Change Detection From Media Sharing Community

No Author Given

No Institute Given

Abstract. From ancient time, the damages or the destructions to the countries caused by the natural disasters were the major issues. Recently, through the improvement of image change detection technologies, social media and the high-resolution images, the damages caused by natural disasters can be analysed in more details to identify the situations in the cities or towns. Many researchers approached to analyse the damage by using the aerial images and the satellites images, but these images are often published to the public after the things settle down. However, when the disasters happen, people want the information of disasters as soon as possible. This research proposes to investigate how the social media images and the image change detection techniques can be used to identify the damages caused by the natural disasters. We first propose a framework that takes advantages of fast clustering and image near duplicate identification for the change detection in disasters. Then we model the social images by exploiting the image tags and location information. Following that, we propose a recursive 2 means algorithm over the new data model, and refine the changes by local interest point-based similarity matching. Finally, we propose a boundary representation model called *relative position annulus* (RPA), which is robust to boundary oration, location shift and editing operations. A RPA matching approach is proposed by extending the dynamic time warping (DTW) measure from series to annulus. Extensive experiments have been done to evaluate the high effectiveness and efficiency of our approach.

1 Introduction

Before, during and after the natural disasters, information about the damages and the current situations are vital for people to make decisions for their next actions. For example, the Nepal earthquake in 2015 did a huge damage to everything, including buildings, roads, infrastructure, and people, which resulted in 8,019 people died and 17,866 people injured; in a Tokyo earthquake, people are still able to walk back home since the earthquake damage the infrastructure, but not the buildings and paths. A recent study [1] found that people would like to know earthquake size and epicentre. Moreover, knowing these information could prevent the secondary and tertiary disasters. It is also found that in Tokyo, only 67.8% of people managed to get back their house on the day of an earthquake, and the rest 32.2% had to become "*homeless*", among them 2% failed to going back home only because they could not find the safe path. People want the information about earthquakes, but there is always the question "How should the people get the information of earthquakes?".

Recently, there are lots of researchers had approached to detect the damages caused by the natural disasters by using the change detection techniques with the aerial images. However, the aerial images consume longer times retrieve and harder to get compare to the other images. On the other hand, social media is pervasive, and updates very quickly especially on large events, e.g. natural disasters, by millions of people all the time. Thus, we investigate the problem of change detection from social media images, so as to let the public aware of the latest situations of natural disasters on the spot. One of the challenges here is the large-scale of the social media images, which makes current change detection techniques infeasible if not impossible. One of the limitation of using the large-scale images with current change detection techniques is the time cost. As existing techniques detect changes based on pixel level comparison [2] without index support or query optimization or both, the time cost for image comparison is high. When applying them to large scale social images, the efficiency issue becomes even unacceptable. Another challenge is the unavailability of some special features, like building shady, used in traditional change detection approaches [3]. In sharing communities, most of social images do not have shady, thus the shady-based matching can not be conducted. Forcing the existing techniques on the social images will cause low detection quality. Finally, traditional change detection over aerial images suppose the image pairs to the same location points are known, which is not true in media sharing communities.

To address these issues, we propose a framework for change detections in sharing communities. First, we represent the metadata of each image as a set of weighted tags, and propose a Social Image Similarity function (SIS) over image metadata and location. Then, we extend the recursive 2-means clustering algorithm []

YL: Should we add this citation now?

from vector space in $L_p - norm$ to key word set space, so Jaccard-based SIS measure can be applied. By deploying this extended recursive 2-means clustering over the whole image dataset, a number of small clusters are generated. Images in the same cluster or neighboring clusters will have the high probability of being the image pairs of the sources. Following that, we conduct PCA-SIFT based matching, which determines if two images are really referring to the same source. Finally, we propose a robust boundary representation that is robust to the view point rotation and other global transformations of the same objects in different images. Based on this representation, the boundary matching between image pair candidates is performed, which decides if a change has happened after disaster.

The contributions of this study are as follows:

- We propose a new similarity function SIS over weighted tags and location of social images, and extend the recursive 2-means clustering for SIS similarity.

- We perform PCA-SIFT-based matching which identifies the image pairs from the same object source.

  YL: I suggest to remove this from the contribution, as it is just the application of PCA-SIFT. ;)

- We propose a robust boundary representation model, based on which the boundary matching between images is conducted to identify the changes happened in disasters.
- Extensive experiments have been conducted over large real social image data collection to evaluate the effectiveness and efficiency of our change detection system.

The rest of the paper is structured as follows: Section 2 reviews the related work; Section 3 presents the framework proposed in this study; Section 4 details the modelling of the social media data; Section 5 presents the proposed change detection algorithm; Section 6 includes the experiment evaluation; Section 7 concludes the paper.

2 Related Work

In this section, we review the existing research closely related to this work, including the image copy detection and change detection.

2.1 Image Copy Detection

Image copy detection identifies the images of the same sources. Typically, image copy detection is done by first extracting the descriptors of local interest points in each image, and counting the number of matched descriptors between two compared ones. Examples on image descriptors include SIFT[4], PCA-SIFT[5], SURF[6], GLOH[7], and Eff^2[8] etc. In [4], Lowe invented SIFT descriptor to find the similarity between images. The SIFT descriptor is extracted by four steps: scale-space extrema selection, keypoint localization, orientation assignment, and keypoint descriptor computation. First, scale-space extrema selection finds the "interest points" in the image by using the Difference-of-Gussian Function(DoG). Then, by keypoint localization, the number of interest points is minimized and noise points are reduced. After that, the orientation assignment find the orientation of the images to ensure the invariance of descriptors with respect to image location, scale and rotation. Finally, a 128-dimensional descriptor vector is computed for each interest point. SIFT descriptor is invariant to the image translation, scaling, and rotation. However, the matching over SIFT can be expensive because of the high dimensionality of the descriptors.

To improve the efficiency of local descriptor matching, different variants of SIFT have been proposed [5–8]. In [5], PCA-SIFT was proposed to reduce the complexity of SIFT. It applies principal component analysis to the normalized gradient patches. PCA-SIFT conducts the operations same as the first three stages in SIFT, which accepts the sub-pixel location, scale and dominant orientations of each keypoint, and extracts a 41×41 patch centered over the keypoint at the given scale, and aligned its dominant orientation to a canonical direction. Different from the descriptor computation in SIFT, PCA-SIFT is obtained by first pre-computing an eigenspace to express the gradient images of local patches, and then projecting the gradient image vector computed for each patch into a 36-dimensional space with the support of the eigenspace. In [6], SURF descriptor was proposed based on the Hessian matrix to approximate the previous descriptor. SURF is a basic Laplacian-based detection, which exploits Gaussian scale-space analysis to localise interest points in the image and over scales. The SURF descriptor is extracted by first fixing a reproducible orientation based on a circular region around the interest point, constructing a square region aligned to the selected orientation, splitting each region into smaller 4×4 sub-regions and computing the sum of Haar wavelet responses and that of absolute response values vertically and horizontally. A descriptor vector for all 4×4 sub-regions of length 64 is obtained as a SURF descriptor. In [7], GLOH was proposed to extend the SIFT descriptor by changing the location grid and using PCA to reduce the size. In [8], Eff^2 was proposed by detecting the interest points of images using Difference of Gaussian over different scales just like SIFT descriptor, and then extracting the information of 8 orientation buckets over each of 3 grid cells around the point. This generates a 72-dimensional vector for each key point. Since PCA-SIFT has the stable performance in all situations as demonstrated in [9] and has lowest dimensionality, we select this descriptor in our image identification.

To match the local descriptor sets of two images, there are mainly two approaches: one-many matching and one to one symmetric matching (OOS). In [4], the similarity between two local descriptor sets is measured by identifying the nearest neighbor of each local descriptor based on Euclidean distance, and calculating the number of their matched key point pairs. Using this approach, multiple key pints in a query image can be matched with a single point of an image data, thus matches over noise key points can be introduced. In [10], Zhao et al. proposed OOS matching based on a cosine distance based partial similarity matching. Using this approach, one key point in a query image can only be matched with a single key point of an image data. As such the matches caused by noise can be excluded. The similarity between two images is determined by the number of matched interest point pairs. As OOS matching achieves better performance in image copy detection, we choose it for our local descriptor similarity measure in this work.

2.2 Change Detection

Approaches have been proposed to detect changes happened at a location during a natural disasters. Traditionally, changes are detected by using pixel-based techniques [2]. Typical pixel-based approaches include *image differencing*, *image regression*, *image rationing*, *vegetation index differencing*, *change vector analysis*, *background subtraction* and *pixelwise fuzzy XOR operator*. All these methods are based on pairwise pixel comparison, which is not robust to the image content shift or noises. Moreover, the pixel-level comparison suffers from high time cost.

Existing literatures have proposed object-based change detection(OBCD) techniques for the geographic data. OBCD compares and detects the changes by using the objects, each of which is the group of pixels contains meaningful data. Normally, OBCD algorithms are applied to satellites, remote sensing and the Synthetic aperture radar (SAR) images and detect the damage and geographical changes from them. Recently, 3D GIS model and Terra-SAR-X are evolved from the GIS and the SAR for change detection. With the improvement of object segmentation, object-based damage detection were well studied for the urban area damage detection. In [11], H.Murakami et al. proposed the simple damage detection method, which identifies changes by subtracting the digital surface model(DSM) from another DSM. In [3], M.Turker et al. used the watershed segmentation to create the segmented building vectors and calculate the shadow area of the segmented building to detect the damages. In [12], L.Matikainen et al. use object-based GIS model data with the overlap analysis algorithm change detection. Recently, L.Gong et al.[13] used VHR Terra SAR-X for finding the changes during earthquake. In [14], J.Tu et al. use the 3D GIS model image to detect the building damages during Beichuan earthquake. The 3D GIS model is exploited to extract the vectors of building images, and the height of a building is estimated using the shadow detection. Different building damage types are detected based on the changes detection of the building area, the height between the pre- and post-disaster, and the building rooftops texture information. Using the satellite, remote and SAR images with damage detection algorithms can achieve high accuracy. Although these images can cover the urban area, there is an issue of retrieving pre-event data sets. Taking all the imagery data for whole country or land is always hard to achieve. There are some areas which are not covered by the images. Media sharing communities provide great sources for capturing disasters during crisis. Thus, it is demanded to conduct the single pre- and post-event image detection in social communities for disaster management. However, existing techniques use typical features for satellite images such as shadow can not be obtained in images from public uploading in media sharing communities. New techniques should be developed for identifying the detection of damages over social images.

3 Framework

In this section, we first define two terms, *change* and *change detection*, and then present the overview of the change detection framework for social images.

Definition 1. *In image processing,* change *is defined as the difference between two pixels or the objects in different images. The* difference *varies in different situations. In this paper, the* difference *is limited to the damages to the roads, the buildings or the infrastructures, which are caused by natural disasters. For instance, when a bridge breaks down during an earthquake, the damage to the bridge will be the* change *in this situation.*

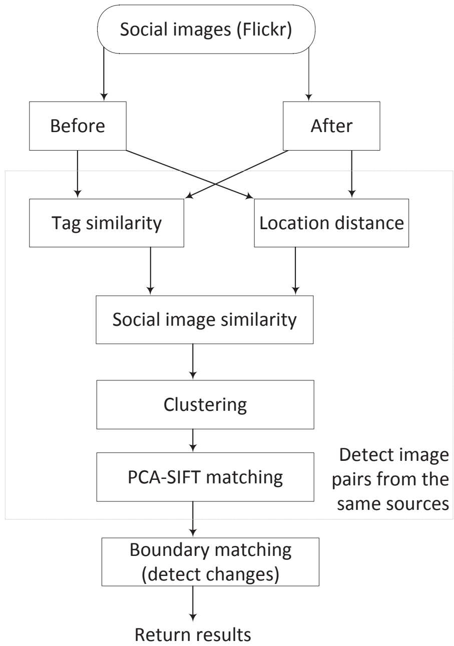


Fig. 1: Framework of our change detection

Definition 2. Change detection *is a process of finding the damages which have been caused by certain natural disasters such as earthquake and flood.*

Figure 1 shows the architecture of our change detection system, which includes the following two components: (1) Finding image pairs from same sources; and (2) Detecting changes from image pairs.

– **Finding relevant image pairs:** The proposed change detection framework accepts a set of keywords as input, which are used to retrieve images from the social media community (e.g. Flickr). Specifically, while retrieving the images, the images are separated into the *before* and *after* natural disasters. The information about retrieved images include the photo ID, tags, and location features. Firstly, we model them to numeric data, then deploy the *Jaccard*-based tag set similarity and the $GreatCircleDistance$ to measure the similarity between tags and that between locations, respectively. Then, these two similarities are fused to obtain the final similarity for two social images, which is called *Social Image Similarity* here. Then, the recursive 2 means algorithm [15] is applied to group similar images into clusters, which significantly reduce the complexity of the problem of change detection based on large-scale social images. After the similar images are grouped together, an One-to-One Symmetric matching (OOS) over PCA-SIFT descriptors of image pairs is applied here to find the similarities between the *before* and *after* images. OOS has been applied to near duplicate video detection, and approved the high effectiveness of detection [16, 17]. The image pairs with high similarity are detected and used for the next step change detection.

– **Detect changes:** For each relevant image pair, we extract the object boundaries of each image, and conduct boundary-based matching. If the objects from two images does not match, we define this as a damage is identified. Otherwise, no damage has happened to the objects contained in the image pair.

With the support of our framework, the comparison between local image descriptors is only performed over the pairs in the same cluster or neighboring ones. This greatly reduces the time cost of relevant image identification. In addition, we can effectively decide which pairs should be considered for the final change assessment. Thus, comparing with shape-based damage assessment [12], our method is more robust.

4 Data modelling

In this section, we present how to model images' tags and location data for change detection from social media images. Specifically, we model them to numeric data first, then measure tags and locations by using two different similarity metrics, respectively. Finally, we fuse them together to obtain the *difference* of social media images.

4.1 Tag-based Similarity

In social media, the posted images normally have some tags, which contain their semantic meanings. Intuitively, the images with similar tag sets have high probability of coming from the same source. Thus, it is necessary to design a similarity function over tag sets of images, based on which the image pairs from the same source can be identified. Jaccard similarity has been successfully used in existing literatures for set matching. Thus, we exploit Jaccard similarity-based measure for tag sets, and focus on how to construct virtual tag set for a group of images, and how to update it in dynamic environment. For single image, its tag set can be modelled as a set of single tag with weight 1. Given a group of images, its virtual tag set is constructed by averaging the weights of each tag appearing in all its images. If a specific tag only appears in part of images, the weight of this tag in the other images of the group will be 0. Given N images, let $\{K_1, K_2, ...K_n\}$ be the tags appearing in the image tag sets, and w_{ij} the weight of tag K_i in image j. This image set can be modelled as a set of weighted tags as below:

$$CentroidKeyword = \{W_1K_1, W_2K_2...W_nK_n\} \tag{1}$$

where W_i is defined as:

$$W_i = \frac{\sum_{j=1}^{N} w_{ij}}{N} \tag{2}$$

Suppose we have 2 images, I_1 and I_2, containing 3 and 5 keywords, where I_1: {"*earthquake*", "*apple*", "*happy*"} and I_2: {"*earthquake*", "*natural*", "*disaster*", "*apple*", "*Nepal*"}. Then, the virtual tag set of this group is {(1)earthquake, (1)apple, (0.5)natural, (0.5)disaster, (0.5)Nepal}. Figure 2 shows an sample from the change detection application[1].

```
Center Text :1
There are 24 tags in center 1 text within 70 photos
Before Calculation: {25 april 2015=70.0, architecture=70.0, art=70.0, bertrand de cam
After Calculation: {25 april 2015=1.0, architecture=1.0, art=1.0, bertrand de camaret

Center Text :2
There are 1261 tags in center 2 text within 430 photos
Before Calculation: {#NZ15=44.0, 108=8.0, 108 Prayers for Kathmandu=126.0, 11=2.0, 11
After Calculation: {#NZ15=0.10232558139534884, 108=0.018604651162790697, 108 Prayers
```

Fig. 2: Centroids Keywords from the appreciation

Given two weighted tag sets A and B, their Jaccard-based tag similarity is defined as:

$$J_t(A,B) = \frac{\|A \cap B\|}{\|A \cup B\|} = \frac{\|A \cap B\|}{\|A\| + \|B\| - \|A \cap B\|} \tag{3}$$

where the $\|A\|$ is the weighted number of tags in set A. Our J_t measure can be used for the matching between images and that of image groups as well.

4.2 Location-based similarity(LDS)

To detect changes from images, it is important to prevent gathering the images having the similar tags but different location. Thus, location information is vital here. The location of an image can be described as a pair of its latitude and longitude. We approximate the location of an group of N images as a pair of average latitude and average longitude over the whole group, which can be computed by:

$$Lat = \frac{1}{N}\sum_{i=1}^{N} Lat_i \tag{4}$$

$$Lon = \frac{1}{N}\sum_{i=1}^{N} Lon_i, \tag{5}$$

YL: Fixed some error here. Please double check these two equations are correct or not.

where Lat denotes latitude and Lon longitude.

Since there are no straight lines on the sphere, we deploy the "Great Circle Distance(GCD)" to calculate the distance between the images, which measures

[1] Sample clusters tags data and weighted tags from application. Where the *Center Text :1* is the first cluster and *Center Text :2* is the second cluster. The line *Before Calculation* and *After Calculation* show the before and after the average calculation.

the shortest distance between two image locations on the surface of a sphere and can be computed by the Haversine Formula [18]. Let a denote Haversine, and c the great circle distance in radians. Then the location distance D is computed by equations 6 (a)-(c).

$$a = sin(Lat/2) + cosLat1cosLat2sin(Lon/2) \tag{6}$$

$$c = 2atan2(\sqrt{a}, \sqrt{(1-a)}) \tag{7}$$

$$D = Rc \tag{8}$$

where R is earth's radius. As suggested by Johor et al. 2013 [19], the location distances are normalised to remove the potential biases among different features. It has been proved that the "Min-Max" standardization had the lowest error rate to k-means algorithms [19]. Thus, we exploit this technique for location distance normalization, which is computed by:

$$D' = \frac{D - min(D)}{max(D) - min(D)} \tag{9}$$

where D denotes the distance between images.

4.3 Social Image Similarity

Once the tag similarity and location distance measures are defined, we define the overall social image similarity (SIS) by fusing the tag-based distance and location-based distance, which is as follows:

$$SIS = \frac{1}{D' + 1} * J_t, \tag{10}$$

whereD' denotes the location distance and J_t the tag-based similarity.

The value of SIS between two images will be in the range from 0 to 1. A larger SIS value indicates a higher probability of the images being closer to each other.

5 Change Detection Over Large Data Sets

In this section, we present the change detection algorithm from the large-scale social images, including how to identify the image pairs from the same source and how to detect the changes in images.

5.1 Image Copy Detection

This section presents how to effectively and efficiently identify the image pairs of the same source (same building or infrastructures etc). Many image copy detection approaches have been proposed and applied for various applications. However, since each image may contain up to thousands of 36-dimensional local descriptors, directly deploying existing technique is obviously not suitable for large image databases due to the high time cost. To improve the efficiency of image copy identification, we propose a process of SIS-based clustering, then deploy a PCA-SIFT-based matching over images in the same or neighboring clusters.

SIS-based clustering Intuitively, images from the same source usually have similarity over the concept level. Thus it is reasonable to cluster social images based on their tag and location information, such that those in a single cluster or neighboring clusters will have high possibility of being from the same source. In this way, the identification of image copy pairs can be simplified as the comparison between images from same or neighboring clusters, which avoids the costly pair-wise image matching over the whole data collection. To do this, we need to find an efficient clustering technique that can be extended to our SIS social image similarity as well. As we operate on large scale image database, the efficiency of clustering process is extremely important. Moreover, we only care about the similarity between images in each cluster, thus permit the overlaps between multiple ones. Thus, we extend the 2-means clustering algorithm that was initially proposed for L_p distance for our SIS similarity considering its low time cost as stated in [15, 20].

Given an image collection, we conduct the clustering by three steps: 1) for each image, we model its metadata as tag set and its location as a pair of latitude and longitude values; 2) we select two images with the lowest similarity as the cluster centres of two initial cluster from the current data collection. Each of the remaining images are assigned to the cluster that has the higher similarity with it. The centre points are recursively recalculated based on Equations 1 and 4, together with the images allocation to clusters based on their similarity, until the new generated centre points are stable; 3) finally, we recursively select a bigger cluster on which the second step is conducted, until the number of clusters reaches to a threshold κ.

In the social media images often there are more than 10 tags on each image and comparing these tags between the images one by one is time-consuming. String hashing has been applied in many applications for improving the system efficiency [21]. Thus we improve our change detection efficiency by selecting a "good" hashing function class. To reduce the time expenses on tags comparison, we use the *djb2* hash function techniques [22], which one of the best hash function techniques for the string data. In *djb2* the hash values are populated by *hash*33 + c*, where the hash is the long data and *c* is the string character. Daniel J. Bernstein uses the magic number 33 to times the hash data, however, he did not explain why it works better than any other constants. In our system, the *djb2* is applied to all the images tags and the hash value is created for each tag. These hash values are used when the J_t similarity is calculated. The first step is making the tags on the image A and the image B to the hash values, and then it compares the hash values of A with those of B. Secondly, if the hash value of A does not exist in that of B, it is considered as unmatched. As such the system does not need to compare all the string tags. This hashing reduces the total number of string comparison in the J_t similarity.

PCA-SIFT-based matching PCA-SIFT-based matching will be used for deciding if two image candidates are referring to the same objects. It will find the similarity between the objects. The main objects are the buildings, statues and other objects which are not moving. Also, the unrelated images selected by the PCA-SIFT algorithm will be removed. The removing occur when the images have the high SIS similarity, but the images are not what the user want. The kept image pairs are passed to change detection stage for assessing if any damages had happened in a natural disaster.

We apply the OOS to the similarity calculation between the local descriptor sets of two images [16]. Given two local interest descriptors, the similarity between them is measured by the $Cosine$ similarity between these two vectors. For two local interest points from two images, they are match pair candidate if the similarity between them is bigger than a threshold value. OOS further check if any one of these two points is the nearest neighbor of the other among all the descriptors of its image. If they are nearest neighbors of each other, they are a real matched pair. The final similarity between two images is calculated by the average similarity of all their matched pairs. To improve the local interest points matching, we use the LIP-IS index proposed in [16] as well. All these ensure that effective and efficient PCA-SIFT-based matching is performed.

5.2 Change Detection

Change detection assesses the damages caused during disasters. Intuitively, two pictures to the same location point contains same buildings or other objects, each is described as a boundary. If there is no damage happened at this place, the object boundaries in two images match. Otherwise, if there exists boundary missing or unmatched from the before image to the after one, the damages could have be caused in the disaster. Therefore, in this section, we propose a new image object boundary modelling together with a novel boundary matching method, which are robust to different image transformations, rotations or editing, for effective change detection.

Model image boundary Existing works model building shadow area boundaries [3], boundary shapes [12] using the coordinates of each pixel falling on the boundary of objects in an image. However, these boundary modelling incurs low effectiveness of detection for social images because of the possible object changes with respect to the viewpoints, rotations, space shift etc. To address this issue, we propose a robust boundary representation, called *relative position annulus* (RPA), which describe each boundary as an annulus of the difference of neighboring edge lengths. Specifically, we exploit sober edge detector to detect a number of boundaries in an image, since it can detect the emphasising edges while reduce the effect of noise edges. Given a boundary consisting of m vertexes $\{v_1, ..., v_m\}$, we represent the boundary as $\{d(v_1, v_2) - d(v_2, v_3), ..., d(v_{m-2}, v_{m-1}) - d(v_{m-1} - v_m), d(v_{m-1}, v_m) - d(v_m, v_1)\}$, where any element can be the start point while other points are ordered clockwise. As such, the RPA representation will be robust to object rotation, viewpoint change and space shift in social images.

Matching boundaries As each image may contain multiple objects, each image is described as a set of relative position annulus of multiple boundaries. To assess the damages in a disaster, we need to do two steps matching: (1) the measure between two RPAs; (2) the measure between two images. To further reduce the influence of small noise objects, we only use the top κ biggest boundaries for boundary comparison between two images.

Given two RPAs, $\mathcal{Q} :< q_1, q_2, ..., q_m >$ and $\mathcal{D} :< v_1, v_2, ..., v_n >$, we measure the similarity between them by extending the DTW [23] for our annulus representation. In the matching, we consider Q as a series, and D as a set of n series, where each series in the set takes v_i $(i = 1, ..., n)$ as the start point of the boundary and the remaining ones are ordered clockwise. Denote the series to v_i as $\mathcal{D}_i$, and its elements as $< v^i_1, ...v^i_n >$, where $v^i_j = v_{((i+j-1) \pmod n)}$. Then the similarity between $\mathcal{Q}$ and $\mathcal{D}_i$ is measured by:

$$SRPA_i(\mathcal{Q}, \mathcal{D}_i) = \begin{cases} 0 & m = m_1 - 1 \; or \; n = n_1 - 1 \\ max\{SRPA_i(\mathcal{Q}_{m-1}, \mathcal{D}_{n-1}) + Sim(q_m, v^i_n), & \\ \quad SRPA_i(\mathcal{Q}_m, \mathcal{D}_{n-1}), & \\ \quad SRPA_i(\mathcal{D}_{m-1}, \mathcal{D}_n)\} & otherwise \end{cases} \tag{11}$$

where Sim is the similarity between q_m and v^i_n computed based on L_1 distance:

$$Sim(q_m, v^i_n) = \frac{1}{1 + |q_m - v^i_n|}. \tag{12}$$

The final boundary distance is decided by finding the maximal DTW between $\mathcal{Q}$ and $\mathcal{D}_i$

$$SRPA = \max_{i=1}^{n} SRPA_i. \tag{13}$$

6 Experiment

In this section, we examine the performance of the proposed method, focusing on its effectiveness and efficiency. Specifically, we answer:

- How does the proposed Social Image Similarity function perform? We answer this by investigate the efficiency of the SIS-based clustering models, the effect of cluster numbers with different types of clustering techniques, and the effect of hash index.
- How effective and efficient of the proposed boundary-based change detection method (*BBCD*)? We compare *BBCD* with existing state-of-the-art algorithms, the shape base change detection(*SBCD*) [12].

6.1 Experimental Setup

The dataset we experimented with is collected from Flickr by focusing on images relevant to the *Nepal earthquake*, which is also known as *Gorkha earthquake*, in 2015. Finally, 10,000 images are collected, which include images *before* and *after* the earthquake. The ground-truth is manually identified by the authors of this paper via careful comparison of all *before* and *after* images. Specifically, for a *ground-truth*, both *before* and *after* image have to contain at least one same building, but the corresponding image contents, angles, resolutions, colours and light effects could be different. This would allow the proposed method to analyse whether the before and the after images are taken in the same location and building.

We conduct the efficiency experiments on 10000 after and before the earthquake images and their features. 4000 after and before the earthquake images are collected for the effectiveness experiment.

YL: this is confusing. How many images you collected? 10000 + 4000? or the 4000 is in the 10000 actually? why do effectiveness only on 4000? for time reason?

6.2 Measure metrics

To evaluate the effectiveness of algorithms, we used two metrics in [**?**], the probability of miss detection and false alarm (*Pmiss* and *Pfa*). Specifically, the *missed detection* mean that the algorithm fails to detect the ground-truth, and the *false alarm* means the detection of non-target pairs. $Pmiss$ and Pfa are defined as follows:

$$Pmiss = \frac{number of missed detections}{number of groundtruth} \tag{14}$$

$$Pfa = \frac{falsealarms}{nontargets} \tag{15}$$

a small value of *Pmiss* and *Pfa* means better effectiveness.

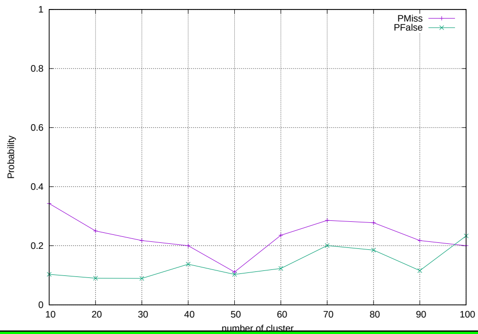
The evaluation of efficiency includes two parts: 1) number of clusters, and 2) comparison with the existing damage detection algorithms. Specifically, for number of clusters part test, it is expected to obtain the best cluster numbers to achieve the smallest *Pmiss* and *Pfa*. We evaluate the efficiency of the proposed approach in terms of the overall time cost of clustering and hash index. We also evaluate the time cost of *BBCD* and *SBCD*, and to make the comparison more precise, we only compare the after the image matching time cost for change detection. Experiment are conducted on [machine information]...

YL: The above paragraph is very confusing, and please double check I rewrite what you want to present correctly.

6.3 Effectiveness

We first examine the effect of number of cluster in *SIS-based clustering* and *OOS+LIP-IS*. After that, we compare the proposed approach with *SBCD* by performing the change detection over the collected Flickr image datasets.

Effect of clustering To examine the effect of the number of clusters K, we vary K from 10 to 100 with a step length 10, and for each cluster, it only looks for image pairs with 3 nearest neighbouring clusters. When calculating $Pmiss$, we only count the ground truth pairs that have more than 10 matches. The results are shown in Fig. 3. It is observed that 1) when K increases from 10 to 50, *Pmiss* decreases from 0.35 to 0.10. After that, *Pmiss* starts increasing when $K > 50$ with slightly drop after $K > 80$. 2) when increasing K, the overall trend of *Pfa* is increasing, which means more false alarms. But, there is a flat period for Pfa is around 0.10 when K is between 40 and 60. This may indicates 50 is good candidate for the number of clusters.



YL: the title of this fig seems not right. It should be "The effect of the number of clusters". And the label for this fig is a duplicate of Fig.5. Please confirm which one is which.

Efficiency comparison

Comparing with existing technique comparing with shape-based approach. (in paper: **automatic detection of buildings and changes in buildings for updating of maps**)

by varying the dataset size from small to big, test the probability of missed detection and probability of false alarm at each dataset size point

6.4 Efficiency

Effect of different clustering techniques Here, we examine the efficiency of different clustering techniques when using them in the proposed change detection method. Specifically, we compare K-means and Recursive 2 Means by varying the number of cluster size from 5 to 100, and then report the average time cost of building the clusters. The corresponding results are shown in Figure 4. It is observed that 1) when K increases, the time cost increases for both techniques; 2) the time cost of K-means increases relatively quicker than Recursive 2 Means, which means Recursive 2 Means is better.

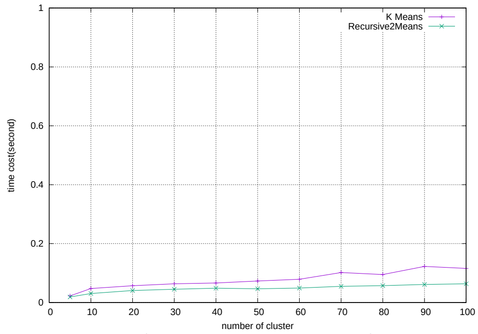
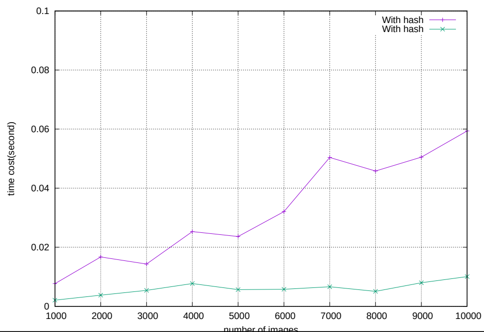


Fig. 4: K-means and recursive 2 means clustering time cost

Effect of hashing index Now, we examine the efficiency of hashing index. Specifically, we compare the time cost of the proposed method with the *djb2* hash function and without it. This is conducted over 1000 to 10000 images, which is shown in Figure 5. It is observed that 1) the method of with hash function is significantly quicker than the counterpart of without hash function, and the more the images, the larger the improvement. 2) while the size of images increase from 1000 to 10000, the time cost for the proposed method with hash function remain relatively small and stable; however, the time cost of the counterpart without hash function increase over 7 times more.



YL: the legend of this fig is not right.

Efficiency comparison Detection efficiency by varying data size this is only for comparing our boundary-based approach and existing shape-based approach.

7 Conclusion

References

1. H. U, S. Naoya, N. Ryota, W. Shuntaro, and H. Hidenori, "Questionnaire survey concerning stranded commuters in metropolitan area in the east japan great earthquake," *Journal of social safety science*, pp. 343–353, nov 2011.
2. M. İlsever and C. Unsalan, *Two-Dimensional Change Detection Methods: Remote Sensing Applications.* SpringerBriefs in Computer Science, Springer London, 2012.
3. M. Turker and E. Sumer, "Building-based damage detection due to earthquake using the watershed segmentation of the post-event aerial images," *Int J Remote Sens*, vol. 29, no. 11, pp. 3073–3089, 2008.
4. D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, pp. 1150–1157 vol.2, 1999.
5. Y. Ke and R. Sukthankar, "Pca-sift: a more distinctive representation for local image descriptors," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, pp. II–506–II–513 Vol.2, June 2004.
6. H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer vision–ECCV 2006*, pp. 404–417, Springer, 2006.
7. K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. on PAMI*, vol. 27, no. 10, pp. 1615–1630, 2005.
8. H. Lejsek, F. H. Ásmundsson, B. T. Jónsson, and L. Amsaleg, "Scalability of local image descriptors: a comparative study," in *MM*, pp. 589–598, 2006.
9. L. Juan and O. Gwun, "A comparison of sift, pca-sift and surf," *International Journal of Image Processing (IJIP)*, vol. 3, no. 4, pp. 143–152, 2009.
10. W.-L. Zhao, C.-W. Ngo, H.-K. Tan, and X. Wu, "Near-duplicate keyframe identification with interest point matching and pattern learning.," *IEEE Trans. on MM*, pp. 1037–1048, 2007.
11. H. Murakami, K. Nakagawa, H. Hasegawa, T. Shibata, and E. Iwanami, "Change detection of buildings using an airborne laser scanner," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 54, no. 2, pp. 148–152, 1999.
12. L. Matikainen, J. Hyypp, E. Ahokas, L. Markelin, and H. Kaartinen, "Automatic detection of buildings and changes in buildings for updating of maps," *Remote Sensing*, vol. 2, no. 5, pp. 1217–1248, 2010.
13. L. Gong, C. Wang, F. Wu, J. Zhang, H. Zhang, and Q. Li, "Earthquake-induced building damage detection with post-event sub-meter vhr terrasar-x staring spotlight imagery," *Remote Sensing*, vol. 8, no. 11, p. 887, 2016.
14. J. Tu, H. Sui, W. Feng, and Z. Song, "Automatic building damage detection method using high-resolution remote sensing images and 3d gis model.," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 43–50, 2016.
15. H. T. Shen, B. C. Ooi, and X. Zhou, "Towards effective indexing for very large video sequence database," in *SIGMOD*, pp. 730–741, 2005.

16. W. Zhao, C. Ngo, H. Tan, and X. Wu, "Near-duplicate keyframe identification with interest point matching and pattern learning," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 1037–1048, 2007.
17. X. Zhou, X. Zhou, L. Chen, A. Bouguettaya, N. Xiao, and J. A. Taylor, "An efficient near-duplicate video shot detection method using shot-based interest points," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 879–891, 2009.
18. C. C. Robusto, "The cosine-haversine formula," *The American Mathematical Monthly*, vol. 64, no. 1, pp. 38–40, 1957.
19. D. U. Ismail Bin Mohamad, "Standardization and its effects on k-means clustering algorithm," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 6, no. 17, pp. 3299–3303, 2013.
20. X. Zhou, X. Zhou, L. Chen, Y. Shu, A. Bouguettaya, and J. A. Taylor, "Adaptive subspace symbolization for content-based video detection," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1372–1387, 2010.
21. X. Zhou, L. Chen, Y. Zhang, L. Cao, G. Huang, and C. Wang, "Online video recommendation in sharing community," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, pp. 1645–1656, 2015.
22. V. G. Martnez, F. H. lvarez, and L. H. Encinas, "State of the art in similarity preserving hashing functions," in *SAM*, pp. 139–145, 2014.
23. D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, AAAIWS'94, pp. 359–370, AAAI Press, 1994.