On Acquisition and Analysis of Visual Data for Crowdsourcing Disaster Response

Hien To, Sasan Tavakkol, Seon Ho Kim, Minh Nguyen, Cyrus Shahabi
Integrated Media Systems Center
University of Southern California
Los Angeles, CA, U.S.A
{hto,tavakkol,seonkim,minhnngu,shahabi}@usc.edu

Abstract—With the popularity and pervasiveness of mobile devices, crowdsourcing data collection and analysis has emerged as an effective and scalable solution to disaster response. This paper addresses the problem of crowdsourcing mobile videos for disasters by identifying two unique challenges of 1) prioritizing visual-data collection and transmission under bandwidth scarcity caused by damaged communication networks and 2) analyzing the acquired data in a timely manner. We introduce a new crowdsourcing framework for acquiring and analyzing the mobile videos utilizing fine granularity spatial metadata of videos for a rapidly changing disaster situation. We also develop an analytical model to quantify the visual awareness of a video based on its metadata and propose the *visual awareness maximization* problem for acquiring the most relevant data under bandwidth constraints. The collected videos are evenly distributed to off-site analysts to collectively minimize crowdsourcing efforts for analysis. Our simulation results demonstrate the effectiveness and feasibility of the proposed framework.

Index Terms—Crowdsourcing, Disaster Response, Geo-tagged Video

1 Introduction

Enhancing situational awareness is of great importance for disaster response and recovery. In the event of disasters, situational awareness can be enhanced by data acquisition and analysis. Data acquisition refers to the efficient collection of data in a timely manner while data analysis represents the process of identifying and understanding the critical incidents from the collected data. Prompt and accurate data acquisition and effective analysis empower decision makers, which in turn can expedite disaster recovery, minimize damages, and potentially save lives. For example, after 2010 Haiti earthquake, the Ushahidi-Haiti project (i.e., Mission "4636") gathered more than 80,000 text messages from on-site users (on-site rescuers, local people, etc.) of which nearly 60,000 were translated into English by Haitians and were sent to the first responders for search and rescue and other emergency activities [11]. Using the collected data, off-site volunteers created a post-disaster map of Haiti to help on-site workers, revealing undamaged roads, bridges, buildings, hospitals, and shelters.

Disaster data can be acquired in many ways such as from automatic sensor readings, reports from field workers and civilians, etc. In contrast to sensor readings, which is limited to a fixed set of locations and suffer from infrastructure damages, crowdsourcing has been shown to be a cost-effective and time-efficient way to acquire disaster data [10], [5] and then to analyze the collected data [20], [24], [28], [30]. Among various media types (text, image, video, graphics, etc.) from multiple data sources, videos and images are most effective in understanding the disaster situation. Videos can be watched and easily understood by international analysts, independent of language and cultural barriers, without wasting time for inaccurate interpretations [11]. However, there is little study in utilizing a large amount of videos, especially from ubiquitous mobile devices, for disaster situations. Hence, the primary focus of this paper is on devising a unified crowdsourcing framework aiming for both collection and analysis of user-generated mobile videos.

There exist platforms for crowdsourcing the mobile video collection along with fine granularity of spatial metadata, such as MediaQ (mediaq.usc.edu) and GeoVid (geovid.org). However, similar to aforementioned studies [10], [5], these platforms neglect to consider **prioritizing data acquisition** and thus may be subject to data overload, which is critical under limited network resources due to catastrophic outage [15], [19], [16]. Data triage is a central issue in disaster data collection since video data is large and often redundant. During the critical first response time, redundant data collection wastes not only communication bandwidth to transmit unnecessary data but also analysts' valuable time for manual verification. In sum, more data do not necessarily mean more effective situational awareness.

Once data are acquired, the next challenge is to analyze the collected data in a timely manner and there exist several studies in this area [20], [24], [28]. However, these studies have focused on processing and integrating data, rather than on assigning analysis tasks to analysts. An effective analysis refers to the **assignment of analysis tasks** to a number of off-site analysts in a balanced way such that no analyst becomes a bottleneck in a collective situational awareness. To the best of our knowledge, the only system that distributes the analysis tasks among the available analysts is GeoQ (geo-q.com/geoq/). GeoQ is a crowdsourcing

platform for disaster data developed by the US National Geospatial-Intelligence Agency. GeoQ statically assigns a certain geographical region to an analyst who will be in charge of analyzing all the corresponding data. In this case, the amount of data within each region represent the workload assigned to each analyst. However, such a static assignment is hardly effective as non-uniform distribution of data may introduce regions with a wide variation of data, leading to unbalanced workload per analyst. Thus, an effective task assignment should consider the data distribution across geospatial regions and among the analysts.

Combining MediaQ and GeoQ, while overcoming their shortcomings, we introduce a new four-step framework that seamlessly fuses fast and efficient data collection with effective analysis (Figure 1). First, to facilitate the real-time data sensing, analysis and consequently time-sensitive decision-making, we propose the so-called *metadata first* mechanism, in which the geospatial metadata of videos such as camera location and camera viewing direction [4] are automatically captured and uploaded at the time videos are taken by on-site users (Step 1). The geospatial metadata, which represent the geographical properties of the captured videos with a far less number of bytes than the actual videos, are transmitted to the server first without delivering the large amount of the corresponding video data. Next, we identify the problem of prioritizing data transmission under bandwidth constraints (Step 2), i.e., only relevant videos selected based on their metadata will be transmitted in a priority order. Thereafter, the collected data are assigned to *analysts* by partitioning a large disaster area into manageable regions, so called *work cell* (Step 3); each work cell and its enclosed collected data are assigned to one analyst. Finally, the analysts watch the videos corresponding to their work cells to identify *incidents* (e.g., a building on fire, a collapsed house, road block) from which they evaluate the importance/urgency of their assigned work cells in the form of an *urgency map* (Step 4), i.e., the higher the assigned value, the more urgent the situation in the cell. Without loss of generality, we focus on earthquakes as one of the most common and also well-studied disasters worldwide. However, the generic framework can be adapted to other kinds of disasters. The earthquake damage area can be obtained from USGS ShakeMap (United States Geological Survey).

The first 24 hours are critical in disaster response; thus, automatically crowdsource data collection in a timely manner is important after a disaster. A promising approach is to automatically crowdsource data in high-priority regions of disaster based on the *urgency map*. To compute the urgency map, we use USGS ShakeMaps to measure the earthquake severity (i.e., near-real-time maps of shaking intensity) and population density as the main factors of spatial importance. The population density is dynamically updated over time to capture human mobility. Besides, we also consider various factors, such as the location of schools, bridges, and residential buildings, which are subjects of interest for a particular group of disaster responders.

Given the urgency map, we develop an analytical model to quantify the situational awareness of a particular video, namely *visual awareness*. In practice, the visual awareness of

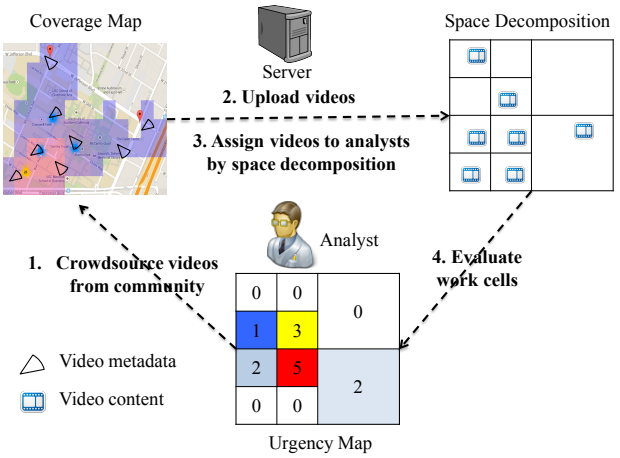


Fig. 1: The crowdsourcing framework for disaster response. The loop is necessarily for dynamic data awareness, which may be critical in a rapidly evolving situation.

a video (or a frame[1]) indicates how relevant the video is to the disaster incidents. Whether the video covers the actual incidents' locations or not is unknown to the control center at the time when only metadata are uploaded. Hence, we define the visual awareness of a video based on the urgency of the video's location and its coverage. Consequently, we define the *Visual Awareness Maximization (VAM)* problem that only selects a set of videos or frames with maximum total visual awareness without exceeding the bandwidth constraints. In other words, this maximizes the amount of useful information obtained from a limited amount of videos delivered under constrained bandwidth.

The bandwidth limit at a given time interval, referred to as *budget*, determines the amount of content that can be uploaded to the server. Our solutions consider the budget constraints with two variations: entire video content needs to be uploaded or individual video frames can be extracted on mobile clients (i.e., keyframes to reduce the data size) and then uploaded. Due to the budget constraints, an approach that simply ranks videos/frames and selects the ones with the highest information does not yield the optimal result. Thus, we study the problem complexity of both variants and prove that they are NP-hard. Particularly, when individual frames can be selected, we propose a solution that minimizes overall redundant coverage of the overlapped frames, therefore, achieve the maximum total visual awareness.

The objective with the partitioning step is to ensure the maximum *collective* visual awareness of the uploaded videos, using any optimal VAM solution. The baseline technique is to use a uniform grid, in which the number of grid cells is determined by the number of analysts. However, the shortcoming of the uniform grid is that, similar to GeoQ, some analysts may be overloaded while the others are underutilized. Therefore, considering the spatial distribution of the videos, we propose two partitioning techniques based on point Quadtree and Kd-tree. These techniques result in an almost equal number of videos assigned to each analyst and thus increases the collective visual awareness of the uploaded videos.

1. A video is represented by a sequence of its frames (Figure 2).

Thus far, we assume that each crowdsourcing iteration in Figure 1 is independent of each other. However, the visual awareness of a video (or a frame) may depend on when they are acquired. Intuitively, an area that has been covered by many analyzed videos should have lower visual awareness than the one covering an uncovered area. Toward that end, we model the visual awareness of a region (or cell) as a function that decays over the number of times the cell being covered. For example, the visual awareness of a cell is inversely proportional to the number of uploaded FOVs covering the cell. This means that the total visual awareness of the cell over time follows harmonic series and thus reaches infinity given an infinite number of frames covering it. Considering multiple crowdsourcing cycles, we measure the total visual awareness across multiple iterations (a.k.a time snapshots). This extension generalizes of VAM as it becomes VAM when there is only one time snapshot.

In a preliminary version of this work [27], we introduced VAM. This article subsumes [27] by extending the framework to multiple snapshots for continuous monitoring the areas of disaster. Moreover, in this article, to evaluate the proposed approaches for multiple time snapshots, we generate synthetic datasets in a more systematic manner using a real-world data from the 2014 South Napa Earthquake. Particularly, we introduced a method to generate the urgency map from multiple data sources. We also included two new real-world datasets for evaluating VAM, GeoUGV [17] and Google Street View [31] and performed new set of experiments for continuous monitoring of disaster area.

To summarize, the specific contributions of this paper are as follows.

(i)) We identify two specific challenges in disaster response, *fast* and *efficient data acquisition* and their *effective analysis* with regards to two existing crowdsourcing projects, MediaQ and GeoQ, and propose a unified crowdsourcing framework to overcome such challenges by leveraging geo-tagged videos.

(ii)) We propose an analytical model to measure the probability that a video covers an actual disaster incident without knowing the locations of the incidents, termed *visual awareness* of the video. We formulate an optimization problem to select a set of videos with the *maximum visual awareness* under bandwidth constraints, and propose to use a dynamic programming algorithm to solve it.

(iii)) We extend our solution to the case where individual video frames can be uploaded. The improved solution minimizes the redundant coverage in overlapped frames, thus, yields up to an order of magnitude higher visual awareness in comparison to the case of transmitting the entire video.

(iv)) We propose adaptive partitioning techniques considering the spatial distribution of the videos to automatically assign the uploaded videos within a particular work cell to the corresponding analyst.

(v)) We conduct experiments on various synthetic and real-world datasets to show the effectiveness and efficiency of the proposed framework. We conclude that the data-dependent partitioning techniques outperform the baseline by an order of magnitude.

The remainder of this paper is organized as follows. In Section 2, we review the related work. Section 3 discusses the preliminaries necessary to present our framework. In Section 4, we introduce the framework and define the constraint optimization problem, VAM. Thereafter, we present an enhancement to the VAM problem in Section 5, followed by our proposed method to calculate the urgency map in Section 6. We present the experimental results in Section 7 and make the conclusion of the paper in Section 8.

2 Related work

Crowdsourcing Disaster Response: Crowdsourcing has been widely regarded as a cost-effective and time-efficiency means in disaster management, especially in data collection and analysis under disaster situations [10], [5], [20], [20], [24], [28], [30]. Firstly, early efforts in disaster data collection focused on geographic information provided voluntarily by individuals [10]. Chu et al. [5] developed a disaster surveillance and response system that provides the global view of the situation of the off-site users (e.g., analysts) with the help of on-site users (field commanders, local people, etc.). Secondly, regarding crowdsourcing the analysis of disaster data, Ortmann et al. [20] conducted a study on processing and integration of data associated with disasters by leveraging Linked Open Data (linkeddata.org). Schulz et al. [24] proposed to combine human and machine intelligence for enhancing the situational picture of the off-site users, resulting in an increased situational awareness. In [28], the authors discussed the feasibility of harnessing collective knowledge in disaster relief and presented a generic system architecture with examples of how this could be accomplished. Yang et al. [30] proposed a platform that provides real-time assistance to on-site users by leveraging off-site users' efforts. Despite efficient data collection and analysis, the credibility of the crowdsourced disaster data is still a major concern [11], [10]. Our study aims to focus on both efficient data collection and effective analysis of user-generated videos concerning disasters to advance capabilities for situational awareness. Fast and efficient data collection is achieved by prioritizing data transmission under limited bandwidth while effective analysis is obtained by evenly distributing the collected data to the analysts.

Anti-disaster Systems: Recently, there has been a growing research interest in improving the resilience and responsiveness of emerging computer systems to facilitate real-time data sensing [15], [19], [16], which is critical for time-sensitive decision-making. In [15], the author presented the infrastructure damage caused by the Great East Japan Earthquake such as transmission cables, mobile stations, etc., and Japan's efforts in restoring such telecommunication network. Liu [16] showed a comparison between typical outages and catastrophic outages caused by disasters, i.e., inaccessible power, damaged or unavailable communication network facilities. In [19], the authors surveyed several studies on resilient information and communication technologies, such as satellite network platform and anti-disaster information distribution platform. Among such systems, unmanned aerial vehicles (UAV) have emerged as effective controlled/autonomously systems in disaster imagery collection, especially in areas that are inaccessible on the road.

For example, Google built aerial drones that can deliver medical equipments, food to people in need across the country. Skybox (skyboximaging.com) developed a satellite acquisition technology to collect real-time satellite imagery and full-motion video from space on demand. In contrast to these studies, our framework shows the possibility of an anti-disaster information distribution platform that collects a vast amount of videos concerning disasters. The collected data, which can comes from various sources, such as mobile phones, UAVs, conventional CCTVs, facilitate comprehension of the situation and better decision-making.

3 PRELIMINARIES

As this study was inspired by the two existing platforms, MediaQ [14] for data acquisition and GeoQ for data analysis, in this section, we introduce them and related concepts.

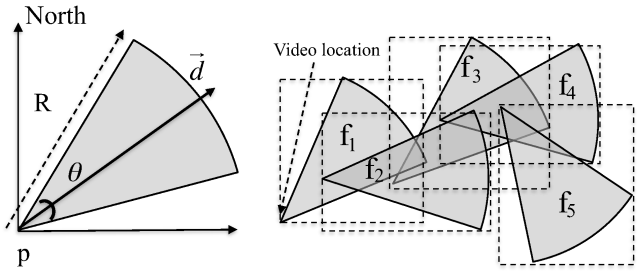
3.1 MediaQ for Data Acquisition

Crowdsourcing disaster data contribute to stages of disaster response in a scalable, cost-effective and real-time manner. However, verifying crowdsourced information is critical for decision-making in disaster response as time lost responding to inaccurate reports may outweigh the benefit of the provided information. For example, less than 6% of the text messages published on the Ushahidi-Haiti crisis map were tagged as "verified" by field reporters [11]. Fortunately, visual data such as images and videos with spatial metadata and content can be verified easily by off-site analysts without the need of the field reporters. Thus, we developed MediaQ for collecting videos with their metadata from community, voluntarily or on-demand manner. With crowdsourcing, off-site analysts can outsource their content requests at particular locations that will automatically generate push messages to nearby workers, i.e., individuals with mobile devices that perform the requests by physically traveling to the specified locations and recording videos.

Geo-tagged Videos: Mobile videos can be captured at a fine granular level (e.g., frames) and their geospatial metadata (e.g., camera location, viewing direction) are transparently associated with each frame. This capability is referred to as geo-tagged videos. Particularly, we represent a video as a sequence of video frames, or frames for short, and each frame is modeled as a field of view (FOV) [4]. In 2D space, the field-of-view of a camera at a particular time forms a pie-slice-shaped area as illustrated in Figure 2a. We formally define a field of view.

Definition 1 (Field of View (FOV)). *A FOV f is denoted as $(p, \overrightarrow{d}, R, \theta)$, in which p is the camera location of <latitude,longitude> coordinates, the camera direction $\overrightarrow{d}$ is obtained based on the orientation angle provided by a digital compass, the camera viewable angle θ describes the angular extent of the scene imaged by the camera. The angle θ is calculated based on the camera and lens properties for the current zoom level, R is the maximum visible distance at which a large object within the camera's field-of-view can be recognized.*

The viewable scene of a camera changes as it moves or changes its orientation. In order to keep track of the FOVs of a moving camera over time, we need to record its location p, direction $\overrightarrow{d}$ and viewable angle θ with a certain frequency and produce time-stamped metadata together with time-stamped video streams. Our meta-data streams are analogous to sequences of $(p, \overrightarrow{d}, R, \theta, t)$ quintuples, where t is the time instant at which FOV information is recorded. Figure 2b depicts FOVs of a 5-seconds video; one frame is sampled per second. For simplicity, we assume that the video location is the first point of a trajectory of video.



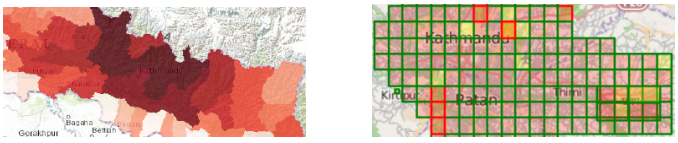
(a) Illustration of FOV (b) MBR estimations of FOVs

Fig. 2: The Field Of View (FOV) model.

As shown in [4], one issue with such a representation is the computational overhead. A more appropriate approach is to define the FOV in the spatial domain with a pie-slice-shaped area and then estimate it with a minimum bounding rectangle (MBR) as shown in Figure 2b. Consequently, we estimate the coverage area of a video as the overall coverage of its FOVs' MBRs. To efficiently compute the coverage of the MBRs, we use *cascaded union*, which is available in various languages such as PostGIS and Python.

3.2 GeoQ for Data Analysis

The disaster data are often analyzed on a crisis map to provide the overview of the disaster situation at the control center. Crisis mapping techniques often evaluate and annotate damage based on a geographical (district) map imported from popular geospatial vector data formats such as Shapefile. For instance, Figure 3a shows the color-coded damage levels in Nepal Earthquake 2015. The darker the color, the more damaged the districts. However, district-based evaluation fails to represent the damage at fine granular level due to rigid pre-defined geographical regions. Therefore, GeoQ uses a grid-based partition of the space to enable fine-grained evaluation as illustrated in Figure 3b.



(a) ArcGIS's district regions (b) GeoQ's work cells

Fig. 3: Crisis mapping tools used in Nepal Earthquake 2015.

GeoQ allows the analysts to collect geographic structured observations across a large area, but manage the work in smaller geographic regions. That is, a disaster area can be partitioned into small regions (e.g., 1km squares), so called

work cells, and be assigned to the analysts. GeoQ also assists the analysts to aggregate and analyze information from the data concerning disasters. The role of the analysts is to evaluate the available data sources in their allocated work cells (e.g., video data from MediaQ, social data from Twitter and Youtube, imagery data from satellites) to determine any relevant incident associated with disasters (e.g, a building on fire, road block) that needs to be marked on the GeoQ's crisis map. To become an analyst, volunteers need to pass required training classes on a particular disaster type (e.g., earthquake, wildfire).

We formally define a work cell and an analyst.

Definition 2 (Work Cell)**.** *A work cell w is a region with an urgency value U that can be rated by an analyst.*

Definition 3 (Analyst)**.** *An analyst is a trusted personnel with expertise in situational crisis. By analyzing data within a work cell, the analyst measures the severity of the disaster and sets an urgency value to his assigned work cell.*

4 Crowdsourcing Disaster Response

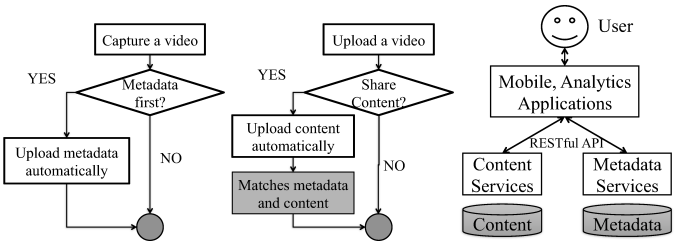
We propose a unified framework that empowers MediaQ and GeoQ, but overcomes their limitations in crowdsourcing disaster data and data analysis. We first focus on efficient mobile video acquisition and transmission.

4.1 Data Acquisition

4.1.1 Acquisition of Video Metadata and Content

First, one critical issue in acquiring videos is the timely delivery of data, especially under a potentially catastrophic disruption of communication during and after a disaster. For example, in 2010 Haiti Earthquake, 85% of Haitians could still access to their mobile phones but 70% of cell towers were destroyed [11]. Therefore, we propose "metadata first" mechanism that prioritizes uploading metadata of videos (the quintuples in Section 3.1) over their content, in which metadata are automatically captured and uploaded when the videos are taken without delivering a large amount of the corresponding video data themselves. The reason for this is to enable time-sensitive acquisition and analysis on the uploaded metadata, such as real-time data sensing, visualization (e.g., video coverage map in Figure 1) and decision-making (e.g., crowdsource more data in sparse-video areas such as the Northeast in Figure 1). Other reasons for separately handling metadata first includes supporting data governance as metadata often lives longer than its content, preserving privacy and strict access control [7].

Due to the small size of metadata with respect to the content[2], they can be transmitted through various channels such as the Internet, SMS and WLAN. The acquired metadata can be used in data management applications, which enable other applications to access the metadata via RESTful APIs. For example, using the RESTful metadata services, a range query can find all video frames that overlap with a user-specified region, or a direction query can find the objects that are covered by a video with a specific viewing direction. This kind of queries is particularly useful when analysts have identified an incident and in search for more videos that cover the event.



(a) Upload metadata (b) Upload content (c) Architecture

Fig. 4: Work-flows for uploading video data in MediaQ.

Second, by leveraging the spatial metadata, the server periodically selects a set of videos to be uploaded, e.g., every five minutes. Once the server identifies the videos, to be discussed in the next sections, their contents can be requested. That is, the server sends push notifications to the mobile phone clients that hold the selected videos, triggering the service to upload the videos. Mobile phone users can choose to opt-in/opt-out the option of automatically transmitting the video content when receiving the notifications; otherwise, the users can directly upload the content. The content once uploaded is matched to the corresponding metadata stored in the server. Figure 5 illustrates the work-flows for uploading video data in MediaQ.

The above selective on-demand data collection efficiently uses network bandwidth by minimizing redundant data transmission. The total volume of uploaded videos, within a time interval, is limited by the network capacity, namely *budget* B. In practice, budget B captures the bandwidth constraints from either the communication network or the cloud server that stores the videos [1]. We formally define the budget.

Definition 4 (Budget)**.** *Budget B is the maximum amount of data (MB) to be transmitted in a given time interval.*

In practice, when bandwidth is not a concern, B can capture the resource constraint on the number analysts, in which each analyst can analyze the maximum amount of video at a given time interval.

4.1.2 Visual Awareness of a Video

Given uploaded metadata, videos will be prioritized for their transmission so that relevant or urgent ones can be delivered first. We develop an analytical model that allows the server to quantify the importance of a particular video or frame, namely *visual awareness*. In practice, the visual awareness of a video indicates the probability that it covers any interesting incidents. Whether the video covers an actual incident or not can be confirmed only when the video is uploaded and evaluated by the analysts. Thus, it is intuitive to define the visual awareness of the video based on its geospatial metadata.

$$VA(v) = area(v)U(v.l) \quad (1)$$

2. The size of metadata of one FOV is around 300 bytes [1], thus, the metadata of a ten-seconds video, with a sampling rate of one FOV per second, is 3KB only. However, the size of a video is typically a few MBytes, which is thousands of times larger than its metadata.

where $area(v)$ is the coverage area of video v, calculated using *cascaded union* as described in Section 3.1; $U(v.l)$ is the *urgency value* at the location of video v, either automatically computed or manually updated by the analyst being assigned to v as will be shown in Section 4.2.2 and 6, respectively. The intuition for Equation 1 is that $VA(v)$ is high if both the urgency of the location $U(v.l)$ and the coverage area of the video $area(v)$ are large. Without loss of generality, in this study, the value of urgency is between 0 and 10 while the unit of area is km^2.

4.1.3 Visual Awareness Maximization

To decide the order of video transmission, the server selects a set of videos with the maximum total visual awareness without exceeding the budget constraint.

Problem 1 (Visual Awareness Maximization (VAM_V))**.** *Given budget B for a time interval and the video set $V = \{v_1, v_2, ...\}$ the budget-constraint maximization of visual awareness is the problem of selecting a set of videos such that the total visual awareness $\sum_{i=1}^{|V|} VA(v_i)d(v_i)$ is maximized while satisfying $\sum_{i=1}^{|V|} size(v_i)d(v_i) \leq B$.*

$d(v_i)$ represents a decision to select the i^{th} video: $d(v_i) = 1$ if video v_i is selected and $d(v_i) = 0$ otherwise. By restriction, we proof that the VAM_V[3] problem is NP-hard by a reduction from the 0-1 knapsack problem [2].

Theorem 1. *The VAM_V problem is NP-hard.*

Proof. Suppose that the maximum weight we can carry in the bag is W. With 0-1 knapsack, given n items, z_1 to z_n where z_i has a value $value(z_i)$ and weight $weight(z_i)$, we need to maximize the sum of the values of the items in the knapsack so that the sum of the weights must be less than or equal to the knapsack's capacity. More formally, we maximize $\sum_{i=1}^{n} value(z_i)d(z_i)$ while satisfying $\sum_{i=1}^{n} weight(z_i)d(z_i) \leq W$ and $d(z_i) \in \{0, 1\}$.

We prove the theorem by providing a one-to-one correspondence from 0-1 knapsack to the VAM_V problem. That is, given an instance of the knapsack problem, there exists a one-to-one mapping to an instance of VAM_V. For every item z_i, we create a video v_i with $VA(v_i) = value(z_i)$ and $size(v_i) = weight(z_i)$. Also, the maximum weight W is mapped to budget B. This simple mapping completes the proof. □

By a reduction from the 0-1 knapsack problem, we can use any algorithm that computes 0-1 knapsack to solve the VAM_V problem. It has been shown in [18] that the greedy algorithm gives 0.5-approximation ratio. Fortunately, there is a pseudo-polynomial time algorithm using dynamic programming to optimally solve 0-1 knapsack. This solution runs in $O(|V|B)$ time and $O(B)$ space, where $|V|$ is the number of videos and B is the budget.

4.2 Data Analysis

Once the data are acquired, they are distributed to analysts who then evaluate them. In this section, we present the problems of task distribution and task analysis in turn, which correspond to Step 3 and 4 in Figure 1, respectively.

3. The subscript V denotes the video-level optimization problem.

4.2.1 Task Assignment

To facilitate timely evaluation on the acquired data, we investigate various partitioning techniques to evenly assign work cells and the enclosed videos to the analysts. We propose to adaptively partition a large disaster region into work cells and automatically assign each work cell to an analyst. As a result, each analyst is assigned the videos within his work cell that have not been yet reviewed. For simplicity, we assume that the disaster region is a rectangle and one analyst is responsible to one and only one work cell. In the following, we present the uniform grid as a baseline and two other techniques based on Quadtree and Kd-tree; both take the spatial distribution of the videos into consideration.

Data-independent Partitions: Given A analysts, we partition the disaster region into an equal-size grid of size $\lfloor\sqrt{A}\rfloor \times \lfloor\sqrt{A}\rfloor$ so that each work cell is assigned to at least one analyst. For example, the disaster region will be split into 8×8 grid given 64 analysts. As a data-independent technique, the equal-size grid may suffer unbalanced allocation of videos, i.e., some work cells have many videos while the others are empty. Consequently, the analysts with empty work cells are idle while the others may be overloaded.

Data-dependent Partitions: To enable balanced assignment, we propose data-dependent techniques based on Quadtree and Kd-tree [23]. The point quadtree algorithm recursively decomposes the space into adaptable cells. A cell is split into four equal quadrants or regions if the number of data points within the cell is larger than a maximum capacity.

We propose an algorithm for space partitioning based on Quadtree and Kd-tree structures with a customized stop condition (Algorithm 1). Unlike the stop condition of the point quadtree, Algorithm 1 terminates when the number of cells is greater than or equal to A (Line 5). This is to ensure that all work cells are assigned to the analysts. Furthermore, at each stage of Algorithm 1, we split the cell with the highest number of videos to maintain balanced workload between the analysts (Line 6). When a parent node is split into four equal quadrants, we move the data from the parent into the corresponding child nodes, NW, NE, SW, SE (Line 9). Finally, Line 11 updates the current number of work cells. Note that a work cell cannot be further partitioned if it has no more than one video (Line 7).

Algorithm 1 QUADTREE (KD-TREE) ALGORITHM

```
1:  Input: uploaded videos U = {u1, u2...}, analyst count A
2:  Initialize root work cell ROOT.data = U
3:  Initialize work cell count cell_count = 1
4:  Initialize priority queue Q = {ROOT}, ranked by video
    count
5:  While cell_count ≥ A and size(Q) > 0:
6:      Work cell with highest video count CELL ← Q
7:      If CELL has more than one video:
8:          Split CELL into four quadrants NW, NE, SW, SE
9:          Move data from NODE to its children
10:         Update queue Q ← Q + {NW, NE, SW, SE}
11:         Update work cell count cell_count ← cell_count + 3
```

The Kd-tree construction algorithm is similar to that of Quadtree, except the splitting criteria in Line 8 needs to be tailored with respect to the point kd-tree algorithm.

Instead of midpoint splitting as Quadtree, median splitting is used, which results in approximately the same number of videos per quadrant. The obvious advantage of Kd-tree over Quadtree and the simple grid is that each analyst has roughly the same number of videos, thus, facilitating concurrent data analysis among them.

Algorithm 2 combines data acquisition and data analysis.

Algorithm 2 VISUAL AWARENESS MAXIMIZATION (VAM_V)

1: Input: video set $V = \{v_1, v_2, ...\}$, budget B, analyst count A
2: Compute visual awareness of each video $VA(v_i)$
3: Perform dynamic programming [18] on V to upload the best videos V_b with maximum visual awareness
4: Distribute videos V_b to A analysts (Algorithm 1)
5: Total uploaded VA: $UploadedVA = \sum_{v \in V_b} VA(v)$

4.2.2 Work Cell Analysis

With GeoQ, the analysts measure the severity of the disaster by analyzing the data within their corresponding work cells, then assign urgency values to them. In the same fashion, we extend the idea of geographic tasking in GeoQ to analyzing the video data. For example, the analysts can count the number of damaged buildings or mark emergency cases by watching their assigned videos. In addition to these specific tasks, analysts can provide an overview of the situation for the decision makers by assigning an urgency value (i.e., priority) to each work cell, e.g., zero means no damage while 10 means severely damaged (urgency map in Figure 1). The urgency values may change over time as more videos are available and to be watched by the analysts.

Due to the complexity of the disaster, we argue that detecting critical incidents, such as fire, flood, smoke and explosion should be semi-automatic. Either the analysts manually watch the uploaded videos in their work cells to identify the incidents, or the server automatically provides descriptions of the events by means of computer vision or machine learning techniques that analyze the videos. These issues are beyond the focus of this work. However, the urgency of the work cells can be automatically recommended based on their *importance* and *geosocial* factors, weighed as follows.

$$U(w) = Importance(w)\alpha + RE(w)\beta \quad (2)$$

While the former one is provided in form of a pre-defined priority map, e.g., nuclear plant areas have higher priority than residence areas, the geosocial factor is represented by region entropy (RE) (entropy of a region is high if many people visit the region). Intuitively, a high-population work cell is more important than the one with fewer people, and the priority of a work cell is high if many people visit, such as schools and hospitals. Location entropy [6], which measures the diversity of unique visitors of a location, can be used to measure the spatial "popularity" of a location. A location has a high entropy if many people visit that location with equal proportions. We extend the concept of location entropy to *region entropy* of a work cell.

For a given work cell w, let O_w be the set of visits to w. Also, let P_w be the set of distinct people that visited w, and $O_{p,w}$ be the set of visits that person p has made to the region w. The probability that a random draw from O_w belongs to $O_{p,w}$ is $P_w(p) = \frac{|O_{p,w}|}{|O_w|}$, which is the fraction of total visits to w that belongs to person p. The region entropy for w is computed as follows:

$$RE(w) = -\sum_{p \in P_w} P_w(p) \times logP_w(p) \quad (3)$$

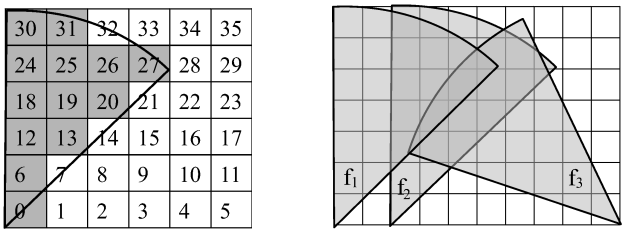
$RE(w)$ can be computed based on any geo-social dataset such as Gowalla[4]. Consequently, we can associate a geosocial priority to every work cell as shown in Equation 2.

5 MINIMUM REDUNDANT COVERAGE

Thus far when a video is selected, the entire video content needs to be uploaded. However, transmitting the content is not only costly but also may render many frames useless, i.e., redundant frames are generated when either users do not move their cameras or videos cover the same area. To reduce the bandwidth usage and therefore maximize the total visual awareness, we propose to upload only keyframes and their metadata to the server and simultaneously minimize redundant coverage of these frames. While the acquisition of metadata and content of a video frame (i.e., an image) is similar to Section 4.1.1, in the following, we focus on identifying the keyframes across all videos.

5.1 Visual Awareness of a Frame

To compute the overlap regions, we divide the space into small grid cells (e.g., 20m squares), so-called *unit cells*; each is identified by a number as shown in Figure 5a. With this discretization, one FOV can be represented by a set of covered unit cells. A unit cell is covered by a FOV if the unit cell's center is within the FOV. For instance, the FOV is represented by the set of gray unit cells in Figure 5a.



(a) Discretization $m \times n$ (b) FOV overlap

Fig. 5: Optimization at the frame level.

Intuitively, visual awareness has composition property, which means that the visual awareness of a FOV is equal to the sum of the enclosed unit cells':

$$VA(f) = \sum_{c \in f} VA(c) \quad (4)$$

where the visual awareness of the enclosed unit cell $VA(c)$ is similar to Equation 1.

$$VA(c) = area(c)\,U(c) \quad (5)$$

where $U(c)$ is the urgency value of the unit cell c.

4. snap.stanford.edu/data/loc-gowalla.html

5.2 Visual Awareness Maximization

To prioritize the video frames for transmission, the server selects a set of frames that maximizes the total awareness without exceeding budget B. With the assumption that the size of all video frames are the same, budget B is equivalent to the maximum number of frames that can be transmitted, $K = \lfloor B/size\ of\ a\ frame \rfloor$. We formally define the problem as follows.

Problem 2 (Visual Awareness Maximization (VAM_F))**.** *Given a budget K for a time interval and a collection of video frames $F = \{f_1, f_2, ...\}$, each frame f_i containing a set of unit cells c_k, the budget-constraint maximization of visual awareness is the problem of selecting a set of frames, denoted by L, such that the total visual awareness of the covered unit cells $\sum_{c_k \in \bigcup_{f_i \in L}} VA(c_k)$ is maximized while satisfying $|L| \leq K$.*

By restriction, we proof that the VAM_F[5] problem is NP-hard by a reduction from the weighted maximum coverage problem (MCP) [12].

Theorem 2. *The VAM_F problem is NP-hard.*

Proof. We proof the theorem by providing a one-to-one correspondence from MCP to the VAM_F problem, or MCP $\leq_p$ VAM_F. Toward that end, given an instance of MCP, denoted by I_m, there exists an instance of the VAM_F problem, denoted by I_v, such that the solution to I_v can be converted to the solution of I_m in polynomial time. The reduction is straightforward by matching from I_m components to I_v components. □

In Figure 5b, given $K = 2$ and the visual awareness of all unit cells is the same, VAM_F selects f_1 and f_3 to minimize redundant coverage.

As MCP is strongly NP-hard, a greedy algorithm is proposed to achieve an approximation ratio of 0.63 [9]. The algorithm chooses a set (i.e., a frame) at each stage that contains the maximum weight (i.e., visual awareness) of *uncovered* elements (i.e., unit cells). Feige and Uriel [9] show that the greedy algorithm is the best-possible polynomial time approximation algorithm for MCP.

Algorithm 3 combines data acquisition and data analysis while minimizing redundant data collection.

Algorithm 3 VISUAL AWARENESS MAXIMIZATION (VAM_F)

1: Input: video set $V = \{v_1, v_2, ...\}$, budget B, analyst count A
2: Partition disaster area into small *unit cells* $\{c_k\}$
3: Compute visual awareness of each *unit cells* $VA(c_k)$
4: Partition each video v_i into a set of FOVs $\{f_1, f_2, ...\} \leftarrow v_i$, F is the set of all FOVs $F = \{f_j\}$
5: Partition each FOV f_j into a set of *unit cells* $\{c_1, c_2, ...\} \leftarrow f_j$
6: Perform max cover [12] on F to upload the best FOVs F_b with maximum visual awareness
7: Distribute FOVs F_b to A analysts (Algorithm 1)
8: Total uploaded VA: $UploadedVA = \sum_{c_k \in \bigcup_{f_j \in L}} VA(c_k)$

5.3 Multiple Time Snapshots

Thus far, we assume the crowdsourcing cycles (illustrated in Figure 1) are reset every time we perform optimization in data collection (Step 2), data distribution (Step 3) data analysis (Step 4). However, the situation after a disaster may change and evolve rapidly and require continuously monitoring the evolving situation. Toward that end, we extend our crowdsourcing framework to consider multiple time snapshots where information, such as urgency map can be updated over time. Subsequently, the visual awareness of a cell is updated correspondingly. Particularly, rather than discarding a covered FOV (i.e., visual awareness is zero) or considering it as a new one (i.e., visual awareness is maximum), we reduce its visual awareness over time. Intuitively, the higher the number of times a FOV has been covered, the smaller its visual awareness. Thus, we model the visual awareness of the FOV as a function that decays over repetitive selections.

Without loss of generality, we assume that the decayed visual awareness of a cell c follows harmonic series. That is, the visual awareness is maximum when it first uploaded, denoted as $VA(c)$. Thereafter, the value decays in the subsequent uploads: $VA(c)/2, VA(c)/3, ...$, etc. One of the reasons we choose harmonic series is that it is a divergent infinite series. We want to model the visual awareness of a cell over time such that its total visual awareness reaches infinity as we have infinite numbers of FOVs covering the cell.

To provide dynamic data awareness, we propose an algorithm to prioritize frame selection over time by decaying its visual awareness as described in Algorithm 4.

Algorithm 4 VISUAL AWARENESS MAXIMIZATION (VAM_M)

1: Partition disaster area into small unit cells $\{c_k\}$
2: Compute the visual awareness of each unit cells $V(c_k)$
3: Total visual awareness $TotalVA = 0$
4: For each time snapshot:
5: Given video set $V = \{v_1, v_2, ...\}$, fixed budget B, analyst count A per time snapshot
6: Apply Algorithm 3 to V to obtain $UploadedVA$
7: Update $TotalVA = TotalVA + UploadedVA$
8: Update visual awareness of the covered *unit cells* $VA(c_k)$

6 GENERATING URGENCY MAPS

We modeled the geospatial visual awareness of a video (or frame) as the product of its coverage area and the *urgency value* at the video's location. As the *urgency value* is computed from an *urgency map*, we present a method to generate the *urgency map*. This urgency map is particularly crucial right after the disaster when the analyzed information from analysts is not yet available. We discretize the disaster area into reasonably small cells and define the urgency matrix as the product of three matrices overlaid on the disaster area namely, *damage*, *exposure*, and *significance matrices*, as given by

$$[urgency] = [damage] \odot [exposure] \odot [significance] \quad (6)$$

where $\odot$ is the elementwise multiplication (aka Hadamard product). The *damage* matrix is obtained from the *intensity map* of an earthquake. The *exposure* matrix captures the expected number of subjects of interest (e.g., nuclear plants, residential buildings, schools, hospitals and bridges). Each subject is assigned a pre-defined priority value, e.g., nuclear

5. The subscript F denotes the frame-level optimization problem.

plant areas have higher priority than residence areas. Hence, the multiplication of the damage matrix and the exposure matrix is the expected value of damage for each subject of interest. Finally, the *significance* matrix captures the significance of different items in each subject category, i.e., a bridge has more average traffic than another. We describe each matrix in details as follows.

6.1 Damage Matrix

We first describe the intensity map, in which the damage matrix is computed from. Among different intensity measures for earthquakes such as peak ground acceleration (PGA), peak ground velocity (PGV), modified Mercalli intensity (MMI), etc. we choose MMI, as it can be better understood by non-experts. MMI indicates the local effects of an earthquake on a scale from I (not felt) to XII (total destruction). For a complete list of MMI levels and associated descriptions see [29]. MMI values depend on the distance from the earthquake epicenter and the local soil characteristics. In our framework, the MMI matrix is automatically obtained from USGS website after a major earthquake. We use the cell size in USGS intensity matrix as the cell size of our discretization.

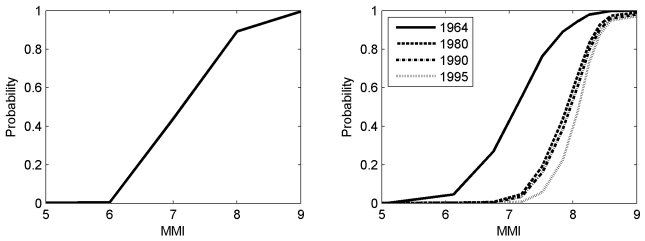
The *damage* matrix for a certain type of structure contains the probability of damage to the structure, given the local intensity of the disaster. The damage itself is normally defined at several levels such as slight, moderate, severe, etc. To simplify the process, we only consider the moderate damage. Therefore, our *damage* matrix is defined as:

$$[damage] = P[moderate\ damage | MMI\ value = i] \quad (7)$$

We use the ATC-13 [22] masonry buildings fragility curve to assemble the *damage* matrix for buildings and schools (Figure 6a). Fragility curves indicate the probability of damage to a structure given the earthquake intensity and damage level. Our damage model for buildings is a simplified modeling as it does not consider the type of buildings in a location; however, this model can be replaced by a more accurate model as needed. The *damage* matrix for bridges is calculated based on the bridge fragility curves defined in [13]. This model proposes a different fragility curves for bridges designed according to 1964, 1980, 1990, or 1995 seismic design codes. Following [21], we use the year of construction of the bridge to account indirectly for variations in the design methods. We assume that all the bridges built after 1995 are designed according to 1995 design code, all the bridges built between 1990 and 1995 are designed according to the 1990 design code, and so on. The fragility curves for bridges are shown in Figure 6b, according to several seismic design codes [13]. Note that the original curves in [13] are with respect to PGA values rather than MMI. We use a conversion equation from [8] to convert PGA to MMI.

6.2 Exposure Matrix

Without loss of generality, in this study, we consider three subjects of interest, including residential building occupants, schools, and bridges. In the last subject, we consider all the highway, light-rail, and railroad bridges located in the disaster area. For each subject, we calculate the urgency matrix. These matrices can be combined using a weighted equation if necessary. For instance, for an earthquake after the midnight, we can assign a weight of zero to schools, as they are expected to be empty at that time of the night. In the current study, we normalize the values of urgency matrix for each subject, and then sum them up with an equal weight to assemble the *total urgency* matrix. We then scale the *total urgency* matrix between 0 and 10. The geospatial information such as schools' locations, bridges' locations, populations etc. are obtained from the well-known Hazus-MH database[6] provide by Federal Emergency Management Agency (FEMA).



(a) Masonry buildings [22]. (b) Bridges [13]

Fig. 6: Moderate damage fragility curves.

To calculate the expected amount of damage to one of our subjects of interest, we must have the local exposure of that subject. The exposure matrix for residential building is filled with the value of the population size in each cell. This means that we consider the occupants exposure rather than buildings' exposure. With a similar logic, we assemble the exposure matrix for schools by counting the number of students in a cell and the exposure matrix for bridges by their number. For example, the exposure is zero if there is no school in that particular cell, or it is the summation of the number of students if there are more than one school in the cell.

We distribute the exposure value of each item in a circular area, instead of dealing with its exact shape. For instance, the population data are available for census blocks which may have arbitrary shapes. The population of each block is redistributed on an equally large circular area using B-Spline weight function [26] instead of dealing with the original shape of the block. This becomes handy to calculate the exposure values in a structured grid based on the scattered data points. Finally, to eliminate the effect of cell size, we divide all the values in exposure matrices by the value of a cell area in km^2. This means that the final values in exposure matrices are of density type, and not absolute values.

6.3 Significance Matrix

The elementwise multiplication of the *damage* and *exposure* matrices calculates the expected value of damage for each subject of interest; however, we also need to consider the significance of different items in each subject category and therefore introducing *significance* matrix is necessary. In particular, we use the number of daily traffic of a bridge as

6. https://www.fema.gov/hazus-mh-overview

its significance value. This means that between two bridges which are exposed to the same level of damage, the one with higher daily traffic has a higher urgency value. For residential buildings and schools, we fill the *significance* matrices with 0 if the cell does not contain, and with 1 if it contains any population or school. The *significance* matrices can also carry the weight of each subject when combining the *urgency* matrices to generate the total urgency map.

6.4 Case Study: 2014 South Napa Earthquake

We study the 2014 South Napa earthquake [25], which was the largest earthquake in the Bay Area since 1990. The maximum MMI for this earthquake was VIII, which corresponds to a condition of severe structural damage. Figure 7 shows the MMI map for this earthquake taken from USGS website. We use this MMI map to generate urgency maps for residential building occupants, schools, and bridges. Figures 8, 9 and 10 show the different layers of maps generated for this earthquake. Logarithmic scale is used for the color-axis for better illustrations. The epicenter of the earthquake was in an urban area; the urgencies are higher close to the epicenter, as the both intensity and exposure were higher there. Figure 11 shows the total urgency map for the disaster area. Considering the logarithmic scale, it can be seen that the area with higher urgency is much smaller than the originally defined disaster area. This helps decision makers to prioritize data collection in the important sub-areas.

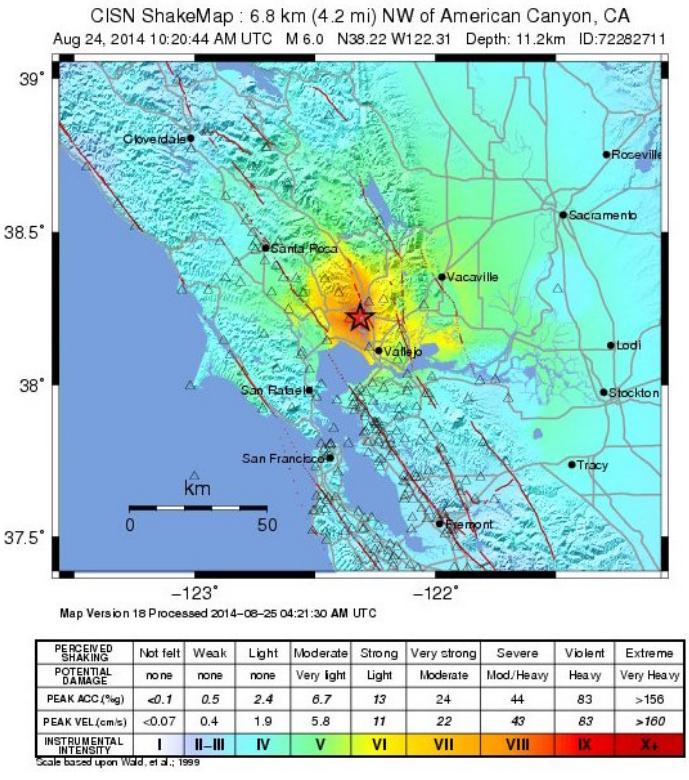


Fig. 7: Intensity map of the 2014 Napa Earthquake from USGS.

7 Performance Evaluation

We conducted several experiments on both synthetic and real-world datasets to evaluate the performance of our proposed approaches. Below, we first discuss our experimental setup and then we present our experimental results.

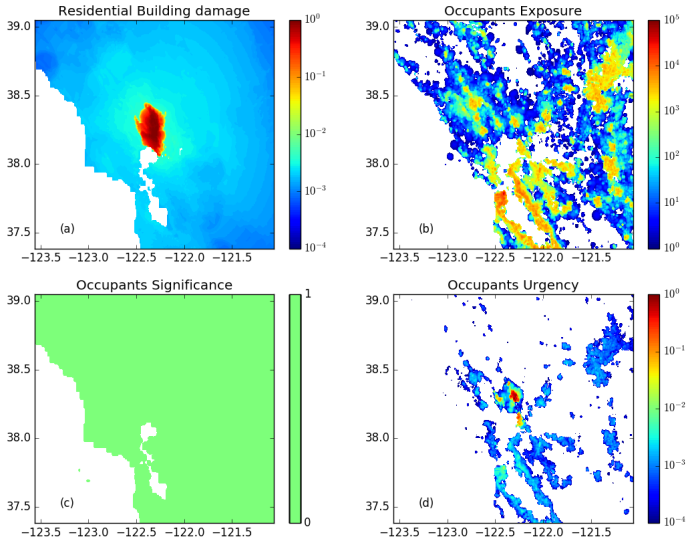


Fig. 8: Damage (a), exposure (b), significance (c), and urgency (d) maps for residential building occupants.

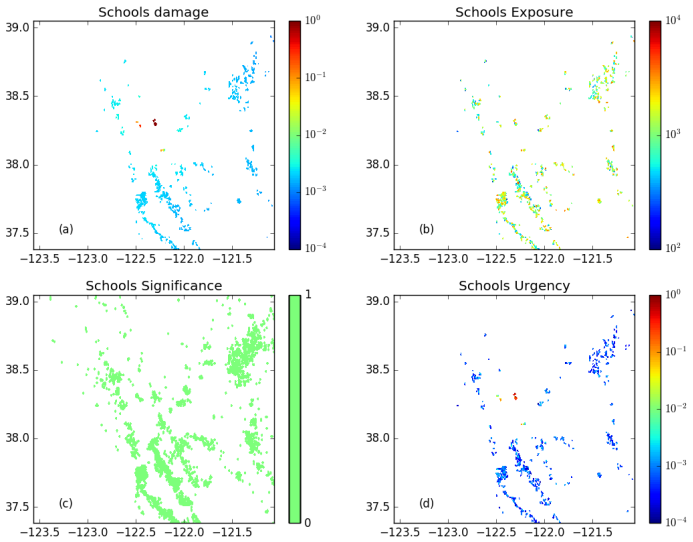


Fig. 9: Damage (a), exposure (b), significance (c), and urgency (d) maps for schools.

7.1 Experimental Methodology

7.1.1 Synthetic Datasets

We used the source code in [3] to generate synthetic video metadata with realistic geospatial properties, based on the behavioral patterns of mobile cameras when they move and rotate. We generated three spatial distributions of the video locations, **Uniform** and **Gaussian** (skewed dataset), in a region of 10×10 square km at Los Angeles, USA. (Figure 12). *Uniform* dataset is randomly generated while *Gaussian* and *Zipfian* datasets follow Gaussian ($\mu = 0$, $\sigma = 0.1$) and Zipfian (skew parameter s = 1) distributions, respectively. We discretized the space into 500×500 unit grid cells; the size of each unit cell is 20 square meter. We used a reasonable assumption of pedestrian camera moving with the speed

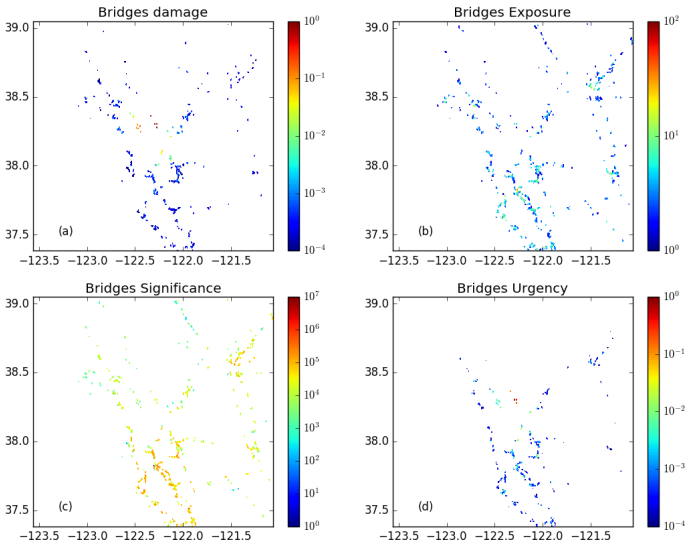


Fig. 10: Damage (a), exposure (b), significance (c), and urgency (d) maps for bridges.

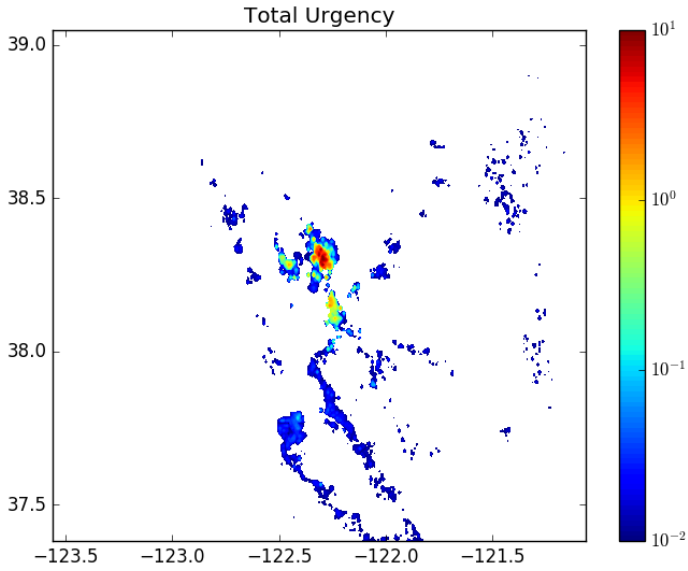


Fig. 11: Total urgency map for the 2014 Napa earthquake.

limit is between 5 and 20 km per hour. We fixed the video sampling rate to one frame per second, which means the metadata of a five-seconds video has five FOVs. We fixed the horizontal viewable angle θ to 60 degrees and the visible distance R to 200 m. We calculated the average rotation (in degrees/s) of the camera at each trajectory point is about 12 degrees while the maximum rotation is 55 degrees.

In all of our experiments, we varied the number of analysts $A \in \{36, 49, \mathbf{64}, 81, 100\}$ and the analyst's capacity $C \in \{2, 3, \mathbf{4}, 5, 6\}$ MBs per time interval. For simplicity, the bandwidth constraint and the analyst constraint are equivalent by setting the bandwidth $B = A \times C$; thus, $B \in \{72, ..., 600\}$ MBs. We fixed the number of videos with metadata only $|V| = 1000$. The video size follows Zipfian distribution with skew parameter s. We varied the skew parameter $s \in \{1.6, 1.8, \mathbf{2.0}, 2.2, 2.4\}$, resulting in the corresponding mean values $\{16.7, 8.2, \mathbf{4.6}, 2.9, 1.4\}$ MB. Default values are shown in boldface. With such default settings, the total coverage area of all FOVs is about 15 square km. Also, we assume one-second video weighs 1MB and the size of each frame image is 100KB. We assigned the urgency value to each unit cell by generating a random number between 0 and 10. All measured results are averaged over ten random seeds.

To generate more realistic dataset, we used real-world data from the **2014 South Napa Earthquake** [25]. USGS provides an application named "Did You Feel It?" so that citizen can report what they feel (e.g., shaking level at their locations) when the earthquake happens. We used the spatial distribution of the reports as shown in Figure 13 to synthesize the spatial distribution of the user-generated videos. Particularly, we first generated a *Uniform* dataset in the area of the earthquake and then kept only the videos in the cells[7] with reports (Figure 13). The number of generated videos in each cell is proportional to the number of reports. We used the first one hour of data after the earthquake, including 72% of the users' reports; discretized it to six equal time snapshots; each corresponds to ten minutes. In addition, we computed the number of videos for a particular time snapshot as being proportional to the number of reports during that time period.

	GeoUGV	GSV
Total number of videos with geo-metadata	2,397	N/A
Average length per video with content (sec)	72.14	N/A
Average camera moving speed (km/h)	4.5	N/A
Average camera rotation speed (degrees/sec)	10	N/A
Total number of users	289	N/A
Average number of videos by each user	8.29	N/A
Total number of FOVs	208,978	3,078
Average number of FOV per second	1.03	N/A
Average number FOV per video	74.16	N/A

TABLE 1: Overview of the real-world datasets.

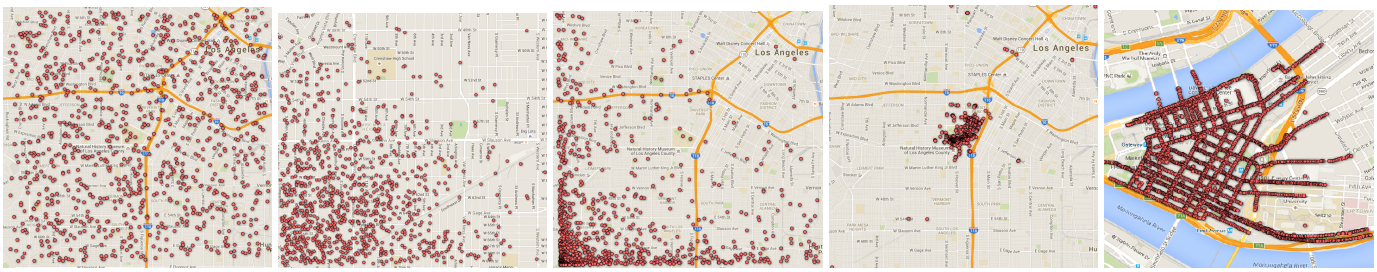
7.1.2 *Real-world Datasets*

We used the real-world geo-tagged mobile video dataset, named **GeoUGV** [17], which has been collected as part of the MediaQ and GeoVid projects. The statistics of the dataset are summarized as follows: 2,397 videos containing 208,978 video frames, that were geo-tagged (i.e., both *GPS locations* and *compass directions* at *fine-grained* intervals), collected by 289 users (e.g., students, faculties) over a period of 10 years (2007–2016). Table 1 shows the overall statistics of the dataset. Most of the videos are recorded by users in a casual walking mode. The camera moving speed is 4.5 km/h on average, and the camera rotation speed is 10 degrees/sec. The average FOV sampling rate is 1.03 FOVs per second, and each video is associated with 74.16 FOVs on average. Note that we filtered only videos captured in the 10×10 square km at Los Angeles, which comprises of 315 videos and 20,444 FOVs. In addition, we used Google Street View [31], which contains 3,078 images with geo-tagged locations and directions, to evaluate our proposed approaches.

7.2 Experimental Results

We evaluate the performance of the proposed techniques in terms of maximizing the so-called effective visual aware-

7. The cell size is 1 km^2.



(a) Uniform (b) Gaussian (c) Zipfian (d) GeoUGV (e) GSV

Fig. 12: Video distributions of the datasets (each video is represented by its first trajectory point).

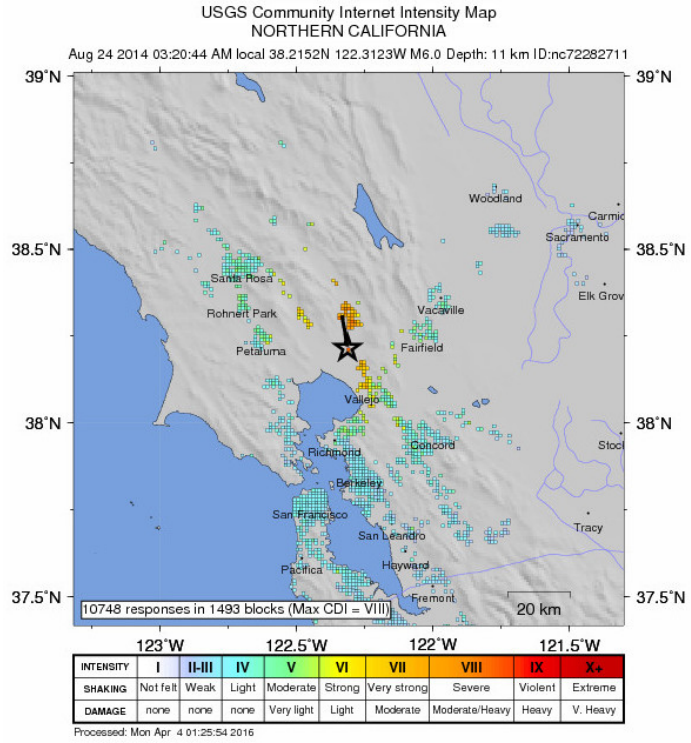
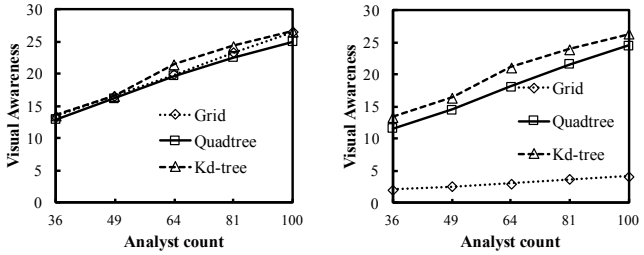


Fig. 13: Users' reports of the South Napa earthquake from USGS.

ness. Effective visual awareness is the total visual awareness of the videos (or images) that are analyzed by analysts; each can review maximum C amount of videos within a single time snapshot. We first present the results of Algorithm 2 where the entire video needs to be uploaded.

7.2.1 Visual Awareness Maximization

Figure 14 illustrates the results by varying the number of analysts A. We observe that with the increase of A, higher visual awareness is obtained. The reason is that the higher the number of analysts (the higher the budget), the larger number of videos are uploaded; leading to the increase in the total visual awareness of the uploaded videos. Also, Kd-tree generally performs best in terms of maximizing visual awareness. As shown on *Uniform* (Figure 14a), Kd-tree moderately increases the visual awareness in comparison to Grid. This is because the partitioning techniques perform similarly well when the videos' locations are uniformly distributed. However, the improvement is seven times higher on *Gaussian* (Figure 14b). The results on the highly skewed dataset (*Zipfian*) can be found in the preliminary version of this work [27]. In the same fashion, Figure 15 shows the similar results when the urgency map is computed based on region entropy (RE) (from Section 4.2.2) rather than randomly generated. The reason is that, unlike Grid and Quadtree, Kd-tree produces roughly equal number of videos per work cell, which contributes to the balanced workload of the analysts (Figure 16). In contrast, it is almost certain that Grid and Quadtree produce empty and over-populated cells. While empty cells waste analyst resources and thus contribute to the smaller visual awareness, highly populated cells are susceptible to redundant coverage of the containing videos. In addition, Kd-tree improves Quadtree due to its median splitting rather than midpoint splitting. However, this improvement gains at the cost of skewness in the produced work cells (i.e., Quadtree creates squared work cells).



(a) Uniform (b) Gaussian

Fig. 14: Varying the number of analysts A.

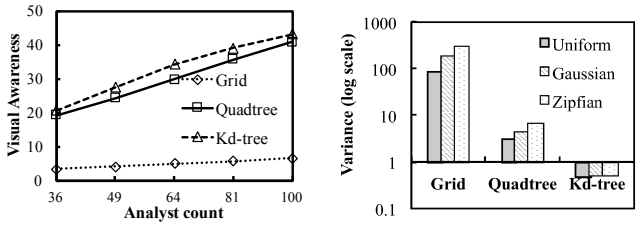
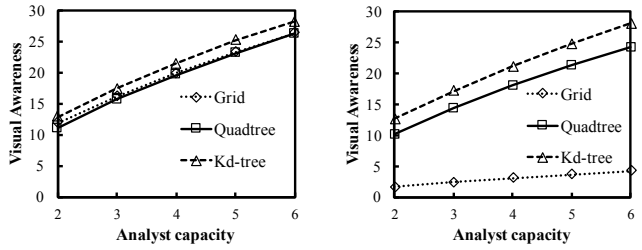


Fig. 15: Gaussian. RE is calculated from Gowalla dataset

Fig. 16: Variance of the number of videos per analyst.

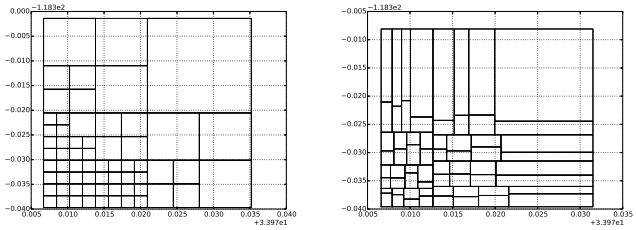
Figure 17 measures the impact of increasing analyst capacity C. As expected, a higher analyst capacity yields

higher visual awareness, as more videos can be effectively analyzed. Also, both Kd-tree and Quadtree outperform Grid in the *Gaussian* dataset (Figure 17b), which shows that Kd-tree and Quadtree adapt better to the non-uniform datasets. Figure 18 depicts this adaptation by showing their structures on various datasets.



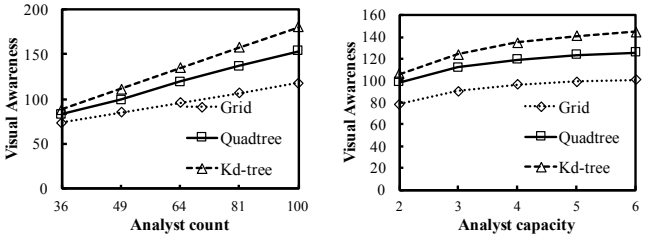
(a) Uniform (b) Gaussian

Fig. 17: Varying the analyst capacity C.



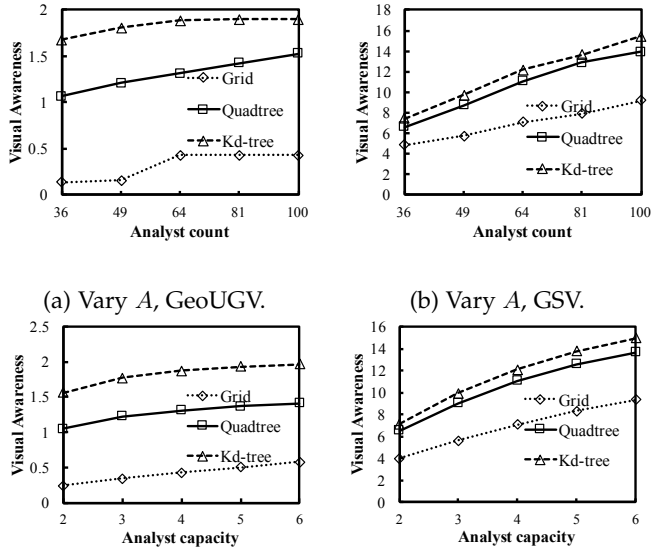
(a) Quadtree, Gau. (b) Kd-tree, Gau.

Fig. 18: Structures of Quadtree, Kd-tree on Gaussian ($A = 64$).



(a) Vary A (b) Vary C

Fig. 19: Performance of greedy algorithm on Uniform.



(a) Vary A, GeoUGV. (b) Vary A, GSV.

(c) Vary C, GeoUGV. (d) Vary C, GSV.

Fig. 20: Results in real-world datasets.

7.2.2 Minimum Redundant Coverage

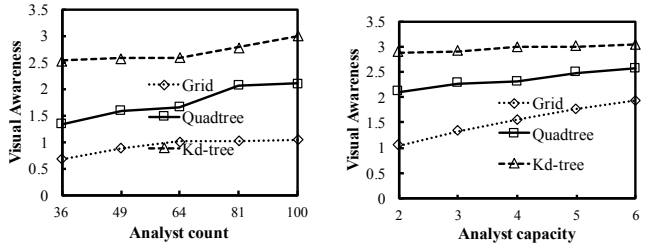
We evaluate the performance of Algorithm 3 from Section 5 where individual video frames can be uploaded. Figure 19a shows the results by varying the number of analysts on *Gaussian* dataset (similar results were observed for *Zipfian*). We observe similar trend as in Figure 14b, except that the obtained visual awareness is an order of magnitude higher. The reason is that, for the same amount of bandwidth and analyst count as in the video-level problem, frame-level optimization selects the frames with minimal overlap and thus maximizes the visual awareness. We also show the results by varying analyst capacity C in Figure 19b and observe similar trends as in Figure 17b.

Figure 20 shows the performance of Algorithm 3 in real-world datasets. The results preserve the trend in the synthetic data except that Kd-tree marginally improves Quadtree with the GeoUGV dataset. This is because most of the videos are captured in a small geographical area (one km squared of USC campus), which is partitioned to a few work cells only.

7.2.3 Case Study of the Napa Earthquake

In this set of experiments, we evaluate the performance of the proposed techniques in a realistic case study of the Napa earthquake described in Section 6.4. We first show the results for one time snapshot (i.e., all videos are given at once); followed by multiple time snapshots (Algorithm 4).

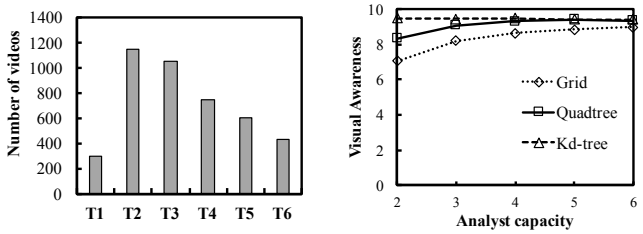
Figure 21b demonstrates the performance of Algorithm 3 in this dataset. As expected, Kd-tree performs the best. We also observe that the visual awareness increases slowly as A and C grows. This is due to the fact that the urgency map provided in Figure 11 is skewed (i.e., there are a few regions with very high urgency); thus, a few analysts who review the videos in such locations already provide high visual awareness.



(a) Vary number of generated videos per time snapshot. (b) Vary analyst capacity per time snapshot.

Fig. 21: Results on single time snapshot.

In addition, Figure 22b shows the overall performance of the proposed techniques in the case of multiple time snapshots, where the number of generated videos per time snapshot is depicted in Figure 22a. We observe higher visual awareness when compared with the prior case. This is because the visual awareness of any unit cell can be counted up to six times while that in the case of a single time snapshot is one and only one.



(a) Number of generated videos per time snapshot. (b) Vary analyst capacity per time snapshot.

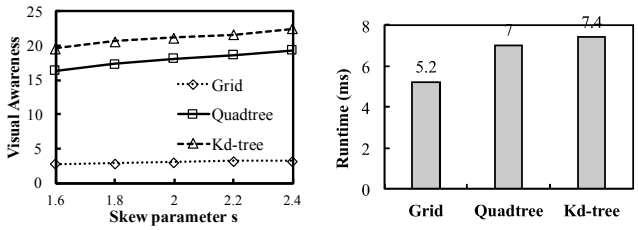
Fig. 22: Results on multiple time snapshots.

7.2.4 The Effect of Video Size

Figure 23a evaluates the performance of the partitioning techniques on *Gaussian* by varying the skew parameter s. The figure shows that increasing s or equivalently decreasing the average video size marginally increases the visual awareness. This unexpected result can be attributed to the fact that regardless of the skew parameter most videos are small in size, which are highly likely to be selected by the server.

7.2.5 Runtime Measurements

Figure 23b compares the construction time of the partitioning techniques (Algorithm 1). Their construction times are small and the differences are insignificant. In addition, the average runtime of the dynamic programming algorithm from Section 4.1.3 is 134 ms while that of the greedy algorithm from Section 5.2 is 15 ms. These results show the practicality of our proposed framework.



(a) Vary s, Gaussian (b) Construction time, Uniform

Fig. 23: The effect of skew parameter s for generating video size and the construction times of the partitioning techniques.

8 Conclusion

We introduced a crowdsourcing framework for collection and analysis of video data under disaster situations. The framework automatically divides a large disaster area into small work cells, each assigned to one analyst. We developed an analytical model to quantify the visual awareness of a particular video or frame and introduced the visual awareness maximization problem. Two problem variants have been studied, one with uploading the entire videos, the other with uploading individual frames to reduce bandwidth usage and avoid redundant coverage. Our experimental results on both synthetic and real-world data demonstrated that the proposed decomposition techniques are effective and the optimization solutions are practical. As future work, we will study crowdsourcing strategies that collaboratively involve both the analysts at the command center and the controlled workers at the disaster site to answer some open questions, including who to ask and where to collect data in disasters.

9 Acknowledgments

This research has been funded by NSF grants IIS-1320149, CNS-1461963 and the USC Integrated Media Systems Center. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the sponsors such as NSF.

References

[1] A. Akdogan, H. To, S. H. Kim, and C. Shahabi. A benchmark to evaluate mobile video upload to cloud infrastructures. In *Big Data Benchmarks, Performance Optimization, and Emerging Hardware*, pages 57–70. Springer, 2014.

[2] R. Andonov, V. Poirriez, and S. Rajopadhye. Unbounded knapsack problem: Dynamic programming revisited. *European Journal of Operational Research*, 123(2), 2000.

[3] S. A. Ay, S. H. Kim, and R. Zimmermann. Generating synthetic meta-data for georeferenced video management. In *Proceedings of the 18th sigspatial international conference on advances in geographic information systems*. ACM, 2010.

[4] S. A. Ay, R. Zimmermann, and S. H. Kim. Viewable scene modeling for geospatial video search. In *Proceedings of the 16th ACM international conference on Multimedia*, 2008.

[5] E. T.-H. Chu, Y.-L. Chen, J.-Y. Lin, and J. W. Liu. Crowdsourcing support system for disaster surveillance and response. In *Wireless Personal Multimedia Communications (WPMC), 15th International Symposium on*. IEEE, 2012.

[6] J. Cranshaw, E. Toch, J. Hong, A. Kittur, and N. Sadeh. Bridging the gap between physical location and online social networks. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*. ACM, 2010.

[7] A. Dey, G. Chinchwadkar, A. Fekete, and K. Ramachandran. Metadata-as-a-service. In *Data Engineering Workshops (ICDEW), 2015 31st IEEE International Conference on*, pages 6–9. IEEE, 2015.

[8] R. T. Eguchi, J. D. Goltz, H. A. Seligson, P. J. Flores, N. C. Blais, T. H. Heaton, and E. Bortugno. Real-time loss estimation as an emergency response decision support system: the early post-earthquake damage assessment tool (epedat). *Earthquake Spectra*, 13(4):815–832, 1997.

[9] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

[10] M. F. Goodchild and J. A. Glennon. Crowdsourcing geographic information for disaster response: a research frontier. *International Journal of Digital Earth*, 3(3):231–241, 2010.

[11] J. Heinzelman and C. Waters. *Crowdsourcing crisis information in disaster-affected Haiti*. US Institute of Peace, 2010.

[12] D. S. Hochba. Approximation algorithms for np-hard problems. *ACM SIGACT News*, 28(2):40–52, 1997.

[13] K. R. Karim and F. Yamazaki. A simplified method of constructing fragility curves for highway bridges. *Earthquake engineering & structural dynamics*, 32(10):1603–1626, 2003.

[14] S. H. Kim, Y. Lu, G. Constantinou, C. Shahabi, G. Wang, and R. Zimmermann. MediaQ: mobile multimedia management system. In *Proceedings of the 5th ACM Multimedia Systems Conference*. ACM, 2014.

[15] M. Kobayashi. Experience of infrastructure damage caused by the great east japan earthquake and countermeasures against future disasters. *Communications Magazine*, 2014.

[16] C.-C. Liu. Distribution systems: Reliable but not resilient? http://sites.ieee.org/pes-enews/2015/05/08/distribution-systems-reliable-but-not-resilient/, 2015.

[17] Y. Lu, H. To, A. Alfarrarjeh, S. H. Kim, Y. Yin, R. Zimmermann, and C. Shahabi. GeoUGV: user-generated mobile video dataset with fine granularity spatial metadata. In *Proceedings of the 7th International Conference on Multimedia Systems*, page 43. ACM, 2016.

[18] S. Martello, D. Pisinger, and P. Toth. Dynamic programming and strong bounds for the 0-1 knapsack problem. *Management Science*, 45(3):414–424, 1999.

[19] Y. Nemoto and K. Hamaguchi. Resilient ict research based on lessons learned from the great east japan earthquake. *Communications Magazine, IEEE*, 52(3), 2014.

[20] J. Ortmann, M. Limbu, D. Wang, and T. Kauppinen. Crowdsourcing linked open data for disaster management. In *Proceedings of the Terra Cognita Workshop on Foundations, Technologies and Applications of the Geospatial Web in conjunction with the ISWC*, pages 11–22. Citeseer, 2011.

[21] R. Ranf, M. Eberhard, and S. Malone. Post-earthquake prioritization of bridge inspections. *Earthquake Spectra*, 23(1):131–146, 2007.

[22] C. Rojahn and R. L. Sharpe. *Earthquake damage evaluation data for California*. Applied technology council, 1985.

[23] H. Samet. *Foundations of multidimensional and metric data structures*. Morgan Kaufmann, 2006.

[24] A. Schulz, H. Paulheim, and F. Probst. Crisis information management in the web 3.0 age. *Proceedings of ISCRAM*, 2012.

[25] U. S. G. Survey. M6.0 - 6km nw of american canyon, california. http://earthquake.usgs.gov/earthquakes/eventpage/nc72282711, 2014.

[26] S. Tavakkol, A. R. Zarrati, and M. Khanpour. Curvilinear smoothed particle hydrodynamics. *International Journal for Numerical Methods in Fluids*, 2016.

[27] H. To, S. H. Kim, and C. Shahabi. Effectively crowdsourcing the acquisition and analysis of visual data for disaster response. In *Big Data (Big Data), 2015 IEEE International Conference on*, pages 697–706. IEEE, 2015.

[28] A. S. Vivacqua and M. R. Borges. Taking advantage of collective knowledge in emergency response systems. *Journal of Network and Computer Applications*, 35(1):189–198, 2012.

[29] H. O. Wood and F. Neumann. *Modified Mercalli intensity scale of 1931*. Seismological Society of America, 1931.

[30] D. Yang, D. Zhang, K. Frank, P. Robertson, E. Jennings, M. Roddy, and M. Lichtenstern. Providing real-time assistance in disaster relief by leveraging crowdsourcing power. *Personal and Ubiquitous Computing*, 18(8):2025–2034, 2014.

[31] A. R. Zamir and M. Shah. Image geo-localization based on multiplenearest neighbor feature matching usinggeneralized graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(8):1546–1558, 2014.