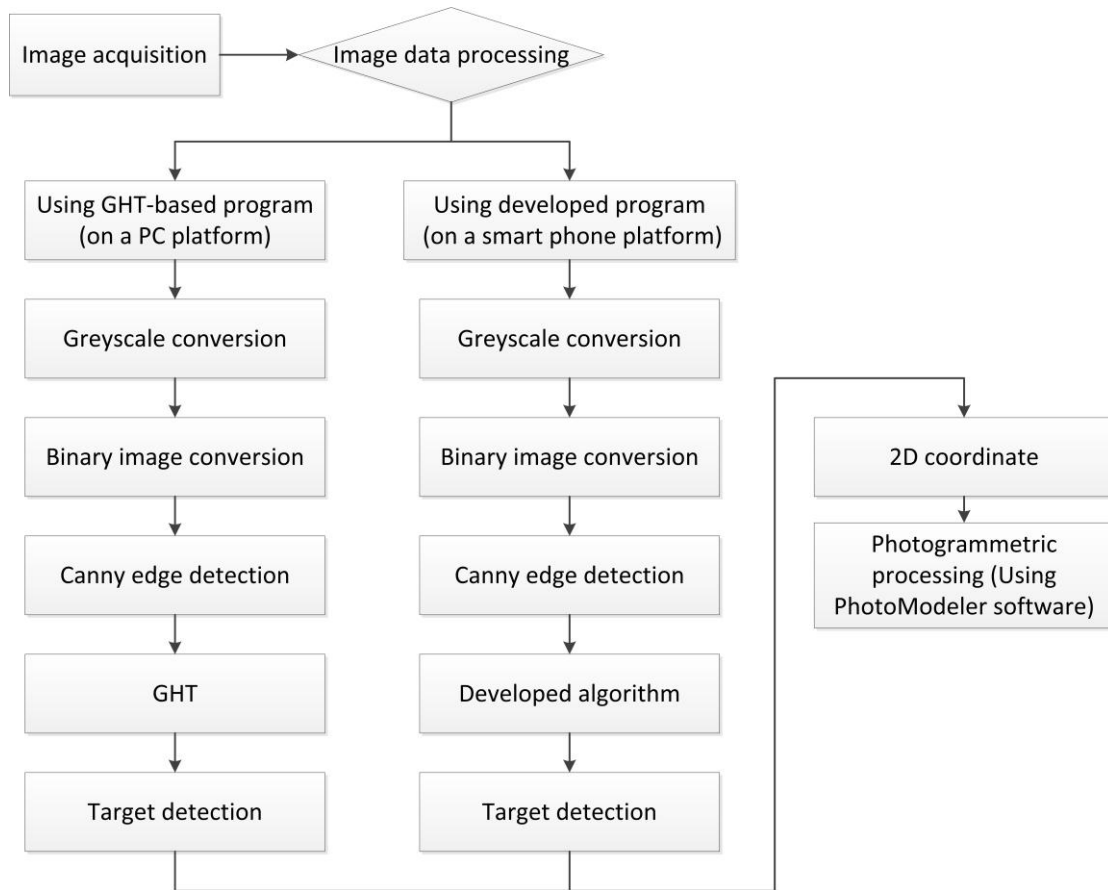


Figure 4-8: Flowchart for comparison of target detection algorithms.

The simulation program converts the colour image to greyscale (as mentioned in § 3.2.7.2). Then the greyscale images will be converted into binary images (the procedure is outlined in § 3.2.7.3) and the edge detection program, which is a design based on the Canny edge detection algorithm, will extract the targets' edge information from the imagery background. The developed mobile platform based target detection program can use the screen touch function to identify the approximate location of the target. The simulation program used GHT to perform edge extraction of targets and further calculated the centre of each target. These 2D coordinates of targets were provided for photogrammetric processing for further comparison.

4.4.2.1 Experimental configuration

Figure 4-9 shows the configuration of the test field; one camera was placed in four different positions and these imaging positions were approximately 1 m away from

the test board. The mobile phone camera was mounted on a mobile phone holder and was fixed on a tripod during the image acquisition process. The camera captured ten images at each position and in total generated 40 images for processing.

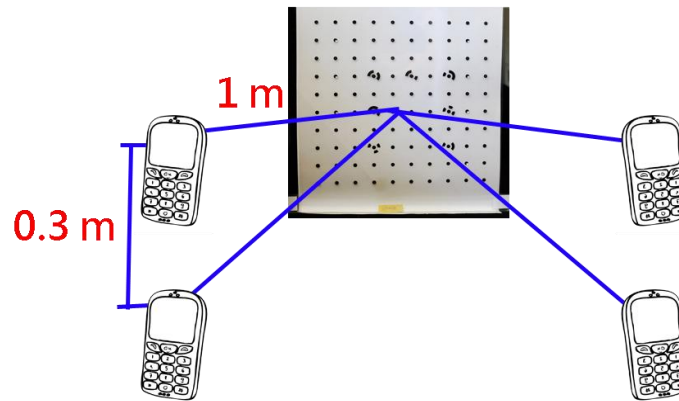


Figure 4-9: Configuration of the test field.

The test boards that were used in this experiment are of glass plates (25 x 25 cm). One glass plate comprises full circle targets (Figure 4-10 (a)) and nine targets on the other glass plate (Figure 4-10 (b)) are replaced by three-quarter circular targets. Unlike the test board that is presented in § 4.2.1 (Figure 4-1), the glass plates for this test are transparent. During the test, these test boards were placed onto a white background to enhance the contrast.

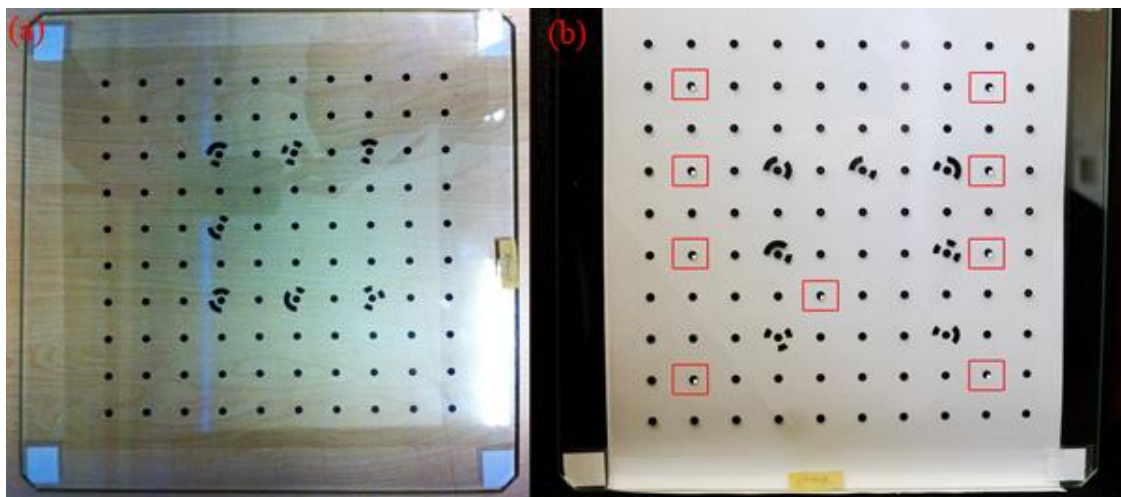


Figure 4-10: Glass plates with two different target patterns.

The target patterns for both test boards were directly printed onto a self-adhesive transparency film in accordance with a fixed size. The three-quarter circular target patterns are modified from those coded targets provided by PhotoModeler. These transparency films were attached onto glass plates for further testing.

The targets' centres were 4.5 mm in diameter (which was about 15 pixels in diameter on the imagery) and the central white dot of each target was 0.1 mm in diameter. The central white dots were provided as measuring target for the P3 analytical plotter.

The distance between two adjustment points was approximately 19 mm (Figure 4-11).

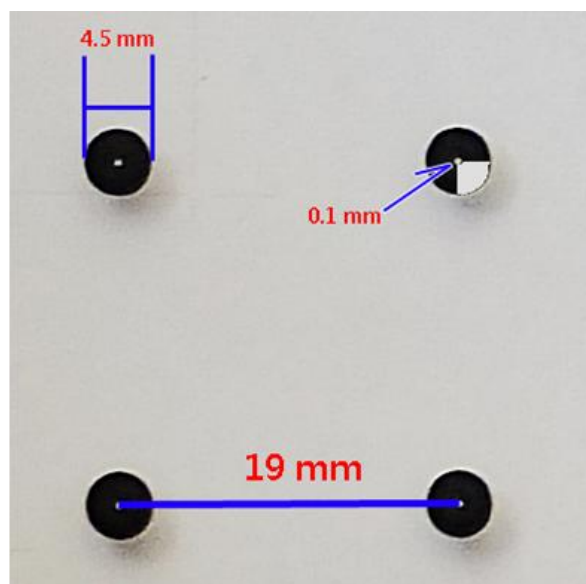


Figure 4-11: Illustration of targets.

In considering the complete system performance, simply drawing conclusions from comparison with other similar measurements is not particularly satisfactory. Therefore, a physical measurement was designed as a contrast benchmark. The two dimensional coordinates of the glass test plate were obtained through manual measurements conducted using a Zeiss P3 analytical plotter. The P3 is a photogrammetric plotting instrument with a measurement system resolution of 1 μm . Although only two dimensional coordinates were observed (with the object measured on a single stage plate of the P3), data measured in this way provided “truth” values of known quality (a precision test, explained below, was performed) for comparison against the

PhotoModeler derived data. Coordinates of all nine targets were measured ten times in the P3 and then averaged in order to improve measurement precision.

In order to avoid the errors that are usually caused during the coordinate transformation process, the comparisons did not compare the differences of coordinates between photogrammetric results and physical measurements. Instead, there are in total 12 distances on both test boards that are selected for comparison (Figure 4-12) and these chosen distances are distributed evenly across the test board.

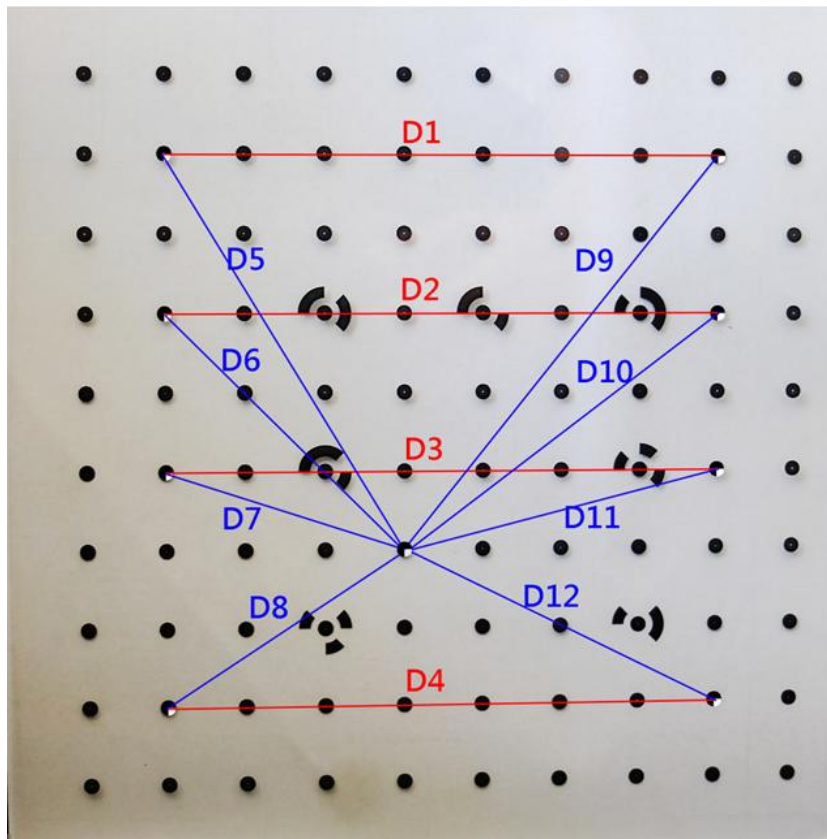


Figure 4-12: The chosen distances for comparison.

4.4.2.2 Test results

Twelve chosen distances were calculated based on the coordinate data. The details of photogrammetric results for both target patterns were summarised in Table 4-2 and Table 4-3.

	Full Circle Target						
	Physical measurement in average (mm)	Photogrammetric results					Relative accuracy for distance measurement
		Max (mm)	Min (mm)	Range (μm)	SD (μm)	RMSE(μm)	
D1	133.974	133.980	133.963	17	6	6	1/22,000
D2	133.941	133.946	133.931	15	5	5	1/27,000
D3	133.893	133.903	133.89	13	6	5	1/27,000
D4	133.946	133.955	133.941	14	4	6	1/22,000
D5	111.246	111.261	111.241	20	9	7	1/16,000
D6	80.623	80.637	80.623	14	5	5	1/16,000
D7	59.975	59.976	59.963	13	4	5	1/12,000
D8	68.784	68.790	68.775	15	6	5	1/14,000
D9	124.525	124.546	124.525	21	11	8	1/16,000
D10	95.546	95.549	95.535	14	7	6	1/16,000
D11	78.866	78.869	78.855	14	5	5	1/16,000
D12	85.125	85.138	85.124	13	5	5	1/17,000
Mean				11	6	6	
Max	133.98		133.963	16	11	8	
Min	59.976		59.963	8	4	5	

Table 4-2: Photogrammetric results of full circle target datasets.

	3/4 Circle Target						
	Physical measurement in average (mm)	Photogrammetric results					Relative accuracy for distance measurement
		Max (mm)	Min (mm)	Range (μm)	SD (μm)	RMSE (μm)	
D1	133.975	133.978	133.970	8	3	4	1/34,000
D2	133.942	133.945	133.939	6	3	3	1/45,000
D3	133.894	133.896	133.890	6	2	2	1/67,000
D4	133.944	133.947	133.939	8	3	4	1/34,000
D5	111.245	111.247	111.237	10	5	5	1/22,000
D6	80.624	80.626	80.620	6	2	4	1/20,000
D7	59.976	59.978	59.970	8	2	3	1/20,000
D8	68.786	68.789	68.782	7	3	4	1/17,000
D9	124.523	124.532	124.519	13	6	6	1/21,000
D10	95.548	95.550	95.541	9	3	4	1/24,000
D11	78.868	78.870	78.864	6	2	3	1/26,000
D12	85.123	85.126	85.120	6	2	4	1/21,000
Mean				11	5	4	
Max	133.978		133.97	9	8	6	
Min	59.978		59.97	3	2	2	

Table 4-3: Photogrammetric results of 3/4 circle target datasets.

In order to further evaluate the differences between the developed image processing algorithm in combination with the dedicated target patterns and the conventional algorithm (full circle targets' results), the standard deviations and root-mean-square errors of these two groups were compared to each other. The comparison of standard deviations was plotted in Figure 4-13 whilst the comparison of root-mean-square errors was shown in Figure 4-14.

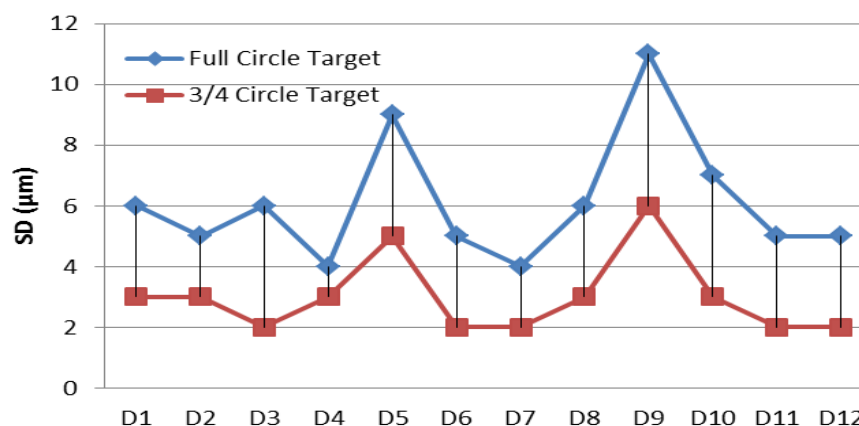


Figure 4-13: Comparison in standard deviations of two different target patterns.

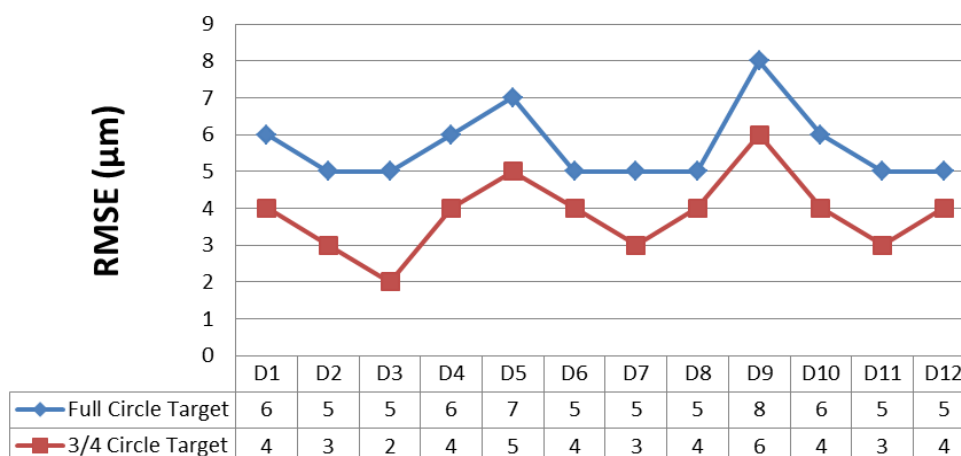


Figure 4-14: Comparison in root-mean-square error of two different algorithms.

When using low resolution images (for example, the images that were captured by a mobile phone camera) to perform photogrammetric processing, the developed image processing algorithm can assist photogrammetric processing software to precisely determine the location of each target. In comparison with the conventional target detection algorithm, the degree of dispersion of photogrammetric outcomes is smaller and the photogrammetric results are closer to physical measurements. The average RMSE of full circle targets is 6 μm. When using 3/4 circle targets in combination with the developed target detection program for testing, the average RMSE was improved from 6 μm to 4μm.

4.5 Summary

This chapter has presented a series of experiments to inspect the critical functions of the mobile platform based structural monitoring system. The sequences of these experiments are listed in Table 4-4.

	Experiments	Main findings
§4-2	Inspection of relationship between the measurement precision and the quantity of imaging stations.	<ul style="list-style-type: none"> The improvement of measurement precision is proportional to the quantity of the camera stations.
§4-3	Inspection of camera synchronisation.	<ul style="list-style-type: none"> During a 62 hours uninterrupted test, the probability of non-synchronised exposures occurring is 0.09% (among 7,439 exposure epochs, there were only seven non-synchronised exposure epochs)
§4-3	Inspection of system recovery mechanism	<ul style="list-style-type: none"> During a 62 hours uninterrupted test, the success rate for system recovery is 100 %.
§4.4	Inspection of optimal target size determination for developed target detection algorithm (in conjunction with 3/4 circle targets)	<ul style="list-style-type: none"> The optimal target size of the developed target detection algorithm is in the range of 11 pixels to 30 pixels.
§4.4	Accuracy assessment of developed target detection algorithm	<ul style="list-style-type: none"> The photogrammetric measurement accuracy is up to 1/67,000 (under laboratory conditions)

Table 4-4: Brief summary of experimental findings.

In terms of the use of this system for long-term structure monitoring of flexible structures, the engineers need to ensure the stability of long-term operation, the synchronisation between imaging stations and the quality of automated operations (which include automatic image acquisition, data transmission and system recovery). The systemic stability was strictly inspected to ensure this developed system can be used for precise long-term structural monitoring applications. Meanwhile, the camera

synchronisation issue was examined to ensure the system is usable for flexible structural monitoring purposes. The inspection of camera synchronisation was introduced in § 4.3.

In terms of the photogrammetric processing functions, the experimental results introduced in § 4.4.1 help operators to determine the optimal target size setting for this mobile platform based image processing module. According to the experimental results, the appropriate range for target size was proposed. In § 4.4.2, the precision of this developed target detection algorithm was examined and found to compare favourably with the GHT approach.

CHAPTER 5. SYSTEM IMPLEMENTATION

5.1. Introduction

The critical functions and integrity of the mobile platform based structural monitoring system were all individually inspected. The experimental configurations and results were presented in the previous chapter. This chapter introduces two experiments which were undertaken to assess the overall operation of this developed system.

5.2. Laboratory experiment – shaking table test

Firstly, the system was inspected under laboratory conditions. A shaking table was chosen as a target object for testing as summarised by Wang et al. (2012). Through the implementation of this experiment, each experimental procedure (the procedures include synchronised image sequence acquisition, image processing, photogrammetric calculation and kinematics behaviour analysis) can be thoroughly examined. Meanwhile, the experiment also tests the capabilities of long-term operation.

5.2.1 *Experimental configuration and workflow*

This experiment lasted for four days; during which time, four smart phone camera stations (the HTC Incredible S; the detail of hardware specification of which was outlined in Table 3-1) were used to simultaneously capture imagery that could be used to examine the performance and remote control capabilities of the system. Meanwhile, four DSLR cameras (Canon 50D with 28 mm f2.8 lens; the detail of hardware

specification was outlined in Table 3-2) were also used to capture imagery in order to provide a benchmark for comparison.

In terms of the experimental configuration, Figure 5-1 shows the configuration of the test site. Eight imaging stations (including four DSLR cameras and four mobile phone cameras) were arranged in the test field to provide convergent coverage of the shaking table. Four digital SLR cameras and four mobile phones were mounted on customized tripod mounts as shown in Figure 5-1 (a).

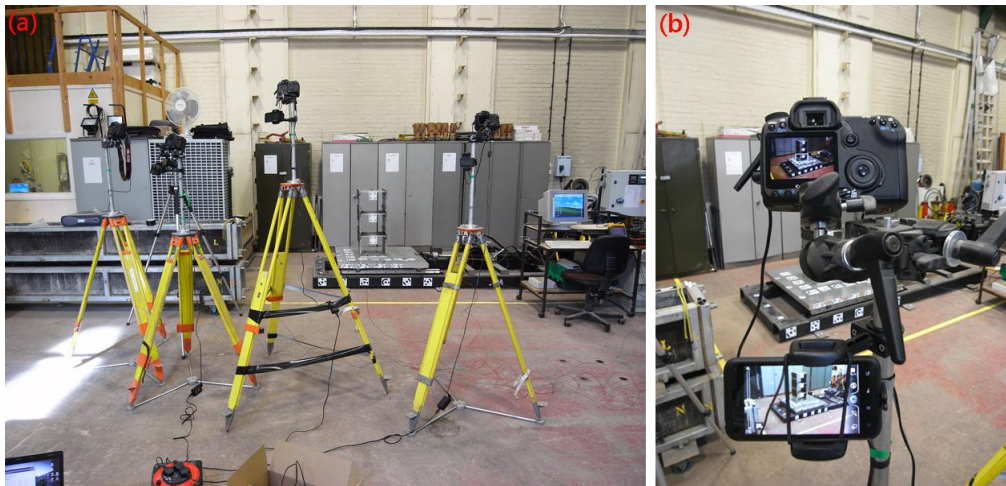


Figure 5-1: (a) Experimental configuration. (b) A close-up photograph of an individual image station.

In order to perform a 4-day experiment, all the digital SLR cameras and the smart phones were connected with 220 AC volt chargers to provide stable power supplies during the experimental duration. Also, the digital SLR cameras were wired into a camera relay panel (Figure 5-2 (c)) which was used to control the shutters for four DSLR cameras. A switch box was used to establish the communication between the central computer and the camera control panel (Figure 5-2 (b)). The switch box was also connected to the shaking table control panel. Through the use of this configuration, the operator could use the central computer to simultaneously control the exposures of all DSLR cameras and the movement of the shaking table.

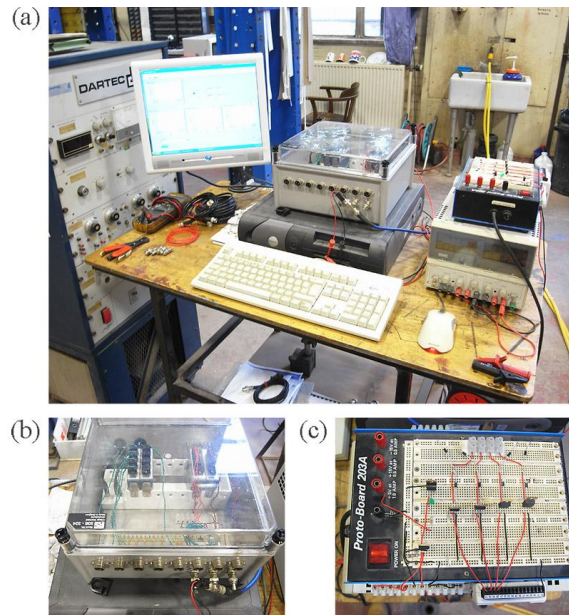


Figure 5-2: (a) Central computer and control component. (b) and (c) show the switch box and the camera relay panel.

The DSLR cameras were simultaneously triggered by a regular pulse every 10 seconds that was generated by a commercial computer programme called DSLR Remote Pro Multi-Camera Version 1.6 produced by Breeze Systems Limited. Through the use of this software, a PC can synchronously control up to 25 cameras. The time lag of exposures from each DSLR camera can be recorded to the PC, the average time lag between the individual DSLR exposures being less than 10 microseconds. Four mobile phone cameras were remotely controlled by a laptop via the multiple-station program. The laptop was also connected to the central computer by means of the switch box. Through the switch box, the PC was synchronised with the laptop and the information (shutter speed, exposure epoch and the absolute time of image capture) from the four smart phone cameras can be sent back to the PC from the laptop where it can be combined with the data from the shaking table and the DSLR cameras to carry out a comprehensive comparison.

Two sample objects were applied as the target modules for the experiment. A rigid box (1.00 m (L) x 0.75 m (H) x 0.60 m (W)) shown in Figure 5-3 (a) was used to provide a benchmark of precision and to give the operator basic information about the performance of the developed system for moving object monitoring under specific

vibration frequencies. The wooden boards were firmly bolted to the metal framework allowing the attachment of photogrammetric targets onto the surfaces.

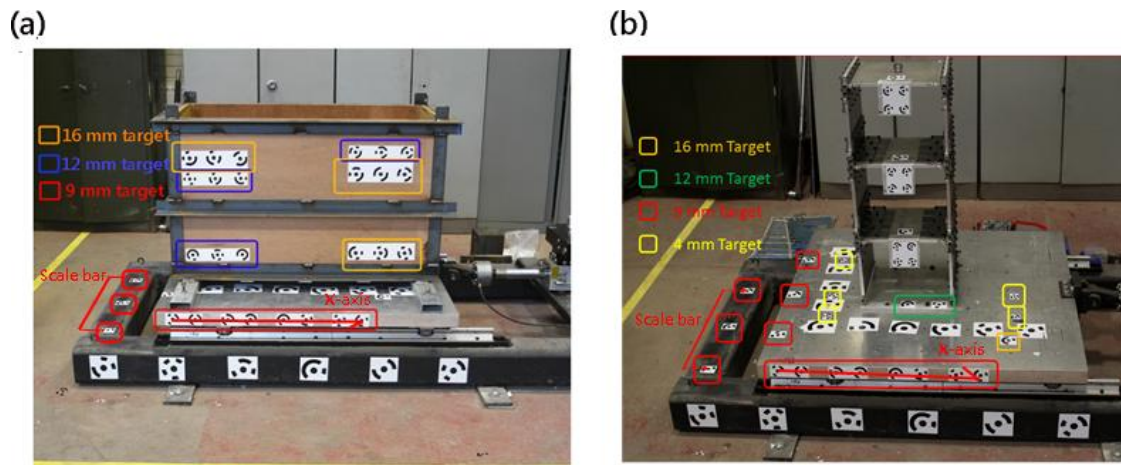


Figure 5-3: Illustration of test objects for shaking table test.

Figure 5-3 (b) shows the test frame which was used to represent a flexible structure. The flexible specimen for the test is a three-story stainless steel moment frame structure (0.30 m x 0.30 m x 0.70 m) which was mounted onto the shaking table to simulate a general flexible structure; for example, a high-rise structure such as a tower block.

This shaking table (1.40 m x 1.40 m x 0.10 m) was driven by the Instron® 3520 series hydraulic power units. This hydraulic pump generates constant power output up to a maximum of 207 bar (approximately equal to 3000 psi). In order to understand whether the change in velocity of an object affects the performance of the mobile phone-based monitoring system or not, two vibration frequencies were applied in this test to generate different velocities of the shaking table platform. The power output of this hydraulic pump was set to 110 bar and 185 bar to maintain the shaking table for a fixed vibration frequency of 0.5 Hertz and 1 Hertz, respectively. Meanwhile, the amplitude was fixed at 0.7 m. During the 4-day experimental period, the shaking table tests were performed on the first and the fourth day. The shaking table and the DSLR camera stations were shut down on the second and the third day, but the mobile phone stations were still operational in order to test the performance of this developed system for long-term operation. Also, the exposure frequency of each mobile phone

was adjusted from thirty seconds per frame to ninety seconds per frame for the second and third days.

The coordinates of the targets which were attached onto the shaking table platform and the rigid box were also measured in three dimensions by a Leica total station (TCRP 1201). These physical measurements were provided as references to compare with the photogrammetric results.

When performing the shaking table test, the frequency of exposure of the DSLR cameras was set to 10 seconds. The state of each camera was monitored by a computer. After every exposure, an electronic pulse was sent back to the desktop computer from each DSLR camera. The computer checked this information automatically to ensure all cameras were firing simultaneously. In the meantime, the frequency of exposure for the mobile phones was three times slower (30 seconds per frame) due to their slower memory buffer. During the test, the multiple DSLR camera stations were launched first, and the time of exposures were sent back to the PC. Next, this information was transferred to the laptop via a switch box. In accordance with the exposure epochs of the DSLR cameras, the self-programmed software synchronised the mobile phone stations via WLAN, sending a script which included the frequency of exposure and the start time for the first exposure (corresponding to the DSLR camera stations). Following this, the mobile phone cameras fired at 30 second intervals until the operator pressed the stop button.

The laptop needs to connect to each smart phone station via wireless network services. Under outdoor operating conditions, the network services are not always available in every test area. This experiment did not use the campus network services in order to simulate the test field where it is not covered by a public network. Instead, a wireless router was connected to the laptop to create a WLAN environment. The configuration of the instruments is illustrated in Figure 5-4.

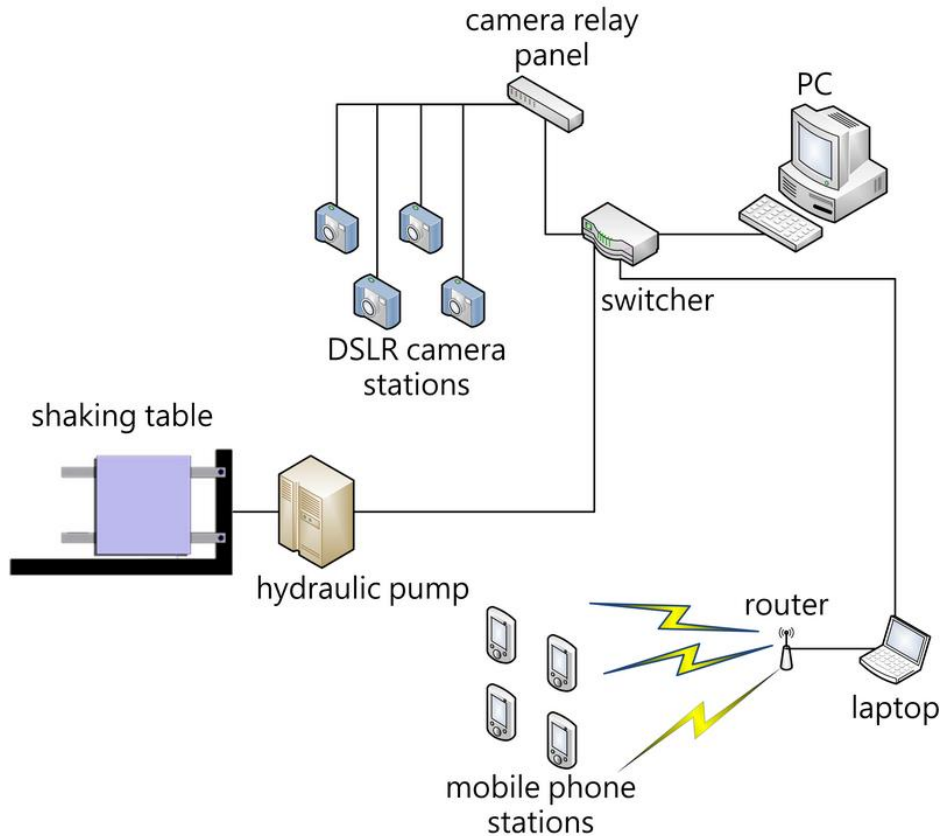


Figure 5-4: Configuration of the shaking table experiment.

Eight imaging stations (comprising four DSLR cameras and four smart phone cameras) were arranged in two different locations to provide convergent coverage of the shaking table, and to evaluate the effect of different network configurations on photogrammetric precision. The network geometry configuration is illustrated in Figure 5-5.

On the first day, the camera stations were arranged relatively close to the shaking table, with H at approximately 2.5 m; whilst the value of B was 1.8 m (b was 0.6 m). This configuration was unaltered for day 2 and day 3. On the fourth day, the values of both H and B were increased to 3.0 m in both cases, with camera convergence altered accordingly.

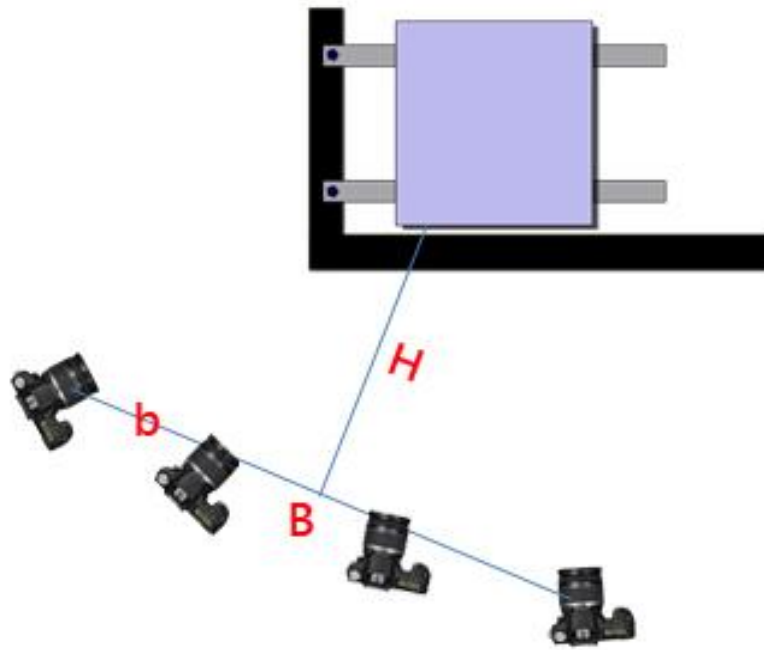


Figure 5-5: Shaking table test. B is the longest distance between two cameras; b is the baseline between any two adjacent cameras and H is the object distance, as measured from the shaking table to the midpoint of B .

The rigid box was comprehensively monitored on the first day. In the first three hours of the first day, the rigid box was fixed onto the shaking table platform for testing. The vibration frequency was 0.5 Hertz for the first hour and for the next two hours it was increased to 1 Hertz. Then, the rigid box was removed and was replaced by the three-story stainless steel moment frame for a four-hour test. The vibration frequency was set to 0.5 Hertz for the first two hours and was adjusted to 1 Hertz frequency for the last two hours of the flexible structural monitoring test.

On the fourth day, the object distance was extended from 2.5 m to 3.0 m. The tests in the fourth day simplified the steps of image sampling. Because the 0.5 Hertz vibration frequency is not rapid enough to cause visible deformation to the flexible structure, both the rigid box and the three-story stainless steel moment frame were put on the shaking table for a 2-hour dynamic test only using 1 Hertz vibration frequency. The details of the experimental configurations are summarised in Table 5-1.

	Day_1				Day_2	Day_3	Day_4	
Photogrammetric network configuration (for both DSLR and smart phone cameras)	H=2.5 m B=1.8 m				H=2.5 m B=1.8 m	H=2.5 m B=1.8 m	H=3.0 m B=3.0 m	
Test model	Rigid box		Flexible structure		Flexible structure	Flexible structure	Rigid box	Flexible structure
Exposure frequency	Mobile phone: 30 sec DSLR: 10 sec				Mobile phone: 90 sec DSLR: shut down		Mobile phone: 30 sec DSLR: 10 sec	
Duration (hours)	2	2	2	2	24	24	2	2
Vibration frequency of shaking table (Hz)	0.5	1	0.5	1	/		1	1

Table 5-1: Experimental configurations for the shaking table test.

5.2.2 Photogrammetric targets

Different sized targets were applied to estimate the effect on the accuracy. There are three different sizes of targets attached onto the rigid box; the largest target size (the diameter of the central dot) is 16 mm whilst the other two sizes are 12 mm, and 9 mm, respectively. The flexible structural test applies one more size of target (Diameter = 4 mm) in addition to the above three dimensions. By the use of equation (3.1) and equation (3.2), the diameter of these targets can be converted into pixels. And these converted results can be used to evaluate whether these target sizes are appropriate for the experimental configuration or not. Because these camera stations were lined up in front of the shaking table, the object distances of imaging stations were slightly different even under the same group of experimental configuration. Therefore, the calculated target sizes in pixels were not identical. The converted results are shown in Table 5-2. As shown in the table, the use of 4 mm targets for smart phone cameras are smaller than the restrictions of PhotoModeler and the use of 12 mm and 16 mm targets are substantially larger than the restrictions of minimum target sized range. These too small and oversized targets were used to evaluate the influence of inappropriate sized targets for photogrammetric measurement precision. Meanwhile,

the target size in pixels changed with the differences of object distances, and the combination of different sized targets (physical size) and different object distances may result in the same number of pixels in diameter. In order to prevent any potential confusion, the remainder of this experiment will use physical size (in mm) to represent each target.

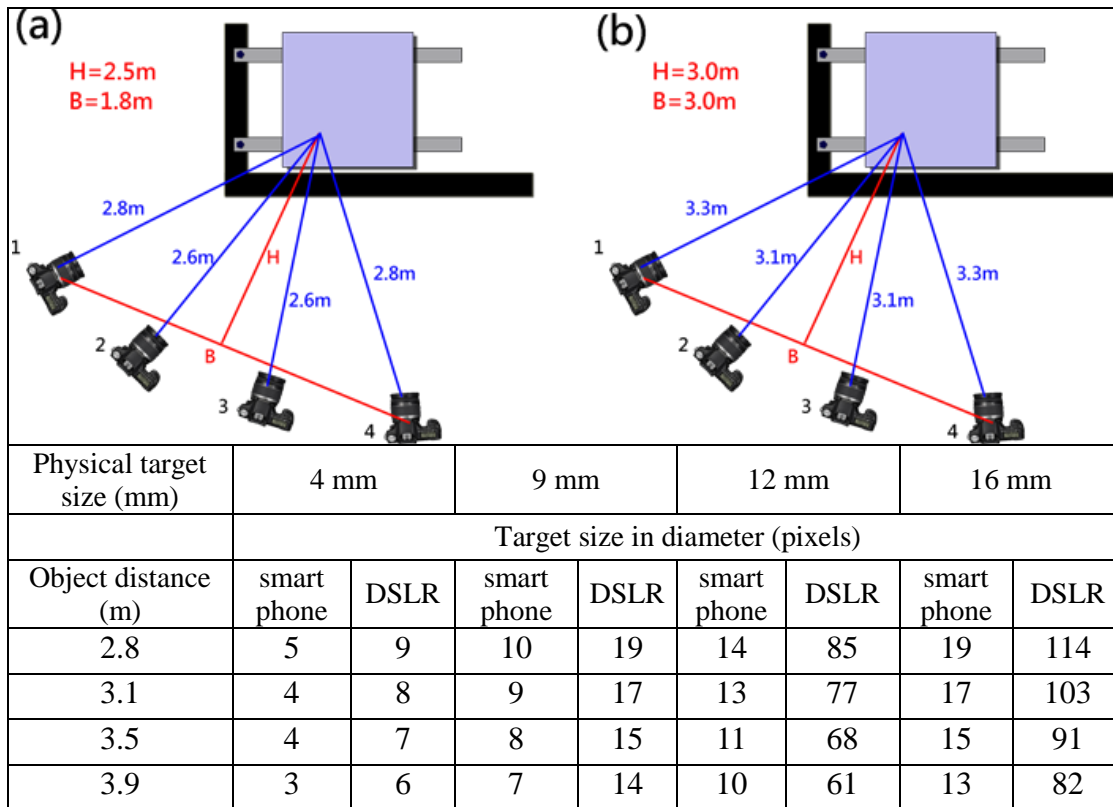


Table 5-2: The total number of pixels in diameter of each target under different experimental configurations.

These targets were evenly distributed across the shaking table surface and these two test frames with the configuration of coded targets as shown in Figure 5-3 (a) and (b). The targets, which were attached onto the three-story stainless steel moment frame, are 9 mm in diameter. Because the stainless steel moment frame did not apply any physical measurement to measure the structural movement, the deformation of the frame can only rely on the calculation from the photogrammetric measurements. In order to estimate the precision, the shaking table was arranged with four different sized targets and the results could be compared with the rigid box measurements. Also,

these different sized targets were arranged in different planes to avoid errors that may have been caused by the depth of field and parallax. In the rigid box test, the distribution of targets was concentrated in the perpendicular plane (perpendicular to the ground). However, most of the targets were attached onto the shaking table platform (parallel to the ground) when the flexible structure test was performed.

The photogrammetric targets were made of two layers of self-adhesive vinyl sheets (Figure 5-6). The black target pattern is attached onto a white vinyl sheet to enhance the contrast. These used a series of coded targets (Figure 5-6 (a)) provided by commercial photogrammetric software, PhotoModeler. As shown in Figure 5-6, two different target patterns were produced for two experiments that were presented in this chapter. The full circle targets (Figure 5-6 (a)) were adopted in this shaking table test whilst the 3/4 circle targets were used in the outdoor experiment that will be introduced in the following sections.

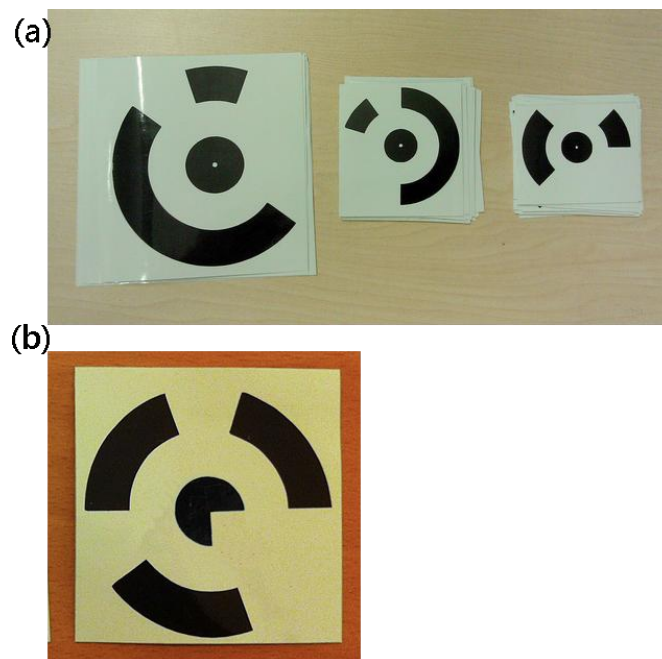


Figure 5-6: Customised target stickers with different target sizes. (a) shows the original PhotoModeler target patterns, and (b) is the customised 3/4 circle target.

5.2.3 *Physical measurement*

For the shaking table test, the x-axis was defined at the shaking table platform using 9 mm targets (for smart phone cameras: the diameter of target centre is about 7 pixels to 10 pixels; for DSLR cameras, the diameter of target centre is 14 pixels to 19 pixels) as the endpoints (Figure 5-3). A scale bar was set onto an outer frame of the shaking table platform (the targets attached onto the outer frame were also 9 mm in diameter). The object coordinates of this scale bar were measured using a Leica total station (TCRP 1201 which has an angular measurement precision of 1'') by the space intersection method. These coordinates of targets were calculated using the STAR*NET least squares software package. The average standard deviations of these measured targets were ± 0.03 mm, ± 0.05 mm and ± 0.03 mm in the X, Y and Z axes, respectively.

5.2.4 *Simulation of structural damage detection*

Finally, a simulation test was undertaken to simulate the use of this multiple-station module for structural health monitoring applications. The movements of the shaking table were controlled by consistent force with fixed amplitude and frequency. The consistent force was generated by a hydraulic pump and it is used to simulate the external forces on buildings. In its normal state, if the external force is less than the permissible amount that a building can withstand, the deformation of the structure is constant within the allowable range (Alfakih, 2007). After a long period of data collection, the engineers can establish the normal strain curve of the structure. If some parts of the structure are damaged, the extent of structural damage will be reflected in the increase of deformation (Fan et al., 2011). As a result, the engineers can monitor the structures health by observing the daily changes (Jiang, 2011).

The previous tests have examined the deformations of the three-story stainless steel moment frame under different conditions (in the absence of external forces or forced by a constant external force). The final stage of the shaking table experiment still applies the same amplitude and frequency, but some mounting bolts of the three-story stainless steel moment frame have been unscrewed to simulate damage to some parts of the structure that results in the loss of the internal force to restrain the deformation of the object (Figure 5-7). The imaging stations captured the image sequences using

the same exposure frequency. The photogrammetric results are then processed to analyse the deformations.



Figure 5-7: Unscrewing some mounting bolts of the three-story stainless steel frame.

5.2.5 Test results

As mentioned above, there are four different sized targets attached onto the shaking table platform for the flexible structure test and three different sizes of targets were applied for the rigid box test. During the four-day experimental period, each DSLR camera station made 4,320 exposures and all these images were captured while the shaking table was moving. Among these 4,320 sets of image data, 4 mm targets appeared in only 2,160 sets of them (because the rigid box test did not use any 4 mm target). In addition, every mobile phone station generated 3,792 images and only 1,890 images were captured in the dynamic test (i.e., in total 1,902 images for each mobile phone camera were captured during the middle two days when the shaking table was not in operation). The image data from each imaging station was sent for photogrammetric processing to calculate the 3D coordinate data of each target. In order to analyse the degree of dispersion between individual targets within the same group (i.e., the coordinate data of the same target), the standard deviation (SD) of this coordinate data was calculated. Table 5-3 and Table 5-4 present the final results.

5.2.5.1 Evaluation of photogrammetric precision

Table 5-3 shows the standard deviations of the X, Y and Z coordinates from the different configurations. In terms of the DSLR cameras, the best results are achieved through the use of the 4 mm target with the 2.5 m object distances (under this configuration, the diameter of targets in pixel were approximately 8 pixels to 9 pixels) and this sized targets accords with the basic requirement of target size of PhotoModeler. The similar target size in pixels of smart phone camera stations was the combination of 2.5 m object distance combined with 9 mm targets (the diameter of targets in pixel were approximately 9 pixels to 10 pixels). Under this configuration, the standard deviations of the X, Y and Z coordinates have the optimal performance than other configurations (specifically refers to the comparison between smart phone stations). The optimal range of photogrammetric target size of 3/4 circle targets has been outlined in § 4.4.1. According to experimental results of this shaking table test, the optimal target sized range of PhotoModeler (combined with the use of full circle targets) is approximately 8 pixels to 10 pixels. Meanwhile, the results of smart phone cameras show a similar trend; both of these two types of imaging stations have larger standard deviations in the direction of the Z-axis.

DSLR camera	B=1.8 m, H=2.5 m			B=3.0 m, H=3.0 m		
	σ_x (μm)	σ_y (μm)	σ_z (μm)	σ_x (μm)	σ_y (μm)	σ_z (μm)
16 mm	27	30	44	18	20	31
12mm	15	14	25	14	13	23
9mm	13	12	23	10	11	21
4 mm	7	7	14	9	9	16
smart phone	B=1.8 m, H=2.5 m			B=3.0 m, H=3.0 m		
16 mm	35	37	50	36	38	51
12mm	23	26	38	24	25	36
9mm	15	16	24	16	17	26
4 mm	21	23	31	23	25	33

Table 5-3: Comparison in standard deviations under the different configurations.

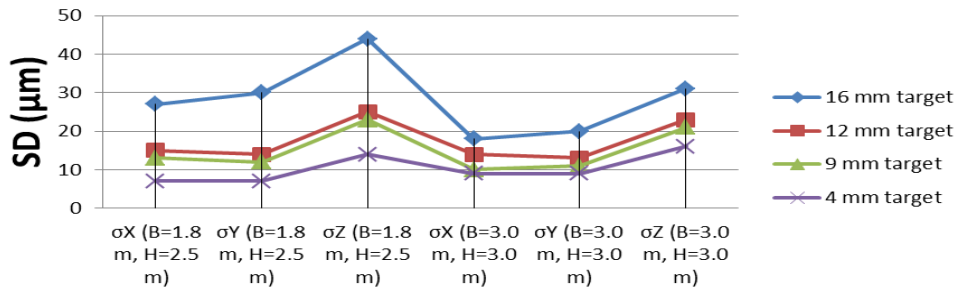


Figure 5-8: Coordinate standard deviations under the different configurations (DSLR).

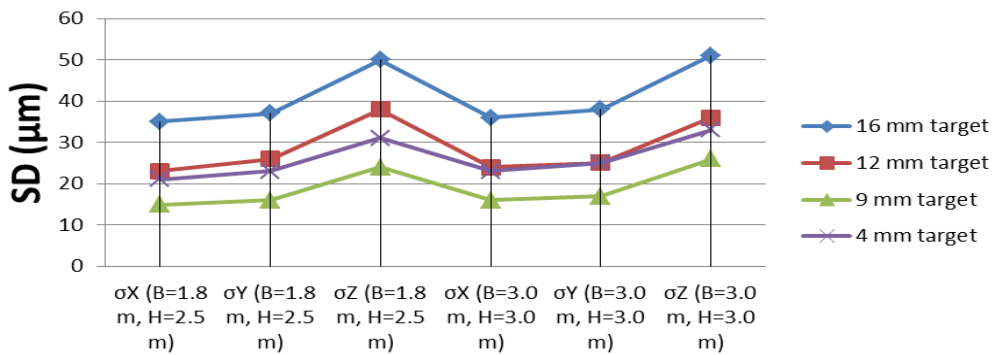


Figure 5-9: Coordinate standard deviations under the different configurations (smart phone).

The rigid box was applied as a measured object in both the shaking table test (dynamic test) and the performance evaluation test for long-term operation (static test). Therefore, the photogrammetric results from the rigid box are used to inspect the difference in performance of the mobile phone-based photogrammetric system between the dynamic object monitoring and the static structural monitoring. In this project, the evaluation only involved the photogrammetric results from the mobile phone stations, because the DSLR cameras were not used to capture the images for the still rigid box. Table 5-4 shows the results of the difference. Unlike Table 5-3 that used all image data for calculation, in this stage; the calculation only involved image data that was using the rigid box as the target object. The process used a total of 9,600 images (2,400 images from each mobile phone station). Among these images, 1,920 of 9,600 images were captured in the shaking table test (the fourth day's image data was not included in the calculation due to the difference of object distance setting)

and the other 7,680 images were captured in the middle of two days when the shaking table was not in operation. Table 5-4 outlines the comparison of these two different motion statuses.

Figure 5-8 and Figure 5-9 have indicated that the use of 9 mm targets (the target diameter in pixels is 7 pixels to 10 pixels) can generate better results than any other target sizes in terms of the mobile platform based system. Therefore, the discussion in this stage is focussed on the results of 9 mm targets. Figure 5-10 illustrates the difference of the use of 9 mm target in two different motion states. The comparison chart shows almost identical results in the X-axis and Y-axis, there being only 1 micron and 2 microns difference in X-axis and Y-axis, respectively. The results in the Z-axis show larger differences than the other axes. The SD of moving object monitoring is 5 microns larger than the stationary state in Z axis.

B=1.8 m, H=2.5 m (Stationary state)			
Rigid box	σ_X	σ_Y	σ_Z
	(μm)	(μm)	(μm)
16 mm	33	35	48
12mm	22	25	36
9mm	14	15	22
B=1.8 m, H=2.5 m (Dynamic state)			
Rigid box	σ_X	σ_Y	σ_Z
	(μm)	(μm)	(μm)
16 mm	38	39	52
12mm	24	27	39
9mm	15	17	27

Table 5-4: Coordinate standard deviations under different motion states.

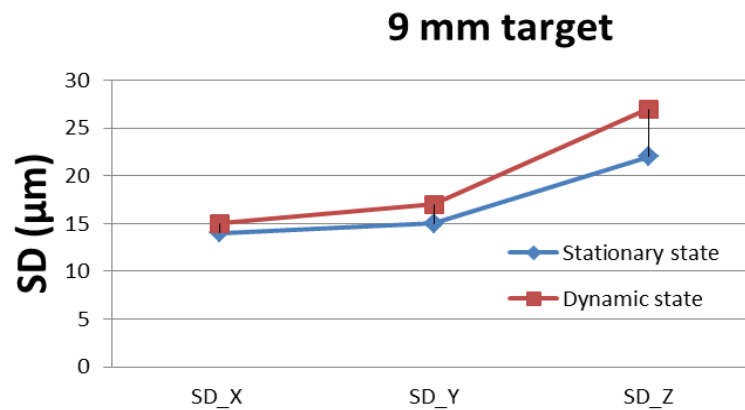


Figure 5-10: Standard deviations of 9 mm targets based on different motion states.

5.2.5.2 Assessment of photogrammetric accuracy in distance measurements

In addition to the inspection of standard deviations, the photogrammetric results are compared with the physical measurements in order to estimate the relative accuracy of this mobile phone-based monitoring system. In terms of the target configuration for flexible structural monitoring testing, thirty-eight targets that were attached onto solid parts of the shaking table (twenty-nine targets on the platform surface and nine targets on the external framework) have been measured by a Leica total station (TCRP 1201). Ten point-to-point distances (Figure 5-11) were then calculated based on the coordinate data. The endpoints of these measurements included different target sizes in order to estimate their relative precisions. The two endpoints of the scaling distance shown in Figure 5-11 used 9 mm coded targets. The yellow measurements used 4 mm targets as endpoints, whilst the green measurements used 16 mm targets. The blue lines represented measurements with 9 mm target endpoints and the single pink line in the middle of the platform used 12 mm coded targets as endpoints. The validation of the photogrammetric measurements was made by comparing these physical measurements with the photogrammetric results. Deviations from the physical measurements were calculated for statistical analysis.

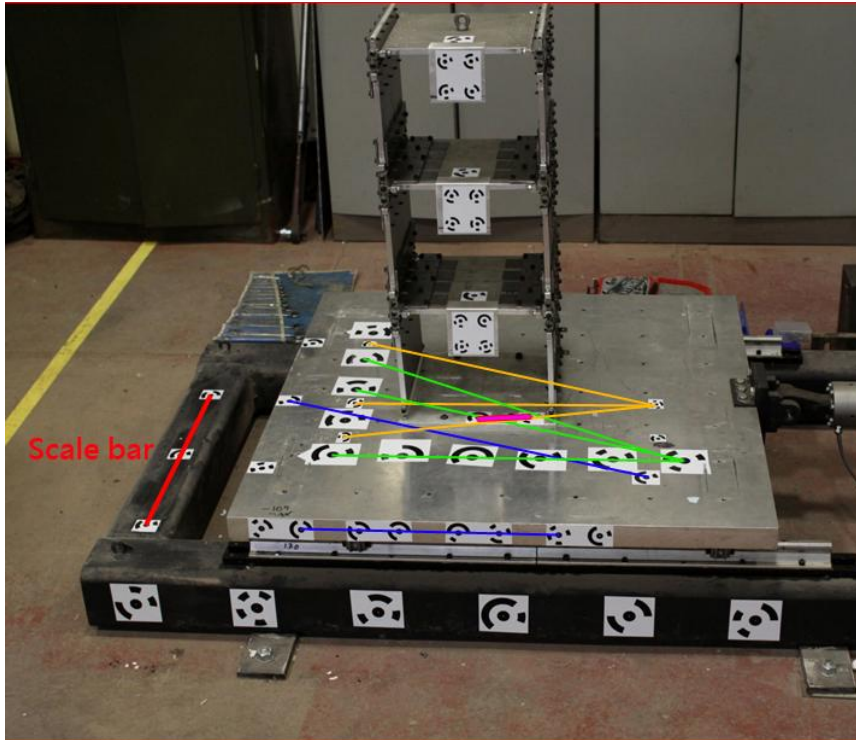


Figure 5-11: Selected measurements for validation comparison.

Figure 5-12 presents results generated from the mobile phone-based camera station data. The result shows the same trend with previous comparisons. The use of 9 mm targets generates better photogrammetric results in both camera network configurations.

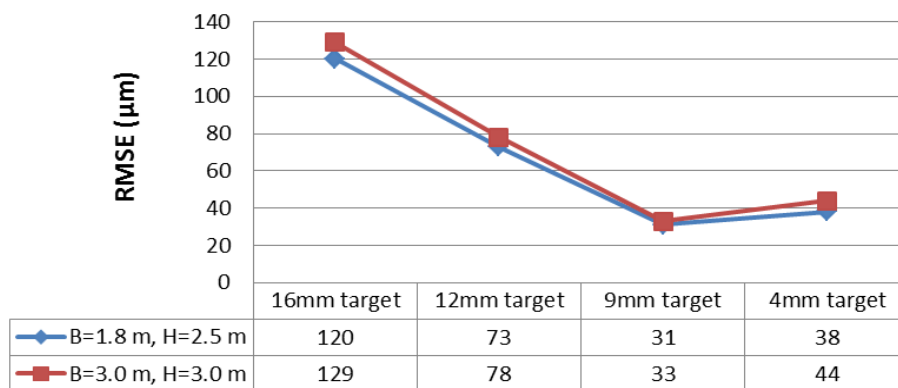


Figure 5-12: RMSE of photogrammetric measurements (smart phone camera stations).

The differences between the photogrammetric results and the physical measurements are getting close when the use of target size is reduced from 16 mm to 9 mm. However, the use of 4 mm targets does not keep improving the photogrammetric results. As shown in Figure 5-11, there are two chosen distances that are using 9 mm targets as endpoints and the longest distance among these two is 867.83 mm and the minimum RMSE is 31 μ m. Therefore, the relative accuracy can be assessed. The relative accuracy of distance measurement is approximately 1/28,000.

5.2.5.3 Demonstration of structural damage detection

The following section presents the results of the structural damage simulation. As shown in Figure 5-13 (a), an aluminium plate, which is perpendicular to the ground, was welded on each layer of the three-story stainless steel moment frame. These aluminium plates are used to represent the corresponding layer that they were attached to. Each aluminium plate has four 9 mm targets on the surface and the configurations of these coded targets are illustrated in Figure 5-13 (c). The structural deformations can be calculated by analysing the displacements of each target. By overlapping the control points which were attached onto the shaking table platform, the structural deformations in different periods can be compared (Figure 5-13 (b) and Figure 5-13 (d)).

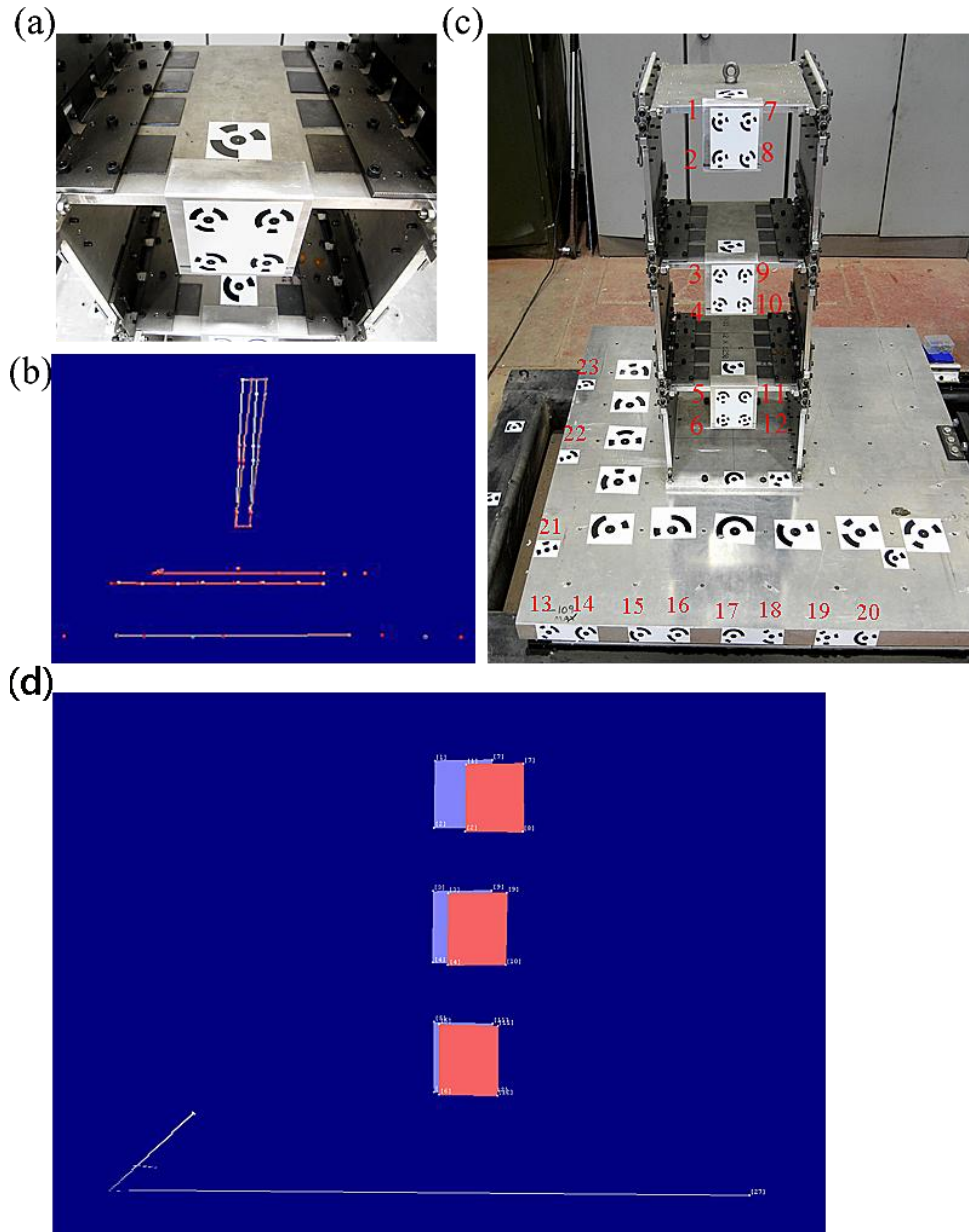


Figure 5-13: Configuration of coded targets for the three-story stainless steel frame.

Table 5-5 shows the comparisons of control points. There are 11 control points attached onto the shaking table platform (Figure 5-13 (c)). The SD of each point was calculated to evaluate the precision of this measurement. In terms of the SD in x-axis, the targets in front of the shaking table (Point_13 to Point_20) have better SD results, the minimum of SD are located on Point_16 and Point_17 (13 microns), and the maximum SD in x-axis in Point_23 (18 microns). The difference between the minimum of SD and maximum SD is 5 microns. The results of SD in the y-axis are

more consistent than the results in the x-axis, the difference between the minimum of SD and maximum SD is 3 microns. The results of SD in the z-axis are still higher than the other axes, the maximum SD is located in Point_23 again and the minimum of SD is still located in the front side of the shaking table (Point_15).

Point ID	SD_X (μm)	SD_Y (μm)	SD_Z (μm)
	(B=3.0 m, H=3.0 m)	(B=3.0 m, H=3.0 m)	(B=3.0 m, H=3.0 m)
13	14	20	31
14	14	21	32
15	15	21	30
16	13	20	33
17	13	22	34
18	15	21	33
19	14	22	36
20	16	20	35
21	14	19	34
22	15	20	36
23	18	21	38
MEAN	15	21	34
MAX	18	22	38
MIN	13	19	30

Table 5-5: Comparisons in standard deviation of control points.

By checking the 240 sets of image data (the experiment lasted for 2 hours and the imaging frequency of each camera was 30 seconds per frame), there were 31 sets of image data captured at the left endpoint and 28sets of image data captured at the right endpoint. These 59 sets of image data were calculated via photogrammetric processing to generate the 3D coordinates and were compared with the previous experimental results. The comparisons are made my comparing the movements form the right endpoint to the left endpoint (by calculating the point to point distances and the results include 0.7 m of amplitude).

	Original (B=3.0 m, H=3.0 m)	Unscrewed (B=3.0 m, H=3.0 m)
Point ID	Displacement (mm)	Displacement (mm)
1	802.482	883.684
2	802.471	883.671
3	776.558	827.569
4	776.534	827.543
5	757.445	785.477
6	757.424	785.491
7	802.493	883.662
8	802.480	883.673
9	776.591	827.556
10	776.540	827.534
11	757.461	785.467
12	757.431	785.456

Table 5-6: Comparisons of displacements under two different experimental configurations.

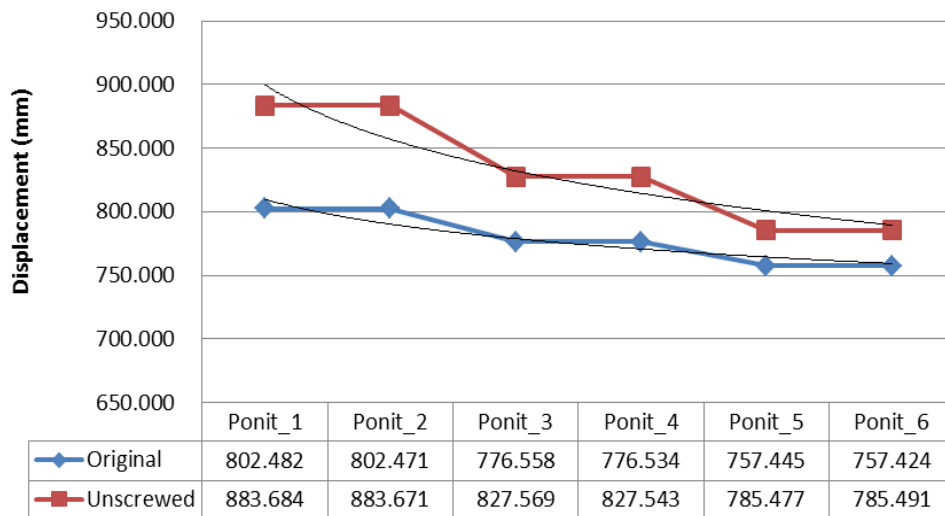


Figure 5-14: Comparisons of displacements under two motion status (from Point_1 to Point_6 on X_axis).

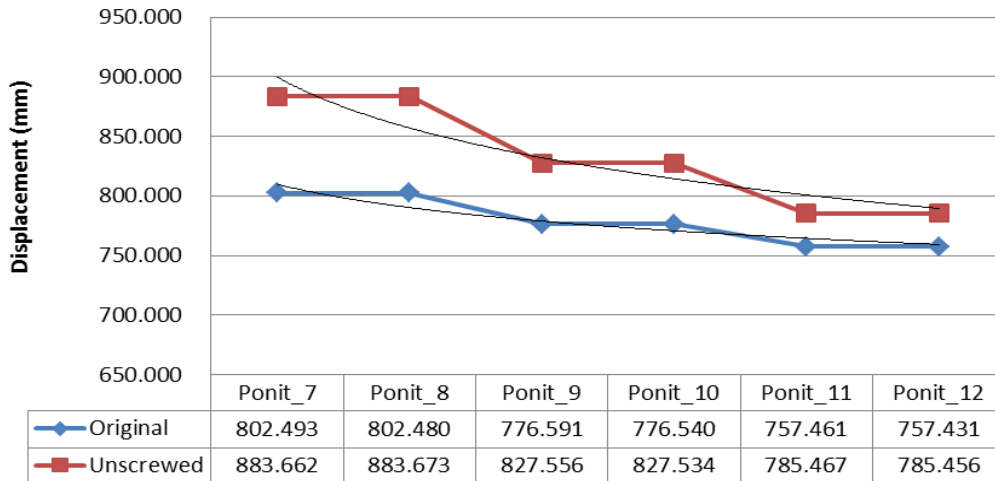


Figure 5-15: Comparisons of displacements under two motion status (from Point_7 to Point_12 on X_axis).

The comparisons of displacements between two different states were listed in Table 5-6 and the trends of targets were plotted in Figure 5-14 and Figure 5-15, respectively. Both of these two states were tested in the same experimental configurations (which mean that the object distance and the distance between cameras are identical, and the amplitude and the vibration frequency are the same). The only difference is the state of the target structure. The control points have been checked and the residuals are limited in a reasonable range (in comparison with the previous tests).

The targets on the aluminium plates were divided into left and right sides. The left side includes the targets from Point_1 to Point_6 and the right hand side is the targets from Point_7 to Point_12. As shown in Figure 5-14 and Figure 5-15, the differences of deformations of these two groups are illustrated. The trend lines denote the difference between the original states and the damaged state. For a long-term monitoring project, the daily (or regular) observations (like the “Original” group shows in Table 5-6) can be used to generate a trend line to describe the structure behaviour in the normal state. The engineer can use this data as a standard to detect the abnormal structural behaviour. If data are collected over a long period, the deformation curve will be much closer to the real behaviour of target structures.

5.3. Outdoor implementation

The individual functions of the mobile phone-based structural monitoring system have been strictly inspected. The performance and precision of the system has been examined in a laboratory environment. The implementation of the “real-world” experiment aims to test the mobile phone-based structural monitoring system under actual operating conditions in the field.

5.3.1 *Experimental configuration*

Site investigation was carried out to identify a suitable “real world” subject to monitor. As shown in Figure 5-16, the target structure selected for monitoring was a 4.60 m (H) x 3.10 m (W) rolling shutter. Under the effect of external force, deformation and displacement of the rolling shutter will take place. Although this outdoor test site was covered by a Newcastle University campus wireless network, the test did not use this network service to perform the wireless system connection. Instead, the test used a wireless router to create a private network for system communication purposes in order to simulate the implementation of this system in a test area without network service.

In this test, four mobile phones were mounted on tripods and were setup 6.5 m away from the rolling shutter to converge on the rolling shutter (two phones on each tripod at different heights; as shown in Figure 5-16). The convergent angle is close to 90 degrees to provide a strong photogrammetric network configuration (Ethrog, 1987). These mobile phones were wirelessly synchronised to a laptop (i.e., the terminal computer). The distance between two tripods was approximately 9.2 m.

A total of 21 photogrammetric targets were attached evenly across the lower half of the rolling shutter (the yellow area in Figure 5-16). Six targets were attached on brick walls on both sides of the door (There are three targets on each side) to provide stationary references for comparison. The coordinates of these stationary targets, which were attached onto brick walls, have been measured in three dimensions by a Leica total station (TCRP 1201) using spatial intersection. These coordinates of targets were calculated using the STAR*NET least squares adjustment software. The

average standard deviations of targets were ± 0.08 mm, ± 0.11 mm and ± 0.09 mm in the X, Y and Z directions, respectively.

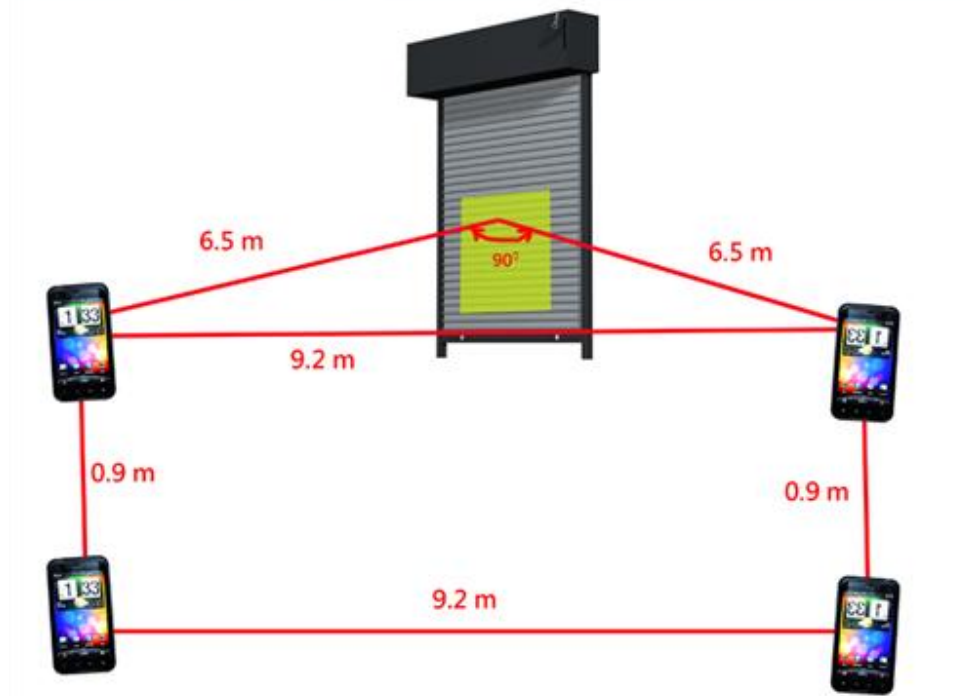
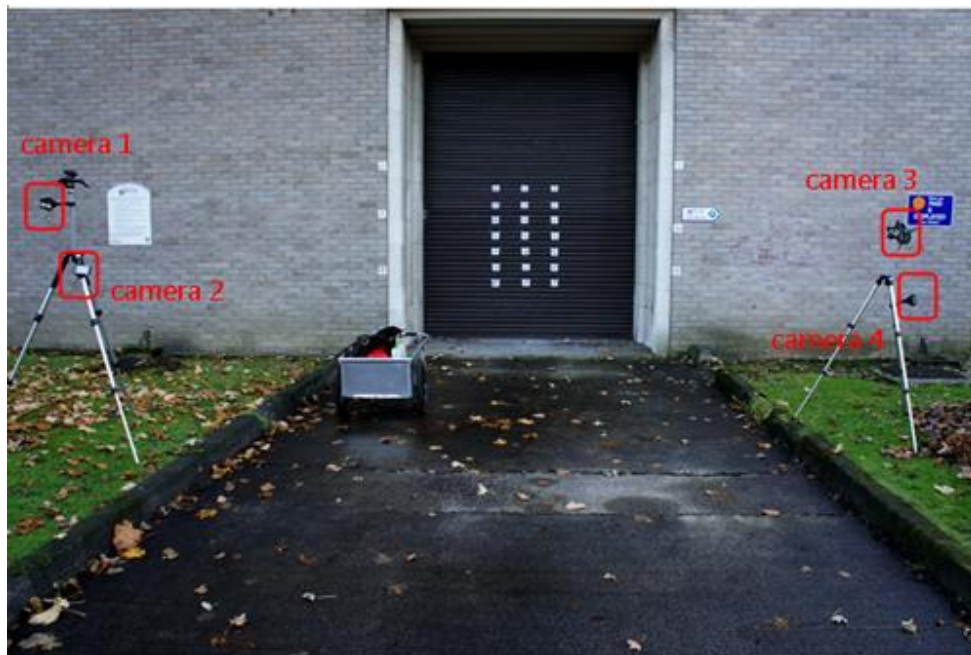


Figure 5-16: Schematic illustration of test site.

5.3.2 Photogrammetric targets

The chosen target size was 26 mm in diameter and the central circle pattern of each

target was modified from a full circle to three-quarter-circle target (Figure 5-6 (b)) which can also be detected by the self-developed target detection program.

Under this configuration, the target size in each photograph can be calculated using equation (3.1), and the result is shown below:

$$I_R = \frac{O_R \times f}{D} = \frac{26 \times 4.57}{6500} = 0.01828(mm) \quad (4.1)$$

Furthermore, by the use of the equation (3.2), the total number of pixels of a target in the sensor can be calculated.

$$P = \frac{F_p \times I_R}{F_w} = \frac{3264 \times 0.01828}{4.6} = 13(pixels) \quad (5.2)$$

According to the experimental results, which were outlined in § 4.4.1, the optimal range of target sizes for the developed target detection algorithm (in conjunction with 3/4 circle targets) is between 11 pixels to 30 pixels. The calculated results (13 pixels) show that this sized target can meet the requirements of both the developed target detection algorithm and the photogrammetric processing software (the instruction manual of PhotoModeler recommends that centre target widths in the photos be at least 8-10 pixels wide).

5.3.3 Operational processes

The experiment lasted roughly two hours. The four smart phone cameras captured synchronised image sequences with the exposure frequency of four frames per minute (one frame every 15 seconds). Every twenty minutes, the rolling shutter was shaken for five minutes by the operator in order to create movement of the rolling shutter.

In terms of the photogrammetric processing, the targets' coordinates were processed using the developed target detection program. These 2D coordinate data were provided for further photogrammetric processing using PhotoModeler. Once the photogrammetric processing was completed, the photogrammetric results of stationary targets were compared with physical measurements to calculate the RMSE values that could be used to evaluate the precision of the developed target detection

algorithm. As shown in Figure 5-17, four distances (S1, S2, S3 and S4) were selected for comparison. Meanwhile, the photogrammetric results of those targets, which were arranged onto the rolling shutter, can also be used to display the dynamic behaviour.



Figure 5-17: The chosen distances for comparison.

5.3.4 Test results

5.3.4.1 Inspection of system synchronisation

In order to test the performance of remote control ability, the terminal computer, which was used to control multiple mobile phone devices, was operated in a laboratory. The communication between mobile phone stations and the terminal computer relied on 3G mobile communication networks (the test area was not covered by an available wireless network). Therefore, the priority of this experiment is to ensure the data transmission and system synchronisation can work correctly. During a two-hour experimental period, each camera captured 473 photographs. The system report shown that, these cameras all worked properly without any failure during these two hours (i.e., the system recovery mechanism was not triggered during this period). In terms of the inspection of system synchronisation, the exposure timings of these four cameras were plotted in Figure 5-18.

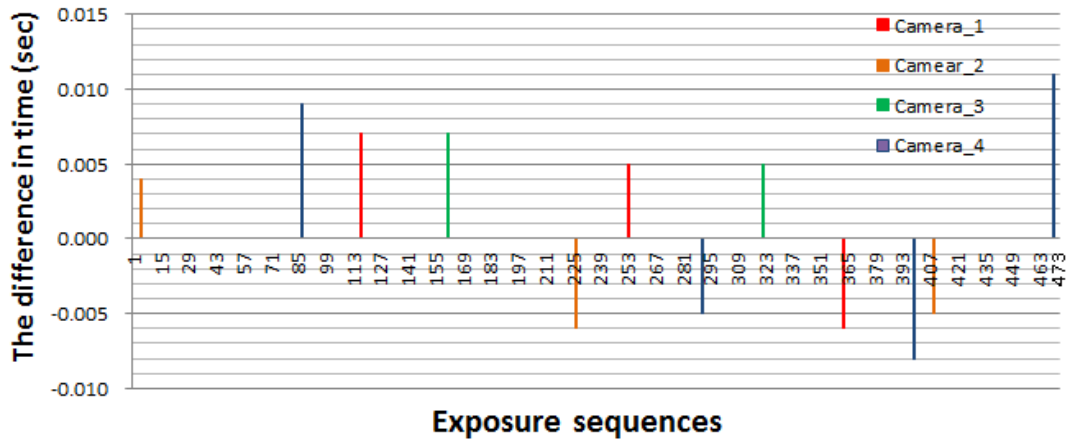


Figure 5-18: the exposure timing of each mobile phone camera.

As shown in Figure 5-18, the results were calculated by comparing the differences between the scheduled exposure timetable and the actual exposure timings. If the actual exposure timing is exactly identical with the scheduled exposure timetable, then the result will be zero. Otherwise, the differences in time will appear in the figure. In comparison, Camera_1 had three non-synchronised exposures during 473 exposure events. Camera_2 had three times non-synchronised exposures. Camera_3 and Camera_4 had two and four times non-synchronised exposures, respectively. But none happened simultaneously. Apart from these twelve non-synchronised exposure epochs, the remaining 461 exposure epochs were accurately triggered following the scheduled exposure timetable and these image data could be used for further photogrammetric processing.

5.3.4.2 Evaluation of photogrammetric precision

Figure 5-19 (a) shows the configuration of photogrammetric targets on the test structure. A total of 21 targets were attached onto the rolling shutter (the serial number of these targets started from PT_1 to PT_21). Six targets were arranged onto the brick wall and the serial number of these targets started from PT_22 to PT_27. The distance from Point_22 to Point_26 was defined a scale bar. The distance from Point_24 to Point_27 was defined as the X-axis and the X-Y plane is parallel to the rolling shutter. Table 5-7 shows the standard deviations in coordinate measurements of six control points that were attached on brick walls.

Point ID	σX (mm)	σY (mm)	σZ (mm)
PT_22	0.307	0.323	0.303
PT_23	0.305	0.329	0.312
PT_24	0.297	0.316	0.289
PT_25	0.293	0.308	0.315
PT_26	0.315	0.322	0.297
PT_27	0.322	0.319	0.307
Average	0.306	0.320	0.304

Table 5-7: Coordinate standard deviations of control points.

5.3.4.3 Assessment of photogrammetric accuracy in distance measurements

In order to evaluate the photogrammetric results, four chosen distances (as shown in Figure 5-19 (b)) were selected to compare with physical measurements. The endpoints of these distances were located on stationary brick walls. The results of comparison are outlined in Table 5-8.

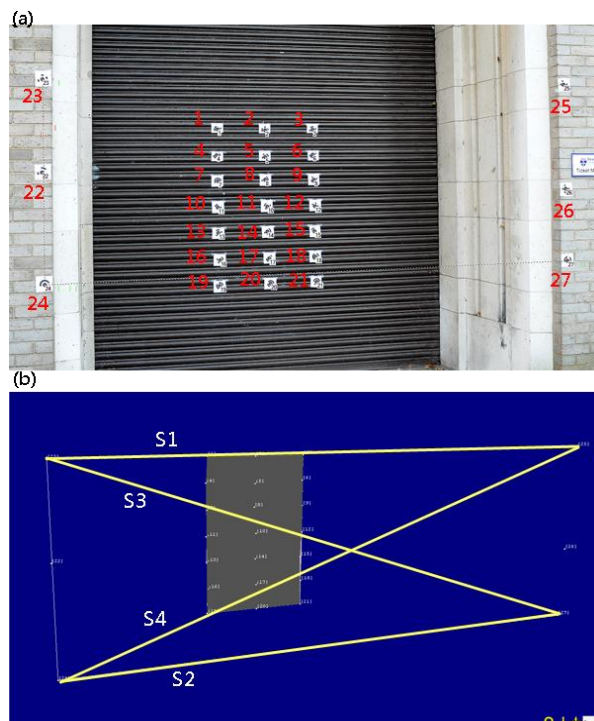


Figure 5-19: Configuration of photogrammetric targets for the rolling shutter test. (a)

is the overall layout of target configuration; (b) illustrates the chosen distances for comparing with the physical measurements.

	S1	S2	S3	S4
Physical measurements (mm)	3636.593	3644.358	5601.133	5613.356
Standard deviation of photogrammetric results (mm)	0.286	0.301	0.327	0.331
RMSE (mm)	0.322	0.332	0.341	0.349
Relative accuracy	1/11,300	1/11,000	1/16,400	1/16,100

Table 5-8: Comparison between the photogrammetric results and physical measurements.

5.3.4.4 3D modelling and deformation tendency

According to the target configuration shown in Figure 5-19, the XY plane is parallel to the rolling shutter. Therefore, the displacements of the rolling shutter can be effectively demonstrated by plotting the Z coordinate of each exposure epoch. Each target on the rolling shutter can generate a displacement curve to describe this target’s dynamic behaviour during the experimental period. Using PT_13 as an example, the movements in Z direction are shown in Figure 5-20.

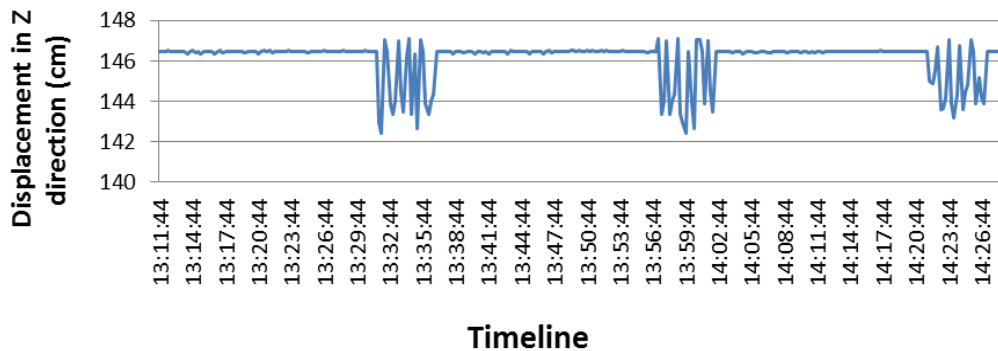


Figure 5-20: Displacement of a rolling shutter in Z direction (PT_13).

Every twenty minutes during the experimental period, the rolling shutter was pushed

(for an approximate five-minute period) by the operator. In the natural state (i.e., does not include those data that were captured when the experimenter was pushing the rolling shutter), the average of the Z coordinate is 1461.087 mm. The additional interference caused a maximum 27.832 mm displacement in Z direction (at the 92nd exposure epoch). Figure 5-21 shows the discrepancy map which was produced by the MATLAB software. The status of bearing force is enhanced by the colour display of different zone blocks.

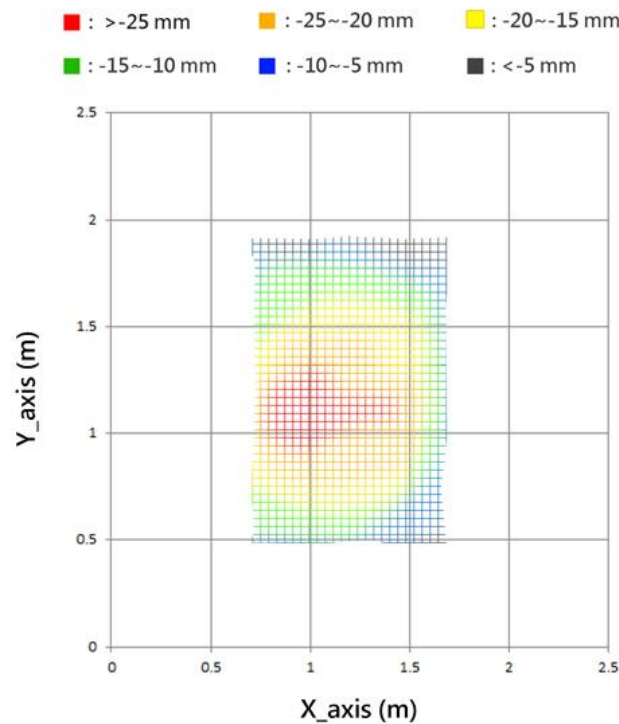


Figure 5-21: Discrepancy map derived from 3D conformal transformation (at the 92nd exposure epoch).

Moreover, these computed photogrammetric results can further be used as reference points to reconstruct a 3D object model. 3D models help users to have realistic impressions of the structures over 2D photographs (Li et al., 2011a). As shown in Figure 5-22, a total of 4 photographs were used to reconstruct this 3D model. There exists some seam lines on the mosaicking boundaries (the right half of the rolling shutter and the top half of the 3D model) due to lack of sufficient image data (that were captured from different angles) and control points around these areas.

Meanwhile, the deformation sequence of the rolling shutter was demonstrated in Figure 5-22 (c). By demonstrating deformation sequences of particular epochs, the engineers can further understand the distortion tendency of target structures.

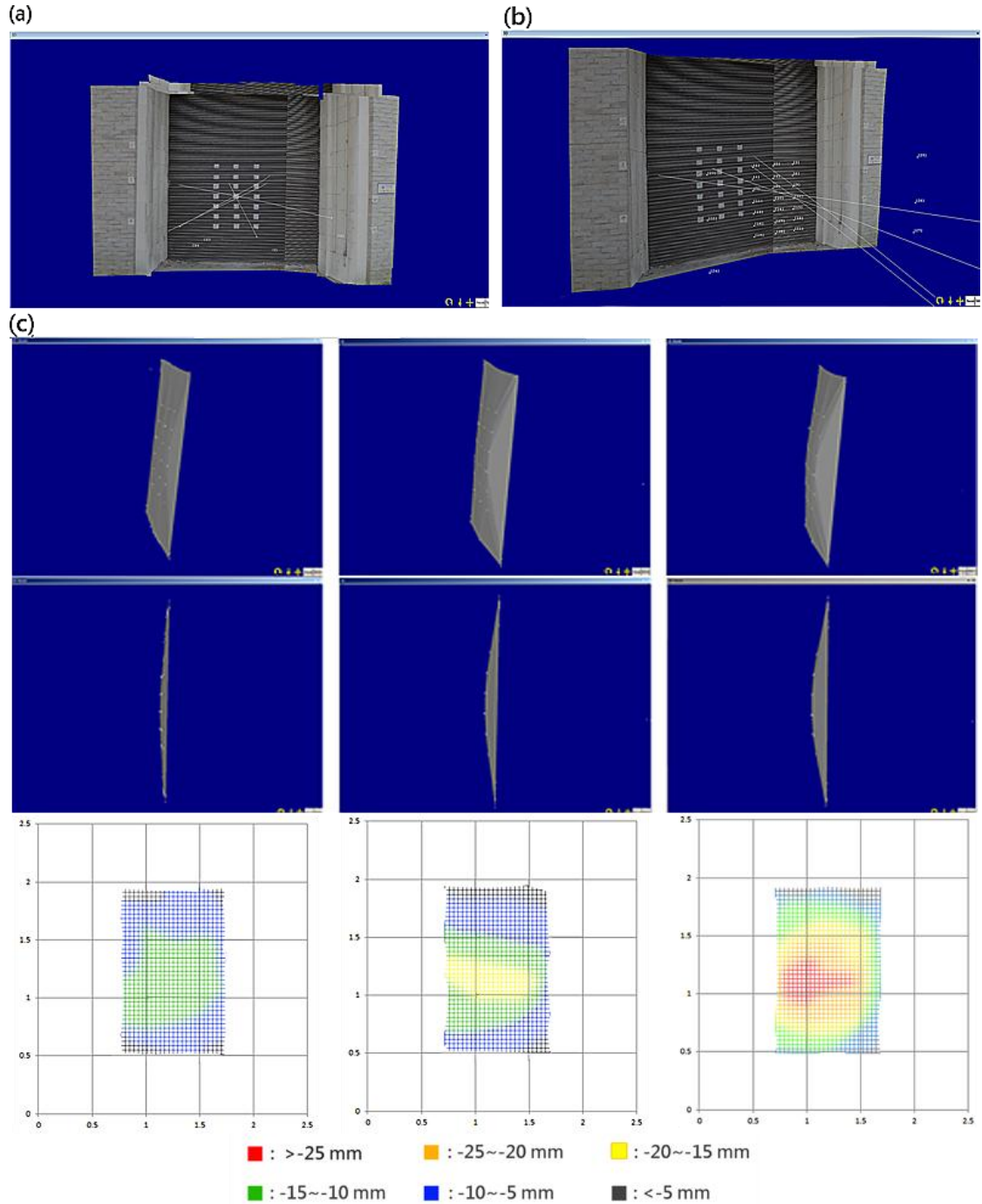


Figure 5-22: Textured 3D model and Deformation sequence of rolling shutter.

5.4. Summary

In this chapter, a shaking table experiment was performed to examine the integration and overall performance of the developed system. This experiment has demonstrated the mobile platform based photogrammetric system for structural monitoring of a moving object (the rigid box monitoring task) and a flexible structure (a three-story stainless steel moment frame structure) in a laboratory environment. The test results clarified the relationship between photogrammetric precision, target size (full circle targets)⁴ and camera network configurations. Meanwhile, the analysis of dynamic structural behaviour was demonstrated in § 5.2.2. A rolling shutter monitoring test further examined this mobile system in a outdoor environment. The outdoor test not only evaluated the system performance and photogrammetric accuracy, but also inspected the remote control mechanism. Finally, based on these calculated photogrammetric results, a 3D model was reconstructed.

⁴ The relationship between the photogrammetric measurement precision and 3/4 circle targets has been presented in §4.4.

CHAPTER 6. DISCUSSION OF RESULTS

6.1 Introduction

In this research, a low-cost, mobile phone-based photogrammetric system has been developed. The system was designed based on photogrammetric principles and exploits the latest mobile phone technologies to present an automatic solution for long-term structural monitoring applications. A series of experiments were undertaken to comprehensively evaluate this developed system. The system was not only evaluated under laboratory conditions; an outdoor monitoring scenario was simulated. Further discussion will be provided in this chapter, which will be drawn from these experimental discoveries. Also, the limitations of the system are outlined.

6.2 System inspection

During the initial phases of this research, a simulation experiment was undertaken to assist the developer in understanding the essential elements of a close-range photogrammetric structural monitoring system. In the discussion chapter, the cushion test is revisited to clarify the improvement, strengths and weaknesses of the developed mobile platform based system.

This experiment (cushion test) was performed using four digital SLR cameras (Canon 50D DSLR camera with a 28 mm prime lens) to monitoring an Ethyl Tetra Fluoro Ethylene (ETFE) foil cushion (as shown in Figure 6-1).

The cushion was inflated using a low pressure air supply, produced by a customised fume extractor, which will be referred to as the air supply. Air pressure was controlled manually via a pressure control manifold, which allowed excess air flow to be bled from the system, thus lowering or increasing the system's pressure. The pressure levels were measured by a pressure transducer that was time stamped and logged within the same system that logs and triggers the photogrammetric camera system. In this way, the photogrammetric data was linked to the cushion pressure. The pressure was also read manually using a manometer. The experimental setup is illustrated in Figure 6-2.

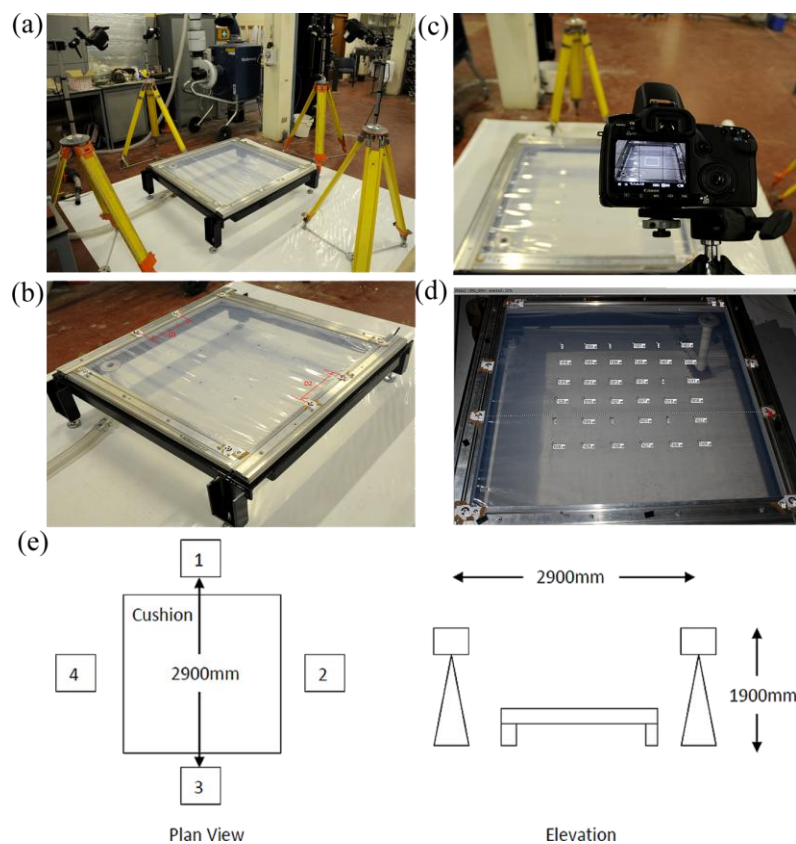


Figure 6-1: The arrangement of test field; (a) entire view; (b) a close-up image of test cushion; (c) The viewing field and (d) the output image data from one of the four image stations, (e) camera positioning.

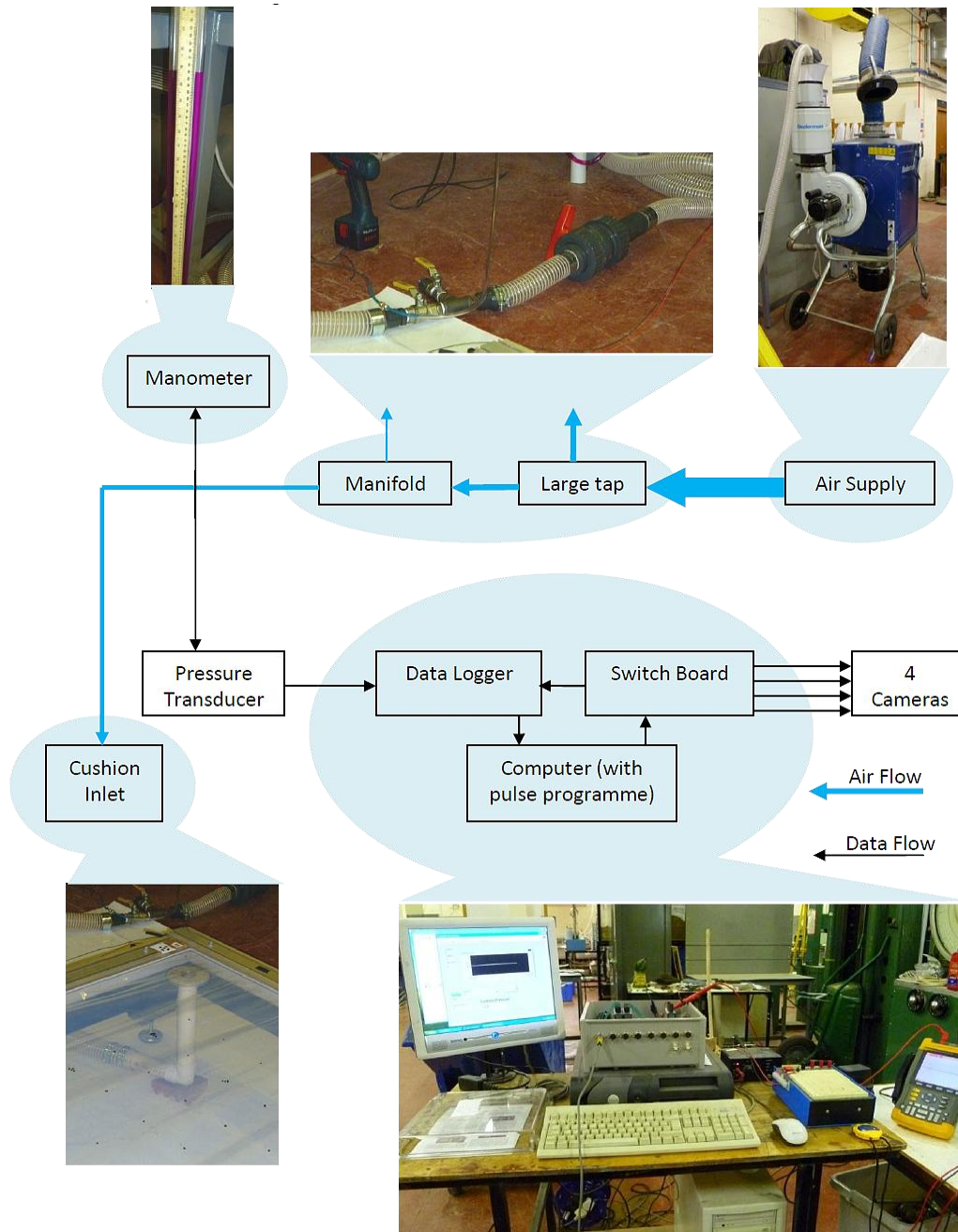


Figure 6-2: Experimental setup for the cushion test.

The cushion was inflated at intervals of approximately 100Pa from 0Pa up to 1400Pa. At each interval, the precise internal pressure was recorded via a pressure-transducer and data logger and the surface co-ordinates were determined via photogrammetry. Each trigger signal was logged and time stamped alongside the pressure data; in this way the pressure is linked to each set of 4 images. The cameras were triggered simultaneously by a regular pulse every 18 seconds, which was generated by a custom

made computer programme produced in Virtual Instrument called Pulse. In order to obtain the dynamic trend of cushion inflation, the photogrammetric results was processed at regular intervals (around every two and a half minutes to three minutes) and the results are shown in Table 6-1 and Figure 6-3.

Time	Pressure		Rise(mm)
	mm H2O	Pa*	
00:00:00	0.0	0.0	0.0
00:02:24	11.4	111.8	11.4
00:08:24	22.3	218.8	25.0
00:11:42	33.4	327.7	31.5
00:15:00	40.7	399.3	34.1
00:18:36	48.8	478.7	39.8
00:21:18	61.7	605.3	40.4
00:25:30	69.6	682.8	45.0
00:27:54	80.3	787.7	46.1
00:31:30	91.1	893.7	49.7
00:34:30	100.4	984.9	50.6
00:37:30	110.7	1086.0	55.2
00:41:06	122.0	1196.8	57.6
00:44:24	130.4	1279.2	59.6
00:47:24	141.0	1389.2	60.8

*1mm of water is equivalent to 9.81Pa

Table 6-1: Photogrammetric measurements of cushion pressure and rise.

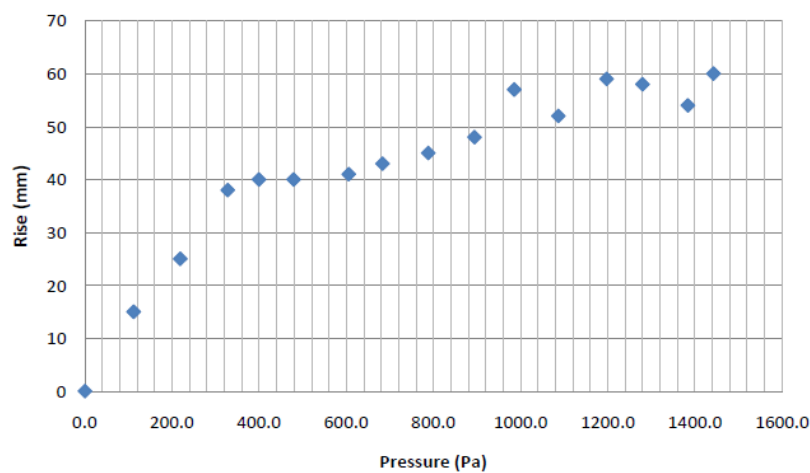


Figure 6-3: Cushion rise against pressure.

This kind of DSLR camera-based photogrammetric system has been extensively used in actual structural monitoring applications (as the related applications have been presented in Chapter 2). Therefore, this system was used as a reference to provide a deeper insight into what requirements a photogrammetric system needs in order to be applied to structural monitoring applications. Through practical manipulation, the results were used to assist in planning the optimal solutions and improving the system performance.

For example, with the exception of one camera, the camera system was successfully fired every 18 seconds throughout the entire experiment. The final result of this cushion test indicated that before the end of the experiment, one of the cameras ran out of power and stopped firing. This affected the photogrammetric results for pressures above 1200Pa (i.e. 1279Pa and 1383Pa). In view of this, this developed system designed a rigorous self-checking mechanism to ensure each remote imaging station would work properly during a long period operation. Meanwhile, two pressures were analysed using only three images instead of 4, which may have reduced the accuracy of the output. Furthermore, because this cushion test lacked reliable physical measurements and other photogrammetric measurements for comparison, the performance and overall precision were unknown. Therefore, when performing the inspections for the mobile platform based structural monitoring system, at least one precise physical measurement or another photogrammetric comparison (using different imaging devices against smart phone cameras) were prepared to generate sufficient data for performance evaluation. In addition, an experiment was performed to explore the relationship between the quantity of imaging stations and the photogrammetric precision. The result has been presented in this thesis and confirmed that the results of photogrammetric processing can be significantly improved by increasing the number of camera stations to a certain number of stations (the use of four or more cameras can significantly improve the photogrammetric result; the improvement trend will become more gentle when the camera stations are increased to six or more).

In terms of the target issue, for this indoor test, a flash gun was required in order to reduce the aperture size, thus increasing the quality of the images. However, a

concern when capturing the photogrammetric images was that there reflections from the surface of the cushion, which were caused by the simultaneous firing of the camera's inbuilt flashes. What is more, the size of the targets and the transparency of the ETFE also had as impact on the success of the photogrammetry, as shown in Figure 6-4. The targets were deemed to be too small and due to the need to attach the reference targets to the immovable frame, a wide angle view had to be utilised in order to capture all targets.

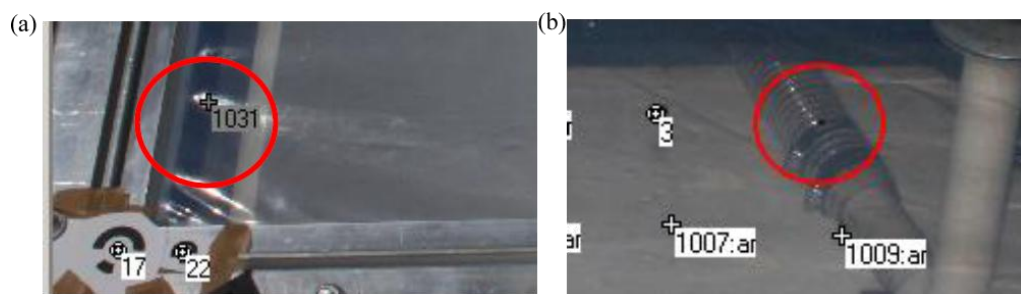


Figure 6-4: Mis-identified target points; (a) reflective point mistaken for target, (b) point cannot be identified due to pipe leading to inlet.

According to this experience, a detailed discussion addressing the determination of optimal target size for close-range photogrammetry, was proposed in this thesis. Moreover, a dedicated target pattern was designed to enhance target recognition rate under a variety of scenarios. A series of experiments have proven that this three-quarter circular black pattern on a white background target can significantly improve the image contrast of photogrammetric targets, even for those images that were captured using low resolution cameras (such as mobile phone cameras). One notable restriction is that these targets cannot be applied onto high ductility surfaces as the target stickers may easily peel off after the object surface has been stretched. In comparison to other structural monitoring measurements that were introduced in Chapter 2, one of the weaknesses of photogrammetric measurements is its reliance on photogrammetric targets. Therefore, photogrammetric measurements are not suitable to be applied on environments that do not provide enough brightness for image acquisition or are unable to attach any photogrammetric targets on to the surface of target objects.

Sampling rate is another restriction. Due to the limitations of the mobile phone

hardware specifications and network transmission rates, the shortest sampling interval of the developed system was ten seconds. The primary concern was the memory usage for wireless data transmission. The detailed memory usage for each related procedure was presented in Chapter 3. When the hardware specifications of handset devices are enhanced, it is expected that the sampling will have significantly improved. Nonetheless, the cushion test, shaking table test and the currently-used sampling rates enabled engineers to create reliable motion models of target structures. Right at this stage, the sampling rate of ten seconds pre-frame is sufficient for long-term sustainability structural monitoring applications. If engineers have the requirement for a more-frequent sampling rate (for example, the video stream of 30 frames per seconds) to record the continuous deformations for target structures, they must replace the wireless transmission with wire transmission, which boosts the sampling rate efficiently.

6.3 Assessment of photogrammetric accuracy

When undertaking a metric monitoring task, accuracy is one of the primary concerns. Focusing on accuracy improvement for photogrammetric measurements, some of the most commonly used approaches have been highlighted in Chapter 2. According to the understanding, improvement of the measurement precision can be achieved by the use of stronger photogrammetric camera network configurations (some examples are presented in Figure 2-18). The photogrammetric precision can also be improved by increasing the quantity of imaging stations (Figure 2-20). Meanwhile, the operators can change the geometric position of imaging stations (i.e. the adjustment of the relative position between imaging stations and target objects); this will help obtain better photogrammetric network configurations (the relationship is shown in equation (2.5)). In this research, a series of experiments have been undertaken to evaluate the system performance according to the aforementioned suggestions. There will be a discussion about this in the next section.

6.3.1 The influence of the quantity of imaging stations on photogrammetric precision

For the experiment that was presented in § 4.2.1, the relationship between the

photogrammetric measurement precision and the quantity of imaging stations was examined. The results indicate that the improvement of measurement precision is proportional to the quantity of the camera stations. The photogrammetric measurement precision is progressively improved with the increase of imaging stations. From the results shown in Figure 4-3, it can be seen that when the imaging stations were increased from 3 cameras to 8 cameras, the standard deviation of distance measurement of smart phone datasets improved by 76 % from 0.029 mm to 0.007 mm. However, improvement slowed down when the imaging stations were increased from 4 cameras to 8 cameras. In terms of the DSLR camera data set, the standard deviation of distance measurement improved by 37% (from 0.008 mm to 0.005 mm) when the imaging stations were increased from 3 cameras to 8 cameras. However, when the imaging stations were increased from 6 cameras to 8 cameras, the standard deviation shows no obvious improvement (the standard deviation (the numbers were rounded to the nearest thousandth) of both groups is 0.005 mm). The results outlined that the increase of photogrammetric imaging stations result in better photogrammetric measurement precision (especially for those imaging stations that were equipped with low resolution imaging devices).

6.3.2 The influence of photogrammetric network configuration and target sizes on photogrammetric precision

In terms of the shaking table test, the statistical data shows that the standard deviations in the Z-axis are obviously higher than X and Y axes. Equation (2.2), (2.3) and (2.4) shows the relationship between the photogrammetric measurement errors of each coordinate axis and imaging station configurations. As mentioned above, the measurement errors in X and Y axes (M_X and M_Y) are proportional to the object distance and the measurement error in Z-axis (M_Z) is proportional to the square of the object distance. These equations outlined that the use of shorter object distance results in a more significant improvement in photogrammetric measurement precision on the Z-axis than on the other two axes (the premise is that the length of baseline is constant). However, the experimental results of the shaking table test did not always show a better precision when using shorter object distance. In this test, the length of baseline and the object distance changed simultaneously and the length of baseline is

proportional to the photogrammetric measurement precision. When using the shorter object distance, the length of baseline also gets shorter. The use of shorter object distance should lead to better photogrammetric results; however, the shorter length of baseline may decrease the photogrammetric precision. To further verify the relationship between the measurement precision, object distance and the length of baseline, more experiments are required (either the object distance or the length of baseline should be the constant value).

In addition to the change of object, distances may significantly affect the photogrammetric measurement precision on the Z-axis; the following sections introduce another possible cause that may explain the larger increments of photogrammetric measurement error in the Z-axis.

The process of photography is the transference of three-dimensional data onto a two-dimensional plane (Li et al., 2011a). During the process, image information in the direction that is perpendicular to a photographic optical axis of the camera, will have the maximum compression (Guidi et al., 2003). When vertical photography is performed, the image information on the Z-axis suffers the highest compression (Gleitsmann and Kappas, 2004). In order to mitigate this phenomenon, the imaging station configuration of the shaking table test did not run parallel to the X-axis (the direction of shaking table platform movement) and the layout was shown in Figure 5-1 (a) and Figure 5-5. For the purposes of improving the strength of the photogrammetric camera network, to reduce the compression of imagery information in the Z-axis direction, these cameras have used low oblique convergent photography (Gleitsmann and Kappas, 2004) for camera station configuration. However, constrained by site limitations (there is too much equipment and heavy machinery in the test site that cannot be moved) and an insufficient number of cameras, it was not possible to significantly improve the strength of the photogrammetric camera network (these four camera stations can only be lined up in front of the shaking table as shown in Figure 6-5).



Figure 6-5: Configuration of imaging stations for shaking table test.

However, the camera network configuration for the outdoor experiment was not as restricted by site constraints and was improved following the recommendations which were outlined in Chapter 2. Firstly, four cameras were arranged in different instrument heights, as recommended by Guidi al. (2003). In terms of the convergent angle between camera stations, Ethrog (1987) recommended that the optimal convergent angle of imaging stations can be obtained when the main error of each axis is the minimum. Equation (6.1), (6.2) and (6.3) show the main error of X-axis, Y-axis and Z-axis.

$$M_x = \frac{1 + \tan \alpha \times \tan \phi}{1 - \tan(\alpha - \phi) \times \tan \phi} \times m_x \sqrt{\left(k_2 - k_1 k_2 \frac{x}{f}\right)^2 + \left(k_1 k_2 \frac{x}{f}\right)^2} \quad (6.1)$$

$$M_y = \frac{\sqrt{\left(\frac{y}{f} \times \tan \phi \times m_x\right)^2 + (m_y \times \sec \phi)^2} \times \sqrt{k_2^2 + 2\left(k_1 k_2 \frac{y}{f}\right)^2}}{1 - \tan(\alpha - \phi) \times \tan \phi} \quad (6.2)$$

$$M_z = \sqrt{2} k_1 k_2 \frac{1 + \tan \alpha \times \tan \phi}{1 - \tan(\alpha - \phi) \tan \phi} m_x \quad (6.3)$$

Where f is the focal length of two cameras (assume both of these two cameras have an identical focal length (i.e., $f_1=f_2=f$)), $P(x, y)$ is the object point, m_1 is the measurement error on X-axis for two images in the image coordinate system, α is the azimuth and ϕ is the convergent angle.

When the convergent angle is 90 degrees, equation (6.1), (6.2) and (6.3) can generate the minimum main errors. For this reason, the convergent angle between two tripods is close to 90 degrees to create the optimal convergence angle. According to the experimental results, which were outlined in § 4.4, the optimal target size for the use of 3/4 circle targets has been clarified and the chosen target size (13 pixels) of the outdoor experiment can meet the restriction. The standard deviations of six control points were summarised in Table 5-7 and the dispersion of standard deviations in each axis is more equally spread in comparison to the shaking table test.

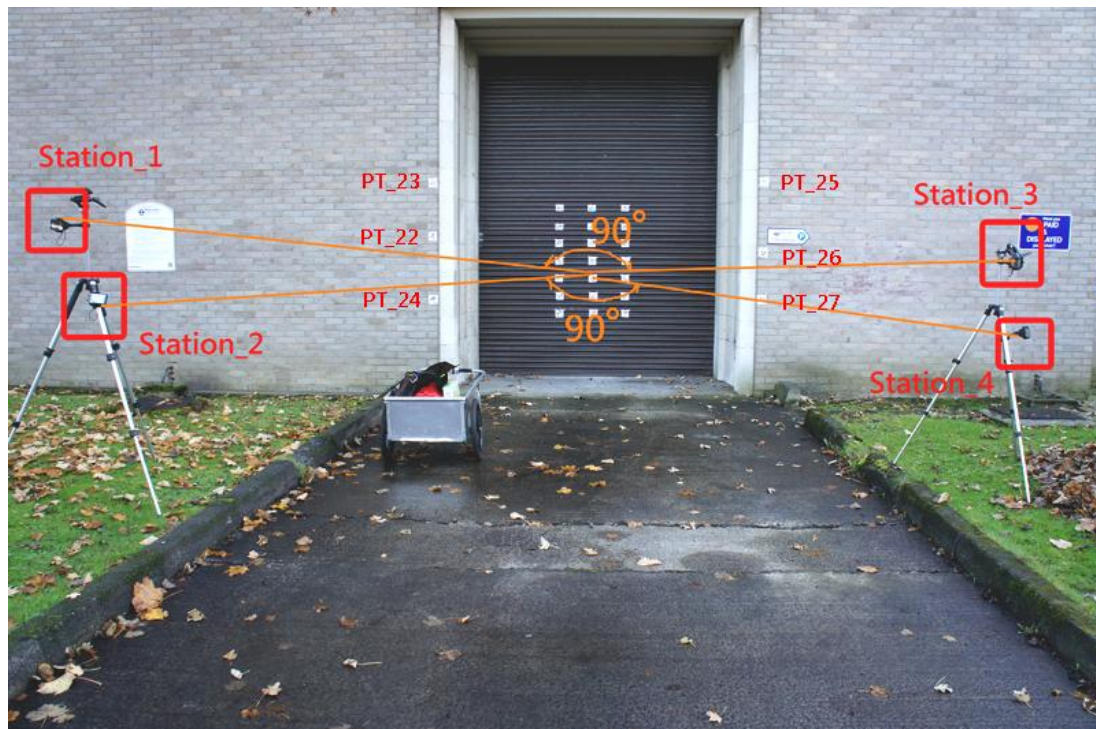


Figure 6-6: Configuration of imaging stations for outdoor test.

The experimental results show that the differences of target sizes led to a remarkable variation of measurement precision. To further investigate the influence of the target sizes on photogrammetric precision, two experiments were undertaken for evaluation. As the experimental results show in § 4.4, when the chosen target size of 3/4 circle target is in the range of 11 pixels to 30 pixels, the developed target detection program has the best performance to identify the centre of each target and can further enhance the photogrammetric measurement precision.

In the shaking table test, four different sized full circle targets were utilised. As shown in Table 5-2 and Table 5-3, the optimal photogrammetric target sizes changed with the different types of cameras (because the use of different cameras could lead to differences in the camera calibration parameters). The optimal target size for Canon 50D digital SLR cameras is 4 mm in diameter. When the approximate object distance was 2.5 m, the DSLR stations have the best precision in coordinate measurement. To further check the target size in pixels, the diameters of 4 mm targets (with the approximate object distance of 2.5 m) were 8 pixels to 9 pixels. When the approximate object distance was expanded from 2.5 m to 3.0 m, the target size in

pixels decreased to 6 pixels to 7 pixels and more than one third of images did not detect the 4 mm targets automatically. This automatic process of target detection was therefore replaced by manual selection and this substitution affected the quality of the 4 mm targets' results by causing larger standard deviations. The same situation occurred in the smart phone stations with the use of the 4 mm targets (the calculated target size in pixels was 3 pixels to 5 pixels). All the images that processed locations of the 4 mm targets were not detected automatically. As a result, the smart phone stations in combination with use of the 4 mm targets generated higher standard deviations than the 9 mm targets.

In terms of the smart phone camera stations, the use of 9 mm targets (the approximate object distance was 2.5 m) generated the best precision among the four different target sizes. To further check the diameter in pixels of the 9 mm target, when the approximate object distance was 2.5 m, the calculated target sizes of mobile phone camera stations were 9-10 pixels. When the approximate object distance was extended to 3.0 m, the calculated target sizes of mobile phone camera stations decreased to 7-6 pixels (which were still within the specified range).

When using a 4 mm target in combination with the approximate object, the distance was 2.5 m; the DSLR camera stations performed the optimal photogrammetric results. In this configuration, the diameters of target sizes in pixels were 8-9 pixels and the optimal target sizes of DSLR camera stations showed almost equal range with the smart phone camera stations. Moreover, these results were consistent with the restrictions of target sizes of PhotoModeler (no less than 8-10 pixels).

The shaking table test also examined the influence of oversized targets on photogrammetric precision. According to the instructions, the target detection function of PhotoModeler was developed based on the generalised Hough transform algorithm. In terms of the use of generalised Hough transform for target detection and target centre evaluation, Lu and Tan (2008) proposed that the precision of target centre evaluation is proportional to the integrity of edge information. When the edge information is fragmentary, the precision of target centre detection will decrease. Dimitrios al. (1999) also stated that the precision of target centre evaluation is inversely proportional to the dimension of the circle (or ellipse). The results of

Table 5-3 support the above-mentioned conclusion. The standard deviations of the 16 mm targets were obviously higher than the 12 mm targets in both the DSLR and smart phone camera stations. As shown in Figure 5-12, the RMSE values of photogrammetric measurements of smart phone camera stations were gradually improving when the target sizes were dwindling. However, smaller sized targets do not provide enough information that limits the localisation accuracy of matching procedures or edge-oriented operators (Luhmann et al., 2007). When the target sizes are smaller than the minimum requirement of photogrammetric processing software, the measurement precision began to reduce.

6.3.3 Performance assessment

A series of experiments have been presented in Chapter 4 and Chapter 5, which were undertaken to evaluate the performance and related accuracy of the mobile platform based monitoring system. In this section, the performance of this developed system is compared with related measurement techniques (which have been outlined in Chapter 2) to determine whether the mobile platform based photogrammetric system is suitable for structural monitoring purposes or not.

As shown in Table 2-1, the suggested accuracy of different measurement techniques was outlined. Among these techniques, the laser scanning measurements, theodolite measurements, architectural and engineering photogrammetry, tachymetric surveying and GPS techniques, etc. have been widely utilised in modern civil engineering. The suggested accuracy of laser scanning measurements is higher than 1mm (the recommended dimensions of target objects are within the range of 5 m to 100 m). The proposed accuracy of theodolite measurements should be in the range of 0.5 mm to 5 mm (the recommended object sizes are between 6 m to 60 m). The suggested accuracy of architectural and engineering photogrammetry is at least 0.1 mm (the recommended object sizes are within the range of 20 m to 500 m). Meanwhile, the tachymetric surveying and GPS techniques are suitable to be used to measure larger objects (the tachymetric surveying: 200 m to 2000 m; GPS measurement can be used to measure the objects (or range of measurement) which are larger than 500 m) and the required accuracy of the tachymetric surveying is 5 mm to 50 mm, whilst the required accuracy of GPS measurements is in the range between 900 mm to 5000 mm.

The relative accuracies of these measurements are summarised in Table 6-2. In addition, Ansari (2005) further proposed that the required accuracy of the instruments which are utilised for structural monitoring applications, should be better than 1/10,000. In terms of the relative accuracy of using close-range photogrammetric measurements for structural monitoring, some close-range monitoring tasks that were performed using DSLR cameras were discussed in Chapter 2. Ozbek al. (2010) performed a large wind turbines monitoring task using close-range photogrammetric measurements (the experiment was performed using 4 DSLR cameras) and the experimental results showed that the relative accuracy was around 1/16,000. For a long-term slope monitoring task, the relative accuracy of distance measurement was up to 1/12,000 (Ohnishi et al., 2006). Jiang and Jauregui (2010) also used DSLR camera-based imaging stations to perform a concrete bridge monitoring project and the overall relative accuracy was approximately 1/15,500.

Measurements	Relative accuracy
Laser scanning	1/2,000~1/5,000
Theodolite	1/12,000~1/120,000
Architectural and engineering photogrammetry	1/25,000~1/100,000
Tachymetric surveying	1/40,000~1/100,000
GPS	better than 1/560

Table 6-2: Relative accuracy of different measurement methods.

In this research, a smart phone platform-based target detection algorithm in combination with dedicated 3/4 circle target patterns, were developed to provide better performance in photogrammetric target identification on the imagery. Under laboratory conditions, the developed target detection algorithm, combined with dedicated 3/4 target patterns, returned the optimal relative accuracy of approximately 1/67,000. Under the same configuration, the relative accuracy of conventional image processing algorithm (combined with normal full circle target patterns) was approximately 1/27,000. In terms of using the mobile platform based monitoring

system for actual monitoring applications, a relative accuracy of 1/28,000 can be obtained in a laboratory environment. This laboratory experiment was performed using original PhotoModeler targets (the centre of each target was a full circle target) and the photogrammetric processing was executed using the PhotoModeler software (including target detection process). In an outdoor environment, a rolling shutter monitoring task was performed. Combined with the use of developed target detection algorithm and dedicated 3/4 targets, the relative accuracy was approximate 1/16,400. In comparison with the information shown in Table 6-2, though the relative accuracy of the outdoor experiment does not meet the suggested accuracy of architectural and engineering photogrammetry, the experimental results have proved that the developed mobile platform based structural monitoring system can generate similar quality photogrammetric results with those DSLR-based photogrammetric monitoring tasks that have been presented in Chapter 2. Moreover, the overall accuracy of this developed monitoring system is qualified for structural monitoring purposes (the relative accuracy of this mobile platform based monitoring system is always better than 1/10,000).

To improve the measurement precision of the low-cost structural monitoring system, some suggestions for improvement have been presented in Chapter 2 and two of them have proven to be effective in this research. The photogrammetric precision can be significantly improved through the increase of imaging stations. In addition, the use of optimal target size has proved to be an effective approach to obtain high precision photogrammetric results.

In terms of the performance of automatic for long-term operation structural monitoring, the high quality system synchronisation mechanism can maintain the synchronisation delay between devices within millisecond level. Thus, multiple imaging stations can simultaneously record the movement sequences of target objects. Meanwhile, 62 hours (which was introduced in §4-3) and 48 hours (which was presented in §5-2; the middle two days of the shaking table test) of uninterrupted tests have proved that the automated monitoring system is capable of long-term operation. The experimental results have provided evidence that the real-time automatic system recovery mechanism can efficiently recover all devices when system failure occurs.

6.3 Potential problems in operation

When inspecting the system performance via a series of experiments, the experimental results have proven that the system is capable for general purposes of structural monitoring in accuracy. In terms of operational performance, some experiences that were obtained from practical experiments were presented in the following sections.

6.3.1 Deficiency in exposure frequency

When performing a photogrammetric-based monitoring task of dynamic objects, the adoption of more intensive sampling rate (i.e., higher exposure frequency) allowing a camera to record more details of movement of target objects. Normally, the required sampling rate of general structural monitoring applications is expected to reach the level of 1 Hz (Armer, 2001). In terms of the mobile platform based structural monitoring system, which was developed in this research, considering the lack of adequate temporary memory space and the less powerful CPU, the image capture frequency is limited (no less than ten seconds per frame, the ideal value is over than fifteen seconds per frame). Whether it is fifteen seconds per frame or ten seconds per frame, these levels of exposure frequencies are inadequate to meet the demands of actual structural monitoring of flexible structures. In the current stage, an alternative to obtain faster sampling rate was developed in this research based on the present conditions of smart phone hardware specifications. As proposed in § 3.3.1, the operators can utilise the grouping function, which was developed as a sub-function of the mobile phone-based structural monitoring system, to create different time gaps between imaging devices and to simulate the use of more intensive exposure intervals. In combination with the use of grouping function and a sufficient number of imaging stations, the system can achieve the basic requirement of 1 Hz sampling frequency.

6.3.2 The degree of automation of system operation

The use of automatic operation can significantly reduce labour costs. Figure 6-7 shows the workflow of this mobile platform based structural monitoring system, also,

the flowchart also indicates whether these procedures can be automatically performed or not.

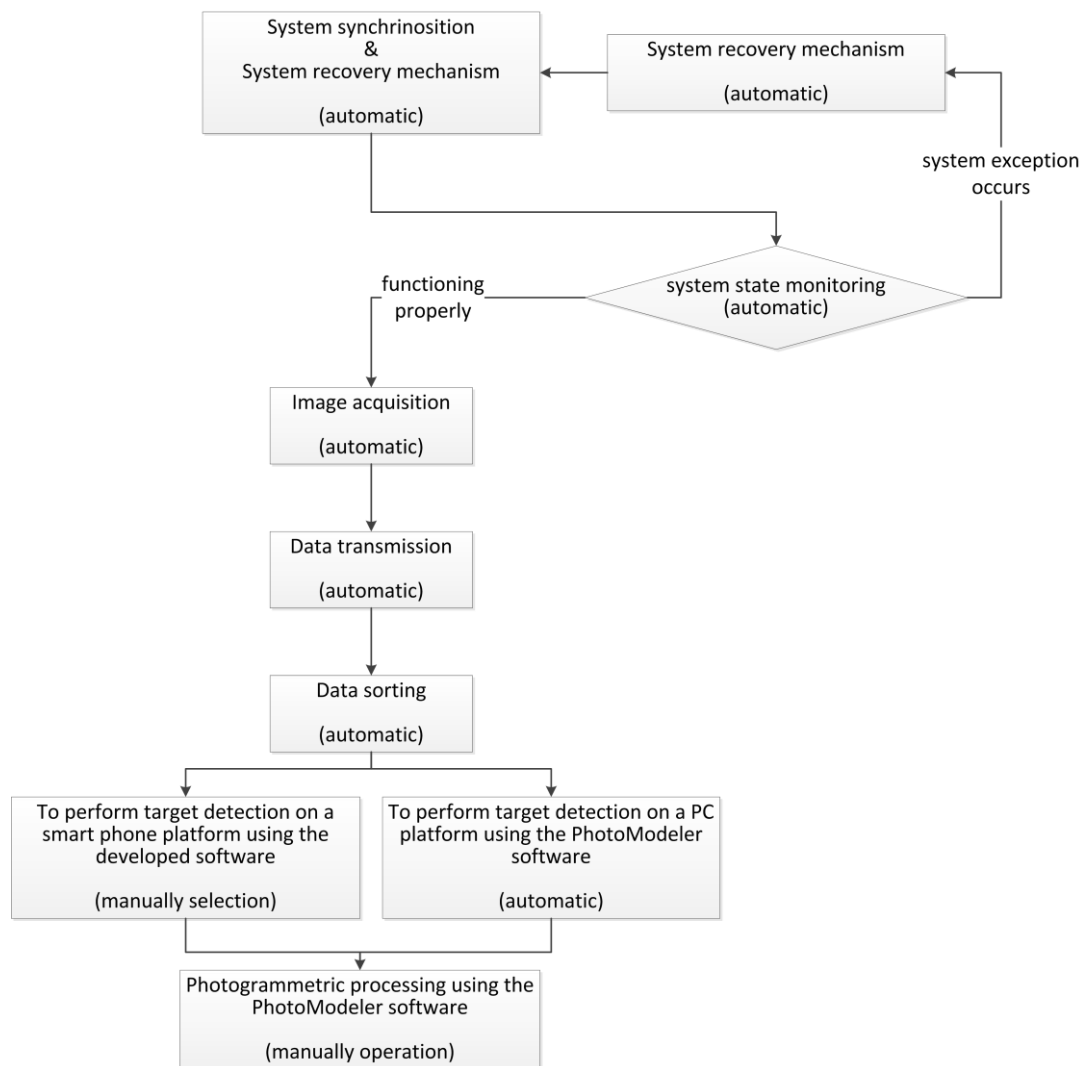


Figure 6-7: Flowchart of using the mobile platform based photogrammetric system for structural monitoring applications.

As shown in Figure 6-7, when using the mobile platform based structural monitoring system in combination with the commercial photogrammetric software package to perform a monitoring task, the operation is automated as much as possible except the photogrammetric processing.

In contrast, by using the developed target detection algorithm to perform target detection process can not only improve the precision of pin-point targeting capability, the use of screen touch function to narrow down the approximate location of each

target can reduce iterative computations and significantly reduce system load (thus, the idea of performing precise target detection process on a smart phone platform can be achieved). The experimental results showed in Table 4-2 and Table 4-3 can be used to compare the photogrammetric precision of the conventional algorithm and the developed target detection application. The average RMSE of the developed target detection application (Combining with the use of 3/4 circle targets) is 2 μm less than the GHT algorithm. Though using the developed target detection application can assist surveyors to precise identify the location of each photogrammetric target when measuring the smart phone camera imagery, the biggest disadvantage is that the operators need to manually point out the approximate location of each target on the smart phone screen. For a long-term photogrammetric-based structural monitoring project, the quantity of captured image data is normally up to thousands. When processing a large amount of image data, this disadvantage will require a lot of labour.

6.4 Summary

According to the experimental results that were presented in Chapter 4 and Chapter 5, a further discussion is provided in this chapter to evaluate these experimental findings, using the photogrammetric principles that were introduced in Chapter 2. These experiments have examined the stability for long-term operation, the quality of system synchronisation and the measurement precision etc. The overall results have proved that the low-cost photogrammetric system is capable of long-term structure monitoring of flexible structures (or rapidly moving objects).

CHAPTER 7. CONCLUSIONS AND FUTURE

WORK

7.1 Introduction

This thesis has demonstrated the development of a mobile platform based structural monitoring system for structural dynamic behaviour analysis, and for stable long-term monitoring applications. The feasibility of the system has been proven through the implementation of an indoor simulation test and an outdoor experiment. Based on the implementation, experimental results and discussions that followed the research, the review of this research, summary contributions and future extensions are summarised in this chapter.

7.2 Summary of work

This research has aimed to propose a cost-effective solution for long-term structural monitoring applications. To achieve this goal, the implementation was divided into three stages following the objectives that were proposed in Chapter 1.

For the first stage presented in Chapter 2, a comprehensive understanding of the most commonly used SHM techniques in the current civil engineering field has been completed and the advantages and disadvantages of each relevant technique has been analysed which addressed Objective 1. Meanwhile, the usage restrictions of each technique were presented. At the end of this stage, the various techniques were

compared with each other. According to the comparison, the most appropriate technique to meet the aim of the development of a low-cost and precise structural monitoring system was determined.

Following the literature review presented in Chapter 2, in consideration of the execution cost, overall accuracy and the potential for long-term operation, a photogrammetric-based structural monitoring system was developed, as presented in Chapter 3, which directly addressed to Objective 2. This system was designed based on close-range photogrammetric principles. By combining the use of low-cost imaging devices (two or smart mobile phone handsets) with in-house control software, a monitoring project can be undertaken within a relatively low budget when compared to conventional methods. In terms of long-term monitoring projects, one of the major concerns is to ensure the monitoring system can always be kept in good working condition. In order to prevent the system from crashing during a long-term monitoring scheme, an automatic system state monitoring program and a system recovery module have been developed to enhance the stability of this low-cost system (§ 3.3.2).

The operations of this mobile phone-based structural monitoring system are fully automatic; also the system can be remotely controlled. This means that operators do not have to visit the site for operation, and saving considerable labour for long-term monitoring tasks. Meanwhile, the potential of a mobile phone platform in terms of the development of photogrammetric processing functions was studied in this research. Based on the existing mobile phone hardware specifications, a dedicated photogrammetric software suite was developed for the Android-based mobile platform (§ 3.4).

A comprehensive evaluation of the mobile phone-based structural monitoring system, via a series of independent experiments as presented in Chapter 4 and Chapter 5, then addressed Objective 3. Each function of the developed system was rigorously inspected to ensure the system could be used for high quality photogrammetric applications and is suitable for long-term structural monitoring purposes (the details of these experiments have been presented in Chapter 4). When the inspections of individual functions were completed, two large scale experiments were undertaken

under laboratory conditions; and in an outdoor environment (Chapter 5). These experiments were performed to evaluate the overall system integration and to identify potential problems (which was outlined in § 6.3) in actual operation for future improvements. By comparing the measurement accuracy with the related close-range photogrammetric research, outlined in Chapter 2, the mobile phone-based structural monitoring system was proven to be an eligible solution for long-term structural monitoring purposes, and this result is in line with achieving Objective 4.

7.3 Research contributions

In terms of the improvements for long-term structural monitoring applications, determining how to efficiently save on labour costs and time have become major concerns. In this research, conventional photogrammetric imaging stations have been replaced by Google Android-based mobile phones. The obvious advantages are that the size of an imaging station can be significantly reduced (i.e., the improvement of portability), whilst the functionality is retained. In comparison to conventional structural monitoring measurements (which were illustrated in Figure 3-6), the convenience of use of the mobile platform based system is significantly improved. The developed system is superior in size, instrument cost. The operators only need to carry a specific number of smart phones (two or more phones) to the test field and fix them on steady pillars to periodically capture the movements of target objects. The operators can save a considerable amount of time and effort with this system setup.

The functionalities of image acquisition, data transmission, system positioning and etc. have been integrated into a compact handset device that have been modified to meet the requirements of close-range photogrammetric purposes. By means of computer programming, the smart phones have been adapted to meet the requirements of long-term, structural monitoring purposes. The entire system was designed based on the Java architecture (the advantages have been introduced in § 3.2.4). By using developed algorithms and dedicated system integration mechanisms, effectiveness has been optimised. Therefore, even though the computing performance of a smart phone is poorer than a personal computer (or a laptop) the system can be smoothly implemented on a smart phone platform. Basically, the operator can use one smart

phone (or any Wi-Fi enabled electronic instrument that is compatible with JAVA language syntax) to remotely control the other handset devices (the remote image acquisition stations). However, in order to store large amounts of image data that were transmitted from each remote imaging station, in this research the control device was replaced with a laptop to provide a larger storage space. Even so, this mobile platform based system (includes a laptop, a wireless router and smart phones) is much smaller in size in comparison to traditional structural monitoring systems.

Because the quality of imaging sensor and lens of current mobile phone cameras cannot compete with digital SLR cameras (commonly used for precise photogrammetric applications) a developed target detection algorithm combined with dedicated target patterns were established on the Android-based mobile platform to improve the precision in terms of the implementation for photogrammetric target identification on low resolution images. An accuracy assessment experiment of this developed algorithm has been undertaken under laboratory conditions. In comparison with one of the most frequently used target detection algorithms - the GHT, the measurement precision in each chosen distance (on a 0.25 x 0.25 m glass plate) has at least 2 μm improvement (the object distance was approximately 1 m). Comparing the photogrammetric results with physical measurements, which were measured by a Zeiss P3 analytical plotter, the returned accuracy in distance measurement can achieve 1/67,000 (in a laboratory environment). When using this system for practical applications, a small-scale dynamic simulation test (the approximate object distances were in the range of 2.5 m to 3.0 m) was performed under laboratory conditions and the optimal relative accuracy in distance measurement of this experiment was approximately 1/28,000 (by combining with the use of conventional target patterns that were generated by the PhotoModeler software), and the outdoor experiment (the approximate object distance was 6.5 m) returned a relative accuracy of approximately 1/16,400 (by combining with the use of dedicated 3/4 circle target patterns). As summarised in Chapter 6, the recommended accuracy for structural monitoring in civil engineering should be better than 1/10,000 (Ansari, 2005). The measurement accuracy of related experiments which were used DSLR cameras as imaging stations were in the range of 1/12,000 to 1/16,000. In contrast, the mobile platform based structural monitoring system is qualified for general structural monitoring

applications in civil engineering.

Focusing on the application of deformation monitoring in dynamic structures, the mobile platform based system mainly focuses on the improvement of the synchronisation between imaging stations and the stability for long-term operation. When the system is in good working conditions, the synchronisation between devices can reach millisecond level that is fast enough for each imaging stations to freeze a structure's (specifically referring to civil engineering structures) instantaneous movements simultaneously. In terms of the quality of camera synchronisation and the stability of this system for long-term operation, a continuous experiment (62 hours; the experiment was presented in § 4.3) was undertaken to examine the performance of this system for long-term operation. According to the experimental results, among 7,439 exposure events, there were only seven non-synchronised exposure epochs appeared during the experiment. The probability of non-synchronised exposures occurring was 0.09%. The stability of the mobile platform based photogrammetric system for long-term operation was inspected in the same experiment. The system recovery mechanism was only launch once among 7,439 exposure events and the success rate for system recovery is 100 %. The results have proven that this automatic system recovery mechanism can efficiently recover all devices when the system failure occurs. When performing a long-term monitoring project, the features of remote control, automated system health monitoring and automatic system recovery deliver significantly reduce labour costs.

The operation of this mobile platform system is fully automatic; what is more is that the operator can remotely change the system settings. An automated system recovery mechanism has been designed to recover these handset devices from system failure. If the devices do not have hardware malfunctions, the operation can be sustained until the operator calls a halt. Thus, the engineers do not need to repeatedly commute between the laboratory and test site for data collection and maintenance. Significant transportation fees and time can be saved.

The data transmission of this system can use any general telecommunications network. In order to ensure the quality of long-term image data transmission between wireless

terminals, this research proposed a lossless segmentation algorithm that was modified from the VQ algorithm. Considering that the quality of the wireless network in every test site is always different, the engineers can adjust the size of the image segment and data transmission rate to optimise the performance of wireless data transmission.

To sum up, the research was completed following the proposed objectives that were outlined in Chapter 1. Based on the requirements of structural monitoring in civil engineering in combination with the mobile phone technologies and photogrammetric principles, a low-cost mobile platform based structural monitoring system was developed. Considering the image quality of smart phone cameras is not as good as high-end digital SLR cameras, the research attempted to maintain and to improve the photogrammetric accuracy by means of adjusting the photogrammetric network geometry following the principles which were proposed in Chapter 2. The experimental results have proven that the photogrammetric accuracy is proportional to the quantity of imaging stations. This research also strictly examined the relationship between the measurement accuracy and photogrammetric target sizes. After a series of experiments, the optimal target sized range for this system was proposed. Meanwhile, a smart phone platform-dedicated target detection application was developed in this research. This developed algorithm can assist surveyors to perform precise target identification process on smart phone camera imagery. By comparing with one of the most commonly used algorithms, the developed target detection algorithm showed good performance in target centre detection on low resolution imagery.

When the individual functions have been inspected, the final objective is to evaluate the overall performance of this developed system for practical applications. Two experiments were undertaken in a laboratory environment and in an outdoor environment, respectively. The results not only verified the mobile platform based structural monitoring system is capable of structural monitoring purposes, but also used different representations to display structural deformations.

7.4 Suggestions for future work

In this research, a series of experimental results have proven that a low-cost,

photogrammetric monitoring system is capable of long-term structural monitoring applications. With the development of hardware specification, the system can still be improved in some specific aspects to satisfy more requirements for different measurement schemes. The following sections proposed some directions and some advice for further research regarding the relevant issue

7.4.1 Considerations of photogrammetric accuracy

This research has proven that the photogrammetric result can be improved by increasing the quantity of imaging stations; also, the use of proper targets for measuring is another effective approach. According to the photogrammetric principles (which were outlined in § 2.2.3), the use of a shorter object distance or a longer baseline can also reduce the photogrammetric measurement errors. The equation (2.5) proposes five variables that may affect the photogrammetric precision. Among these five variables, a scale ratio can be derived when the mean object distance is divided by the focal length of the lens. According to equation (2.5), the photogrammetric precision is essentially proportional to the strength of the camera station configuration and the average number of exposures at each station. Meanwhile, the photogrammetric precision has an inverse relationship with the scale ratio and the image coordinate standard error. The better precision of photogrammetric target localisation, the better measurement accuracy can be achieved (Marshall, 1989). For majority of target centre detection algorithms, the optimal approximation of each target centre can be converged by means of iterative computations (Pratt, 2012). As shown in Figure 7-1, by performing repeated iterative computations, the approximation will move closer to the ideal “true” centre of the ellipse.

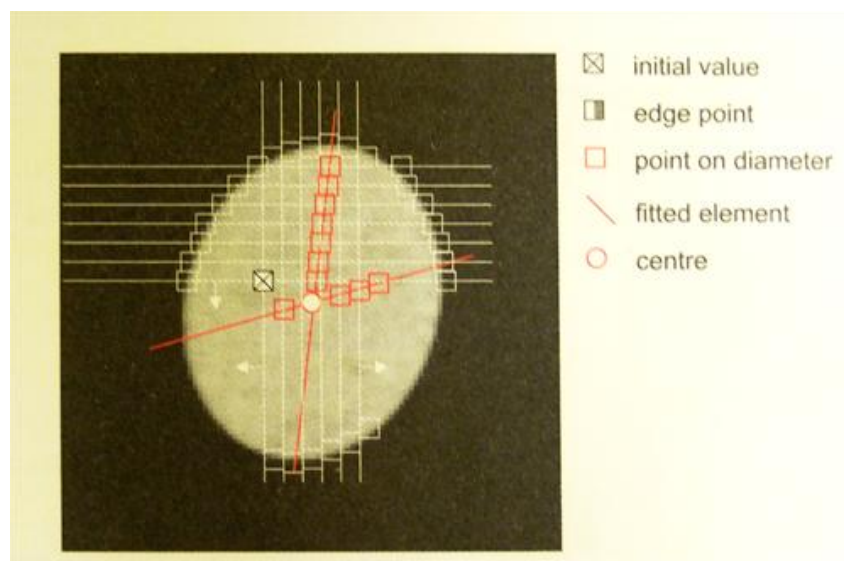


Figure 7-1: Principle of ellipse operator (the Zhou ellipse operator) (Luhmann et al., 2007).

Constrained by the smart phone hardware specifications, the smart phone devices in the current market cannot perform too complex iterative computations. Therefore, the precision of using conventional image processing algorithms for target localisation may not as accurate as performing the same procedures on a personal computer. In considering the image quality of smart phone cameras is relatively low than professional DSLR cameras, the blurred edge of targets may also decrease the precision of target centre identification (however, this disadvantage can be improved via the use of proper target sizes). For this reason, a smart phone platform-dedicated target detection application was developed in this research to compensate the inadequacy of smart phone hardware.

With the development of mobile phone technologies, the imaging sensors and mobile phone hardware specifications are constantly improving. The improvement of powerful CPU and abundant memory will allow operators to perform more complex algorithms on a smart phone platform. Therefore, the conventional imaging processing algorithms can therefore be completely converted to a smart phone (do not need to be simplified). Under these circumstances, the GHT algorithms can be smoothly performed on a smart phone to provide precise target identification. Based on these conditions, some image processing operators can be adopted to improve the

performance of target detection for GHT. Figure 7-2 shows the empirical relationship between target size and achievable point accuracy whilst the typical application areas for different operators are outlined. These methods are the most commonly used for detection of ellipse centres. According to the use of different target sizes, the operators can chose a proper method to convert it into the GHT algorithm.

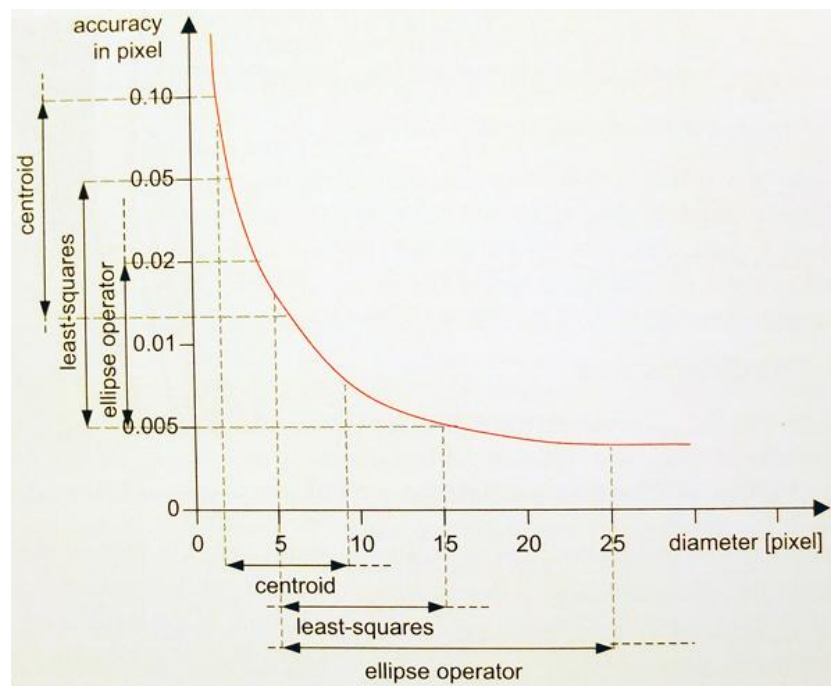


Figure 7-2: The relationship between different point measurement operators and the measurement accuracy of target centre detection (Luhmann et al., 2007).

7.4.2 OS considerations

In order to keep improving the performance of the developed system for more complex photogrammetric applications, the system should be transferred to a more powerful handset device. Therefore, the system needs to be modified to adapt to the new operating system. The low-cost, mobile phone-based, photogrammetric software package was developed and based on the platform of the Android OS V2.2. Although the Android OS provides wide compatibility for the majority of Android-based apps, the applications still needed a slight adjustment between different versions. Normally, incompatibility between different handset models and OS versions happens due to the different screen resolutions and the integration of the built-in custom software. Unlike

the Apple iPhone, Android phones are not exclusively produced by Google Inc. The third-party manufacturers such as the HTC Corporation, Samsung Group, LG Corporation, Motorola Inc., and etc. produce a variety of Android-based handset models for customers every season. Each handset model has its own hard specification (for example, each mobile phone model may have a different screen size, camera resolution, imaging sensor size, CPU, internal memory size, etc.); also, most of these mobile phone manufacturers apply their customized graphical user interface (Custom GUI) on their handset products to optimise the original Android operating interface (for example, the HTC-Sense, Samsung-TouchWiz, Motorola-MotoBlur, etc.). These differences between modules make the integration more difficult than the Apple iPhone⁵. If the self-programmed software package is applied to different Android phones, which are made by different manufacturing companies, or different Android OS versions, the operator needs to adjust the differences of hardware specifications in the software. Also, the format of the software package needs to correspond to the customised graphical user interface of the new mobile phone.

In terms of the differences of screen sizes, each mobile phone application has its own default display format. If the mobile phone's screen size doesn't fit the software's default setting, then the first problem will be the display issue. The use of smaller screens causes an obstruction in the display as the shortage of space won't be able to show the complete software interface. On the contrary, if the screen size is bigger than the programs setting then the operating window cannot fill the entire screen.

If the programmers need to change the default setup, manually allocate the system resources and memory space, or have the background programs to attain privileged control, the mobile phone has to perform a 'root access'. The root access is a process that aims to unlock the software limitations of an Android platform (the implementation of root access is analogous to running the "jailbreak" procedure on the Apple iOS operating system). The implementation of the rooting process releases the full administrator permissions to programmers. However, the rooting access is an unofficial recognition procedure, the mobile phone will lose the factory warranty if it

⁵ In terms of the Apple iPhone, each generation has only one hardware specification.

has performed the rooting procedure. In addition, performing the rooting procedure incorrectly may crash the handset OS and switch the mobile phone straight into protection mode (i.e. the mobile phone is locked and under such circumstances, the operator needs to recover the device back to the original default setting or seek out the manufacturer for technical support when the recovery does not take effect).

Instead of performing the root access to obtain the full administrator permissions of the OS, using the debugging tool in Android SDK to gradually revise the application errors is another slow but effective alternative. Though the app can be developed and tested on the dedicated Android emulator, the actual operating performance on the mobile phone is unpredictable because of the details of the customized GUI settings are unknown. The solution is to synchronise the mobile phone with the Android SDK interface and perform the app on the mobile phone directly to discover the system errors and let the error messages be sent back to the Android emulator for further debugging (as shown in Figure 7-3). The repeated debugging process helps the programmer to adjust the optimal memory allocation.

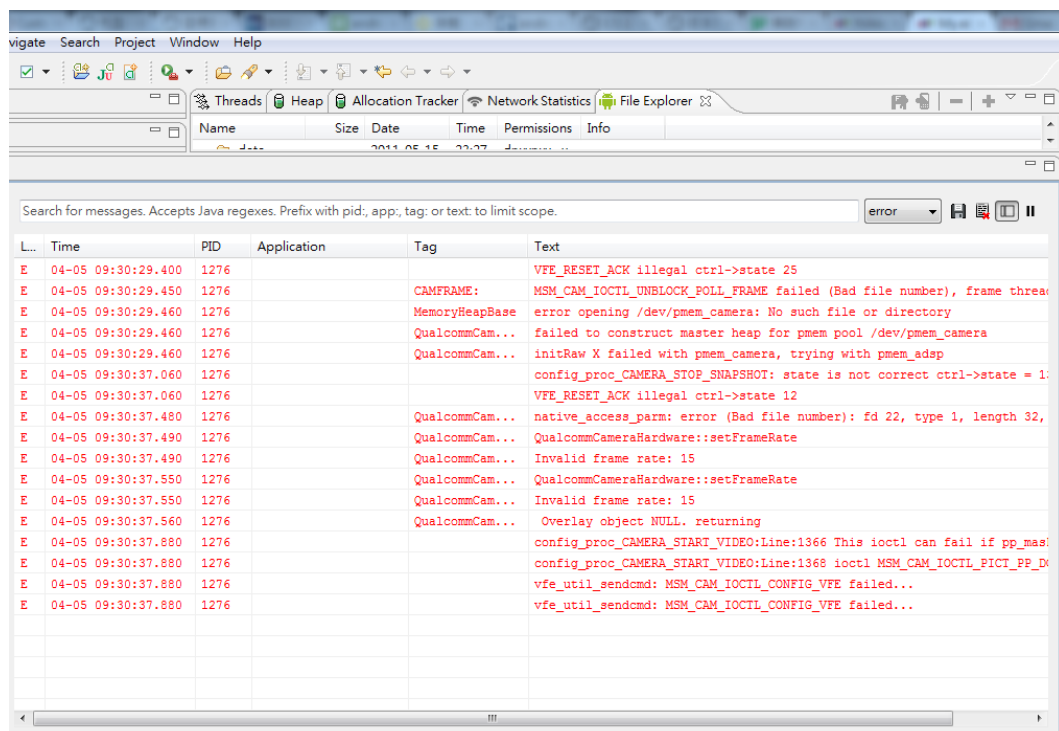


Figure 7-3: Illustration of error detection module.

7.4.2 The development of a stable power supply for long-term monitoring purposes

Modern structural monitoring projects use a large number of electronic instruments (Wu et al., 2011). For a long-term monitoring task, the power issue has long been a major concern (Kottapalli et al., 2003). The power supplies for the electronic devices which were involved in this system are the same as those used for normal household electrical appliances (110V~240V). Therefore, mobile phones and laptops can easily be charged with 220 AC volt chargers (in the UK) or other power supplies with the same voltage levels (such as an external battery pack).



Figure 7-4: Illustration of the external battery pack.

According to the previous test results, a 5,000 mAh Li-ion external battery pack (Sanyo KBC-L2, as shown in Figure 7-4) can provide approximately five-hour continuous electricity supply to a mobile phone (the HTC Incredible S). The power output of this external battery pack is steady but it does not last for too long. In order to pursue a stable and long-lasting source of electricity, a self-generating electrical power system is needed. For this reason, a solar charger may be able to deliver a reliable solution to solve the power supply issue. The easiest approach is to use the solar charger. The solar chargers for small appliances on the current market cannot effectively manage power. The operators cannot directly apply them in unattended

operations due to the charge-discharge mechanism being too simple. Once the electronic instruments are connected to the compact solar charger, the solar chargers charge the connected electronic devices directly; regardless of they need to be charged or not. In order to effectively control solar power, the use of a large capacitor and a controlling program for power management are prerequisite (Ali, 2010). Controlled by a power manager program, the solar charger can save the generated electricity in a large capacitor and distributing electricity to a mobile phone only when it is in a low battery state. Figure 7-1 shows a prototype of a solar energy generating system (SEGS) which was developed by the author.

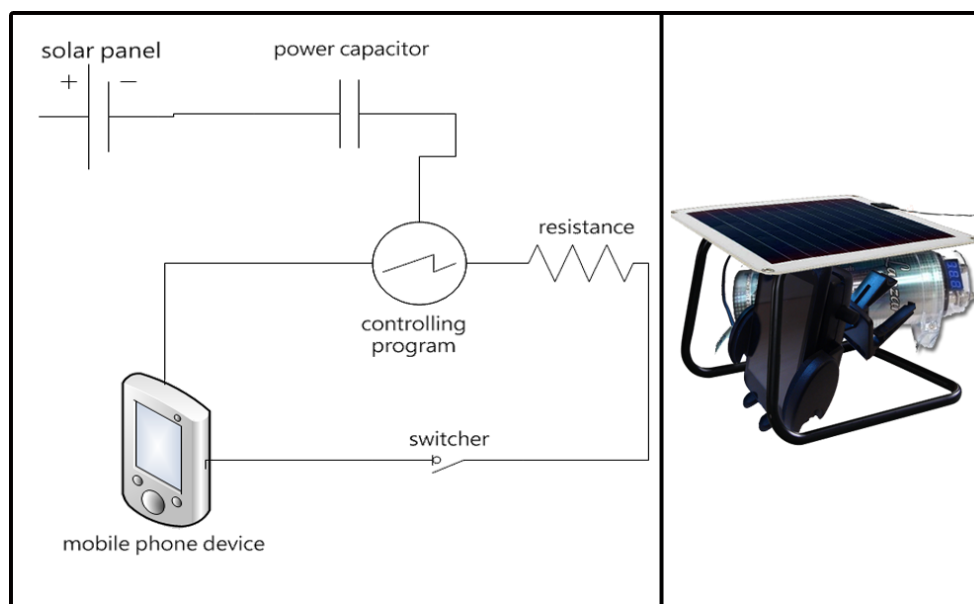


Figure 7-1: Illustration of the developed solar energy generating system.

In actual use, these imaging stations are likely to be placed outdoors. Therefore, the composition of the energy generating system should be as simple as possible and be able to take into account the function of being weatherproof. As the illustration shows in Figure 7-1, the SEGS is composed of a solar panel (17.5 cm (L) x 16 cm (W) x 0.8 cm (W)), and a 12 Volt power capacitor. The solar panel is placed on the top of a tubular frame to provide a maximum light receiving area. If the budget permits, the operator can use more solar panels to cover the other three sides of tubular frame (leaving only the front side for imaging acquisition purpose) to obtain more light-receiving areas and to avoid the phone from getting wet in bad weather.

Using this solar panel to fully charge the power capacitor takes more than six hours and the charged power capacitor can provide enough electric power to fully charge a mobile phone once. The use of a larger solar panel can certainly increase the charging speed. Furthermore, the operator can use a large-capacity power capacitor to store more backup electric power. As shown in the circuit configuration (Figure 7-1), the timing of discharge of the power capacitor is controlled by the controlling software program. The programmer can improve the system by modifying the controlling program to distribute electricity more efficient.

7.4.3 Performance improvement of the low-cost monitoring system

In terms of long-term structural monitoring applications, surveyors can compare the latest photogrammetric data with the previous results to inspect the state of health of target objects. In this research, the experimental results have demonstrated the inspection process in § 5.2.2. However, the comparison relies on manual operation. The follow-up research can focus on the development of the automated real-time early warning system. The researcher can develop a software suite to store and manage the sequential photogrammetric data, which is generated by this mobile phone-based monitoring system. When the latest data is inserted into the database, the early warning system not only archives it with the old file, but also automatically compares it with the previous data. If the comparison has shown some abnormal situations (i.e., deformation greater than the permissible value) the real-time early warning system can automatically provide appropriate warnings.

In this research, the potential of a mobile phone platform for the development of photogrammetric processing functions was studied. The ultimate aim of this module is to develop a complete photogrammetric software suite. At the current stage, this module has been able to precisely calculate the targets' 2D coordinates. In addition, the position and orientation of a mobile phone can be recorded while the mobile phone camera is capturing an image. These results have provided basic information for photogrammetric processing. With the development of mobile phone hardware specifications, the new models will feature more power CPU and larger memory space. These improvements can be applied to deal with repeated iterative calculations.

Thus, the more precise photogrammetric calculation can be achieved. On this foundation, developers will be able to transfer more complex photogrammetric calculations (such as, direct linear transformation (DLT), self-calibration bundle adjustment, Hough transform and 3D modelling etc.) to a mobile phone platform.

7.5 Concluding remarks

In terms of the implementation of long-term structural monitoring, the stability of long-term operation, the total implement costs (including the labour costs, equipment costs and maintenance fees etc.) and the measurement accuracy have long been three major concerns. To minimise the system load during the process of wireless image transmission, the image segmentation algorithm was modified based on the VQ algorithm. This modification can effectively minimise the usage of smart phone temporary memory to reduce the probability of system crashes during a long-term operation. At the current stage, the processes of image acquisition, data transmission, data sorting, system health monitoring and system recovery are now fully automated. A large amount of implement costs can be significantly reduced. The developed system health monitoring mechanism can automatically detect non-synchronised exposure epochs and system exceptions; in combination with the automatic system recovery mechanism, the probability of non-synchronised exposures occurring during a long-term operation is approximately 0.09% whilst the success rate for system recovery is up to 100 %. The system stability can therefore be maintained.

Focusing on the use of photogrammetric measurements for deformation monitoring in dynamic structures, the photogrammetric accuracy is closely correlated with the photogrammetric network geometry, image quality, camera synchronisation, etc. A change of photogrammetric network geometry results in the change of target size in the imagery and this research has proven that the use of inappropriate sized targets will decrease the precision of photogrammetric measurements. According to the use of different target patterns and target detection algorithms, the optimal range for target size was proposed in this research to assist subsequent users to determine appropriate target sizes for related photogrammetric applications. When using this low-cost photogrammetric-based structural monitoring system to monitor deformation of civil engineering structures, the synchronisation between devices can reach millisecond level. This is fast enough for each imaging stations to freeze a structure's instantaneous movements simultaneously. To improve the precision of target detection on low resolution imagery (which was captured by a mobile phone camera), a

dedicated algorithm and target patterns were developed. Combined with the use of developed target detection algorithm and dedicated 3/4 circle targets, the relative accuracy could attain 1/16,400 in an outdoor environment and the overall accuracy of this mobile platform based photogrammetric system has proven that it is qualified for long-term structural monitoring purposes.

**APPENDIX I SYSTEM OPERATION OF
MULTIPLE-STATION MODULE**

A1.1. Environment settings and system operation

Figure A-1 shows the start screen of the command program on the terminal computer. When the program is opened, the first setting that the operator has to configure is the setting that determines the quantity of imaging stations needed.

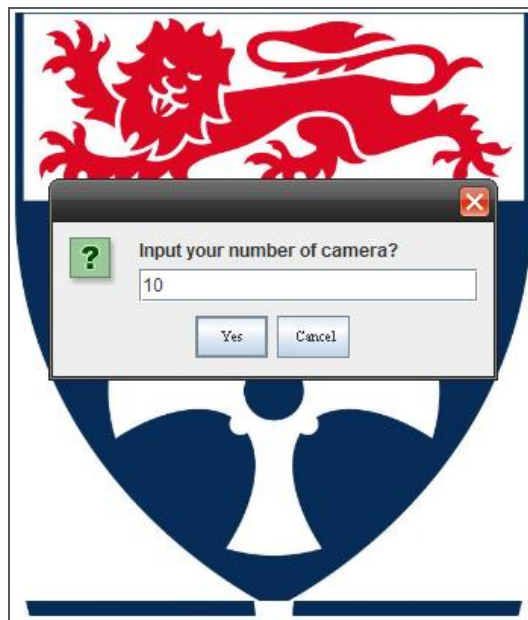


Figure A-1: Illustration of start screen.

In accordance with the input value, the program creates several control columns

(shown in Figure A-2). Each control column is assigned to manage one mobile phone station. The terminal computer locates these imaging stations by tracking the specific IP addresses of each mobile phone.

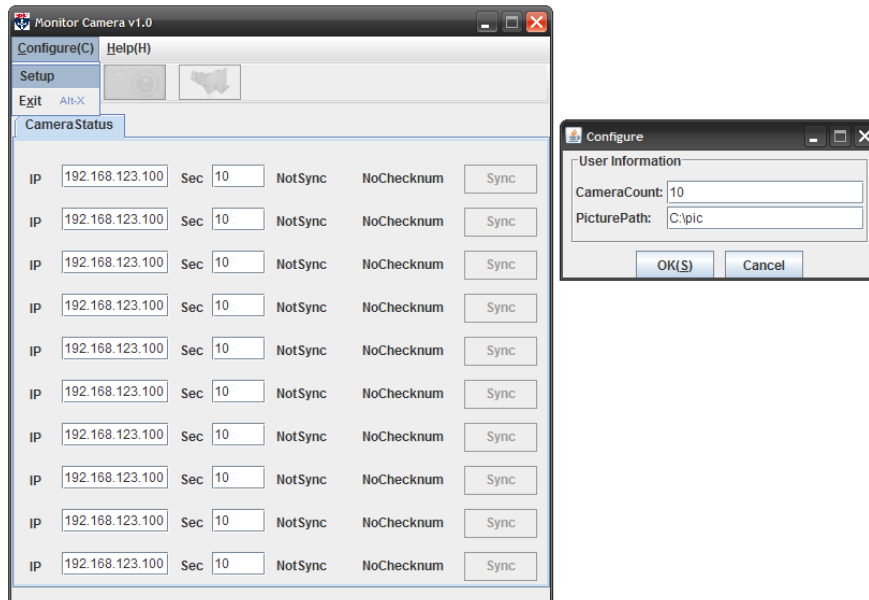
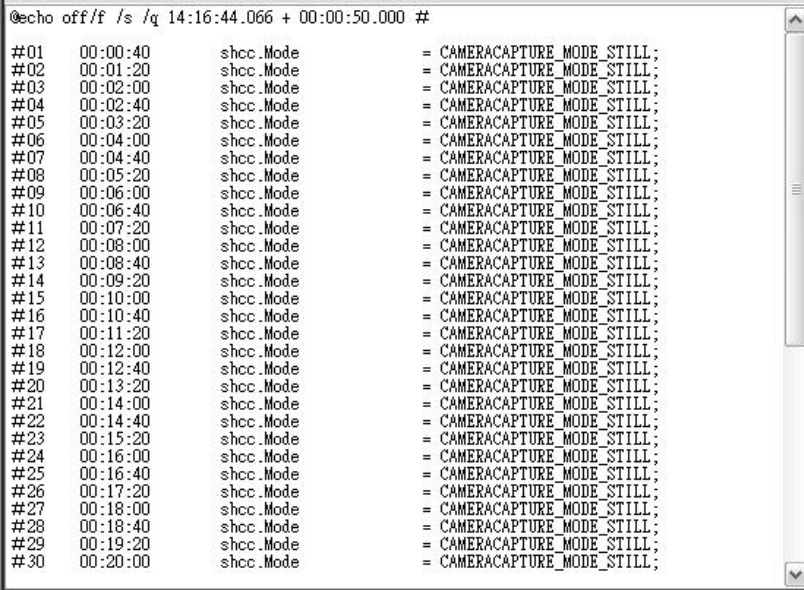


Figure A-2: The main control panel of the multiple-station module.

The operator can assign an exposure frequency to each individual mobile phone station from the terminal computer. Following the given exposure frequency, the terminal computer generates a series of scripts, sending these scripts to each camera station sequentially in order to automatically perform the synchronised image acquisition process.

Reliable synchronised image acquisition process is based on precise time synchronisation between system devices. As mentioned, the image stations synchronously capture image sequences following a series of scripts (each script includes the commands of the exact timing of each exposure, automatic system check, editing system health report and real-time system recovery etc.) that are automatically edited and are sent from the terminal computer. Also, each command is given a specific execution timestamp. Figure A-3 shows the command script. The contents of the script indicate the sequence of actions (the first column), the execution time (the second column), the instruction properties and the corresponding source code (the

third and fourth column in Figure A-3). The first row in Figure A-3 shows two different time stamps – the first one is 14:16:44.066 and the second one is 00:00:50.000. The first time stamp is the current time of the terminal computer when the script is prepared. The second time stamp means the first action in the script will be performed after fifty seconds. The value of “fifty seconds” is the default value which is considering the time spent on data transmission and decoding. The exposure interval is decided by the operator when setting up the software.



```

@echo off/f /s /q 14:16:44.066 + 00:00:50.000 #
#01 00:00:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#02 00:01:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#03 00:02:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#04 00:02:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#05 00:03:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#06 00:04:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#07 00:04:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#08 00:05:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#09 00:06:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#10 00:06:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#11 00:07:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#12 00:08:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#13 00:08:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#14 00:09:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#15 00:10:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#16 00:10:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#17 00:11:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#18 00:12:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#19 00:12:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#20 00:13:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#21 00:14:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#22 00:14:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#23 00:15:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#24 00:16:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#25 00:16:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#26 00:17:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#27 00:18:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#28 00:18:40 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#29 00:19:20 shcc.Mode = CAMERACAPTURE_MODE_STILL;
#30 00:20:00 shcc.Mode = CAMERACAPTURE_MODE_STILL;

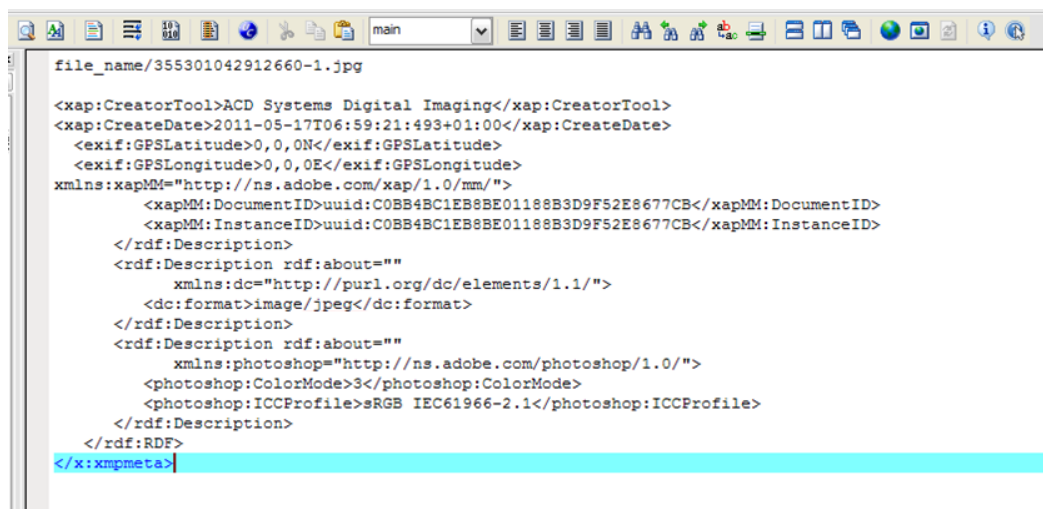
```

Figure A-3: Illustration of the command script.

Each script contains fifty directives and the computer will automatically repeat to generate another fifty-action-script with successive time sequences if the operator has not cancelled the operation. The mobile phone holds these unimplemented directives in temporary memory space. In order to preserve more memory space for system buffering, and also considering the phones processing performance, this system only uses fifty directives in a script.

The terminal computer monitors the operation on mobile phones continuously and sends a new script to each mobile phone when the old script reaches the thirtieth action. During the period when the mobile phones are processing the final twenty actions, a background program, which was developed on the handset platform, decodes the latest script and prepares it for processing once the actions of the previous

script are completed. In order to enable precise system synchronisation, the synchronisation of the internal clocks between devices is critical. Because every action is performed on mobile phones it follows the specific timeline that is recorded in the script. These mobile phone stations share the same script. If the time-of-day information of each mobile phone and the command computer is exactly the same, the mobile phones will be able to perform actions simultaneously. When the computer automatically checks the real-time status reports (Figure A-4) from each mobile phone, the failure of system synchronisation can be detected if any individual device has not performed a specific action at the scheduled time.



```
file_name/355301042912660-1.jpg

<xap:CreatorTool>ACD Systems Digital Imaging</xap:CreatorTool>
<xap:CreateDate>2011-05-17T06:59:21:493+01:00</xap:CreateDate>
  <exif:GPSLatitude>0,0,0N</exif:GPSLatitude>
  <exif:GPSLongitude>0,0,0E</exif:GPSLongitude>
  xmlns:xapMM="http://ns.adobe.com/xap/1.0/mm/"
  <xapMM:DocumentID>uuid:C0BB4BC1EB8BE01188B3D9F52E8677CB</xapMM:DocumentID>
  <xapMM:InstanceID>uuid:C0BB4BC1EB8BE01188B3D9F52E8677CB</xapMM:InstanceID>
  </rdf:Description>
  <rdf:Description rdf:about=""
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    <dc:format>image/jpeg</dc:format>
  </rdf:Description>
  <rdf:Description rdf:about=""
    xmlns:photoshop="http://ns.adobe.com/photoshop/1.0/"
    <photoshop:ColorMode>3</photoshop:ColorMode>
    <photoshop:ICCProfile>sRGB IEC61966-2.1</photoshop:ICCProfile>
  </rdf:Description>
  </rdf:RDF>
</x:xmpmeta>
```

Figure A-4: Illustration of the status report; the details of image number, image format, exposure time, ISO, shutter speed, GPS position, etc. are all listed in it.

The recovery of the system from synchronisation failure is as important as performing the system synchronisation process. The developed software provides an automatic system health detection module which includes the real-time system health monitoring function and the automated failure recovery module. When the failure recovery module is enabled, the CCP (clock correction program) will be triggered simultaneously to recalibrate the devices internal clocks.

As has been mentioned, the computer continuously monitors the status of each mobile phone station by checking the status reports which are sent from the handsets after every exposure via 3G or the Wi-Fi network. The status reports record the details of

image number, the precise time info of the exposure, shutter speed, aperture, ISO, GPS information and camera orientation. When the camera receives these reports, the auto-decode process will be undertaken in the background. The system health monitoring program extracts the information from these reports and makes comparisons. The recovery module will be triggered if the health monitoring program detects either the exposure time from each camera is not synchronised or the photographic parameters⁶ (including the shutter speed, ISO and aperture) are different. Once abnormalities have been detected, the recovery program will use the CCP to refresh the computers system clock first, whilst the CCP sends commands to every mobile phone to refresh the time information from the chosen time source. Once the clocks of all the devices are recalibrated, the computer will generate a new script using the same parameters and settings; sending the script to each camera to continue the previous task. If the computer still detects any abnormalities from any individual imaging station in the next two consecutive reports after they have been recovered, the recovery program will send a request to force every mobile phone to reboot the image capture program. After these procedures, if these cameras still cannot fire simultaneously or the shutter speeds are not the same, the system health monitoring program will present a warning window on the computer screen to notify the operator that manual intervention is required.

For a long-term photogrammetric operation, the performance of camera synchronisation has always been one of the most important issues that photogrammetrists have to pay attention to. In the research, the camera is controlled by a self-programmed interrupter program. The interrupter has been used in many fields such as mechanics, electronics, medical etc. (Slade, 2007). In electrical engineering, the interrupter is used as an electronic switch to interrupt the flow of a steady direct current (Stewart, 2004). In this mobile phone-based photogrammetric system, the interrupter program is responsible for controlling the cameras sensors input current (i.e. the interrupter program is applied as an input current limiter). The exposures of the camera sensor are precisely controlled by electronic signals

⁶ Following the default setting, the camera ISO and aperture are fixed at 200 and F 4.0 respectively. The shutter speed is decided using the parameters from a specific mobile phone of which IP address is always designated as 192.168.0.101.

following the exposure timings and exposure lengths (shutter speeds) that are recorded in the script. Figure A-5 shows a code fragment which was excerpted from the interrupter program. The forty-seventh row indicates that the actions of the interrupter are following the “clock time”. In this program, the “clock time” means the exposure timings on the script. When the mobile phone receives the script which was sent from the computer, the interrupter program records these exposure timings simultaneously and takes this information as the basis for exposures. If the internal clock of each mobile phone has been adjusted to the same standard, the cameras can therefore expose at the same time. The capability of the interrupter program directly affects the performance of camera synchronisation.

```

30
31 public final class Interrupter {
32
33     private static final AtomicLong counter = new AtomicLong(0);
34
35     private final Thread threadToInterrupt;
36     private Thread interrupterThread;
37     private Clock time = now();
38
39     private Interrupter(Thread threadToInterrupt) {
40         this.threadToInterrupt = threadToInterrupt;
41     }
42
43     public static Interrupter interrupt(Thread thread) {
44         return new Interrupter(thread);
45     }
46
47     Interrupter using(Clock time) {
48         if (interrupterThread != null)
49             throw new IllegalStateException("Controlling time after events have been put in motion will have no affect");
50         this.time = time;
51         return this;
52     }
53
54     public Interrupter after(final Duration duration) {
55         final Timeout timeout = timeout(duration, startStopWatch());
56         interrupterThread = new Thread(new Runnable() {
57             public void run() {
58                 try {
59                     waitUntil(timeout);
60                     if (!interrupterThread.isInterrupted()) {
61                         Interrupter.this.threadToInterrupt.interrupt();
62                     }
63                 } catch (InterruptedException e) {
64                     currentThread().interrupt();
65                 }
66             }
67         }, "Interrupter-Thread-" + counter.incrementAndGet());
68         interrupterThread.start();
69         return this;
70     }
71

```

Figure A-5: A source code fragment of the interrupter program.

A1.2. System synchronisation and time correction

With the characteristics of high quality camera synchronisation, structural movements can be captured simultaneously by each imaging station. Furthermore, a fast shutter speed and high sampling rate can be achieved under good illumination conditions to record more information of structural deformation during the sampling period. The features mentioned above provide this mobile phone-based photogrammetric system with superior flexibility for structure monitoring.

In terms of structural monitoring of flexible structures, in order to produce a series of high quality image pairs for photogrammetric processing, not only the resolution of imaging devices but also the reliable synchronisation between each imaging device is essential. Highly synchronised camera station networks should be able to ensure that the camera shutter of each camera can be launched simultaneously for every exposure. And under the circumstances, the images that are captured at the same exposure epoch can precisely represent the instantaneous and dynamic characteristics of the object structure. In order to achieve this objective, a series of software developments and decision making processes were developed. The development process is divided into two main parts – the time adjustment for terminal computer and each mobile phone device, respectively. The calibration procedures on these two parts can be performed at the same time if both of these two devices are able to connect to a public network. Otherwise, the system will calibrate the time information of the computer and set the computer clock as a standard to synchronise each mobile phone station. In actual operation, the network configuration for each test area is different and the system has developed several modules allowing users to correct the system clock under different conditions. The details of these processes are described below.

A1.2.1 Time adjustment on the terminal computer

In order to synchronise these electronic devices, to unify the internal clock of each individual image station is priority. When the internal clock of every instrument has been unified, all imaging stations can simultaneously perform commands that are sent from the terminal computer following the specific timeline. For this reason, a stable, high precision and interoperable time standard is necessary.

The self-programmed software provides a fully functional procedure for time adjustment based on the NTP. The first step to synchronise the terminal computer with the mobile phone stations is to synchronise the system clock of the terminal computer with a remote NTP server. Figure A-6 shows the start-up screen for the time adjustment module. This module is modified form an open source code that can be downloaded from the official NTP website. And this module has been integrated into the developed software which was developed in this research. Once the operator

finishes the initial setting at first use, the program will automatically perform in the background until the user needs to change any option. By using this module, a user can adjust the clock manually or synchronise the computer clock with NTP.

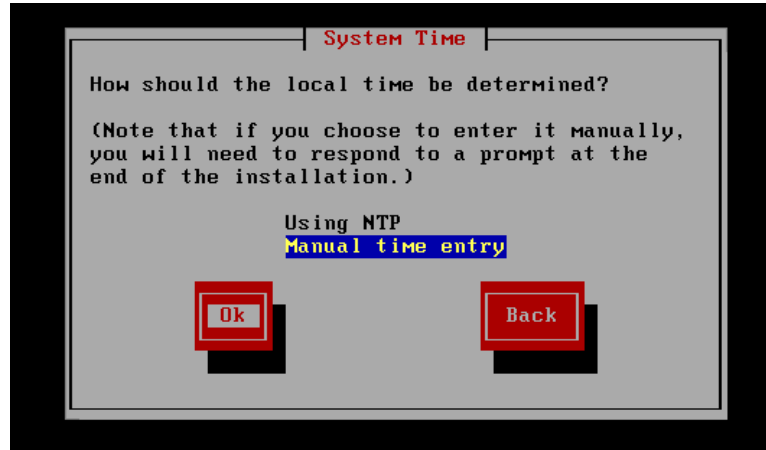


Figure A-6: Setting the computer clock by the use of the self-programmed time adjustment module.

When the user chooses to synchronise the computer's clock with NTP, the background program can automatically extract related information from the ISP (Internet Service Provider) to define the located time zone of the client computer. The information from ISP includes the DNS (Domain Name System), IP information and some internet services. Following this information, the background program can determine the located time zone and will present a confirmation window as shown in Figure A-6.

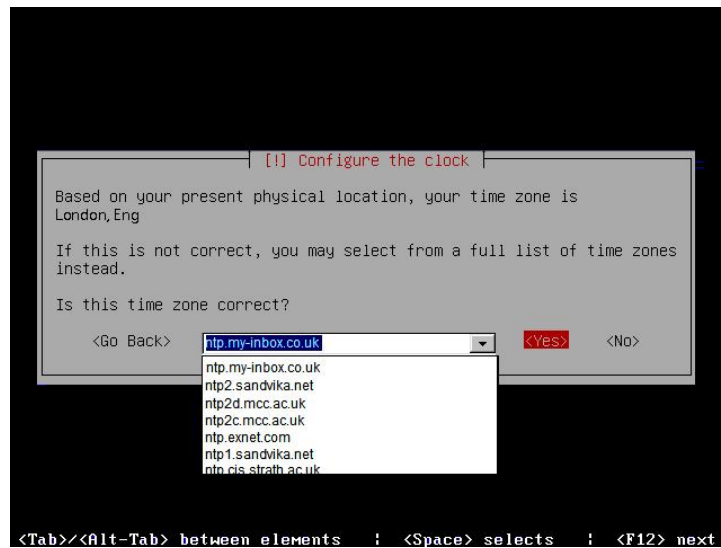


Figure A-6: The confirmation window for time zone setting.

The operator can inspect the answer for further confirmation whether the time zone is correct or not. If the listed time zone is incorrect then the “No” option is selected to open another window that allowing the user to manually insert the IP address of a proper NTP server. Alternatively, the operator can select a server from a the drop-down list and press “Yes” to connect to the chosen NTP server (Figure A-7) if the listed time zone is correct. The program will show a quality report of the server connection on the screen till the connection process is complete. By analysing the results from the final report, the user can decide to rerun the process with another NTP server or not.

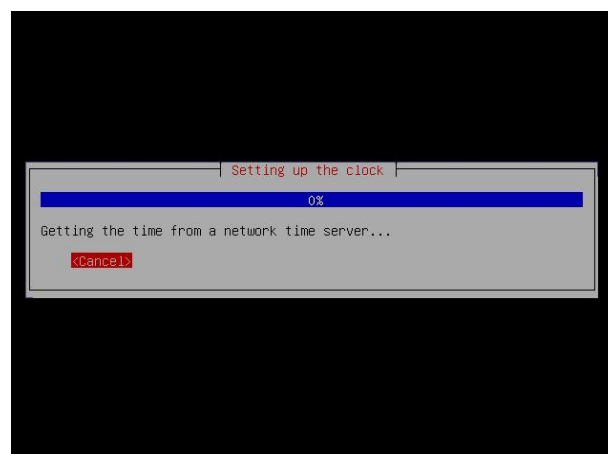


Figure A-7: NTP server connection.

The final setting of the time correction on the terminal computer is to insert an update frequency for continuous updating of the time information from the NTP server. Following the update frequency, the background program refreshes the time information of the computer periodically and automatically to ensure the internal clock always synchronises with NTP. In actual operation, the internal clock of electronic products gain or lose about 5 to 15 seconds per day (Lombardi, 2000). There are 86,400 seconds in a day. Assume that the time lag of electronic devices achieve the maximum of fifteen seconds per day. Come down on average, in every second these electronic devices generate 0.00017 second time error, which means those devices cause 0.001 second time lag within every six seconds. The precision of the system can display up to millisecond level. If the update frequency is less than six seconds, the system should not be able to detect the time lag. However, 6-second update-frequency is too intensive to the system as it consumes too much network usage and cause heavy computing load to mobile phone platform. Therefore, the default value for the update frequency is set as 30 seconds. The operator can check the quality of clock-synchronisation from the system report. If the CCP interrupts the operation too frequently, this means the system needs a faster update frequency to maintain the synchronisation between devices. Then, the operator can re-adjust the parameter manually from the terminal computer to improve the system performance.

A1.2.2 Time adjustment on the mobile phone platform

The next procedure for clock synchronisation is to synchronise the mobile phones with the same NTP server and the terminal computer. By sharing the same time information from the same NTP server, the mobile phones and the terminal computer can therefore synchronously perform the monitoring commands and system adjustment based on the same time signal.

The low-cost monitoring system which was developed in this research allows users to remotely perform structural monitoring tasks via a terminal computer. In terms of practical application, the terminal computer does not have to be located in the same test field with the mobile phone stations. Depending on whether the terminal computer and the mobile phone stations are arranged in the same area (or time zone)

or not, there are two options to synchronise the mobile phone stations with the NTP. The first method is to operate the connecting process on each mobile phone individually. The operator can manually choose the NTP server on each mobile phone using a self-programmed, mobile phone-based NTP connecting program. Alternatively, the operator remote controls the NTP server connection of mobile phone stations from the terminal computer. These applications are outlined in the following section.

If the terminal computer and the mobile phones are not arranged in the same test field, the use of the same NTP server may not be a good option due to the difference of the network quality and the connectivity to a specific NTP server from difference areas. Under these circumstances, within the developed software there exists a time adjustment module on the mobile phone. By applying this module, the program automatic updates the time information from the designated NTP server. Figure A-8 illustrates the setting screen of the mobile phone-based, time adjustment program. Similar with the adjustment program on the PC platform (the Microsoft Windows XP OS), the mobile phone program also provides a drop-down NTP server list. Also, these NTP servers are covered by the time zone where the mobile phone is located. The mobile phone offers three different methods to define the time zone. If the mobile phone has equipped with a SIM card, the location of the mobile phone can be triangulated by the telecommunications base stations and the GPS satellites (Figure A-9) using a technique named as Assisted GPS (A-GPS), and the appropriate time zone can be defined based on the location of the mobile phone. If the mobile phone doesn't have a SIM card, the user can use the mobile phone built-in GPS sensor to calculate the location and time zone. Otherwise, the mobile phone has to connect to the Internet and access the World Wide Web, then using the information from ISP to determine the location and time zone.

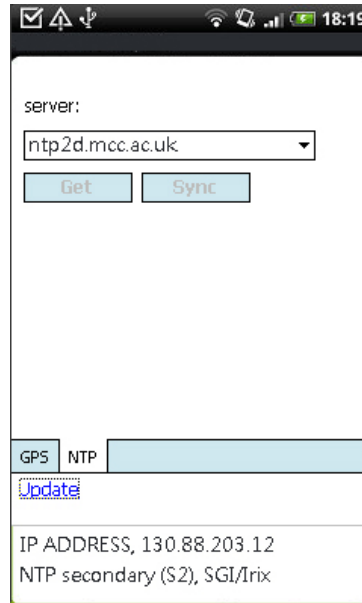


Figure A-8: The mobile phone-based time adjustment program.

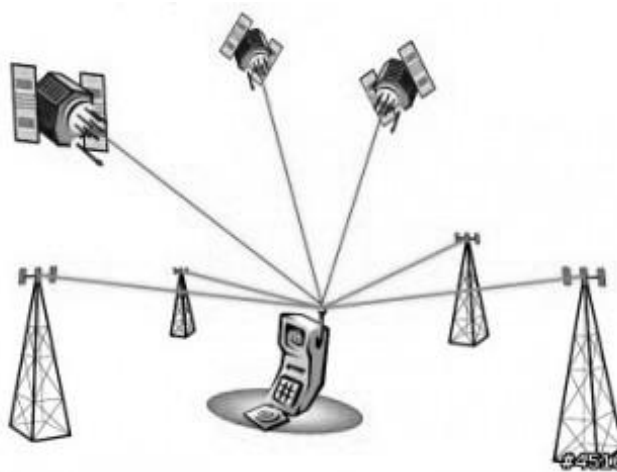


Figure A-9: Illustration of A-GPS (Diggelen, 2009).

The operations mentioned above present the procedures for system clock correction, and those methods are applied under the circumstances when one of the devices can connect to a public network to synchronise the internal clock with the chosen server. A special operating module to adjust the mobile phone's internal clock was developed on a PC platform. This module is applied in the test field where has no public networks allowing the mobile phone to access the World Wide Web, whilst the GPS signal is blocked and the mobile phone does not insert a SIM card into it. Under such circumstances, mobile phones will not be able to connect to the public network to

synchronise with any NTP server or the remote terminal computer. The alternative is to create a local closed network which is purely provides the connectivity between the terminal computer and mobile phones for device synchronisation and data transmission. This local closed network is a restricted network as it cannot connect to the Internet. Therefore, the terminal computer has to be arranged near the mobile phone-based imaging stations and all of these devices have to be covered by the regional closed network. In order to establish a local area network (LAN), the terminal computer is connected with a wireless router to create a wireless LAN (WLAN). The closed network allows Wi-Fi devices to connect to each other under the coverage of the wireless router's signal.

Following the WLAN configuration, there are two different approaches to adjust the internal clocks of these mobile phones. The method of choice is depending on whether the terminal computer can access the Internet or not. A terminal computer with the capability to connect to the Internet allows correction to NTP servers for time correction (Figure A-10 (a)). In this situation, the command computer can periodically refresh its' system clock form a chosen NTP server and deliver the latest time information to each mobile phone camera instantaneously via the regional closed network. The whole process relies on comprehensive control by an automatic time correction module form the developed software. Before the beginning of the image capture process, the operator must perform a simple setup on terminal computer platform. The operator has to select a proper NTP server for connection and insert the refresh rate (update frequency). The only difference between these two modules is that the operator needs to choose to adjust the internal clock of each mobile phone from the terminal computer at the final step (Figure A-12).

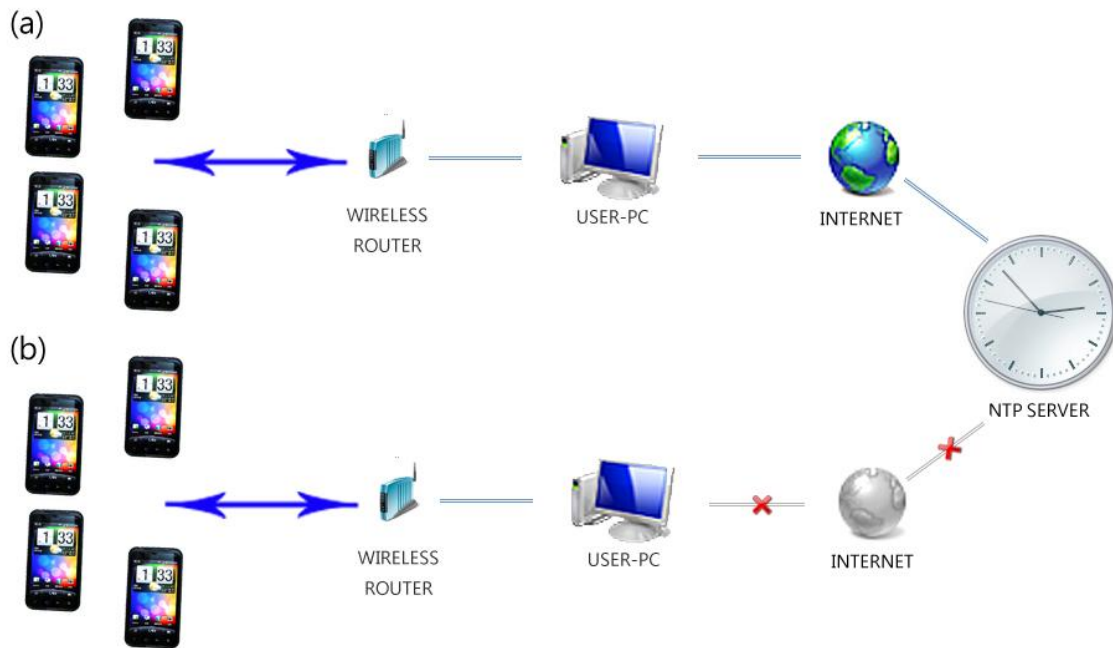


Figure A-10: Two methods to adjust the internal clock of each mobile phone.

Once the parameters have been inserted, the automatic clock correction program (hereinafter referred to as CCP) will start to synchronise each handset's system clock. The procedure uses the traditional "two-way message exchange" to detect the clock drift and calculate the offset between devices for time correction. Firstly, the CCP sends an auto-response message to each mobile phone. As the handset receives the message, an auto-execute Java script will be triggered and will reply to the terminal computer immediately. The return signal is read and interpreted by the computer. The computer calculates the round-trip time between the computer itself and each mobile phone (the computer distinguishes them by different IP addresses) and counting the transmission offset to each mobile phone (Figure A-11) (assume the transmission speed and round-trip-time are consistent and the same, the transmission offset is defined as the mean of the round-trip time) (Kyoung-Lae et al., 2007).

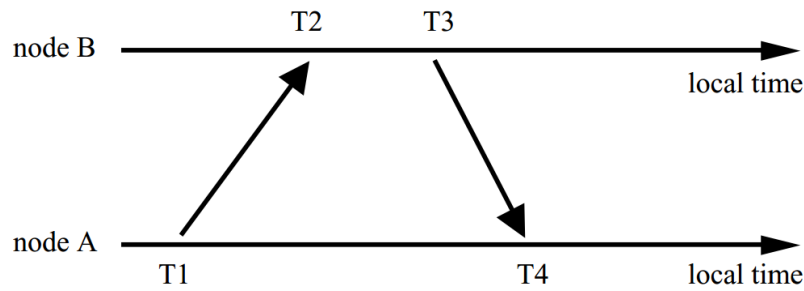


Figure A-11: Two way message exchange between two nodes (Sivrikaya and Yener, 2004).

If the local network quality is not stable or there are too many barriers existing between the handset devices and the computer as they may reduce the reception quality, the operator can repeat the offset calculation process for obtaining more samples to evaluate a more reliable offset value. Afterward the self-programmed software package on the terminal computer automatically generates scripts and sending to each mobile phone following different IP addresses. The unique script contains Java codes, the calculated offset and the time information of both the computer and the individual handset device. Each mobile phone has a different calculated offset and script due to the difference of transmission speed and round-trip-time. The transmission offset reveals the extent of time delay between the computer and the specific mobile phone station. According to the offset, each mobile phone will automatically adjust its internal clock to compensate the time delay. Once the process has been completed the program will be hidden in the background and will automatically rerun the whole process following the refresh rate that was inserted initially. Anytime when the computer updates the time information from the NTP, the mobile phones refresh their internal clock simultaneously. By the use of this algorithm operate as in the client/server modes, the resulting accuracy usually can be maintained at a few milliseconds on an Ethernet (Mills, 1995).

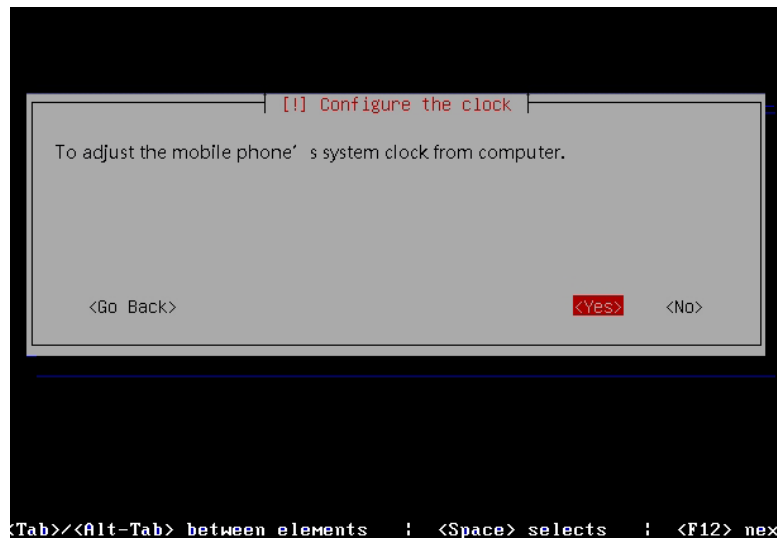


Figure A-12: The confirmation of the standard for synchronisation.

In addition to the above operating configuration, there is still a possibility that both of mobile phones and the terminal computer cannot connect to the internet (such as the illustration shown in Figure A-10 (b)). Under such circumstances, the computer will not be able to connect to any one of the NTP servers for time correction. Then the system therefore must use the terminal computer's system clock as an alternative standard to synchronise the mobile phones' clocks. The operation is similar to the method introduced in the previous paragraph. Instead of refreshing the computer's system clock from any NTP server, the operator need to choose "Manual time entry" (Figure A-6) at the beginning when using the time adjustment module to adjust the computer's system clock. Afterwards a setup window displays on the screen which allows the user to manually adjust the computer's system clock. The final step of system clock configuration is to confirm the setting using the computer's system clock as a standard time (Figure A-13). Till the manual input processes are finished, the CCP will be triggered to automatically calculate the transmission offsets from the computer to each handset and editing scripts for each camera station. Following the scripts, every mobile phone can synchronise the internal clock with the computer.



Figure A-13: To confirm the mobile phones are going to use the computer's time information as the standard for time adjustments.

By updating the time information from the NTP via the terminal computer or refreshing the mobile phone's internal clock directly using the terminal computer's time information, the performance and the quality of time correction are unpredictable. The performance is dependent on the quality and the stability of a local closed network. The speed and reliability of the computer to send and receive data packets determines the stability of the Internet connection (Stallings, 2006). To perform the clock synchronisation process under a stable network, the calculated round-trip time and the transmission offset are consistent and reliable. The use of a reliable transmission offset for processing leads to higher quality of time correction and synchronisation between devices. On the contrary, the transmission offset will become uncertain if the network quality is poor or the receiving speed of any individual mobile phone is unstable.

The other problem which may affect the reliability of the processing is the clock drift issue on the personal computer. Without the real-time, high precision time-of-day information from the NTP server to refresh the electronic devices' system clock, the alternative solution is to use the terminal computer's system clock as a standard. The system clock of a computer can be divided into two categories – the software and hardware clocks (Berger, 2005). The software clock is programmed into a

programmable interval timer (PIT) and this PIT is embedded in a larger chip to perform timing and counting functions. There are two drawbacks of a software clock which decrease the precision. First of all, if there is any operation need to involve the function of the PIT, the interrupt request causes the clock to gain or lose time. Furthermore, the software clock shuts down when the computer is turned off and all of its time-of-day information will be deleted. Till the computer is turned back on, the software clock starts running again and using the time information from the hardware clock for initialisation. Therefore, the software clock cannot be operated independently. In terms of the hardware clock in a computer, the clock signal is created by the continuous mechanical resonance of a vibrating crystal oscillator. However, the timing uncertainty of the crystal oscillator is affected by its own manufacturing quality, the given tolerance, the configuration of peripheral component, and the operating temperature. In actual operation, the hardware clocks gain or lose about 5 to 15 seconds per day (Lombardi, 2000). These disadvantages cause irregular clock drift and affect the time synchronisation error directly.

Without the precise correction from the NTP servers, the time synchronisation within electronic devices is difficult to maintain to a high standard (the millisecond level). The alternative of using the terminal computer's time information as a standard can provide an exclusive standard for other electronic devices to follow, but this standard is difficult to maintain during long-term operation as it suffers from clock drift. In order to keep the system in stable and precise time synchronisation status, connecting to a stable and precise third-party source (such as the NTP servers) for real-time updating of the internal clock is essential. The terminal computer's time information should only be used as a standard if there is not any network access point (NAP) allowing users to access the Internet service providers (ISPs) for time correction.

Once the user has completed the time correction and the clock synchronisation procedures between the computer and mobile phones, all the electronic devices of the system are following the same timeline. By applying the same time sequence in tandem with the highly quality system synchronisation, the orders and commands which are sent from the computer can therefore be performed on each mobile phone synchronously.

APPENDIX II THE GENERALISED

HOUGH TRANSFORM (GHT)

A2.1. GHT for target detection

The original Hough transform algorithm is focussed on the identification of lines in the image (Zhipeng et al., 2011). After years of development, the improved Hough transform (also known as the generalised Hough transform (GHT)) has been extended to identifying positions of arbitrary shapes, most commonly circles or ellipses (Goneid et al., 1997).

The use of the Hough transform for target detection is to calculate the possible trajectory of the reference point in the parameter space and to filter out the peak from the strongest imagery signals. Therefore, the results are not affected by discontinuity edge patterns (Xiangqi et al., 2010); that means even if the object detection program does not determine a complete object pattern, the Hough transform can still restore the edge features in accordance with the point to point correlations (Hinton and Boyce, 1993).

In order to detect the irregular shape which cannot be parameterized (without an analytical model), Ballard (1981) first proposed the generalised Hough transform in 1981 and used the equation (5.1) to define a generalised shape.

$$a = \{y, s, \theta\} \tag{5.1}$$

$$y = (xr, yr) \tag{5.2}$$

$$s = (sx, sy) \tag{5.3}$$

Where:

The origin of the generalised shape is represented by “y”, “ θ ” is orientation of the generalised shape and “s” are two scale factors of x-axis and y-axis.

The equation (5.1) includes five parameters. In order to simplify the calculation, the parameter “s” can be regarded as a scalar whilst the equation (5.1) is turned into a four-dimensional equation. This generalised shape can also be described as the following equations.

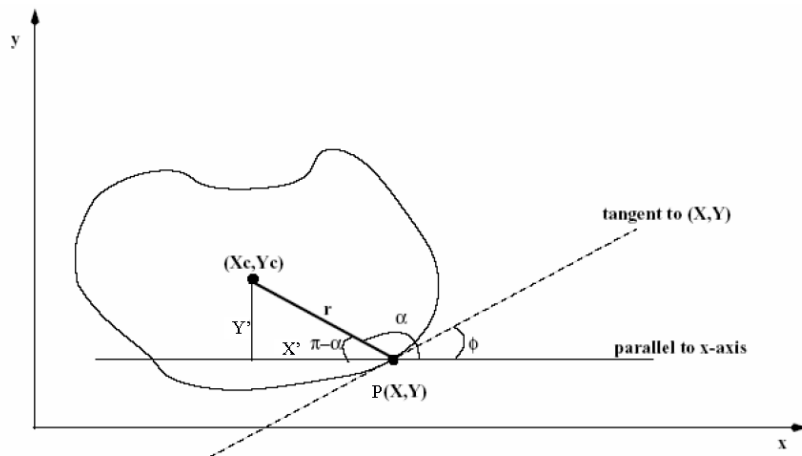


Figure A-14: Illustration of a generalised shape; where (X_C, Y_C) and r denote the two coordinates of the centre of gravity and the distance between the edge point P to the centre of gravity (if the generalised shape is a circle, then the parameter “ r ” can be represented as the radius) (Leavers, 1992). The angle α is the gradient direction.

In Figure A-14, the point P could be any point which is located onto the edge of the generalised shape. The gradient of the position along the x direction makes an angle (Φ) with the x -axis. And the centre of gravity of the generalised shape is represented by equation (5.4) and (5.5) (Lee et al., 1992).

$$X_C = X + r \cos(\alpha) \tag{5.4}$$

$$Y_C = Y + r \sin(\alpha) \quad (5.5)$$

Also

$$r = \text{sqr}t\left[(X - X_C)^2 + (Y - Y_C)^2\right] \quad (5.6)$$

$$\alpha = \tan^{-1}\left[\frac{(Y - Y_C)}{(X - X_C)}\right] \quad (5.7)$$

The edge information has been detected by a developed program which is based on the Canny edge detection algorithm. The edge information is provided for this GHT calculation. By applying these data, the point P becomes a known quantity. Also, the critical values (which means the maximum and minimum values of the parameter space) of the pattern boundary are known quantities. Each point on the edge corresponds to the location of the two-dimensional parameter space and can be defined as a one-dimensional matrix as shown in equation (5.8).

$$P(X, Y) = [(X_C, Y_C), r] \quad (5.8)$$

In order to define a closed curve, an R-table is necessary (Figure A-15). The use of the GHT algorithm for target edge detection relies on repeated calculations to approach the true value of the target edge. The repeated calculations are using an accumulator which uses the R-table as an index table to perform the calculation. In a parameter space, different combination of the Euclidean distance r and gradient direction α may result in the same Φ value and these possible combinations are listed in the R-table. When dealing with the edge detection operation, the program randomly designates a reference point (e.g., (X_P, Y_P)) and this reference point is located within the maximum and minimum critical values. Next, the program uses this reference point to process an R-table by the use of equation (5.6) and (5.7) (Ji et al., 2011). During the calculation process, the temporary edge information is using the results from the Canny edge detection.

ϕ_1	$(r_{11}, \alpha_{11}), (r_{12}, \alpha_{12}), \dots, (r_{1n}, \alpha_{1n})$
ϕ_2	$(r_{21}, \alpha_{21}), (r_{22}, \alpha_{22}), \dots, (r_{2n}, \alpha_{2n})$
ϕ_3	$(r_{31}, \alpha_{31}), (r_{32}, \alpha_{32}), \dots, (r_{3n}, \alpha_{3n})$
...
ϕ_k	$(r_{k1}, \alpha_{k1}), (r_{k2}, \alpha_{k2}), \dots, (r_{kn}, \alpha_{kn})$

Figure A-15: R-table (Ji et al., 2011).

When the R-table has been established, the program uses the R-table as an index table to retrieve the edge information. At this stage, the reference point has been adjusted to (X_{P+1}, Y_{P+1}) by the accumulator. When the process is finished, the program will repeat the process again using the (X_{P+2}, Y_{P+2}) as a new reference point. This process will be continued until the points which are surrounded inside the graphic range have been calculated. By evaluating the statistical results, the local maxima in each edge point will be defined as the best approximation. Once the edge point has been decided, the program will use equation (5.4) and (5.5) combined with the R-table to compute the centre of the target. And then, the GHT target detection process can be finished.

APPENDIX III

SOURCE CODE

A 3.1. Main program

```
package com.camera;

import java.awt.*;
import java.util.*;
import java.awt.event.*;
import java.io.BufferedInputStream;
import java.io.BufferedOutputStream;
import java.io.DataOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.net.InetAddress;
import java.net.ServerSocket;
import java.net.Socket;
import javax.swing.*;

public class Main extends JFrame
{

    int CameraNum = 1;

    private Main my = this;

    static int CameraTakePictureNumber = 0;
    //ProfileDialog pd = new ProfileDialog();
    static SplashScreen splasher = new SplashScreen();
    private JFrame frame;
    static public String IMEI;
    private int count = 0;
```

```
private boolean [] syncT;
public String root_path = "/home/shulong/camera/";
private int CTakePicture;

//TOP
protected JMenu jMenuItemFile = null;
protected JMenu jMenuItemHelp = null;
protected JMenuBar jMenuItemBar = null;
protected JMenuItem jMenuItemOption = null;
protected JMenuItem jMenuItemSetup = null;
protected JMenuItem jMenuItemExit = null;

//Middle
protected JPanel jPanelMiddle = null;
protected JPanel jPanelToolBarBar = null;
protected JPanel jPanelToolBarBarDetail = null;

protected JToolBar jToolBarBar = null;

protected static JLabel jLabelCameraTotal = null;
protected static JLabel jLabelCameraStatus = null;

protected JButton jButtonSync = null;
protected JButton jButtonTakePicture = null;
protected JButton jButtonSyncData = null;

//Bottom
protected JTabbedPane jTabbedPane = null;
protected JPanel jPanelCameraTabContext = null;
protected JPanel jPanelCameraTabContextDetail = null;
protected JPanel jPanelInCameraTab = null;
private JTextField[] jTextF;
private JLabel[] jSpLabel;
protected JLabel[] jLabelSyunStatus;
protected JLabel[] jLabelCheckSumStatus;
protected JButton[] jbn;
protected MonitorCameraSocket[] newSocket;
protected int[] CameraStatus;
protected boolean[] TakePictureStatus;

protected JLabel jLabelReceiver = null;
protected JLabel jLabelSubject = null;
protected JPanel jPanel3 = null;
protected JPanel jPanel1 = null;
protected JScrollPane jScrollPane = null;

protected JMenuItem jMenuItemAbout = null;
```

```
protected static JTextField jTextFieldSender = null;
static JTextField jTextFieldReceiveDate = null;
static JTextField jTextFieldReceiveSubject = null;
protected static JTextField jTextFieldReceiveSender = null;
protected static JTextArea jTextArea1 = null;
private JPanel jContentPane = null;
private JTabbedPane jTabbedPane1 = null;
private JPanel jPanel9 = null;
private JTextField jTextField = null;
private JTextField jTextField1 = null;
private JTextField jTextField2 = null;
private JTextField jTextField3 = null;
private JTextField jTextField4 = null;
private JTextField jTextField5 = null;
private JTextField jTextField6 = null;
private JDialog jDialogAbout = null;
private JPanel jContentPane1 = null;
private JLabel jLabelName1 = null;
private JLabel jLabelName2 = null;
private JLabel jLabelAddress1 = null;
private JLabel jLabelAddress2 = null;
protected JButton jButtonArrowLeft = null;
protected JButton jButtonArrowRight = null;
protected JLabel jLabelStatus = null;

public Main(int num, String path)
{
    super();
    CameraNum = num;
    root_path = path;

    initialize();
    splasher.hide();
}

public void updateUI() {
    SwingUtilities.updateComponentTreeUI(this);
}

public void initialize()
{
    int _programNum = CameraNum;

    JTextF = new JTextField[_programNum];
    JSpLabel = new JLabel[_programNum];
    jLabelSyunStatus = new JLabel[_programNum];
    jLabelCheckSumStatus = new JLabel[_programNum];
}
```

```
jbn = new JButton[_programNum];
newSocket = new MonitorCameraSocket[_programNum];
CameraStatus = new int[_programNum];
TakePictureStatus = new boolean[_programNum];
syncT = new boolean[_programNum];

for (int i=0; i<CameraNum; i++)
{
    syncT[i] = false;
}

this.setSize(508, 558);
this.setContentPane(getJPanelMiddle());
this.setJMenuBar(getJMenuBar());
this.setTitle("MonitorCamera v1.0");
this.setResizable(false);

try
{
    ClassLoader cl = this.getClass().getClassLoader();
    ImageIcon image1 = new ImageIcon("images/LOGO.jpg");
    setIconImage(image1.getImage());
}
catch(Exception x)
{
}
}

public JMenuBar getJMenuBar() {
    if (jMenuBar == null) {
        jMenuBar = new JMenuBar();
        jMenuBar.add(getJMenuFile());
        jMenuBar.add(getJMenuHelp());
    }
    return jMenuBar;
}

public JMenu getJMenuFile() {
    if (jMenuFile == null) {
        jMenuFile = new JMenu();
        jMenuFile.add(getJMenuItemSetup());
        jMenuFile.add(getJMenuItemExit());
        jMenuFile.setText("Configure(C)");
        jMenuFile.setMnemonic(java.awt.event.KeyEvent.VK_C);
    }
    return jMenuFile;
}
```

```

public JMenu getMenuHelp() {
    if (jMenuHelp == null) {
        jMenuHelp = new JMenu();
        jMenuHelp.add(getJMenuItemAbout());
        jMenuHelp.setText("Help(H)");
        jMenuHelp.setMnemonic(java.awt.event.KeyEvent.VK_H);
    }
    return jMenuHelp;
}

public JMenuItem getMenuItemSetup() {
    if (jMenuItemOption == null) {
        JMenuItemOption = new JMenuItem();
        JMenuItemOption.setText("Setup");
        JMenuItemOption
            .addActionListener(new java.awt.event.ActionListener() {
                public void
actionPerformed(java.awt.event.ActionEvent e)
                {
                    ProfileDialog pd = new ProfileDialog(my);
                    Dimension dlgSize = pd.getPreferredSize();
                    Dimension frmSize = getSize();
                    Point loc = getLocation();
                    pd.setLocation((frmSize.width - dlgSize.width) / 2
                        + loc.x, (frmSize.height - dlgSize.height)
                        / 2 + loc.y);
                    pd.show();
                }
            });
    }
    return JMenuItemOption;
}

public JMenuItem getMenuItemOption() {
    if (jMenuItemOption == null) {
        JMenuItemOption = new JMenuItem();
        JMenuItemOption.setText("Sync");
        JMenuItemOption
            .addActionListener(new java.awt.event.ActionListener() {
                public void
actionPerformed(java.awt.event.ActionEvent e)
                {
                    CameraTakePictureNumber = 0;

                    int port = 12121;
                    for (int i=0; i<CameraNum; i++)
                    {

```

```

newSocket[i] = new MonitorCameraSocket(i,
my);

newSocket[i].SetAddressPort(JTextF[i].getText() , port);
    newSocket[i].SetFunction(1);
    }

//Start
for (int i=0; i<CameraNum; i++)
{
    newSocket[i].start();
}

int count = 0;
while (true)
{
    if (newSocket[count].isAlive() == true)
    {
        count++;
    }

    if (CameraNum == count) break;
}

int sync = 0;
sync = CameraStatus[0];
for (int i=1; i<CameraNum; i++)
{
    if (CameraStatus[i] < sync)
    {
        sync = CameraStatus[i];
    }
}

//Check
for (int i=0; i<CameraNum; i++)
{
    if (CameraStatus[i] != sync)
    {
        jLabelSyunStatus[i].setText("NoSync");

        newSocket[i] = new
MonitorCameraSocket(i, my);

        newSocket[i].SetAddressPort(JTextF[i].getText() , port);

        newSocket[i].SetFunction(3);

```



```

newSocket[i].start();

count = 0;
while (true)
{
    if (newSocket[count].isAlive() ==
false)
    {
        count++;
    }

    if (CameraNum == count) break;
}
}
else
{
    jLabelSyunStatus[i].setText("Sync");
    newSocket[i] = new

MonitorCameraSocket(i, my);

newSocket[i].SetAddressPort(JTextF[i].getText() , port);

newSocket[i].SetFunction(2);

newSocket[i].start();
syncT[i] = true;
jbn[i].setEnabled(false);

jLabelCheckSumStatus[i].setText(Integer.toString(CameraStatus[i]));
}

}
System.out.println("done.");

}
});
}
return jMenuItemOption;
}

public JMenuItem getJMenuItemExit() {
    if (jMenuItemExit == null) {
        jMenuItemExit = new JMenuItem();
        jMenuItemExit.setText("Exit");
        jMenuItemExit.setMnemonic(java.awt.event.KeyEvent.VK_X);
        jMenuItemExit.setAccelerator(javax.swing.KeyStroke.getKeyStroke(
            java.awt.event.KeyEvent.VK_X,
java.awt.Event.ALT_MASK,

```

```

        false));
    JMenuItemExit
        .addActionListener(new java.awt.event.ActionListener() {
            public void
actionPerformed(java.awt.event.ActionEvent e) {
                System.exit(0);
            }
        });
    }
    return JMenuItemExit;
}

public JTabbedPane getJTabbedPane() {
    if (jTabbedPane == null) {
        jTabbedPane = new JTabbedPane();
        jTabbedPane.addTab("CameraStatus", getJPanelTabCameraStatus());
        //jTabbedPane.addTab("Setting", getJPanel8());
        jTabbedPane.setBounds(new Rectangle(1, 45, 501, 442));
    }
    return jTabbedPane;
}

public JPanel getJPanelTabCameraStatus() {
    if (jPanelCameraTabContext == null) {
        jPanelCameraTabContext = new JPanel();
        jPanelCameraTabContext.setLayout(null);
        jPanelCameraTabContext.add(getJPanelCameraTabContext(), null);
    }
    return jPanelCameraTabContext;
}

public JPanel getJPanelCameraTabContext() {
    if (jPanelCameraTabContextDetail == null) {
        jPanelCameraTabContextDetail = new JPanel();
        jPanelCameraTabContextDetail.setLayout(null);
        jPanelCameraTabContextDetail.setBounds(new Rectangle(14, 2, 501,
442));
        jPanelCameraTabContextDetail.add(getJPanelInCameraTab(), null);
        //jPanelCameraTabContextDetail.add(getJPanel3(), null);
    }
    return jPanelCameraTabContextDetail;
}

//Button Listener
static class ButtonListener implements ActionListener{

    private int bindex;
    private Main subclass;

```

```

public ButtonListener(int index, Main my)
{
    bindex = index;
    subclass = my;
}

public void actionPerformed(ActionEvent evt)
{
    int port = 12121;
    int timeout = 10000000;

    subclass.newSocket[bindex] = new MonitorCameraSocket(bindex,
this.subclass);

    subclass.newSocket[bindex].SetAddressPort(subclass.JTextField[bindex].getText() ,
port);
    subclass.newSocket[bindex].SetFunction(3);

    int count = 0;
    while (true)
    {
        if (subclass.newSocket[count].isAlive() == true)
        {
            count++;
        }

        if (3 == count) break;
    }
}

private int SetJPanel(final int i, JPanel TabJPanel)
{
    JSpLabel[i] = new JLabel();
    JSpLabel[i].setText("IP");
    JSpLabel[i].setBounds(new Rectangle(0, i*40 + 20, 45, 27));

    JTextField[i] = new JTextField();
    JTextField[i].setBounds(new Rectangle(30, i*40 + 20, 180, 22));

    JTextField[i].setText("192.168.123.101");

    JLabelSyunStatus[i] = new JLabel();
    JLabelSyunStatus[i].setText("NoSync");
    JLabelSyunStatus[i].setBounds(new Rectangle(30 + 190, i*40 + 20 , 60,
27));
}

```

```

jLabelCheckSumStatus[i] = new JLabel();
jLabelCheckSumStatus[i].setText("NoChecknum");
jLabelCheckSumStatus[i].setBounds(new Rectangle(30 + 190 + 65, i*40 +
20, 100, 27));

```

```

jbn[i] = new JButton();
jbn[i].setText("Sync");
jbn[i].setBounds(new Rectangle(30 + 190 + 65 + 100, i*40 + 20, 90, 27));
jbn[i].addActionListener(new ButtonListener(i, this));
jbn[i].setEnabled(false);

```

```

TabJPanel.add(JSpLabel[i], null);
TabJPanel.add(JTextF[i], null);
TabJPanel.add(jLabelSyunStatus[i], null);
TabJPanel.add(jLabelCheckSumStatus[i], null);
TabJPanel.add(jbn[i], null);

```

```

return 0;

```

```

}

```

```

public JPanel getJPanelInCameraTab() {
    if (jPanelInCameraTab == null) {
        jPanelInCameraTab = new JPanel();
        jPanelInCameraTab.setLayout(null);
        jPanelInCameraTab.setBounds(new Rectangle(0, 0, 501, 442));
        jPanelInCameraTab.setName("jPanelInCameraTab");

        for (int i=0; i<CameraNum; i++)
        {
            SetJPanel(i, jPanelInCameraTab);
        }
    }
    return jPanelInCameraTab;
}

```

```

public JPanel getJPanelMiddle() {
    if (jPanelMiddle == null) {
        jLabelStatus = new JLabel();
        jLabelStatus.setBounds(new Rectangle(1, 487, 501, 16));
        jLabelStatus.setText("");

        jPanelMiddle = new JPanel();
        jPanelMiddle.setLayout(null);
        jPanelMiddle.add(getJPanelMidToolBar(), null);
        jPanelMiddle.add(getJTabbedPane(), null);
        jPanelMiddle.add(jLabelStatus, null);
    }
}

```

```

        return jPanelMiddle;
    }

    public JPanel getJPanelMidToolBar() {
        if (jPanelToolBarBar == null) {
            GridLayout gridLayout = new GridLayout();
            gridLayout.setRows(1);
            gridLayout.setVgap(0);
            gridLayout.setHgap(0);
            jPanelToolBarBar = new JPanel();
            jPanelToolBarBar.setBounds(new Rectangle(0, 0, 501, 37));
            jPanelToolBarBar.setLayout(gridLayout);
            jPanelToolBarBar.add(getJToolBarBar(), null);
        }
        return jPanelToolBarBar;
    }

    public JToolBar getJToolBarBar() {
        if (JToolBarBar == null) {
            JToolBarBar = new JToolBar();
            JToolBarBar.add(getJPanelToolBarBarDetail());
        }
        return JToolBarBar;
    }

    public JPanel getJPanelToolBarBarDetail() {
        if (jPanelToolBarBarDetail == null) {
            JLabelCameraStatus = new JLabel();
            JLabelCameraStatus.setBounds(new Rectangle(350, 19, 120, 15));
            JLabelCameraTotal = new JLabel();
            JLabelCameraTotal.setBounds(new Rectangle(350, 1, 120, 15));
            jPanelToolBarBarDetail = new JPanel();
            jPanelToolBarBarDetail.setLayout(null);
            jPanelToolBarBarDetail.add(getJButtonSync(), null);
            jPanelToolBarBarDetail.add(getJButtonTakePicture(), null);
            jPanelToolBarBarDetail.add(getJButtonSyncData(), null);
            jPanelToolBarBarDetail.add(jLabelCameraTotal, null);
            jPanelToolBarBarDetail.add(jLabelCameraStatus, null);
        }
        return jPanelToolBarBarDetail;
    }

    public void SyncCallback(int index, int timeout, String timestamp)
    {
        if (timeout == 0)
        {
            jbn[index].setEnabled(true);
            jLabelSyunStatus[index].setText("error");
        }
    }

```

```

        jLabelCheckSumStatus[index].setText(newSocket[index].error_string);
    }
    else
    {
        CameraStatus[index] = Integer.parseInt(timestamp);

jLabelCheckSumStatus[index].setText(Integer.toString(CameraStatus[index]));

    }
}

public void NeedSyncCallback(int index, int timeout, String timestamp)
{
    if (timeout == 0)
    {
        jbn[index].setEnabled(true);
        jLabelSyunStatus[index].setText("error");
        jLabelCheckSumStatus[index].setText(newSocket[index].error_string);
    }
    else
    {
        CameraStatus[index] = Integer.parseInt(timestamp);

jLabelCheckSumStatus[index].setText(Integer.toString(CameraStatus[index]));
        syncT[index] = false;
        jbn[index].setEnabled(false);
    }
}

public void TakePictureCallback(int index, int timeout, String err)
{
    if (timeout == 0)
    {
        jbn[index].setEnabled(true);
        jLabelSyunStatus[index].setText("error");
        jLabelCheckSumStatus[index].setText(newSocket[index].error_string);
    }
    else
    {
        jLabelCheckSumStatus[index].setText(err);
        if (err.equals("error"))
        {
            CTakePicture = 1;
        }
    }
}
}

```

```

public JButton getJButtonSync()
{
    if (JButtonSync == null) {
        JButtonSync = new JButton();
        JButtonSync.setIcon(new ImageIcon("images/sync.png"));
        JButtonSync.setToolTipText("Sync");
        JButtonSync.setBounds(new Rectangle(1, 0, 58, 33));
        JButtonSync.setMnemonic(java.awt.event.KeyEvent.VK_S);
        JButtonSync.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent ae)
            {
                jLabelStatus.setText("");
                CameraTakePictureNumber = 0;

                int port = 12121;
                for (int i=0; i<CameraNum; i++)
                {
                    newSocket[i] = new MonitorCameraSocket(i, my);
                    newSocket[i].SetAddressPort(JTextField[i].getText()
port);

                    newSocket[i].SetFunction(1);
                }

                //Start
                for (int i=0; i<CameraNum; i++)
                {
                    newSocket[i].start();
                }

                int count = 0;
                while (true)
                {
                    if (newSocket[count].isAlive() == true)
                    {
                        count++;
                    }

                    if (CameraNum == count) break;
                }

                int sync = 0;
                sync = CameraStatus[0];
                for (int i=1; i<CameraNum; i++)
                {
                    if (CameraStatus[i] < sync)
                    {
                        sync = CameraStatus[i];
                    }
                }
            }
        });
    }
}

```

```

    }

    //Check
    for (int i=0; i<CameraNum; i++)
    {
        if (CameraStatus[i] != sync)
        {
            jLabelSyunStatus[i].setText("NoSync");

            newSocket[i] = new MonitorCameraSocket(i, my);
            newSocket[i].SetAddressPort(JTextF[i].getText() ,

port);

            newSocket[i].SetFunction(3);

            newSocket[i].start();

            count = 0;
            while (true)
            {
                if (newSocket[count].isAlive() == false)
                {
                    count++;
                }

                if (CameraNum == count) break;
            }
        }
        else
        {
            jLabelSyunStatus[i].setText("Sync");
            newSocket[i] = new MonitorCameraSocket(i, my);
            newSocket[i].SetAddressPort(JTextF[i].getText() ,

port);

            newSocket[i].SetFunction(2);

            newSocket[i].start();
            syncT[i] = true;
            jbn[i].setEnabled(false);

            jLabelCheckSumStatus[i].setText(Integer.toString(CameraStatus[i]));
        }

        }
        jLabelStatus.setText("Sync Success.");
    }
    });
}

```



```

        return JButtonSync;
    }

    private JButton getJButtonTakePicture()
    {
        if (JButtonTakePicture == null) {
            JButtonTakePicture = new JButton();
            JButtonTakePicture.setBounds(new Rectangle(70, 0, 58, 33));
            JButtonTakePicture.setMnemonic(KeyEvent.VK_R);
            JButtonTakePicture.setIcon(new
ImageIcon("images/take_picture.png"));
            JButtonTakePicture.setToolTipText("TakePicture");
            JButtonTakePicture
                .addActionListener(new java.awt.event.ActionListener() {
                    public void
actionPerformed(java.awt.event.ActionEvent e) {
                        jLabelStatus.setText("");
                        try {
                            int port = 12121;
                            CTakePicture = 0;
                            for (int i=0; i<CameraNum; i++)
                            {
                                newSocket[i] = new
MonitorCameraSocket(i, my);

                                newSocket[i].SetAddressPort(JTextField[i].getText() , port);
                                newSocket[i].SetFunction(4);

                                newSocket[i].SetCount(CameraTakePictureNumber);

                            }

                            //Start
                            for (int i=0; i<CameraNum; i++)
                            {
                                newSocket[i].start();
                            }

                            int count = 0;
                            while (true)
                            {
                                if (newSocket[count].isAlive() == true)
                                {
                                    count++;
                                }

                                if (CameraNum == count) break;
                            }
                        }
                    }
                });
        }
    }

```

```

System.out.println(Integer.valueOf(CTakePicture));

        if (CTakePicture == 1)
        {
            for (int i=0; i<CameraNum; i++)
            {
                newSocket[i] = new
MonitorCameraSocket(i, my);

                newSocket[i].SetAddressPort(JTextF[i].getText() , port);
                newSocket[i].SetFunction(4);

                newSocket[i].SetCount(CameraTakePictureNumber);

            }

            for (int i=0; i<CameraNum; i++)
            {
                newSocket[i].start();
            }

        }
        else
        {
            CameraTakePictureNumber++;

            System.out.println("CameraTakePictureNumber: " +
CameraTakePictureNumber);
        }

    } catch (Exception ec) {
    }

    jLabelStatus.setText("Take Picture Success.");
}
});
JButtonTakePicture.setMnemonic(java.awt.event.KeyEvent.VK_R);
}
return JButtonTakePicture;
}

public JButton getJButtonSyncData() {
    if (JButtonSyncData == null) {
        JButtonSyncData = new JButton();
        JButtonSyncData.setIcon(new ImageIcon("images/syncdata.png"));
    }
}

```

```

JButtonSyncData.setToolTipText("SyncData");
JButtonSyncData.setBounds(new Rectangle(140, 0, 58, 33));
JButtonSyncData
.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent e)
    {
        jLabelStatus.setText("");
        final LoadData load = new LoadData();
        load.endValue = CameraTakePictureNumber * CameraNum;

        final JProgressView lb = new JProgressView("sync data");
        lb.setLoadObj(load);
        lb.startLoad();

        //start
        Thread tl = new Thread()
        {
            LoadData lobj = load;
            int recieve_port = 12124;
            int port = 12121;

            public void run()
            {
                try {
                    //Open Socket Waiting
                    ServerSocket serverSkt = new
ServerSocket(recieve_port);
                    while (CameraTakePictureNumber > count)
                    {
                        //prepare directory
                        String fileName = root_path + "/" +
count + "/" + Main.IMEI + "-" + count + ".jpg";
                        File directory = new File(root_path +
"/" + count);

                        BufferedInputStream inputStream;
                        BufferedOutputStream outputStream;

                        if (!directory.exists())
                            directory.mkdirs();

                        System.out.println("fileName: " +
fileName);

                        for (int i=0; i<CameraNum; i++)
                        {
                            if (syncT[i] == false) continue;

```

```

//Send message for request
Socket client = new Socket();
InetSocketAddress isa = new
InetSocketAddress(JTextF[i].getText() , port);
client.connect(isa, 10000);
DataOutputStream out = new
DataOutputStream(client.getOutputStream());

out.writeUTF("getPicData");

out.writeUTF(Integer.toString(count));

//waiting pic data
Socket clientSkt =
serverSkt.accept();

System.out.printf("%s connect to
\n",
clientSkt.getInetAddress().toString());

InputStream =
new
BufferedInputStream(clientSkt.getInputStream());

OutputStream =
new
BufferedOutputStream(new FileOutputStream(fileName));

int bufferSize = 1024;
byte[] buffer = new
byte[bufferSize];

int length;

while((length =
{
    outputStream.write(buffer,
    0, length);
})

outputStream.flush();
outputStream.close();
InputStream.close();
clientSkt.close();
lobj.currentProecess++;
}

```

```

        count++;
    }

    serverSkt.close();
    lb.stopLoad();
    lb.clear();
    jLabelStatus.setText("Sync      Picture
Success.");

    } catch (Exception ec) {
    }
    }
};

    tl.start();

    }
});
}
return JButtonSyncData;
}
public JMenuItem getJMenuItemAbout() {
    if (jMenuItemAbout == null) {
        jMenuItemAbout = new JMenuItem();
        jMenuItemAbout.setText("About");
        jMenuItemAbout.setMnemonic(java.awt.event.KeyEvent.VK_A);

        jMenuItemAbout.setAccelerator(javax.swing.KeyStroke.getKeyStroke(
            java.awt.event.KeyEvent.VK_A,
            java.awt.Event.ALT_MASK,
            false));
        jMenuItemAbout
            .addActionListener(new java.awt.event.ActionListener() {
                public void
actionPerformed(java.awt.event.ActionEvent e) {
                    getJDialogAbout();
                    Dimension dlgSize = getJDialogAbout()
                        .getPreferredSize();
                    Dimension frmSize = getSize();
                    Point loc = getLocation();
                    getJDialogAbout()
                        .setLocation(
                            (frmSize.width - dlgSize.width)
/ 2
                                + loc.x,
                                (frmSize.height
-
                                dlgSize.height)
/ 2 + loc.y);
                    jDialogAbout.show();

```

```
        }
    });
}
return jMenuItemAbout;
}

private JPanel getJContentPane() {
    if (jContentPane == null) {
        GridBagConstraints gridBagConstraints2 = new GridBagConstraints();
        gridBagConstraints2.fill = GridBagConstraints.BOTH;
        gridBagConstraints2.gridy = 0;
        gridBagConstraints2.weightx = 1.0;
        gridBagConstraints2.weighty = 1.0;
        gridBagConstraints2.gridx = 0;
        jContentPane = new JPanel();
        jContentPane.setLayout(new GridBagLayout());
        jContentPane.add(getJTabbedPane1(), gridBagConstraints2);
    }
    return jContentPane;
}

private JTabbedPane getJTabbedPane1() {
    if (jTabbedPane1 == null) {
        jTabbedPane1 = new JTabbedPane();
        jTabbedPane1.addTab("", getJPanel9());
    }
    return jTabbedPane1;
}

private JPanel getJPanel9() {
    if (jPanel9 == null) {
        GridBagConstraints gridBagConstraints9 = new GridBagConstraints();
        gridBagConstraints9.fill = GridBagConstraints.BOTH;
        gridBagConstraints9.gridy = 6;
        gridBagConstraints9.weightx = 1.0;
        gridBagConstraints9.gridx = 0;
        GridBagConstraints gridBagConstraints8 = new GridBagConstraints();
        gridBagConstraints8.fill = GridBagConstraints.BOTH;
        gridBagConstraints8.gridy = 5;
        gridBagConstraints8.weightx = 1.0;
        gridBagConstraints8.gridx = 0;
        GridBagConstraints gridBagConstraints7 = new GridBagConstraints();
        gridBagConstraints7.fill = GridBagConstraints.BOTH;
        gridBagConstraints7.gridy = 4;
        gridBagConstraints7.weightx = 1.0;
        gridBagConstraints7.gridx = 0;
        GridBagConstraints gridBagConstraints6 = new GridBagConstraints();
        gridBagConstraints6.fill = GridBagConstraints.BOTH;
```

```
        gridBagConstraints6.gridy = 3;
        gridBagConstraints6.weightx = 1.0;
        gridBagConstraints6.gridx = 0;
        GridBagConstraints gridBagConstraints5 = new GridBagConstraints();
        gridBagConstraints5.fill = GridBagConstraints.BOTH;
        gridBagConstraints5.gridy = 2;
        gridBagConstraints5.weightx = 1.0;
        gridBagConstraints5.gridx = 0;
        GridBagConstraints gridBagConstraints4 = new GridBagConstraints();
        gridBagConstraints4.fill = GridBagConstraints.BOTH;
        gridBagConstraints4.gridy = 1;
        gridBagConstraints4.weightx = 1.0;
        gridBagConstraints4.gridx = 0;
        GridBagConstraints gridBagConstraints3 = new GridBagConstraints();
        gridBagConstraints3.fill = GridBagConstraints.BOTH;
        gridBagConstraints3.gridy = 0;
        gridBagConstraints3.weightx = 1.0;
        gridBagConstraints3.gridx = 0;
        JPanel9 = new JPanel();
        JPanel9.setLayout(new GridBagLayout());
        JPanel9.add(getJTextField(), gridBagConstraints3);
        JPanel9.add(getJTextField1(), gridBagConstraints4);
        JPanel9.add(getJTextField2(), gridBagConstraints5);
        JPanel9.add(getJTextField3(), gridBagConstraints6);
        JPanel9.add(getJTextField4(), gridBagConstraints7);
        JPanel9.add(getJTextField5(), gridBagConstraints8);
        JPanel9.add(getJTextField6(), gridBagConstraints9);
    }
    return jPanel9;
}

private JTextField getJTextField() {
    if (jTextField == null) {
        jTextField = new JTextField();
    }
    return jTextField;
}

private JTextField getJTextField1() {
    if (jTextField1 == null) {
        jTextField1 = new JTextField();
    }
    return jTextField1;
}

private JTextField getJTextField2() {
    if (jTextField2 == null) {
        jTextField2 = new JTextField();
    }
}
```

```
    }
    return jTextField2;
}

private JTextField getJTextField3() {
    if (jTextField3 == null) {
        jTextField3 = new JTextField();
    }
    return jTextField3;
}

private JTextField getJTextField4() {
    if (jTextField4 == null) {
        jTextField4 = new JTextField();
    }
    return jTextField4;
}

private JTextField getJTextField5() {
    if (jTextField5 == null) {
        jTextField5 = new JTextField();
    }
    return jTextField5;
}

private JTextField getJTextField6() {
    if (jTextField6 == null) {
        jTextField6 = new JTextField();
    }
    return jTextField6;
}

private JDialog getJDialogAbout() {
    if (jDialogAbout == null) {
        jDialogAbout = new JDialog(this);
        jDialogAbout.setTitle("About MonitorCamera");
        jDialogAbout.setSize(new Dimension(236, 141));
        jDialogAbout.setContentPane(getJContentPane1());
    }
    return jDialogAbout;
}

private JPanel getJContentPane1() {
    if (jContentPane1 == null) {
        GridBagConstraints        gridBagConstraints13        =        new
GridBagConstraints();
        gridBagConstraints13.gridx = 1;
        gridBagConstraints13.anchor = GridBagConstraints.WEST;
    }
}
```



```

        gridBagConstraints13.insets = new Insets(0, 5, 0, 0);
        gridBagConstraints13.gridy = 1;
        jLabelAddress2 = new JLabel();
        jLabelAddress2.setText("");
        GridBagConstraints gridBagConstraints12 = new
GridBagConstraints();
        gridBagConstraints12.gridx = 1;
        gridBagConstraints12.insets = new Insets(0, 5, 0, 0);
        gridBagConstraints12.gridy = 0;
        jLabelAddress1 = new JLabel();
        jLabelAddress1.setText("");
        GridBagConstraints gridBagConstraints11 = new
GridBagConstraints();
        gridBagConstraints11.gridx = 0;
        gridBagConstraints11.gridy = 1;
        jLabelName2 = new JLabel();
        jLabelName2.setText("");
        GridBagConstraints gridBagConstraints10 = new
GridBagConstraints();
        gridBagConstraints10.gridx = 0;
        gridBagConstraints10.gridy = 0;
        jLabelName1 = new JLabel();
        jLabelName1.setText("");
        jPanel1 = new JPanel();
        jPanel1.setLayout(new GridBagLayout());
        jPanel1.add(jLabelName1, gridBagConstraints10);
        jPanel1.add(jLabelName2, gridBagConstraints11);
        jPanel1.add(jLabelAddress1, gridBagConstraints12);
        jPanel1.add(jLabelAddress2, gridBagConstraints13);
    }
    return jPanel1;
}

public static void main(String[] args)
{
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    splashr.setLocation(
        (screenSize.width - 400) / 2,
        (screenSize.height - 344) / 2);
    splashr.setSize(new Dimension(344,400));
    splashr.show();

    Properties p = new Properties();
    FileInputStream in;
    try {
        in = new FileInputStream("profile.properties");
        p.load(in);
    } catch (FileNotFoundException e) {

```

```
        e.printStackTrace();
    }
    catch (IOException e) {
        e.printStackTrace();
    }

    int CameraNum = Integer.valueOf(p.getProperty("CameraNum"));
    String root_path = p.getProperty("PicturePath");

    Main app = new Main(CameraNum, root_path);
    app.addWindowListener(new java.awt.event.WindowAdapter() {
        public void windowClosing(java.awt.event.WindowEvent evt) {
            System.exit(0);
        }
    });
    Dimension frameSize = app.getSize();
    if (frameSize.height > screenSize.height) {
        frameSize.height = screenSize.height;
    }
    if (frameSize.width > screenSize.width) {
        frameSize.width = screenSize.width;
    }
    app.setLocation((screenSize.width - frameSize.width) / 2,
        (screenSize.height - frameSize.height) / 2);
    app.setVisible(true);
}
}
```

References

- Abdel-Ghaffar, H. S. (2002), 'Analysis of synchronization algorithms with time-out control over networks with exponentially symmetric delays', *Communications, IEEE Transactions on*, 50 (10), pp. 1652-1661.
- Abt, F., Blug, A., Nicolosi, L., Dausinger, F.; Höfler, H., Tetzlaff, R. and Weber, R. (2011), 'Real time closed loop control of full penetration keyhole welding with cellular neural network cameras', *Journal of Laser Micro/Nanoengineering*, 6 (2), pp. 131-137.
- Adamos, C. and Faig, W. (1992), 'Hough transform in digital photogrammetry', In: *International Archives of Photogrammetry and Remote Sensing*. Washington, USA, 29 (B3), pp. 250-254.
- Ahonen, T. (2008), *Mobile as 7th of the mass media: cellphone, cameraphone, iPhone, smartphone*. 1 ed: Futuretext pp.
- Akca, D. and Gruen, A. (2009), 'Comparative geometric and radiometric evaluation of mobile phone and still video cameras', *The Photogrammetric Record*, 24 (127), pp. 217-245.
- Akkose, M., Bayraktar, A. and Dumanoglu, A. A. (2008), 'Reservoir water level effects on nonlinear dynamic response of arch dams', *Journal of Fluids and Structures*, 24 (3), pp. 418-435.
- Alba, M., Fregonese, L., Prandi, F., Scaioni, M., Valgoi, P., (2006), 'Structural monitoring of a large dam by terrestrial laser scanning', In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Dresden, Germany, 36 (5), pp. 6.
- Alfakih, A. Y. (2007), 'On dimensional rigidity of bar-and-joint frameworks', *Discrete Applied Mathematics*, 155 (10), pp. 1244–1253.
- Ali, Q. I. (2010), 'Design and implementation of a mobile phone charging system based on solar energy harvesting', In: *Energy, Power and Control (EPC-IQ), 2010 1st International Conference on*. Nov. 30 2010-Dec. 2 2010. pp. 264-267.
- Allen, S., Graupera, V. and Lundrigan, L. (2010), *Pro smartphone cross-platform development: iPhone, Blackberry, Windows Mobile and Android development and distribution*. 1 ed New York, United States: Apress, pp.288.
- Anna, S.-C., Adam, C. and Michel, M. (2005), 'Use of deformation monitoring results

-
- in solving geomechanical problems—case studies', *Engineering Geology*, 79 (1–2), pp. 3-12.
- Ansari, F. (2005), *Sensing issues in civil structural health monitoring*.1 ed New York, United States: Springer, pp.528.
- Arias, P., Ordóñez, C., Lorenzo, H., Herraiez, J. and Armesto, J. (2007), 'Low-cost documentation of traditional agro-industrial buildings by close-range photogrammetry', *Building and Environment*, 42 (4), pp. 1817-1827.
- Armer, G. (2001), *Monitoring and assessment of structures*.10 ed United Kingdom: Taylor and Francis, pp.pp. 105.
- Asif, S. Z. (2011), *Next generation mobile communications ecosystem: technology management for mobile communications*.1 ed New Jersey, United States: John Wiley & Sons, pp.432.
- Atkinson, K. B. (2001), *Close range photogrammetry and machine vision*.2 ed Caithness, UK: Whittles Publishing, pp.371.
- Axelsson, P. (1999), 'Processing of laser scanner data—algorithms and applications', *ISPRS Journal of Photogrammetry and Remote Sensing*, 54 (2–3), pp. 138-147.
- Ay, S. U. (2008), *Large format CMOS image sensors: performance and design*.1 ed Saarbrücken, Germany: VDM Verlag, pp.
- Babchenko, A. and Maryles, J. (2007), 'Graded-index plastic optical fiber for deformation sensing', *Optics and Lasers in Engineering*, 45 (7), pp. 757-760.
- Bailey, M. and Cunningham, S. (2009), *Graphics shaders: theory and practice*.1 ed Oxfordshire, United Kingdom: A K Peters, Ltd., pp.
- Baldi, P., Cenni, N., Fabris, M. and Zanutta, A. (2008), 'Kinematics of a landslide derived from archival photogrammetry and GPS data', *Geomorphology*, 102 (3–4), pp. 435-444.
- Ballard, D. H. (1981), 'Generalising the Hough transform to detect arbitrary shapes', *Pattern Recognition*, 13 (2), pp. 16.
- Baltsavias, E. P. (1999), 'A comparison between photogrammetry and laser scanning', *ISPRS Journal of Photogrammetry and Remote Sensing*, 54 (2–3), pp. 83-94.
- Baofeng, P. and Gang, L. (2011), 'Monitoring and control technology in cantilever construction of continuous beam bridge', In: *Consumer Electronics*,

Communications and Networks (CECNet), 2011 International Conference on. 16-18 April 2011. pp. 1418-1422.

Berger, A. S. (2005), *Hardware and computer organization*.1 ed Amsterdam, Netherlands: Elsevier, pp.

Beug, M. F., Moser, H. and Ramm, G. (2012), 'Dynamic bridge standard for strain gauge bridge amplifier calibration', In: *Precision Electromagnetic Measurements (CPEM), 2012 Conference on.* 1-6 July 2012. pp. 568-569.

Bock, W. J., Wisniewski, R. and Wolinski, T. R. (1992), 'Fiber-optic strain-gauge manometer up to 100 MPa', *Instrumentation and Measurement, IEEE Transactions on*, 41 (1), pp. 72-76.

Bogle, S. and Sankaranarayanan, S. (2011) In: *Electro/Information Technology (EIT), 2011 IEEE International Conference on.* 15-17 May 2011.

Braga-Neto, U. and Goutsias, J. (2004), 'Grayscale level connectivity: theory and applications', *Image Processing, IEEE Transactions on*, 13 (12), pp. 1567-1580.

Brannan, J. and Ward, B. (2011), *iOS SDK programming a beginners guide*.1 ed New York, United States: McGraw-Hill Osborne Media, pp.

Brooker, G. (2008), *Introduction to sensors for ranging and imaging*.1 ed Hertfordshire, United Kingdom: Scitech pp.

Bruno, R., Conti, M. and Gregori, E. (2008), 'Throughput Analysis and Measurements in IEEE 802.11 WLANs with TCP and UDP Traffic Flows', *Mobile Computing, IEEE Transactions on*, 7 (2), pp. 171-186.

Bulur, G. and Sahin, A. B. (2012), 'Determination of depth and position of buried objects with Optical Fiber Sensor integrated Ground Penetrating Radar', In: *Signal Processing and Communications Applications Conference (SIU), 2012 20th.* 18-20 April 2012. pp. 1-3.

Burger, W. and Burge, M. J. (2007), *Digital image processing: an algorithmic introduction using Java*.1 ed New York, United States: Springer, pp.566.

Cavalin, P., Oliveira, L. S., Koerich, A. L. and Britto, A. S. (2006), 'Wood defect detection using grayscale images and an optimized feature set', In: *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on.* 6-10 Nov. 2006. pp. 3408-3412.

Chen, C.-Q., Koh, S.-N. and Sivaprakasapillai, P. (1995), 'VQ codebook design

-
- algorithm based on partial GLA', *Electronics Letters*, 31 (21), pp. 1803-1805.
- Chen, Q., Zhao, L., Lu, J., Kuang, G., Wang, N. and Jiang, Y. (2012), 'Modified two-dimensional otsu image segmentation algorithm and fast realisation', *Image Processing, IET*, 6 (4), pp. 426-433.
- Chen, W. W., Che-Hao, C., Ming-Ko, C., Pei-Shan, H., Wan-Ting, C., Ya-Lan, C. and Yi-Wen, C. (2010), 'Landslide site reconstruction with terrestrial laser scanning', In: *Geoinformatics, 2010 18th International Conference on*. 18-20 June 2010. pp. 1-5.
- Cheng-Min, L., Jyh-Horng, L., Chyi-Ren, D. and Chang-Ming, W. (2011) In: *Innovations in Bio-inspired Computing and Applications (IBICA), 2011 Second International Conference on*. 16-18 Dec. 2011.
- Chintalapudi, K., Fu, T., Paek, J., Kothari, N., Rangwala, S., Caffrey, J., Govindan, R., Johnson, E. and Masri, S. (2006), 'Monitoring civil structures with a wireless sensor network', *Internet Computing, IEEE*, 10 (2), pp. 26-34.
- Choi, S.-J., Kim, H.-H. and Pan, J.-K. (2012), 'A multiplexed structure of intensity based fiber optic sensor', In: *Opto-Electronics and Communications Conference (OECC), 2012 17th*. 2-6 July 2012. pp. 178-179.
- Chongchong, Y., Jia, Z., Li, T. and Jinyan, W. (2010), 'Distortion data research of bridge structure health monitoring based on LS-SVM classification', In: *Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on*. 23-25 June 2010. pp. 118-123.
- Chun-Ming, T. and Hsi-Jian, L. (2002), 'Binarization of color document images via luminance and saturation color features', *Image Processing, IEEE Transactions on*, 11 (4), pp. 434-451.
- Connor, J. J. and Faraji, S. (2012), *Fundamentals of structural engineering 2 ed* New York, United States: Springer, pp.1143.
- Cooke, B. and Williams, P. (2009), *Construction planning, programming and control*. 3 ed New Jersey, United States: John Wiley & Sons, pp.504.
- Corin, R., Etalle, S., Hartel, P. H. and Mader, A. (2007), 'Timed analysis of security protocols', *Journal of Computer Security*, 15 (6), pp. 32-40.
- Dan, W., Bao-jin, P. and Qun-fei, X. (2010), 'A building structure health monitoring system based on the characteristic of TFBG', In: *Optical Communications and Networks (ICOON 2010), 9th International Conference on*. 24-27 Oct. 2010. pp. 95-98.

-
- Deitel, P. J., Deitel, H. M., Deitel, A. and Morgano, M. (2011), *Android for Programmers: An App-Driven Approach*.1 ed New Jersey, United States: Prentice Hall, pp.
- Deng, Z. X. Q. (2003), *GPS application in building construction*.2 ed China: China Building Industry Press pp.656.
- Diego, G.-A., Pablo, R.-G. and Javier, G.-L. (2009), 'An automatic procedure for co-registration of terrestrial laser scanners and digital cameras', *ISPRS Journal of Photogrammetry and Remote Sensing*, 64 (3), pp. 308-316.
- Diggelen, F. v. (2009), *A-GPS: Assisted GPS, GNSS, and SBAS*.1 ed London, United Kingdom: Artech House, pp.
- Dimitrios, I., Walter, H. and Laine, A. F. (1999), 'Circle recognition through a 2D Hough Transform and radius histogramming', *Image and Vision Computing*, 17 (1), pp. 15-26.
- Dong-Hyun, K., Bo-Suk, Y. and Sang-Bum, L. (2011), '3D boiler tube leak detection technique using acoustic emission signals for power plant structure health monitoring', In: *Prognostics and System Health Management Conference (PHM-Shenzhen), 2011*. 24-25 May 2011. pp. 1-7.
- Egels, Y. and Kasser, M. (2002), *Digital photogrammetry*.1 ed United Kingdom: Taylor & Francis, pp.56.
- Elaldi, F. (2005), 'An overview for structural health monitoring of composites in aerospace applications', In: *Recent Advances in Space Technologies, 2005. RAST 2005. Proceedings of 2nd International Conference on*. 9-11 June 2005. pp. 309-314.
- Erfeng, Z. and Yongqiang, J. (2008), 'Dam Deformation Monitoring Model and Forecast Based on Hierarchical Diagonal Neural Network', In: *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*. 12-14 Oct. 2008. pp. 1-4.
- Ethrog, U. (1987), 'Network design optimization in non-topographic photogrammetry: 31 March–2 April 1987, Baltimore-USA', *Photogrammetria*, 42 (3), pp. 138-139.
- Fan, S., Cui, W. and Li, S. (2011), 'Study on Control Method to Improve the Performance of Dynamic Response of Shaking Table for Earthquake Simulation', In: *Measuring Technology and Mechatronics Automation (ICMTMA), 2011 Third International Conference on*. 6-7 Jan. 2011. 3 pp. 431-436.

-
- Fekete, S., Diederichs, M. and Lato, M. (2010), 'Geotechnical and operational applications for 3-dimensional laser scanning in drill and blast tunnels', *Tunnelling and Underground Space Technology*, 25 (5), pp. 614-628.
- Feng, W. H. (1991), *Close Range Photogrammetry: the state of the object shape and motion photography method* 1 ed Wuhan, China: Wuhan University Press, pp.339.
- Fraser, C. S. (1997), 'Digital camera self-calibration', *ISPRS Journal of Photogrammetry and Remote Sensing*, 52 (4), pp. 149-159.
- Fraser, C. S. (1981), 'Accuracy aspects of multiple focal setting self-calibration applied to non-metric cameras', *Photogrammetria*, 36 (4), pp. 121-132.
- Fraser, C. S. and Edmundson, K. L. (2000), 'Design and implementation of a computational processing system for off-line digital close-range photogrammetry', *ISPRS Journal of Photogrammetry and Remote Sensing*, 55 (2), pp. 94-104.
- Gen, M., Tsujimura, Y. and Ishizaki, S. (1996), 'Optimal design of a star-LAN using neural networks', *Computers & Industrial Engineering*, 31 (3-4), pp. 855-859.
- Gersho, A. and Gray., R. M. (1992), *Vector quantization and signal compression* 1 ed New York, United States: Springer pp.732.
- Ghazali, R., Sukri, A. E. E., Latif, A. R. A., Rasam, A. R. A., Latif, Z. A. and Samad, A. M. (2011), 'Evaluating the relationship between scanning resolution of laser scanner with the accuracy of the 3D model constructed', In: *Control System, Computing and Engineering (ICCSCE), 2011 IEEE International Conference on*. 25-27 Nov. 2011. pp. 590-595.
- Gili, J. A., Corominas, J. and Rius, J. (2000), 'Using Global Positioning System techniques in landslide monitoring', *Engineering Geology*, 55 (3), pp. 167-192.
- Gleitsmann, L. and Kappas, M. (2004), 'Digital multi-image photogrammetry combined with oblique aerial photography enables glacier monitoring survey flights below clouds in Alaska', In: *Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International*. 20-24 Sept. 2004. 2 pp. 1148-1151.
- Goggin, G. (2006), *Cell phone culture: mobile technology in everyday life*.1 ed London, United Kingdom: Routledge, pp.
- Goneid, A., El-Gindi, S. and Sewisy, A. (1997), 'A method for the Hough transform detection of circles and ellipses using a 1-dimensional array', In: *Systems, Man,*

-
- and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on. 12-15 Oct 1997. 4 pp. 3154-3157 vol.4.*
- Gordon, B., Chaddha, N. and Meng, T. H. Y. (1994), 'A low-power multiplierless YUV to RGB converter based on human vision perception', In: *VLSI Signal Processing, VII, 1994., [Workshop on]. 1994. pp. 408-417.*
- Gozalvez, J. (2008), 'First Google's android phone launched [Mobile Radio]', *Vehicular Technology Magazine, IEEE*, 3 (4), pp. 3-69.
- Guidi, G., Beraldin, J. A., Ciofi, S. and Atzeni, C. (2003), 'Fusion of range camera and photogrammetry: a systematic procedure for improving 3-D models metric accuracy', *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33 (4), pp. 667-676.
- Haddad, T., Rahbar, K. and Meyer, P. (2010), 'Cluster TDEV a new performance metric for timing over packet networks', In: *Precision Clock Synchronization for Measurement Control and Communication (ISPCS), 2010 International IEEE Symposium on. Sept. 27 2010-Oct. 1 2010. pp. 141-145.*
- Haksoo, C., Sukwon, C. and Hojung, C. (2008), 'Structural health monitoring system based on strain gauge enabled wireless sensor nodes', In: *Networked Sensing Systems, 2008. INSS 2008. 5th International Conference on. 17-19 June 2008. pp. 211-214.*
- Hawa, M., Taifour, S., Qasem, M. and Tuffaha, W. (2012), 'A dynamic cross-layer routing protocol for Mobile Ad hoc Networks', *AEU - International Journal of Electronics and Communications*, 66 (12), pp. 996-1005.
- Hinton, C. J. and Boyce, J. F. (1993), 'Circle, ellipse and parabola parametrisation using integral transforms', In: *Hough Transforms, IEE Colloquium on. 7 May 1993. pp. P6/1-P6/4.*
- Hogari, K., Yamada, Y. and Toge, K. (2010), 'Design and performance of ultra-high-density optical fiber cable with rollable optical fiber ribbons', *Optical Fiber Technology*, 16 (4), pp. 257-263.
- Holst, G. C. and Lomheim, T. S. (2011), *CMOS/CCD sensors and camera systems.2* ed Washington, United States: SPIE Press, pp.355.
- Hu, J. and Liu, X. (2011), 'Design and Implementation of Tailings Dam Security Monitoring System', *Procedia Engineering*, 26 (0), pp. 1914-1921.
- Hyun-Kwan, L. and Seong-Lyun, K. (2009) In: *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE. 5-8 April 2009.*

-
- Irvine, W. and Maclellan, F. (2005), *Surveying for construction*.5 ed New York, United States: McGraw-Hill Higher Education, pp.464.
- ITU (2012), *Recommendation ITU-R BT.601-4*. Available at: <http://www-inst.eecs.berkeley.edu/~cs150/Documents/ITU601.PDF>, the ITU Radiocommunication Assembly (Accessed: 23th May 2012).
- Jafar, I. and Hao, Y. (2007), 'Multilevel component-based histogram equalization for enhancing the quality of grayscale images', In: *Electro/Information Technology, 2007 IEEE International Conference on*. 17-20 May 2007. pp. 563-568.
- Jeunhomme, L. B. and Thompson, B. J. (1989), *Single-mode fiber optics: principles and applications*.2 ed United States: CRC Press, pp.312.
- Ji, Y., Mao, L., Huang, Q. and Gao, Y. (2011), 'Research on Object Shape Detection from Image with High-Level Noise Based on Fuzzy Generalized Hough Transform', In: *Multimedia and Signal Processing (CMSP), 2011 International Conference on*. 14-15 May 2011. 1 pp. 209-212.
- Jiang, C. (2011), 'Perspectives on long-term performance assessment of bridges based on structural health monitoring system', In: *Electric Technology and Civil Engineering (ICETCE), 2011 International Conference on*. 22-24 April 2011. pp. 553-556.
- Jiang, R. and Jauregui, D. V. (2010), 'Development of a digital close-range photogrammetric bridge deflection measurement system', *Measurement*, 43 (10), pp. 1431-1438.
- Jindal, A. and Liu, M. (2012), 'Networked Computing in Wireless Sensor Networks for Structural Health Monitoring', *Networking, IEEE/ACM Transactions on*, 20 (4), pp. 1203-1216.
- Jipping, M. J. (2007), *Smartphone operating system concepts with Symbian OS: a tutorial guide*.1 ed New Jersey, United States: John Wiley & Sons, pp.354.
- Jones, R. E., Naden, J. M. and Neat, R. C. (1988), 'Optical-fibre sensors using micromachined silicon resonant elements', *Control Theory and Applications, IEE Proceedings D*, 135 (5), pp. 353-358.
- Jui-Teng, C. and Chi-Kuei, W. (2011), 'DSM measurement for exposed gravel surface using close-range photogrammetry with multi-scale images', In: *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*. 24-29 July 2011. pp. 471-474.
- Jun, Z. and Jinglu, H. (2008), 'Image Segmentation Based on 2D Otsu Method with Histogram Analysis', In: *Computer Science and Software Engineering, 2008*

- Junjie, B., Damei, Z. and Xiuyan, Z. (2011), 'Structural health monitoring of smart civil structures based on fiber Bragg grating sensing technology', In: *Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on.* 8-10 Aug. 2011. pp. 635-638.
- Kaasalainen, S., Kukko, A., Lindroos, T., Litkey, P., Kaartinen, H., Hyyppa, J. and Ahokas, E. (2008), 'Brightness measurements and calibration with airborne and terrestrial laser scanners', *Geoscience and Remote Sensing, IEEE Transactions on*, 46 (2), pp. 528-534.
- Karbhari, V. M. and Ansari, F. (2009), *Structural health monitoring of civil infrastructure systems.* 1 ed Cambridge, United Kingdom: Woodhead Publishing Ltd, pp.552.
- Kee, E., Johnson, M. K. and Farid, H. (2011), 'Digital Image Authentication From JPEG Headers', *Information Forensics and Security, IEEE Transactions on*, 6 (3), pp. 1066-1075.
- Kim, T. (1992), 'Side match and overlap match vector quantizers for images', *Image Processing, IEEE Transactions on*, 1 (2), pp. 170-185.
- Konstantinidis, G., Wilcox, P. D. and Drinkwater, B. W. (2007), 'An Investigation Into the Temperature Stability of a Guided Wave Structural Health Monitoring System Using Permanently Attached Sensors', *Sensors Journal, IEEE*, 7 (5), pp. 905-912.
- Kottapalli, V. A., Kiremidjian, A. S., Lynch, J. P., Carryer, E., Kenny, T. W., Law, K. H. and Lei, Y. (2003), 'Two-tiered wireless sensor network architecture for structural health monitoring', In: *SPIE 10th Annual International Symposium on Smart Structures and Materials.* San Diego, United States, pp. 8-19.
- Kyoung-Lae, N., Chaudhari, Q. M., Serpedin, E. and Suter, B. W. (2007), 'Novel Clock Phase Offset and Skew Estimation Using Two-Way Timing Message Exchanges for Wireless Sensor Networks', *Communications, IEEE Transactions on*, 55 (4), pp. 766-777.
- Kyung-Tae, L., Gil-Yong, L., Jung-Oh, C., Renzhe, W. and Sung-Hoon, A. (2010), 'Design and fabrication of a smart flexible structure using Shape Memory Alloy wire (SMA)', In: *Biomedical Robotics and Biomechanics (BioRob), 2010 3rd IEEE RAS and EMBS International Conference on.* 26-29 Sept. 2010. pp. 599-603.
- Leavers, V. F. (1992), *Shape detection in computer vision using the Hough transform.* 1 ed New York, United States: Springer, pp.201.

-
- Lee, B., Roh, S. and Park, J. (2009), 'Current status of micro- and nano-structured optical fiber sensors', *Optical Fiber Technology*, 15 (3), pp. 209-221.
- Lee, H. M., Kittler, J. and Wong, K. C. (1992), 'Generalised Hough transform in object recognition', In: *Pattern Recognition, 1992. Vol.III. Conference C: Image, Speech and Signal Analysis, Proceedings., 11th IAPR International Conference on.* 30 Aug-3 Sep 1992. pp. 285-289.
- Leica_Geosystems (2010), *Automated high performance total station TCA2003.* Available at: <http://www.leica-geosystems.com>, Leica Geosystems (Accessed: 06/07/2012).
- Levine, J. (1999), 'Time synchronization over the Internet using an adaptive frequency-locked loop', *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 46 (4), pp. 888-896.
- Li, A.-r. and Liu, M.-j. (2010), 'Model structure health monitoring', In: *Software Engineering and Service Sciences (ICSESS), 2010 IEEE International Conference on.* 16-18 July 2010. pp. 734-737.
- Li, G.-k., Gao, F. and Wang, Z.-g. (2011a), 'A photogrammetry-based system for 3D surface reconstruction of prosthetics and orthotics', In: *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE.* Aug. 30 2011-Sept. 3 2011. pp. 8459-8462.
- Li, R., Li, G., Li, Z. and Tan, Y. (2009), 'Triangulation of point clouds based on contact measurement system', In: *Computer-Aided Industrial Design & Conceptual Design, 2009. CAID & CD 2009. IEEE 10th International Conference on.* 26-29 Nov. 2009. pp. 755-757.
- Li, W. and Wang, C. (2011), 'GPS in the Tailings Dam Deformation Monitoring', *Procedia Engineering*, 26 (0), pp. 1648-1657.
- Li, W., Zhiming, S., Kai, Y. and Chaoyong, P. (2011b), 'A self-adaptive image enhancing method based on grayscale power transformation', In: *Multimedia Technology (ICMT), 2011 International Conference on.* 26-28 July 2011. pp. 483-486.
- Li, X., Yu, H., Huang, H. and Zhang, D. (2011c), 'Numerical simulation and analysis of spectral characteristic for weak fiber Bragg grating', In: *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on.* 12-14 Aug. 2011. 5 pp. 2435-2438.
- Lin, S.-Y., Mills, J. P. and Gosling, P. D. (2008), 'Videogrammetric monitoring of as-built membrane roof structures', *The Photogrammetric Record*, 23 (122), pp. 128-147.

-
- Linde, Y., Buzo, A. and Gray, R. (1980), 'An Algorithm for Vector Quantizer Design', *Communications, IEEE Transactions on*, 28 (1), pp. 84-95.
- Ling, W., Zhangming, L. and Qiang, F. (2011), 'Analysis on deformation characteristic of the overlying high-voltage tower under the construction of the complex lager section tunnels', In: *Electric Technology and Civil Engineering (ICETCE), 2011 International Conference on*. 22-24 April 2011. pp. 1019-1022.
- Loh, K. J., Hou, T. C., Lynch, J. P. and Kotov, N. A. (2007), 'Nanotube-based sensing skins for crack detection and impact monitoring of structures ', In: *the 6th international workshop on structural health monitoring*. Stanford, CA, USA, pp. 1685-1693.
- Lombardi, M. (2000), *Computer time synchronizatio*. Available at: <http://tf.nist.gov/service/pdf/computertime.pdf>, National Institute of Standards and Technology (Accessed: 20 Dec. 2011).
- Lu, W. and Tan, J. (2008), 'Detection of incomplete ellipse in images with strong noise by iterative randomized Hough transform (IRHT)', *Pattern Recognition*, 41 (4), pp. 1268-1279.
- Luhmann, T., Robson, S., Kyle, S. and Harley, I. (2007), *Close range photogrammetry: principles, techniques and applications*.1 ed New Jersey, United States: John Wiley & Sons, pp. 57.
- Lutes J., Chrzanowski A., Bastin G. and C., W. (2001), 'DIMONS software for automatic data collection and automatic deformation analysis', In: *the 10th FIG International Symposium on Deformation Measurements*. California, United States, 2 pp. 101-109.
- Ma, S. and Yang, W. (2010), 'Design of spatial deformation sensor based on the technologies of Compliant Mechanisms and Fiber Bragg Grating', In: *Mechanic Automation and Control Engineering (MACE), 2010 International Conference on*. 26-28 June 2010. pp. 5728-5732.
- Maas, H.-G. and Hampel, U. (2006), 'Photogrammetric techniques in civil engineering material testing and structure monitoring.', *Photogrammetric Engineering and Remote Sensing*, 72 (1), pp. 39-46.
- Madeira, S., Gonçalves, J. and Bastos, L. (2010), 'Photogrammetric mapping and measuring application using MATLAB', *Computers & Geosciences*, 36 (6), pp. 699-706.
- Marchand-Maillet, S. (1999), *Binary digital image processing*.1 ed Massachusetts, United States: Stephane Marchand-Maillet, pp.

-
- Marshall, A. R. (1989), *Network design and optimization in close range photogrammetry*.1 ed New South Wales, Australia: University of New South Wales, pp.289.
- Mason, S. (1995), 'Expert system-based design of close-range photogrammetric networks', *ISPRS Journal of Photogrammetry and Remote Sensing*, 50 (5), pp. 13-24.
- McManigle, J. E., Stebbing, R. V. and Noble, J. A. (2012), 'Modified Hough transform for left ventricle myocardium segmentation in 3-D echocardiogram images', In: *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*. 2-5 May 2012. pp. 290-293.
- Mesch, F. (2007), 'Structures for self-monitoring of measurement systems', In: *SICE, 2007 Annual Conference*. 17-20 Sept. 2007. pp. 1681-1686.
- Mills, D. L. (1991), 'Internet time synchronization: the network time protocol', *IEEE Transactions on Communications*, 39 (10), pp. 1482-1493.
- Mills, D. L. (1995), 'Improved algorithms for synchronizing computer network clocks', *Networking, IEEE/ACM Transactions on*, 3 (3), pp. 245-254.
- Mills, D. L. (1999), 'Cryptographic authentication for real-time network protocols', *AMS DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 45 pp. 135-144.
- Mills, D. L. (2011), *Network time synchronization: the Network Time Protocol on earth and in space*.2 ed United States: CRC Press, pp.466.
- Min, S. Y. (2008), *Modern geodetic measurements*.2 ed Wuhan, China: Wuhan University Press, pp.368.
- Ming-Chih, L., Tien-Yu, T., Cheng-Pei, T. and Yu-An, C. (2010), 'The Non-Contact Landslides Monitoring System for Long-Distance', In: *Internet Technology and Applications, 2010 International Conference on*. 20-22 Aug. 2010. pp. 1-4.
- Mora, P., Baldi, P., Casula, G., Fabris, M., Ghirotti, M., Mazzini, E. and Pesci, A. (2003), 'Global Positioning Systems and digital photogrammetry for the monitoring of mass movements: application to the Ca' di Malta landslide (northern Apennines, Italy)', *Engineering Geology*, 68 (1-2), pp. 103-121.
- Nakamura, J. (2005), *Image sensors and signal processing for digital still cameras*.1 ed United States: CRC Press, pp.335.

-
- National Instruments Corporation (1998), *Strain gauge measurement – a tutorial* [Online]. Available at: http://www.eidactics.com/Downloads/Refs-Methods/NI_Strain_Gauge_tutorial.pdf (Accessed: 22/07/2012).
- Noppe, N. and Decréton, M. (1993), 'Strain gauges in a nuclear environment', *Materials & Design*, 14 (6), pp. 339-341.
- Ohnishi, Y., Nishiyama, S., Yano, T., Matsuyama, H. and Amano, K. (2006), 'A study of the application of digital photogrammetry to slope monitoring systems', *International Journal of Rock Mechanics and Mining Sciences*, 43 (5), pp. 756-766.
- Ozbek, M., Rixen, D. J., Erne, O. and Sanow, G. (2010), 'Feasibility of monitoring large wind turbines using photogrammetry', *Energy*, 35 (12), pp. 4802-4811.
- Pappa, R. S., Jones, T. W., Black, J. T., Walford, A., Robson, S. and Shortis, M. R. (2002a), 'Photogrammetry methodology development for gossamer spacecraft structures', In: *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. Denver, United States, 36 (8), pp. 12-21.
- Pappa, R. S., Jones, T. W., Black, J. T., Walford, A., Robson, S. and Shortis, M. R. (2002b), 'Photogrammetry methodology development for gossamer space structures', *Sound and Vibration*, 36 (8), pp. 12-21.
- Partridge, C. and Pink, S. (1993), 'A faster UDP [user datagram protocol]', *Networking, IEEE/ACM Transactions on*, 1 (4), pp. 429-440.
- Paul, K. and Kundu, T. K. (2010) In: *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*. June 29 2010-July 1 2010.
- Petrou, M. and Petrou, C. (2010), *Image Processing: The Fundamentals*. 2 ed New Jersey, United States: John Wiley & Sons, pp.818.
- Petzold, C. (2010), *Programming Windows Phone 7: Microsoft XNA framework edition*. 1 ed United States: Microsoft Press pp.
- Petzold, C. (2011), *Microsoft silverlight edition: programming for Windows Phone 7* 1 ed United States: Microsoft Press, pp.
- PhotoModeler, 2010. http://www.photomodeler.com/products /photomodeler/articles _and_reports.htm (accessed: 17th April 2010).
- Ping, Y. and Venetsanopoulos, A. N. (1993) In: *Circuits and Systems, 1993.*

- Plankis, B. J., Horning, M., Ponto, N. and Brown, L. K. (2007), 'Designing a Dependable and Fault-Tolerant Semiautonomous Distributed Control Data Collection Network With Opportunistic Hierarchy', *Oceanic Engineering, IEEE Journal of*, 32 (2), pp. 400-407.
- Poudel, U. P., Fu, G. and Ye, J. (2005), 'Structural damage detection using digital video imaging technique and wavelet transformation', *Journal of Sound and Vibration*, 286 (4-5), pp. 869-895.
- Pratt, W. K. (2012), *Digital image processing*. 1 ed New Jersey, United States: John Wiley & Sons, Inc., pp.816.
- Qilin, Z. and Dalin, L. (2010), 'Modular Structural Health Monitoring System for Large Span Spacial Structures', In: *Software Engineering (WCSE), 2010 Second World Congress on. 19-20 Dec. 2010.* 1 pp. 152-155.
- Raguse K. and Heipke C. (2006), 'Photogrammetric synchronization of image sequences', In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.* Dresden, Germany, 36 (5), pp. 254-259.
- Ramana, Y. V. and Sarma, L. P. (1981), 'Simple and sensitive strain gauge displacement transducer', *Review of Scientific Instruments*, 52 (9), pp. 1417-1419.
- Rich, J. R. (2004), *How to do everything with your smartphone, Windows Mobile Edition* 1 ed New York, United States McGraw-Hill Osborne Media, pp.
- Rivera, E., Thomson, D. J. and Polyzois, D. (2002), 'Structural health monitoring of composite poles using fiber optic sensors', In: *Electrical and Computer Engineering, 2002. IEEE CCECE 2002. Canadian Conference on. 2002.* 1 pp. 479-482
- Rodrigues, C., Félix, C., Lage, A. and Figueiras, J. (2010), 'Development of a long-term monitoring system based on FBG sensors applied to concrete bridges', *Engineering Structures*, 32 (8), pp. 1993-2002.
- Sariff, S. S. b. T., Shintarou, T. and Takayama, S. (2011), 'GPS implementation in sensing node network system for landslide disaster', In: *SICE Annual Conference (SICE), 2011 Proceedings of. 13-18 Sept. 2011.* pp. 2014-2018.
- Schukar, V. G., Kusche, N. and Habel, W. R. (2012), 'How Reliably do Fiber Bragg Grating Patches Perform as Strain Sensors?', *Sensors Journal, IEEE*, 12 (1), pp. 128-132.

-
- Senior, J. (2008), *Optical fiber communications: principles and practice* 3 ed New Jersey, United States: Prentice Hall, pp.1128.
- Seunghee, P., Chung-Bang, Y. and Inman, D. J. (2006), 'A Self-contained Active Sensor System for Health Monitoring of Civil Infrastructures', In: *Sensors, 2006. 5th IEEE Conference on.* 22-25 Oct. 2006. pp. 798-802.
- Shabtai, A., Fledel, Y., Kanonov, U., Elovici, Y., Dolev, S. and Glezer, C. (2010), 'Google Android: A Comprehensive Security Assessment', *Security & Privacy, IEEE*, 8 (2), pp. 35-44.
- Sivrikaya, F. and Yener, B. (2004), 'Time synchronization in sensor networks: a survey', *Network, IEEE*, 18 (4), pp. 45-50.
- Slade, P. G. (2007), *The vacuum interrupter: theory, design, and application.*1 ed Florida, United States: CRC Press, pp.
- Soloviev, A. and De Haag, M. U. (2010), 'Monitoring of Moving Features in Laser Scanner-Based Navigation', *Aerospace and Electronic Systems, IEEE Transactions on*, 46 (4), pp. 1699-1715.
- Song, C., Wang, K. and Zheng, L. (2008), 'An Image Processing System Research on Target and Rendezvous Status Identification of Missile and Plane', In: *Image and Signal Processing, 2008. CISP '08. Congress on.* 27-30 May 2008. 4 pp. 525-527.
- Stallings, W. (2006), *Data and computer communications.*8 ed New Jersey, United States: Prentice Hall, pp.
- Stefanescu, D. M. (2011), 'Strain gauges and Wheatstone bridges - basic instrumentation and new applications for electrical measurement of non-electrical quantities', In: *Systems, Signals and Devices (SSD), 2011 8th International Multi-Conference on.* 22-25 March 2011. pp. 1-5.
- Stewart, S. (2004), *Distribution switchgear.*1 ed Herts, United Kingdom: The Institution of Engineering and Technology, pp.
- Sun, H., Yuan, S., Zhao, X., Zhou, H. and Lianq, D. (2010), 'Technology of structure damage monitoring based on multi-agent', *Systems Engineering and Electronics, Journal of*, 21 (4), pp. 616-622.
- Syahmi, M. Z., Aziz, W. A. W., Zulkarnaini, M. A., Anuar, A. and Othman, Z. (2011), 'The movement detection on the landslide surface by using Terrestrial Laser Scanning', In: *Control and System Graduate Research Colloquium (ICSGRC), 2011 IEEE.* 27-28 June 2011. pp. 175-180.

-
- Takewaki, I., Nakamura, M. and Yoshitomi, S. (2007), *System identification for structural health monitoring*. Southampton, United Kingdom: WIT Press, pp.272.
- Thomas, M. J. (2011), *3G Mobile Broadband Performance Figures for UK*. Available at: http://stakeholders.ofcom.org.uk/binaries/research/telecoms-research/bbspeeds/2010/Mobile_BB_performance.pdf, The Office of Communications (Accessed: 05/03/2012).
- Tsujikawa, K., Tajima, K. and Zhou, J. (2005), 'Intrinsic loss of optical fibers', *Optical Fiber Technology*, 11 (4), pp. 319-331.
- Umbaugh, S. E. (2011), *Digital image processing and analysis: human and computer vision applications with CVPITools.2* ed United States: CRC Press, pp.
- Uren, J. and Price, W. F. (2010), *Surveying for engineers.5* ed Basingstoke, United Kingdom: Palgrave Macmillan, pp.816.
- Wakefield, T., McNally, D., Bowler, D. and Mayne, A. (2007), *Introduction to mobile communications: technology, services, markets.1* ed London, United Kingdom: Auerbach Publications, pp. 456.
- Wang, C.-H., Mills, J. P., Gosling, P., Bridgens, B. and Grisdale, R. (2010), 'Monitoring the testing, construction and as-built condition of membrane structures by close range photogrammetry.', In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Newcastle, UK, 36 (5), pp. 592-596.
- Wang, C.-H., Mills, J. P. and Miller, P. E. (2012), 'Automated low-cost photogrammetry for flexible structure monitoring', In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. 22(5), Melbourne, Australia, pp. 6.
- Weizheng, Z., Tao, J. and Mingqin, H. (2010) In: *Image and Signal Processing (CISP), 2010 3rd International Congress on*. 16-18 Oct. 2010.
- Westoby, M. J., Brasington, J., Glasser, N. F., Hambrey, M. J. and Reynolds, J. M. (2012), 'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications', *Geomorphology*, 179 (0), pp. 300-314.
- Wigley, A., Moth, D. and Foot, P. (2007), *Microsoft Mobile development handbook.1* ed United States: Microsoft Press, pp. 607.
- Wild, G., Allwood, G. and Hinckley, S. (2010), 'Distributed sensing, communications, and power in optical Fibre Smart Sensor networks for structural health

-
- monitoring', In: *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2010 Sixth International Conference on.* 7-10 Dec. 2010. pp. 139-144.
- Wood, B. (2009), *Building maintenance*. 1 ed New Jersey, United States: John Wiley & Sons, pp.328.
- Wu, L., Wang, T. and Wei, Z. (2010), 'GPS Height Conversion Based on Genetic Neural Network', In: *Computational and Information Sciences (ICCIS), 2010 International Conference on.* 17-19 Dec. 2010. pp. 503-506.
- Wu, M. and Chen, C. W. (2003) In: *Sensors, 2003. Proceedings of IEEE.* 22-24 Oct. 2003.
- Wu, S., Lin, C., Zhang, Z. and Li, Z. (2011) In: *Electric Technology and Civil Engineering (ICETCE), 2011 International Conference on.* 22-24 April 2011.
- Wylie, C. R. (2008), *Introduction to projective geometry*. 1 ed New York, United States: Dover Publications, pp.
- Xiangqi, H., Sasaki, T., Hashimoto, H. and Inoue, F. (2010), 'Circle detection and fitting using laser range finder for positioning system', In: *Control Automation and Systems (ICCAS), 2010 International Conference on.* 27-30 Oct. 2010. pp. 1366-1370.
- Xinlian, L., Litkey, P., Hyypä, J., Kaartinen, H., Vastaranta, M. and Holopainen, M. (2012), 'Automatic Stem Mapping Using Single-Scan Terrestrial Laser Scanning', *Geoscience and Remote Sensing, IEEE Transactions on*, 50 (2), pp. 661-670.
- Xinzhong, W. and Chuanxi, L. (2011), 'GPS displacement monitoring system of Pingsheng Bridge', In: *Electrical and Control Engineering (ICECE), 2011 International Conference on.* 16-18 Sept. 2011. pp. 5944-5947.
- Xiongyao, X. and Chenchao, Z. (2012), 'Non-destructive evaluation of shield tunnel condition using GPR and 3D laser scanning', In: *Ground Penetrating Radar (GPR), 2012 14th International Conference on.* 4-8 June 2012. pp. 479-484.
- Xu, C. and Hu, Q. (2009), 'Dynamic structural monitoring using the kinematic positioning of a robotic total station', In: *Electronic Measurement & Instruments, 2009. ICEMI '09. 9th International Conference on.* 16-19 Aug. 2009. pp. 2-69-2-73.
- Yang, J. (2010), 'Discussion on Section Construction Techniques for Continuous Bridge of Large-Span Concrete', In: *Optoelectronics and Image Processing (ICOIP), 2010 International Conference on.* 11-12 Nov. 2010. 1 pp. 436-438.

-
- Yilmaz, H. M., Yakar, M., Gulec, S. A. and Dulgerler, O. N. (2007), 'Importance of digital close-range photogrammetry in documentation of cultural heritage', *Journal of Cultural Heritage*, 8 (4), pp. 428-433.
- Yinghui, X., Qingming, Z. and Qiancong, P. (2007), '3D Data Acquisition by Terrestrial Laser Scanning for Protection of Historical Buildings', In: *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*. 21-25 Sept. 2007. pp. 5971-5974.
- Yongdae, K., Youngdeok, K., Chulsub, L. and Sejin, K. (2010), 'Thin polysilicon gauge for strain measurement of structural elements', *Sensors Journal, IEEE*, 10 (8), pp. 1320-1327.
- Young, L. and Garde, G. (2007), 'A practical approach for scientific balloon film strain measurement using photogrammetry', In: *AIAA Balloon Systems Conference*. Williamsburg, VA, American Institute of Aeronautics and Astronautics, Inc, pp. 11-17.
- Yuan, K. X. (1991), *Surveying and mapping: mapping project supervision science 1* ed Wuhan, China: Wuhan University Press, pp.447.
- Yun-shan, C., Hui-bin, G., Wan-xin, S. and Min, W. (2010), 'Grayscale control algorithm based on optokinetic nystagmus', In: *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on*. 20-22 Aug. 2010. 2 pp. 253-256.
- Zhang, Y., J. Tsai, G. P., Li, M. Q. F. and Bachman, M. (2010), 'MOEMS acoustic sensors for structural health monitoring.', In: *IEEE Sensors 2010* Waikoloa, HI, pp. 2311-2315.
- Zhipeng, W., Jinshi, C., Hongbin, Z., Kegesawa, M. and Ikeuchi, K. (2011), 'Object detection by common fate Hough transform', In: *Pattern Recognition (ACPR), 2011 First Asian Conference on*. 28-28 Nov. 2011. pp. 613-617.
- Zhu, Y.-H., Qi, H. and Wang, G.-X. (2011), 'Temperature characteristic of fiber Bragg grating', In: *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*. 12-14 Aug. 2011. 8 pp. 4020-4022.