MONITORING DYNAMIC MOVEMENTS WITHIN THE INTERIOR OF BUILDINGS USING CAMERA

Kai-Uwe Doerr∗
Dept. of Electrical Engineering and Computer Science
University of California
Irvine, California 92697
Tel: 949-824-2818, Fax: 949-824-3203
Email: kdorr@uci.edu

Falko Kuester
Dept. of Electrical Engineering and Computer Science
University of California
Irvine, California 92697
Tel: 949-824-9320, Fax: 949-824-3203
Email: fkuester@uci.edu

Tara C. Hutchinson
Dept. of Civil and Environmental Engineering
University of California
Irvine, California 92697
Tel: 949-824-2166, Fax: 949-824-2117
Email: thutchin@uci.edu

Derek Nastase
Dept. of Civil and Environmental Engineering
University of California
Irvine, California 92697
Email: dnastase@uci.edu

ABSTRACT
Previous experiences during earthquake events emphasize the need for new technologies for real-time monitoring and assessment of facilities with high value nonstructural elements such as equipment or other contents. Moreover, there is a substantial limitation in our ability to rapidly evaluate and identify potential hazard zones within a structure, exposing rescue workers, society and the environment to unnecessary risks. A real-time monitoring system, integrated with critical warning systems, would allow for improved channeling of resources and informed decision making processes for rescue workers and building owners. In recognition of these issues, in this paper, we describe a methodology for image-based tracking of seismically induced motions. The methodology includes the calibration, acquisition, processing, and analysis tools for seismic assessment. We especially focus on the issue of providing a reliable feature/object detection and tracking methodology for an image sequence from a single camera view.

Keywords: Seismic motions, image acquisition, image processing, tracking algorithms

INTRODUCTION
After an earthquake takes place, the general situation inside a building is unknown. While the primary structure of a building may perform well in an earthquake, it is very likely that extensive nonstructural damage occurred. Rooms may be inaccessible, nonstructural elements such as shelves or desks can block entrances, water pipelines can be demolished, while electronic components like computers may still be supplied with power. The situation can be even more critical with hazardous materials housed in physical, chemical and medical laboratories or industry complexes. These uncertain conditions, in combination with the risks of secondary effects like fire and aftershocks, result in unnecessary safety risks for first responders and rescue teams. Information and a deeper knowledge about the general behavior of nonstructural elements would help to guide rescue teams and develop adequate action strategies.

Investigating the general object behavior during a seismic event will also enhance the development of effective design
methodologies for expensive equipment and methods for storing them. Solving these issues will help to decrease economic losses which could otherwise force smaller companies out of business if the equipment can not be purchased again.

However, the large diversity of nonstructural elements, with various purposes, increase the difficulty of gaining a basic understanding of their behavior. To understand nonstructural element behavior, the three dimensional response of an object in varied conditions under a variety of motions needs to observe. Traditional methods of monitoring element behavior using discrete, pointwise measurement instruments render 3D response measurement difficult. An alternative methodology is necessary to pursue the problem such that nonstructural behavior can be accurately interpreted. In recognition of these issues, in this paper, we describe a methodology for image-based tracking of seismically induced motions. We focus especially on two major components for such a tracking system. We describe our approach of an adequate system hardware setup, capable of collecting and storing image sequences with reasonable resolution and update rates, before we focus on our methodology of a reliable 2D feature detecting and tracking pipeline as a key component for all vision based tracking systems. Once the reliable identification of object features in an image is achieved, commonly known multiple view geometry algorithms can be applied to extract and calculate the 3D position of an object/feature. We will introduce our 2D image processing pipeline and provide preliminary results from actual field test data.

MONITORING / ACQUISITION SYSTEM

The designed acquisition system uses a unique hardware and software layout involving a multi-threaded process, which bypasses conventional hardware image frame grabbers. It internally (within the software) acquires, synchronizes and time stamps the collected images. In the following sections, we describe the hardware and software components for the acquisition system.

HARDWARE COMPONENTS

The image-based capture system consists of four high-speed charged-couple-device (CCD) cameras, connected to a server-style PC with extended storage and network capabilities. All cameras operate at a resolution of 658x494 pixels and are capable of acquiring images at 80 frames per second. Table 1 presents a summary of the major hardware equipment items used in the acquisition system.

At maximum acquisition speed and resolution, the cameras produce data at a rate of 24 MB/s each. The pipeline for data transfer from the camera to the storage device is shown in Fig. 1 to highlight possible bottlenecks. In this case, the percentage values represent the utilization of the components best-case bandwidth and clearly highlights the system components that have to be treated carefully. It can be seen clearly seen that the most computationally expensive task occurs when data is transferred to the hard drive.

Data will first be passed through the FireWire bus, which has a maximum throughput of 50 MB/s. Since the amount of data first produced by the four cameras already exceeds the bandwidth of a 32bit PCI bus (max. 80 MB/s), a server-style motherboard with a triple PCI bus configuration was selected. This setup allows two of the four FireWire cards to be attached to a 64bit PCI-X bus (max 480MB/s), while the other two are placed on its 32bit PCI bus. Next, the data is transferred to the CPU, across the main memory modules involving data transmission through the 533 MHz Front Side Bus and the 266 MHz DDR Ram memory bus. The bandwidth of both busses is much higher than the demands of the application, therefore, these busses do not present any limitation, even if the images have to cross both busses twice.
After moving across the processor and RAM modules, the data must be stored for post-process analysis. For this purpose, the third PCI-X bus is linked with the SCSI bus and interfaced with fast SCSI hard drives. Although current hard drives have an advertised transfer rate of 60 MB/s or greater, these speeds can unfortunately only be attained while reading information. Benchmark tests showed that sustained transfer rates of 27 MB/s per disk while saving can be achieved, using fast 10,000 RPM SCSI hard drives each dedicated to store data from a single camera.

SOFTWARE DESIGN

The processing pipeline consists of three major building blocks for (1) image capture, (2) data archival and (3) image processing and data analysis. The image capture process provides software-based camera synchronization, image acquisition and time-stamping, while the data archival components handle the organization, partitioning and storing of arbitrarily long video sequences. Digital image/video data is directly acquired from the CCD camera through its IEEE 1394 (FireWire) interface. The implemented multi-threaded framework avoids unnecessary wait states and allows proper control of timing and sequencing of the images captured. One thread is responsible to trigger image acquisition for the entire camera array, while four additional threads (one per camera) manage data transfer from a camera specific data buffer to an AVI stream stored on a camera system disk. All of the storing threads run with real time priority to ensure acquired images are time stamped and merged into the respective AVI stream as soon as they become available. The image capture is synchronized with the image transfer threads through multiple image buffers. One of these image buffer threads is available for each camera and accessed by the corresponding transfer thread, allowing temporary storage of up to 256 acquired images, in the event that system resources or bandwidth become limited for a short period of time. System performance test, showed for multiple trials that this architecture allows the capture of video data at full resolution and frame rate for the described camera array. Figure 2 shows the performance test with four cameras connected to the system. Summary statistics with various camera setups indicate excellent system performance, with mean frame rates of 79.99 attainable in each trial case.

Finally, the post processing stage implements different feature detection and tracking algorithms in a pipeline approach, allowing multiple streams to be processed in parallel to obtain the desired 3D feature information. Therefore we used and extended the Open Source Computer Vision Library (OpenCV) [Intel Research Group [1]].

The implemented pixel-based image processing chain (PIPC) identifies trackable objects using the well known Good features to track algorithm based on the work of Shi and Tomasi [2]. To track the identified object features, we apply the pyramidal implementation of the Lucas and Kanade [3] feature tracker as described by Jean-Yves Bouguet [4] (see Eqn. (1)).

$$\epsilon(d) = \epsilon(d_x, d_y) = \sum_{x=d_x-w_x}^{d_x+w_x} \sum_{y=d_y-w_y}^{d_y+w_y} (I(x, y) - J(x + d_x, y + d_y))^2$$

This tracking algorithm is based on the optical flow approach in combination with a similarity detection in the 2D neighborhood by minimizing the residual function $\epsilon(d)$ in a predefined integration window of the size $(2w_x + 1) \times (2w_y + 1)$, where $I(x, y)$ and $J(x, y)$ are representing two sequenced images, $u_x$ and $u_y$ define a 2D image coordinate of a feature and $d_x, d_y$ are the optical flow components in x- and y-directions, respectively. The algorithm is enhanced by calculating this residual in respect of the optical flow $d = [d_x, d_y]$ in each level of the image convolution pyramid. This calculated optical flow vector is subsequently passed to the next level as an initial feature position guess, down to the original image level calculation.

Before applying the tracking routine in our PIPC, the images are passed through a real-time image correction filter that removes radial and tangential distortion introduced by the camera lenses. To calculate the needed distortion parameters (Camera Matrix) we have developed a self calibration sequence for the camera based on a moving chessboard pattern. To assure the quality of the calibration, the detected parameters of the calibration are validated by calculating the re-projection error for each chessboard image presented during the calibration sequence.

To enhance the achievable resolution for object feature detection and the calibration sequence, we use a gradient based sub-pixel finding algorithm described in OpenCV [1] to refine corner locations. The implemented PIPC (see Fig.3) starts the image data processing with the correction for radial and tangential lens distortion. Subsequently, to identify trackable object features in the scene and to reduce computation time, we use a mask-based feature identification approach. The object mask can be determined either by user-defined regions in the acquired image, or

---

**Figure 2.** PERFORMANCE TESTS DATA STORING WITH FOUR CAMERAS CONNECTED TO THE ACQUISITION SYSTEM
by referencing an image to a precalculated model of the clean background (scene without the to be observed objects). Features of interest are then identified within the object mask in each image. The detected object features are then stored and passed to the 2-dimensional (2D) feature tracking algorithm described above, which calculates the movement of each feature over the processed image sequence.

FIELD BUILDING EXPERIMENT IN SHERMAN OAKS, CALIFORNIA

Teamed with the University of California, Los Angeles, a full-scale vibration experiment was conducted on a vacant structure damaged during the 1994 Northridge Earthquake, and used to evaluate the image-based tracking methodology. The building of interest is a four-story office building located in Sherman Oaks, California. The field investigation had two primary objectives: (1) to characterize the seismic response of an important class of equipment and building contents within the building interior and (2) to test the image-based tracking methodology, as applied to monitoring the response of these equipment and contents. A research team from UCLA conducted full-scale vibration tests on this building by mounting linear and eccentric mass shakers to the roof of the building and using NEES\textsuperscript{1} mobile testing equipment. The shakers were installed on the roof to induce excitations varying from step-functions, to scaled versions of actual ground motions, operating primarily in the north/south and east/west directions (E. Yu et al [5]).

Inside 3 different rooms, of the building we installed benchshelf systems representing our test environment (see Fig. 4). Each test setup in the selected rooms contained a data acquisition computer system, four cameras, and a set of analog sensors. All cameras were facing the countertop-self system observing the same scene from different angles To simulate a typical laboratory working environment, the countertop-self system was equipped with common laboratory components like microscopes and filled chemical glassware (see also Fig. 4). The analog string potentiometer sensors were placed on top of the counter and attached to the ”to be observed” equipment. Within the building, all of the computer systems were routed into a network box, which was in turn connected to a wireless base station that communicated with the control computer system outside of the building.

EXPERIMENTAL RESULTS

Vision-Based tracking systems under seismic event conditions face additional challenges. Besides uncontrollable light conditions and unpredictable occlusion effects due to falling objects, the cameras themselves are affected by the seismic event. To overcome the shaking camera problem, a reference system in the scene has to be defined to which all detected object movements can be referenced. The chessboard patterns mounted on counter and shelf, as shown in Fig. 4, define such a reference system throughout the collected image sequences. Each of the collected image sequences collected consists of a number of AVI video stream blocks. Each block covers about 82.5 seconds of the test duration with a data size of about 2GB.

To test the capability of our algorithms to reliably detect and track features in a 2D image sequence under these conditions, we used a straightforward approach to extract relative object movement. By subtracting the pixel displacement information of the reference object (chessboard square intersection) from the displacement of an object feature, we extracted the general displacement information stored in a single camera image sequence in the X and Y direction.

Figure 5 shows the detected pixel displacement in the X direction (Image Coordinate system $X_I, Y_I$) through our PIPC for

\textsuperscript{1}Network for Earthquake Engineering Simulation.
the reference system and an object feature \((f_{0})\) on the microscope. Most of the shown displacement in these plots is caused by the movement of the camera itself due to the shaking of the mounting wall inside the building. To calculate real 3D position information an approach has to be found to distinguish between camera displacement and real feature displacement. However with a straightforward approach it is possible to extract qualitative 3D tracking information. The signals in the first part of the sequence appear very similar, which states that no relative movement between reference system and object feature \((f_{0})\) took place. At the end of the sequence, the signals show different displacements which have to be interpreted as the feature on the microscope moving in relation to the reference system. To compare the detected relative pixel displacement with the analog sensor data we used a geometrical approach to convert the image coordinate displacement to the rotated desktop coordinate system \(X_d, Y_d\). Assuming that the mean displacement occurs in \(X_d\) direction and the perspective distortion is minimal, metric displacement data can be calculated when the distance between feature and camera is known. By measuring these distances in the field and applying a fourth order, bandpass, butterworth filter to both data sets, the signals shown in Figure 6 can be extracted. The comparison of the signal from the analog potentiometer sensor with the displacement data from the camera shows that we were able to detect the starting point of the microscope movement at a similar time step. It also pinpoints that the pixel subtraction approach is clearly not usable to filter out the camera movement totally. The low frequency amplitude overlay in the camera plot shows this effect obviously. The amplitude in the camera data also increases over time to about 15% above the measured amplitude from the string potentiometer sensor. This effect reflects the fact that the measured camera object distance has changed over the sequence due to the shaking of the mounting wall.

However this method of generating a 3D position information is not usable as a precise measurement method for 3D feature tracking, it shows that the introduced 2D PIPC is capable to reliably detect and track object features over time without losing any feature. To overcome the shaking camera problem and to analyze the collected data properly we introduced a methodology for calculating the 3D camera position relative to a reference pattern from a single camera image (Doerr, Hutchinson and F. Kuester [6]). This methodology uses the known geometric information from a chessboard pattern to recalculate the perspective transformation matrix (PPTM) of a CCD camera for each image in an AVI stream. Since the PPTM will be calculated in the reference coordinate system (here Dynamic Reference System) two equation systems for each camera view will be available to solve the reverse transformation from image coordinates to real 3D po-
Applying this methodology to the collected data from the field test will enable us to use information from all four cameras of the acquisition system to calculate a 3D feature position.

The PIPC also provides another method to identify general movement in a video stream by detecting pixel changes according to a predefined image in the sequence (background image/model). Detecting these changes over time leads to a trace on the mask history image for all moving object. Figure 7 shows the scene before and after the earthquake simulation. On a first look at both images the difference is not obvious, but the trace image (Fig. 8) shows the displacement history of objects between these two time stamps. The white areas highlights the region in an image where the background image differs more than a given threshold value from the actual image. The gray marked areas represent these regions over the whole sequence. All moving objects will leave this kind of trace in a history image, no matter of the object size or color.

This important kind of information can not be detected with conventional analog sensors. With just one glance on this history image, first response forces would be able to detect were objects have fallen and in which direction. Once the fallen objects are detected, potential hazardous zones, for example in chemical laboratories, can be identified.

CONCLUSION

In this paper we presented our approach to setup an image data acquisition system. The experimental evaluation of the system demonstrates that decent data acquisition rates for a vision based tracking system of about 80fps are achievable using commodity components in a unique hardware setup. We demonstrated also the capability of our pixel-based image processing chain (PIPC) to reliably detect and track image features under seismic event conditions using real field test data sets collected during earthquake simulation test in Sherman Oaks, California.

ACKNOWLEDGMENT

Support for this work was provided by the National Science Foundation, Civil and Mechanical System Division (grant number 0340540), where Dr. Steven McCabe is our program manager. Additional support was provided under a joint program between the Consortium of Universities for Research in Earthquake Engineering (CUREE) and Kajima Corporation. Assistance was provided by Mr. Bob Kazanjy, Development Engineer, and other research staff of the Structural Engineering Test Hall (SETH) and the Visualization and Interactive System (VIS) Group at the University of California, Irvine. The above financial and other support is greatly appreciated.

REFERENCES