Large-Scale Damage Detection Using Satellite Imagery

Lionel Gueguen Raffay Hamid
DigitalGlobe Inc.
12076 Grant Street, Thornton, Colorado, USA
{lgueguen, mhamid}@digitalglobe.com

Abstract

Satellite imagery is a valuable source of information for assessing damages in distressed areas undergoing a calamity, such as an earthquake or an armed conflict. However, the sheer amount of data required to be inspected for this assessment makes it impractical to do it manually. To address this problem, we present a semi-supervised learning framework for large-scale damage detection in satellite imagery. We present a comparative evaluation of our framework using over 88 *million images collected from* $4,665$ *KM^2 from* 12 *different locations around the world. To enable accurate and efficient damage detection, we introduce a novel use of hierarchical shape features in the bags-of-visual words setting. We analyze how practical factors such as sun, sensor-resolution, satellite-angle, and registration differences impact the effectiveness our proposed representation, and compare it to five alternative features in multiple learning settings. Finally, we demonstrate through a user-study that our semi-supervised framework results in a ten-fold reduction in human annotation time at a minimal loss in detection accuracy compared to manual inspection.*

1. Introduction

Each year, hundreds of catastrophic events impact vulnerable areas around the world. Assessing the extent of damage caused by these crises is crucial in the timely allocation of resources to help the affected populations. Since disaster-locations are usually not readily accessible, the use of satellite imagery has emerged as a valuable source of information for estimating the impact of catastrophic events.

However, currently these assessments are mostly done by analyzing the pre- and post-event images of distressed areas by human photo-interpreters, making it a labor-intensive and expensive process. It is therefore important to scale-up damage detection to larger areas accurately and efficiently. Our work is a step towards solving this problem.

In the following, we summarize some of the key challenges that need to be addressed in this regard, and how our work contributes towards them:

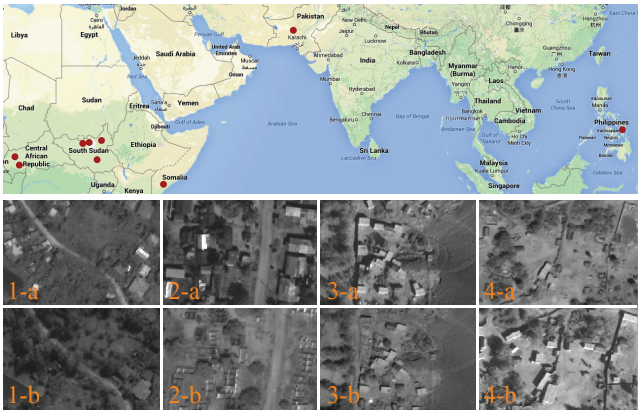


Figure 1: (Top) Areas of interest shown with red dots. Each area represents multiple local regions. We considered 12 local regions, spanning $4,665$ KM^2. (Bottom) Before and after imagery for different events shown in the two rows. These include (1) Typhoon (Phillipines, Oct. 2013), (2) Armed conflict (Central African Republic, Dec. 2013), (3) Earthquake (Pakistan, Sept. 2013), and (4) Internally displaced people's shelters (Somalia, May 2013).

1- Comprehensive Data-Set: Thus far, there has been a lack of comprehensive labeled data-set that could be used to explore automatic damage detection at scale. To this end, we present a benchmark data-set of 86 pairs of pre- and post-event satellite imagery of distressed areas covering $4,665$ KM^2 with the associated ground truth of damaged regions acquired by expert interpreters. This data-set was collected by using the satellites of DigitalGlobe Inc. Our data-set covers 12 different regions from around the world, and spans a wide range of terrains and climates, with a variety of damage types (see Figure 1). This data-set enables us to rigorously explore and make generalizable conclusions about the various facets of the problem at hand.

2- Appropriate Feature Choice: The scale of our problem naturally presents an accuracy-efficiency tradeoff for the features being considered. To this end, we introduce the use of trees-of-shapes features [28] in the bag-of-visual-words model [16] that focuses more on the shape character-

istics of a scene, as opposed to its edge attributes (as done by other popular descriptors *e.g.*, SIFT [18]). Our results show that this difference proves to be quite important to detect damaged areas accurately. We present a thorough empirical analysis for the effectiveness of our proposed scheme, and compare it to multiple alternatives.

3- Label Acquisition Cost: Given the high skill-set required from the photo-interpreters to assess the damage accurately, acquiring reliable ground-truth labels is particularly challenging for our problem. This high label acquisition cost makes it important to explore the various learning paradigms that could utilize the labeled data effectively. To this end, we present a thorough comparison of different learning strategies, including supervised, unsupervised and semi-supervised methods. Our results suggest the use of semi-supervised learning as a good trade-off between the label-acquisition cost and the detection accuracy. We present a user-study of the photo-interpretation efficiency provided by our framework, and report a ten-fold speed-up compared to an exhaustive manual inspection, at a minimal loss in detection accuracy.

In the following, we begin by reviewing the relevant previous works on damage detection. We then present the details of our benchmark data-set in § 3. In § 4, we go over the feature-sets and learning approaches we consider in our analysis, and present our experiments and results in § 5.

2. Related Work

The problem of damage detection is an instance of the broader problem-class generally referred to as novelty or change detection. Applying statistical approaches to solve novelty detection is a well-studied topic [20], and has been explored in multiple research-areas. Both supervised [17] and unsupervised [13] approaches have been tried to solve novelty detection, depending on the expected variance in the occurring novelty as well as the cost of label acquisition. More recently, there has been a growing interest in exploring semi-supervised learning approaches [2] to efficiently bootstrap the learning and label curation processes in a coupled way by having a set of humans in the loop. This learning model has also been tried at larger scales, using crowds for the purposes of label acquisition and curation [26].

Within the context of using satellite imagery, the main focus so far has been on unsupervised approaches [4] [12]. These techniques attempt to localize the outliers that correspond to scene changes [21] [23]. While effective in capturing generic changes, these approaches struggle with detecting the relevant ones.

There have also been supervised algorithms proposed in the past to provide accurate decisions regarding the occurring change [23]. In particular, these methods have focused on the relevant (as opposed to general) change detection [5]. However, most of these techniques rely on direct comparisons of pixel spectral responses, which can be adversely affected by sensor-misalignments and differences in acquisition geometries of the satellite imagery.

There has been a recent focus on using local image descriptors [25] in order to be more robust to data variabilities. These studies tend to show that the combination of local descriptors with supervised learning leads to high accuracies. However, this direction can be challenging to scale-up in cases where the label-acquisition cost is steep.

Our semi-supervised damage detection framework attempts to combine the benefits of previous supervised and unsupervised approaches, by requiring fewer labeled data and achieving higher detection accuracy simultaneously.

3. Data-Set Details

We now present details of our benchmark data-set.

3.1. Sources

To compile a comprehensive data-set for exploring large-scale damage detection, we had to rely on multiple data-sources. We obtained the different areas of interest (AOIs) around the world from the United Nations Institute for Training and Research (UNITAR/UNOSAT) [1], which is responsible for publishing maps of major disaster events. These maps provide geo-located points indicating relevant changes on the ground. We used this information to build ground-truth for 12 AOIs, the geographic layout of which is shown in Figure 1. We used this ground-truth information to evaluate each of our considered features and learning methods. Different types of crisis events were considered in our AOIs including armed conflicts, earthquakes, typhoons, and refugee-camp developments (see Figure 1 for example cases). We obtained the pre- and post-event images of our AOIs from DigitalGlobe's archive. We extracted pairs of gray-scale panchromatic images of submetric resolution, such that the images fully cover our AOIs.

3.2. Variabilities

In the context of damage analysis, the pre- and post-event images are likely to be captured from different sensors, which can have different ground resolutions. To incorporate sensor-variability in our data-set, we selected our 86 image-pairs from the four of DigitalGlobe'satellites: QuickBird, WorldView-1, WorldView-2 and GeoEye-1 with pixel-resolution of 0.61m^2, 0.5m^2, 0.46m^2, and 0.41m^2, respectively. To simplify the matching between sensors, we down-sampled all images to 1m resolution. The variability in our sensor combinations is given in Table. 1.

Another important source of variability in satellite images is the acquisition angles (sun, satellite elevation and azimuth), as it effects the directions and lengths of the casted

Satellites	QB-2	WV-1	WV-2	GE-1
QB-2	1	-	-	-
WV-1	9	14	-	-
WV-2	3	23	6	-
GE-1	5	11	9	5

Table 1: Four of the six of DigitalGlobe's satellite are used in our 86 image-pairs. Here the per-pixel resolutions of QuickBird (QB-2), WorldView-1 (WV-1), WorldView-2 (WV-2) and GeoEye-1 (GE-1) are 0.61m^2, 0.5m^2, 0.46m^2, and 0.41m^2, respectively.

shadows. While it would help to have image-pairs with similar acquisition angles, however given the tight time constraint of damage analysis campaigns, images most readily available generally have to be used. These images therefore do not always meet the similar acquisition angles constraint. To incorporate this variability in our data-set, we consider different acquisition angles for our 86 image-pairs, and this variability is illustrated in Figure 2. Because of high positional precision of on-board localization systems, the image pair are expected to be aligned with a displacement which rarely exceeds ± 5 meters.

3.3. Size

Our collected data-set covers a total area of 4665 KM^2, and contains 9.25 Billion 11-bit pixels. Furthermore, the area of significant damages labeled by expert photo-interpreters of UNITAR/UNOSAT is spread over 174 KM^2, which corresponds to 3.74% of our 12 AOIs. In our experiments, we found that using regions of 50×50m area (equivalent to 50×50 pixel window) sliding over the AOI with a stride of 10×10m (equivalent to 10×10 pixels) to be optimal. This level of granularity results in more than 86 million 50×50 image chips. To the best of our knowledge, the size and variability of our data makes our work the largest analysis of automatic damage detection ever published to date.

4. Computational Framework

We propose a semi-supervised learning scheme with humans in the loop to efficiently and accurately perform damage detection. Given a pair of image-strips, we extract the features of their (50×50) pixel windows sliding with a 10 pixel stride. Features from both strips are concatenated and used to learn a shared space that can accurately represent the notion of damage in the scene. Based on the damages detected in this unsupervised manner, we show the detected areas to a set of human observers who provide feedback labels regarding the true and false detections. We then use these labels in a supervised setting to finally learn the damage detection classifier. Our framework overview is presented in Figure 3. We now present the details for each of our framework steps.

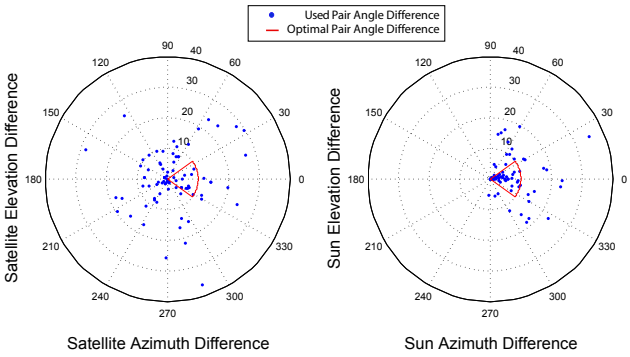


Figure 2: Scatter plot showing differences of sun and satellite angles (optimal versus used) for our 86 image strips.

4.1. Feature Extraction

Areas assessed for disaster-related damages usually have multiple man-made structures, each with their signature geometric shapes. This necessitates the use of features that can efficiently and accurately encode object shapes [11].

We therefore propose to use shape-distributions [10] [28] (SD) as features in our learning framework. Our work is the first to propose the usage of shape distributions in a bag-of-visual-words paradigm for the problem of large-scale damage detection, and validates its effectiveness in an in-depth manner. Using this extraction mechanism, we decompose each image-chip into its multiple constituent shapes, which are clustered and used in a bags-of-visual-words setting. We now present the details of our extraction mechanism.

4.1.1 Tree of Shapes

We particularly adopt the tree-of-shapes [22] representation for shape distributions, which organizes a given image into its *upper* and *lower* level-sets [8] based connected components. We define the upper level-set χ of a gray-scale image $u : \Omega \mapsto \mathbb{N}$ for a level-threshold λ as:

$$\chi_\lambda(u) = \{p \in \Omega \mid u(p) \geq \lambda\}. \quad (1)$$

The lower level set Ψ_λ can similarly be defined by inverting the above inequality. Connected components of level sets $\{\chi_\lambda(u) \mid \lambda \in \mathbb{N}\}$ are a lossless representation of u and provide its segment (as opposed to edge) based representation.

By construction, the components of the lower level set correspond to the holes of the components of the upper-level set, and vice-versa. Both these sets can be incorporated in one non-redundant tree structure called the tree of shapes, where the components are hierarchically nested and the lower and upper components are connected depending

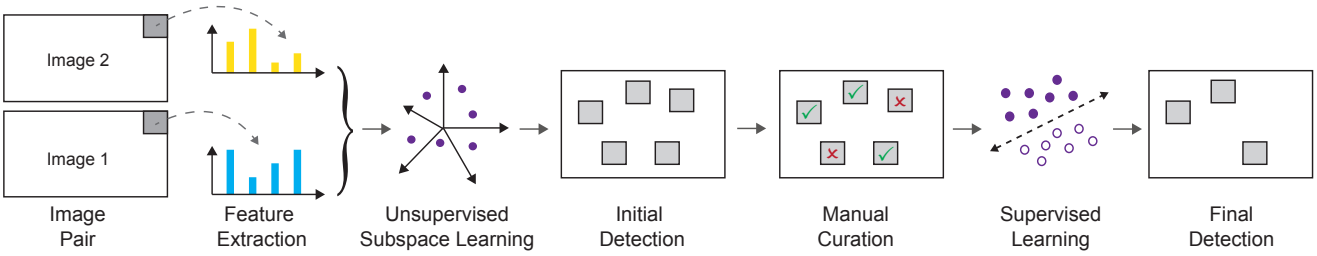


Figure 3: Given a pair of image-strips, we extract features of their overlapping windows, which are then used to learn a shared sub-space. Based on the changes detected in this unsupervised manner, we show the detected areas to a set of human-observers to obtain feedback from them. We use this feedback in a supervised setting to learn a damage detection classifier.

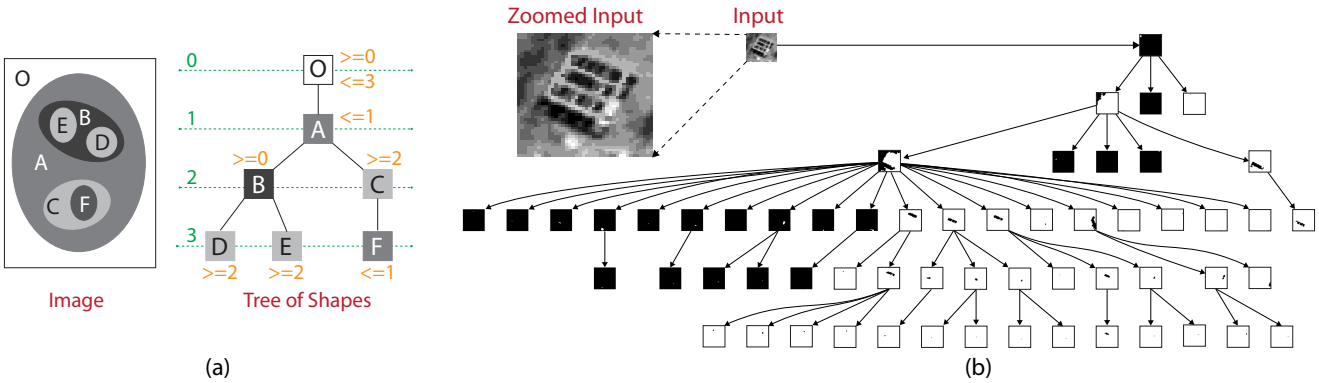


Figure 4: (**a**)- Illustration of tree of shapes using an input image with four gray-scale values. (**b**)- A 50×50 example image-chip of a roofless building is decomposed into its equivalent tree of shapes. The illustration shows the decomposition in upper (white) and lower (black) level sets components. These components add up to reconstruct the input chip in a lossless manner.

on the holes they fill (see Figure 4 for illustration). We follow the quasi-linear algorithm [9] for the construction of shape-trees, which enables us to extract them efficiently.

4.1.2 Shape Attributes

We characterize each shape extracted by a tree-of-shapes by its contrast, spectral response, and moment-based shape descriptors. We compute the contrast as the difference between the grey levels of a node and its parent in a tree-of-shapes. The spectral response is obtained as the average gray-level values for each shape-component. We derive our shape descriptors from the 2nd and 3rd order central moments [10] of a node. In particular, we use four shift-invariant shape descriptors, *i.e.*, area, eccentricity, and the first two Hu's moments [14]. These descriptors can be computed efficiently by exploiting the nesting property of nodes in a tree of shapes [9], requiring only a single pass over an input image for all the extracted shapes.

4.1.3 Descriptor Encoding

Having computed the shape attributes for all the shapes extracted in an image, we apply K-Means clustering on their respective shape-attributes in order to compute a shape-codebook. This codebook is used to perform vector quantization on the extracted shape-attributes in order to convert them into their respective codes, *i.e.*:

$$\arg\min_{\mathbf{C}} \sum_{i=1}^{\mathrm{N}} ||\mathbf{s}_i - \mathbf{B}\mathbf{c}_i||^2 \qquad (2)$$
$$\text{such that}, ||\mathbf{c}_i||_{l^0} = 1, ||\mathbf{c}_i||_{l^1} = 1, \text{and}, \mathbf{c}_i \succeq \mathbf{0}, \forall i$$

Here $\mathbf{s}_i$ represents the i-th extracted shape, $\mathbf{B}$ represents the shape codebook, and $\mathbf{c}_i$ represents the i-th code extracted. In practice, for each shape this is done by searching for its nearest neighbor entry in the codebook, and converting it into its sparse quantized representation. We finally take a weighted average for the codes of all the shapes extracted from an image. The weight of a shape is determined by the fraction of pixels it contains in an image. This process is repeated for each overlapping image-chip in the pre- and post-event image strip.

4.2. Subspace Learning

Based on our extracted shape-codes, we use linear canonical correlation analysis [23] to learn a subspace that accu-

rately encodes the notion of change between image strips. Let $\mathbf{c}_i$ and $\mathbf{d}_i$ be the feature vectors from the corresponding windows of pre- and post-event image-strips. Let $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_N]^T$ be the concatenated matrix of feature vectors from the pre-event strip. Similarly, we define the concatenated feature matrix $\mathbf{D}$ as $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_N]^T$. The canonical correlation analysis attempts to re-project $\mathbf{C}$ and $\mathbf{D}$ into $\tilde{\mathbf{c}} = \mathbf{Ca}$ and $\tilde{\mathbf{d}} = \mathbf{Db}$, such that $\tilde{\mathbf{c}}$ and $\tilde{\mathbf{d}}$ have maximum correlation, *i.e.*, our objective function is:

$$\tilde{\mathbf{a}}, \tilde{\mathbf{b}} = \arg\max_{\mathbf{a},\mathbf{b}} \frac{\mathbf{a}^T\mathbf{C}^T\mathbf{Db}}{\sqrt{\mathbf{a}^T\mathbf{C}^T\mathbf{Ca}}\sqrt{\mathbf{b}^T\mathbf{D}^T\mathbf{Db}}} \tag{3}$$

This optimization can be treated in terms of generalized eigen-values [3], where $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{b}}$ correspond to the top eigen vector for the above problem. We can find the top k eigen vectors of this problem and concatenate them to form $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$. Finally, the damage indicator (DI) is computed as the L_2 norm of the projected space difference:

$$\text{DI}(\mathbf{c}_i, \mathbf{d}_i) = \|\mathbf{c}_i^T\tilde{\mathbf{A}} - \mathbf{d}_i^T\tilde{\mathbf{B}}\| \tag{4}$$

Difference values diviating sufficiently from zeros do not fit the global mapping of features, and therefore correspond to the parts of the scene where changes have occurred.

4.3. Semi-Supervised Learning

We show the detection results based on the aforementioned unsupervised subspace learning to human photo-interpreters, who decide about the detections being true or false positives. Treating these samples as positive and negative classes, we stack up their corresponding shape-distribution codes and use them to train a linear support vector machine (SVM). For this work, we used linear SVM [6] with L1-regularization and L2-loss function. The learned classifier is then used for final detection on test images.

5. Experiments and Results

We now present the results of our experiments. We begin by explaining the features used and the evaluation metrics employed in our analysis.

5.1. Compared Features

We consider the following features in our comparisons:

1-Gray-Scale Distributions (GSD): Gray scale distributions represent histograms of gray scale values in an image-chip. These features have been extensively employed for change detection [21, 25], and serve as our baseline feature. We tried different quantization granularities for GSD and found 50 as optimal number of bins for our problem.

2-Optical-Flow (O-Flow): Treating damage detection as a pixel-flow problem, we use the magnitudes of optical-flow fields [19] as an estimate of the amount of damage occurred.

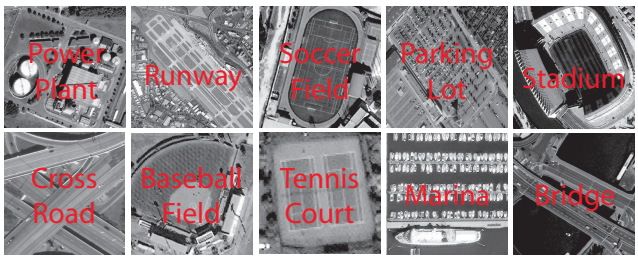


Figure 6: Subset of object classes used to fine-tune CNN.

3-Bag-of-Words with SIFT (B-SIFT): We tried SIFT [18] in a bag-of-visual-words [16] settings to see how much edge-based scene characteristics help in damage detection. For each image-chip, we tried different number of grid and patch sizes and found 8 and 16 to be their optimal values. We also tried different codebook sizes, and found 512 to be sufficiently representative. Finally, we tried spatial-pooling with 2 levels of pyramids, but found it not helpful, most likely due to the small chip-size. Our reported results therefore do not incorporate spatial-pooling.

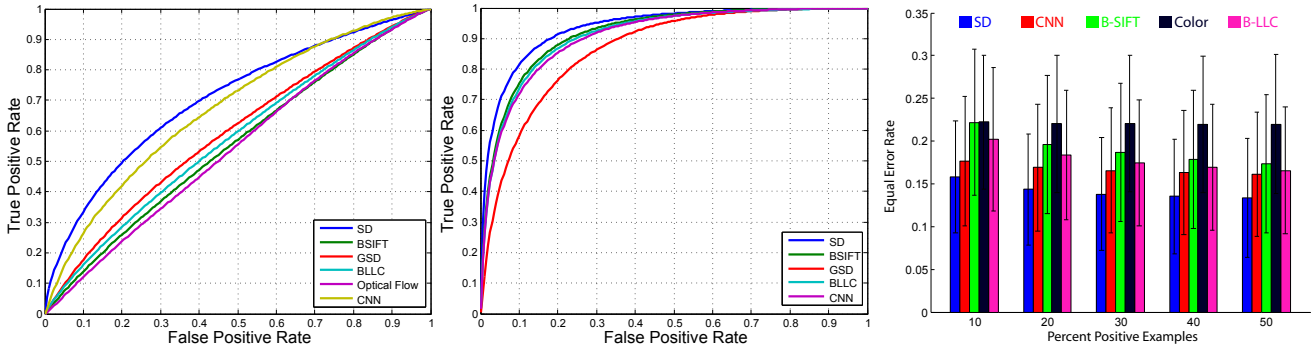
4-Bag-of-Words with LLC (B-LLC): To improve the encoding quality of B-SIFT, we also tried locality-constrained linear coding (LLC) [27]. The values for grid, patch and codebook sizes were set the same as they were for B-SIFT, *i.e.*, 8, 16, and 512 respectively. Similar to B-SIFT, we found spatial-pooling not to be helpful for B-LLC, and therefore did not incorporate it in our reported results.

5-Convolutional Neural Networks (CNN): To test the effectiveness of more non-linear feature-maps, we used Convolutional Neural Networks. We employed the pre-trained model from the ImageNet Large-Scale Visual Recognition Challenge 2012 [24], using the Caffe framework [15].

We fine-tuned this model by presenting it with images of man-made objects captured from satellites. We collected $\sim$ 150 thousand images of 20 object-classes from more then 250 cities around the world. Examples for some of these classes are shown in Figure 6. We rotated these images eight way around their centers, and used the resulting 1.2 Million images to fine tune our pre-trained model. The output feature dimensionality for the model was set to 1024.

5.2. Evaluation Metrics

Using the ground truth collected from our AOIs, we use the receiver operating curves (ROCs) to evaluate the different damage indicators. Note that true positive rate (TPR) represents the number of relevant damages detected over the total number of changes. Similarly, the false positive rate (FPR) represents the area of false-alarms with respect to the area of no damages. For a given (TPR, FPR) tuple, the associated damage indicator covers an area of $f \cdot \text{TPR} + (1 - f) \cdot \text{FPR}$,



(a) (b) (c)

Figure 5: (a) Average ROCs provided for 6 sets of features used in combination with the unsupervised canonical correlation analysis are shown. (b) Average ROC curve of the 5 sets of features combined with the SVM-L algorithm are presented. (c) Mean and standard deviation of the equal error rate are represented as a function of number of positive examples.

Feature Type	Unsupervised	Supervised
SD	0.339±0.088	0.123±0.068
CNN	0.360±0.120	0.161±0.072
B-SIFT	0.462±0.049	0.149±0.071
B-LLC	0.447±0.057	0.156±0.074
GSD	0.429±0.072	0.205±0.077
O-Flow	0.473±0.029	-

Table 2: Means and standard deviations of the Equal Error Rate for the 6 sets of considered features.

where f is the fraction of damaged area. Note that the quantity $f \cdot \text{TPR} + (1-f) \cdot \text{FPR}$ is also the relative search space size that needs to be examined by a human photo-interpreter during curation. We use the Equal Error Rate (EER) as a summary of the performance of an ROC curve, which is given as the ROC point satisfying $\text{FPR} = (1 - \text{TPR})$.

5.3. Feature and Learning-Strategy Comparison

The average ROC curves for our unsupervised and supervised settings are shown in Figures. 5-a and b respectively, with their corresponding Equal Error Rate given in Table. 2. For the the supervised case, the training data was generated by using 50% of the positive examples randomly sampled from ground-truth, with an equal number of negative examples sampled from areas not overlapping with the damaged areas. The reported results were obtained by testing the imagery over the remaining unused area.

Note that our proposed use of shape distribution (SD) features performs best in both supervised and unsupervised settings. Furthermore, it is the most consistent feature across all 86 image pairs, with accuracy variation of $\sim 7\%$ averaged across supervised and unsupervised settings. CNN features rank second in the unsupervised setting, while B-SIFT, B-LLC, and CNN all give comparable accuracies in the supervised setting.

Note that the ERR of 0.12 in Table 2 for our framework implies a 12% miss-rate (or 88% TPR). Given the expected fraction of damaged area f in our data as 3%, the expected reduction of search space provided by our framework is 1 - $[0.03 \times 0.88 + (1 - 0.03) \times 0.12] = 85.7\%$ of the full area.

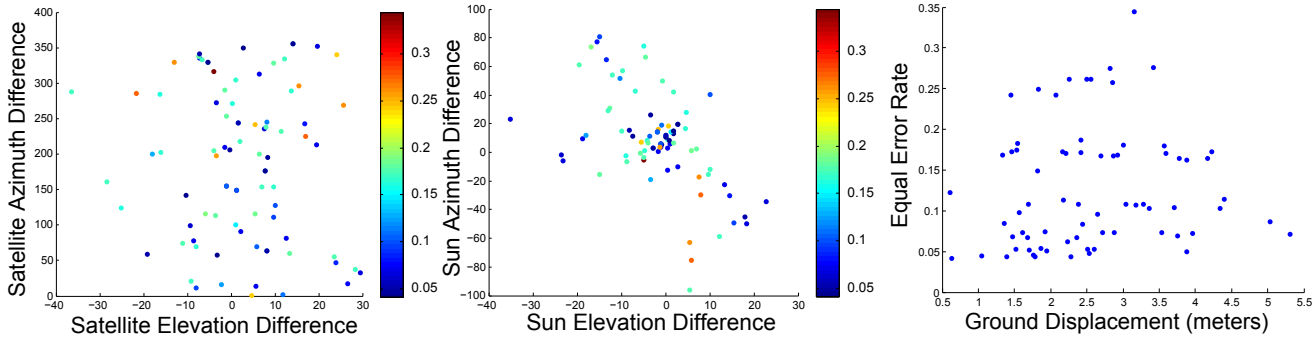
Figure 8 shows some example instances of damages detected and missed by our system. Most of the missed damages were subtle, naturally making them challenging to automatically detect. For cases where the damage occurred over buildings or well-defined structures, our framework was able to detect them with high precision. Our system tends to make mistakes for damages of more amorphus nature, such as mud-huts, or damages covering only a small area in an image chip such as an isolated house.

5.4. Effect of Training Sample Size

We present the average EER as a function of percentage of ground-truth positive instances in Figure 5-c. Besides GSD, we observe monotonic accuracy-improvements with the number of positive training examples for each feature. For SD features in particular, the EER improves by 0.6125% on average for every 10% increase of labeled training data size. This shows that even with very few labeled training data, the SD features are capable to disambiguate between damaged-versus-non-damaged areas effectively.

5.5. Run-Time Analysis

The processing time of our framework per 1-million image chips is given in Table. 3. For these times, we ensure that the image chips are co-located and overlapping such that acceleration such as image integral based filtering [7] can be employed. Times reported in Table 3 were obtained using Intel(R) Xeon(R) CPU E5 $-$ 2687W v2 @ 3.40GHz. In the



(a) (b) (c)

Figure 7: Effects of satellite, sun-angle differences and image misalignment on SD features in supervised setting.

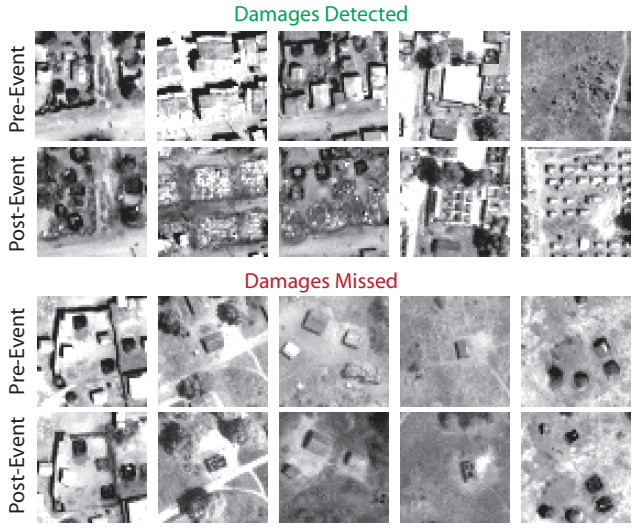


Figure 8: Example damaged areas detected by our framework as well as the ones our framework missed.

	Flow	GSD	B-SIFT	B-LLC	SD	CNN
Times	10	170	124	550	778	10406

Table 3: Compute-time (seconds) for features and unsupervised change indicator per million window-pairs.

case of optical-flow and CNN, GPU based implementations were used on an NVidia Tesla K20 card.

Note that while optical-flow is the fastest to compute, it performs quite poorly in accuracy. This is why we only consider optical flow in an unsupervised setting, and not in the supervised one. B-SIFT performs both faster and better than GSD. B-LLC is comparable to B-SIFT in accuracy while being less efficient. Finally, our proposed use of SD features takes 13 times less time than CNNs, while giving significantly better accuracy than all alternate features.

5.6. Effect of Acquisition Variability

The average EER using SD features as a function of sun, satellite angle-differences and image misregistration are plotted in Figure 7.

It is evident that satellite angle-differences do not impact our framework's accuracy, while the sun angle-differences seem to matter more. This is because sun angles impact shadows which can significantly vary image appearance. We also computed the average EER for each sensor combination given in Table 1. The standard deviation for this is 0.03, indicating that our approach is robust to using pre- and post-event imagery from different sensors. Finally, for each image pair, we compute the homography using SIFT key point. The homography's translation component has been retained, and its norm is depicted against the EER in Figure 7. The scatter plot shows independence between both x- and y-axes, highlighting the robustness of our approach to image displacements.

5.7. Photo-Interpretation Speed-Up

To quantify the time reduction during photo-interpretation using our framework we conducted a user-study in which we used two images each of size 7500×11250 pixels. Each image was split into $33,750$ non-overlapping image chips of 50×50 pixels. First, multiple experts were asked to scan the image-chips exhaustively and select the ones that contained damaged areas. In the second case, a change indicator was computed using our proposed framework, based on which we sampled image-chips to be shown to the human experts and decided which chips contain damage. The times and accuracies for both cases are given in Table 4.

Note that curation guided by the change indicator of our framework actually improves the false-positive rate by 76%. At the same time however, guided-curation produces a 15% reduction in the true-positive rate. Most of the damaged areas are however co-located, and therefore recovering from this loss in recall is easily achievable in practice. Overall,

	Time (s)	FPR	TPR
Exhaustive	41767	0.03	0.88
Guided	4048	0.007	0.74

Table 4: Times and performances of photo interpretation.

we obtain a ten-fold speed-up factor by applying guided curation as opposed to exhaustive search. This result highlights the importance of using a semi-supervised learning framework for the problem of large-scale damage detection.

6. Conclusions and Future Work

We presented a comprehensive analysis for the problem of large-scale damage detection using satellite imagery. We presented a novel use of hierarchical shape features in bags-of-visual words setting, and demonstrated its accuracy and efficiency advantages over multiple alternatives.

Going forward, we plan to improve the encoding scheme used in our current framework from hard quantization to one involving multiple soft-assignments. Furthermore, we plan on incorporating approximate sub-space learning mechanisms to further improve the efficiency of the unsupervised part of our framework. Finally, we plan to apply our damage-detection framework to a larger class of changes, such detecting urbanization patterns, and harbor and border monitoring.

References

[1] United Nations Institute for Training and Research, url-http://www.unitar.org/unosat/maps. 2

[2] G. Blanchard, G. Lee, and C. Scott. Semi-supervised novelty detection. *JMLR*, 11:2973–3009, 2010. 2

[3] M. Borga. Canonical correlation: a tutorial. *On line tutorial http://people. imt. liu. se/magnus/cca*, 4, 2001. 5

[4] L. Bruzzone and D. Prieto. Automatic analysis of the difference image for unsupervised change detection. *IEEE TGARS*, 38(3):1171–1182, May 2000. 2

[5] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Rojo-Alvarez, and M. Martinez-Ramon. Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection. *IEEE TGARS*, 46(6):1822–1835, June 2008. 2

[6] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM TIST*, 2(3):27, 2011. 5

[7] F. C. Crow. Summed-area tables for texture mapping. *SIGGRAPH*, 18(3):207–212, Jan. 1984. 6

[8] L. C. Evans, J. Spruck, et al. Motion of level sets by mean curvature i. *J. Diff. Geom*, 33(3):635–681, 1991. 3

[9] T. Geraud, E. Carlinet, S. Crozet, and and Laurent Najman. A quasi-linear algorithm to compute the tree of shapes of nD images. In *ISMM*, 2013. 4

[10] L. Gueguen. Classifying compound structures in satellite images: A compressed representation for fast queries. *IEEE TGARS*, 53(4):1803–1818, April 2015. 3, 4

[11] L. Gueguen, M. Pesaresi, A. Gerhardinger, and P. Soille. Characterizing and counting roofless buildings in very high resolution optical images. *IEEE GRSL*, 2012. 3

[12] L. Gueguen, P. Soille, and M. Pesaresi. Change detection based on information measure. *IEEE TGRS*, 49(11):4503–4515, 2011. 2

[13] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, and G. Coleman. Detection and explanation of anomalous activities: Representing activities as bags of event n-grams. In *IEEE CVPR*, 2005. 2

[14] M.-K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962. 4

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 5

[16] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE CVPR*, volume 2, pages 2169–2178, 2006. 1, 5

[17] W. Lee and S. J. Stolfo. A framework for constructing features and models for intrusion detection systems. *ACM transactions on Information and system security (TiSSEC)*, 3(4):227–261, 2000. 2

[18] D. G. Lowe. Object recognition from local scale-invariant features. In *IEEE ICCV*, 1999. 2, 5

[19] B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 81, pages 674–679, 1981. 5

[20] M. Markou and S. Singh. Novelty detection - a review: statistical approaches. *Signal processing*, 83(12), 2003. 2

[21] G. Mercier, G. Moser, and S. Serpico. Conditional copulas for change detection in heterogeneous remote sensing images. *IEEE TGARS*, 46(5):1428–1441, May 2008. 2, 5

[22] P. Monasse and F. Guichard. Fast computation of a contrast-invariant image representation. *IEEE Transactions on Image Processing*, 9(5):860 –872, may 2000. 3

[23] A. Nielsen. The regularized iteratively reweighted mad method for change detection in multi- and hyperspectral data. *IEEE Tran. Image Processing*, 2007. 2, 4

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014. 5

[25] C. Vaduva, T. Costachioiu, C. Patrascu, I. Gavat, V. Lazarescu, and M. Datcu. A latent analysis of earth surface dynamic evolution using change map time series. *IEEE TGARS*, 51(4):2105–2118, April 2013. 2, 5

[26] S. Vijayanarasimhan and K. Grauman. Large-scale live active learning: Training object detectors with crawled data and crowds. *IJCV*, 108(1-2):97–114, 2014. 2

[27] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *IEEE CVPR*, pages 3360–3367, 2010. 5

[28] G.-S. Xia, J. Delon, and Y. Gousseau. Shape-based invariant texture indexing. *IJCV*, 88(3):382–403, 2010. 1, 3