# Smartphone Detection of Collapsed Buildings During Earthquakes

**Aku Visuri**
University of Oulu
Oulu, Finland
aku.visuri@oulu.fi

**Zeyun Zhu**
University of Oulu
Oulu, Finland
zeyun.zhu@ee.oulu.fi

**Denzil Ferreira**
University of Oulu
Oulu, Finland
denzil.ferreira@oulu.fi

**Shin'ichi Konomi**
Kyushu University
Fukuoka, Japan
konomi@artsci.kyushu-u.ac.jp

**Vassilis Kostakos**
The University of Melbourne
Melbourne, Australia
vassilis.kostakos@unimelb.edu.au


*UbiComp/ISWC'17 Adjunct* , September 11–15, 2017, Maui, HI, USA

ACM 978-1-4503-5190-4/17/09…$15.00
https://doi.org/10.1145/3123024.3124402

## Abstract
The leading cause of death during earthquakes is the collapse of urban infrastructures and the subsequent delay of emergency responders in identifying and reaching the affected sites. To overcome this challenge, we designed and evaluated a crowdsensing system that detects collapsed buildings using end-user smartphones as distributed sensors. We present our evaluation of smartphones' accuracy in inferring a building collapse by detecting falls onto solid surfaces, and estimating the false positive rate. Further sensors can present more detailed information about each potential collapse event. We conduct simulations to identify strategies for dealing with false-positive data under scenarios of varying population density. Potential building collapses can be determined with 95% accuracy given 10 connected devices within a 125m radius, increasing to 99.99% for 50 devices. End-user devices can proactively offer valuable help to emergency responders during earthquakes, potentially saving lives.

## Author Keywords
Emergency / disaster management; infrastructure sensing; mobile devices

## ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous

## Introduction

During the modern era (1900-), the main cause of death during earthquakes is the collapse of buildings, accounting for over 75% of the casualties [1]. In January 1995, "The Great Hanshin Earthquake", also known as the Kobe Earthquake, hit Kobe, Japan. In the highly populated area, over 5000 people were killed, mostly due to collapsed housing. During the Athens 1999 earthquake, 127 of 143 fatalities were attributed to collapsed buildings, and during the Tōhoku earthquake in 2011, more than 100K buildings were destroyed and more than 10K people lost their lives.

High-fidelity seismic sensor stations are deployed for early detection of earthquakes, but little effort has been put into detecting collapsed buildings. Identifying collapsed buildings mostly involves image processing and aerial photographs, which can introduce delays of hours. This makes planning rescue operations challenging, inefficient, and this can increase the mortality rate [6].

Commodity every-day devices equipped with sensing capabilities are increasingly available, with an estimate 87% of connected devices by 2017 being tablets and smartphones. In our work, we take advantage of the sensing capabilities of smartphones, and introduce a system designed to identify potential building collapses during catastrophic earthquakes. We design an Android application to monitor for fall events (*i.e.,* devices falling from surfaces like desks), which often occur in buildings hit by earthquakes. We stream said fall events to a centralised server from multiple smartphones, that function collectively as a crowdsensing system.

## Contribution

Smartphones are leveraged during earthquakes by systems such as "ShakeMaps" [8], providing real-time estimates using traditional seismic sensing networks. Another approach is leveraging crowdsourcing from smartphones, *e.g.*, the "Did You Feel It" application [7], but data entry is often the first priority for users in the affected areas. Lastly, desktop and portable computers can be equipped with MEMS accelerometers, which are leveraged by the Quake-Catcher Network [2], but identifying seismic events from everyday use is challenging.

Although smartphone sensors' precision is not on par with that of hi-fidelity bespoke sensors [4], they are much more densely distributed in urban areas where fatalities are likely, and in greater number [3]. Previous work has leverage smartphones for fall detection, as falls are a major health risk, especially among elderly people. Our work achieves fast detection of potential building collapses, which is the leading cause of death during earthquakes. This is an issue that surprisingly few projects have addressed, despite its *direct* impact on loss of life. Our method is based on crowdsensing and uses smartphone built-in sensors to detect isolated fall events, followed by server-side clustering analysis to account for false-positives and detect regions with potentially collapsed buildings. The intuition behind our work is that although individuals may drop their phones by accident, a large number of devices dropping simultaneously at the same location, and *e.g.,* remaining stationary afterwards, indicating they were left behind or buried in rubble, may be strongly indicative of a collapsed building and can help in planning rescue operations.

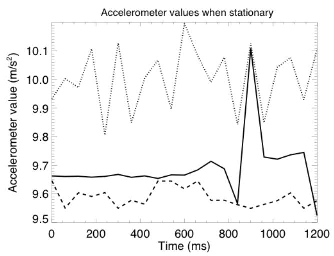


**Figure 1:** Accelerometer values for three stationary devices.

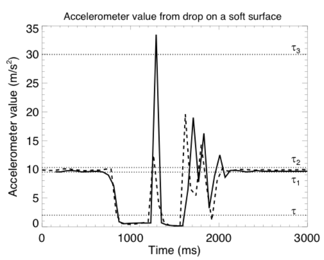


**Figure 2:** Device falling on a soft surface, *e.g.*, a cushion.

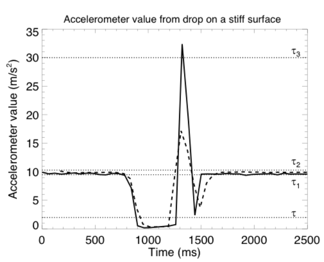


**Figure 3:** Device falling on a stiff surface, *e.g.*, hard floor.

## Design

We design a two-tier system architecture to detect collapsed buildings based on sensor data from mobile devices. This two tier architecture forms the entire Collapse Detection System (CDS). In practice, a cluster of servers would be required to overcome the limitations of performing cluster analysis and support increasing amounts of concurrently connected devices.

*Collapse Detection*

We contacted researchers from the geology department of our University to gather more insights in the different types of earthquakes, their characteristics, and how they affect buildings. Based on these discussions, we inferred that building collapse is most likely to occur in a scenario where the earthquake causes sideways movements in the earth and subsequently, in buildings. During these types of earthquakes objects, such as people or furniture, move in a similar fashion. This causes smaller objects situated on top of them to also move and fall down. This type of behaviour can be seen in both simulated[1] as well as high magnitude real-world earthquakes[2].

We designed a rule-based algorithm to detect device falls. We decided on a rule-based approach to reduce the battery consumption caused by unnecessarily complex computation being run on the device constantly. On a high level, our algorithm relies on individual devices detecting a fall event and remaining stationary, which suggests that they have dropped from a surface. Subsequently, our system performs clustering across the aggregated device data to confirm whether a potential building collapse has actually taken place.

Each fall event follows a structure that consists of three stages. Figure 1 shows the accelerometer values from stationary devices, and Figure 2 and Figure 3 showcase the accelerometer values when a device falls on a soft or stiff surface, respectively. We can identify the 'freefall' period (900ms to 1250ms in Figure 2), the moment when the device has an impact with the surface (1250ms to 1500ms), and the resulting stationary period (1500ms-) that indicates that a device fell down. We use three longitudinal thresholds – T(low), T(high), and T(mid) - to identify these three stages in tandem, with a prerequisite minimum time threshold for each stage to verify a fall from a required height, and that the device remained stationary after the fall. If these conditions are met we can assume that the device has fallen.

We evaluate the fall detection algorithm in a laboratory setting by pushing three different devices (LG G3, OnePlus One, Samsung Galaxy S2) down from two heights (15cm and 70cm). Falls registered from 15cm are false positives (<8% false positive rate), and falls registered from 70cm are considered true positives. Drops to a stiff surface were identified with 98.7% (N = 50) accuracy and drops to soft surfaces with 95.6% (N = 50) accuracy. The only inaccurately identified cases were with the Galaxy S2 – which was also the oldest model in our tests. We also analysed the false negative rate (cases where no fall event was registered), and noticed that a strong rotating movement caused

[1] World's Largest Earthquake Test: Simpson Strong-Tie, Magnitude 7.5, https://www.youtube.com/watch?v=hSwjkG3nv1c

[2] Material from a 7.0 magnitude earthquake in Haiti, January 12 2010: https://www.youtube.com/watch?v=Xr6CFIHJvhw

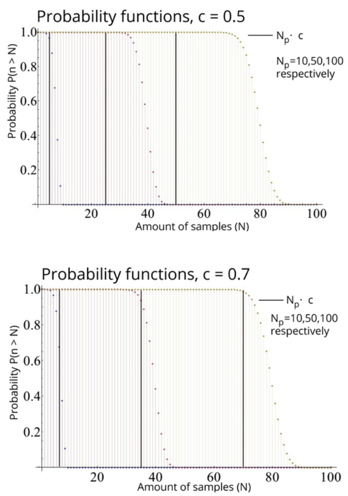


**Figure 4.** Binomial distribution for different ratios (c=0.5, c=0.7) for verifying a collapse event within a region.

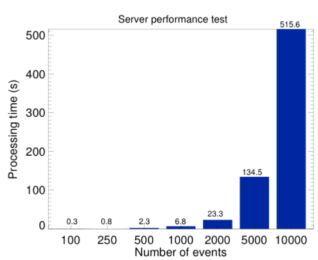


**Figure 5.** Processing time with increasing amount of simultaneous fall events.

difficulties to the algorithm to identify a fall event, and the false negative rate was 24% for these events (N = 50). For the non-rotating falls, the false negative rate was zero.

We then evaluated the algorithm *in-the-wild* by recruiting 7 young adults (6 male, 1 female, 22-33 years old) and deployed our software on their personal smartphones for a period of one week. A total of 9 false positive fall events were registered during this period. This is equal to a rate of 1 false-positive event per device per 5.4 days. Participants reported that activities such as walking with the phone in a pocket, or in a backpack, biking, small drops or jumps did not trigger any false positive events. The cause of the reported fall events during the test period was unknown.

Additionally, shortly after recruiting our participants, we asked them to jump 50 times in place with their device in their pockets or hands in an attempt to prompt false positives. This activity triggered three false-positive earthquake events, each when the device was held in from 20cm height, without resulting in false-positive events. No false-positives were registered when participants were asked to run, including running up and down stairs with phones in their hands or pockets. No false positives were detected when participants were asked to vigorously shake their phones or otherwise attempt to trigger false positives. Vibrations from the device's motors did not trigger a false-positive event.

*Server-side analysis*

Each smartphone is connected to a server, and transmits a timestamp and location coordinates periodically. The locations of the devices are clustered using the *k*-means clustering algorithm. The value of k is selected according to the maximum distance between devices so that the generated regions have a radius of 125m (approximate size of a building block). If a smartphone registers a fall event, it sends the information along with a timestamp and location coordinates to the server.

Once fall event is registered on the server, we confirm that a potential building collapse event has occurred by analysing the number of reported collapsed events and the devices within this region. As our system still contains a (low) probability for a false negative event, we formulate how many devices within a region need to report a fall event to verify a potential building collapse. Using binomial distribution with different ratios (c=0.5 or c=0.7) of devices registering a fall in a region to the number of devices within that region, and different number of devices (N = 10, 50, 100).

We plotted the value c for each of the three distributions in Figure 4. A collapsed building true positive detection is most likely during an earthquake for N = 50 and N = 100, and for c=0.5 and c=0.7. If the number of devices in a cluster is 10, this implies a 30% chance that an actual earthquake would not be detected as such. On the other hand, using c=0.5, the probability of fulfilling the condition is larger than 95% (*i.e.,* only a 5% chance of false-positive detection). Hence, to obtain a reliable building collapse detection for cases where N = 10, we adopted c= 0.5. Note that as the value of c *increases*, the rate of false-positives *decreases*, but there is also increased chance of non-detection of a real collapse if the number of phones in a cluster is small.

Finally, we evaluate the processing time to cluster and analyse an increasing number of fall events. The computation tasks is performed on a t2.small AWS instance (CentOS 6.6, single core 2.6GHz CPU, 2GB of RAM). Figure 5 shows the processing time getting incrementally higher as the device count increases. Based on these computation times, the maximum number of devices connected to a single server with equal modest capabilities should not exceed 5000 since the computation time of more than 2 minutes quickly makes the system obsolete for its intended purposes. In practice, multiple and bigger servers could be clustered together to keep the clients of the same geographical region connected, thus following a distributed server architecture instead of a centralized server approach.

## Discussion and Limitations

In this paper, we present a building collapse detection system, aimed at speeding up emergency response in the aftermath of earthquakes. Our system relies on citizen's smartphones as distributed sensors, thus taking advantage of their widespread availability, and much higher geographic density than bespoke earthquake monitors.

We did not manage to completely eliminate false-positives on clients, due to the variability of everyday activities. Our results show that a false positive may be triggered on average every 5.4 days on each device. For a city with a million citizens, this means about 127 false positive fall events per minute.

A key constraint to consider is the availability of internet connectivity after catastrophic events. Much of the subsequent communication network overload can be attributed to increased number of phone calls in the aftermath of an earthquake. In that regard, our system's propensity to immediately report drop events is likely to take place before individuals begin to make calls in the aftermath, and hopefully avoid the congestion period, and the data packages communicated by our implementation are small. Clearly, there is little our system can do if the catastrophic event causes physical damage to network connectivity, but these issues are also being constantly researched and improved [5].

Other limitations are also evident, as registering multiple smartphone falls do not necessarily indicate a building collapse. Further information should be collected from such devices to verify that they are in an environment that suffered from a building collapse. Sensors, such as light, temperature, gyroscope, or magnetometer can describe the environment and allow the device – if still connected – to further validate whether the device resides within a collapsed building. *E.g.*, increasing temperature, low light values, and the device laying down in an abnormal posture can be indications of the device being under rubble, or close to a fire. Such measurements should be carefully evaluated, and analysing each would likely be a publication on their own.

## Conclusion

Our work outlines a crowdsensing system for detecting building collapses during earthquakes. Our system uses citizen's smartphones as distributed sensors. We demonstrate the performance of the system under varying scenarios, and optimise the analysis of the collected data to reduce the rate of false-positives. Our work contributes to the growing body of literature demonstrating how crowdsensing can be used to study geographic phenomena. In our case, a timely analysis and identification of collapsed buildings can potentially save lives in the aftermath of severe earthquakes.

## References

1. AW Coburn, RJS Spence and A Pomonis. Year. Factors determining human casualty levels in earthquakes: mortality prediction in building collapse. In *Proceedings of the 10th World Conference on Earthquake Engineering*, 5989-5994.
2. Elizabeth S Cochran, Jesse F Lawrence, Carl Christensen and Ravi S Jakka. 2009. The quake-catcher network: Citizen science expanding seismic horizons. *Seismological Research Letters*, 80 (1). 26-30.
3. D. Ferreira, V. Kostakos and I. Schweizer. 2017. Human Sensors on the Move. In Loreto, V., Haklay, M., Hotho, A., Servedio, V.C.P., Stumme, G., Theunis, J. and Tria, F. eds. *Participatory Sensing, Opinions and Collective Awareness*, Springer International Publishing, 9-19.
4. Denzil Ferreira. 2013. AWARE: A mobile context instrumentation middleware to collaboratively understand human behavior *Department of Computer Science & Engineering*, University of Oulu.
5. Takeo Ogawara, Yoshihiro Kawahara and Tohru Asami. Year. Disaster-tolerant authentication system for NDN using hierarchical ID-based encryption. In *Network Protocols (ICNP), 2013 21st IEEE International Conference on*, IEEE, 1-2.
6. Carl H. Schultz, Kristi L. Koenig and Eric K. Noji. 1996. A Medical Disaster Response to Reduce Immediate Mortality after an Earthquake. *N Engl J Med*, 334 (7). 438-444. 10.1056/NEJM199602153340706
7. David J Wald, Vincent Quitoriano, Lori A Dengler and James W Dewey. 1999. Utilization of the Internet for rapid community intensity maps. *Seismological Research Letters*, 70 (6). 680-697.
8. David J Wald, Vincent Quitoriano, Thomas H Heaton, Hiroo Kanamori, Craig W Scrivner and C Bruce Worden. 1999. TriNet "ShakeMaps": Rapid generation of peak ground motion and intensity maps for earthquakes in southern California. *Earthquake Spectra*, 15 (3). 537-555.