Information Fusion to Automatically Classify Post-event Building Damage State

Xiaoyu Liu[1], Lissette Iturburu[2], Shirley J. Dyke[1,*], Ali Lenjani[1], Julio Ramirez[2] and Xin Zhang[2]

[1] School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, United States;
[2] Lyles School of Civil Engineering, Purdue University, West Lafayette, IN 47907, United States;
* Correspondence: sdyke@purdue.edu

Abstract

Post-event reconnaissance missions are conducted after each major natural hazard event to collect valuable and perishable data. Teams of engineers and scientists are deployed to collect data, and in particular visual data (images), to support particular lines of inquiry, or to identify new lines of inquiry, that may lead to new knowledge about the best practices for the design of civil infrastructure. Visual data, combined with computer vision methods, can be a valuable tool for accelerating and automating these processes. Together they provide the means to more easily use the data, and organize the data sets so that they can be discovered in a search and reused. The focus of this paper is the development of an automated technique to classify the overall damage state of a building based on a typical set of reconnaissance images collected from a single building in the field. The motivating task is the collection of data and classification of damage into broad categories, such as those needed for computing the Hassan index [1]. The method adopts a naïve Bayes fusion algorithm [2] to combine the data, and an integrated sampling technique to reduce the computational time without compromising the quality of the results. Validation is performed using 29,543 past reconnaissance images from 720 buildings in different parts of the world that was collected, in part, for determination of the Hassan index.

Keywords: Information fusion, Naïve Bayes fusion, Post-event reconnaissance, Building damage state classification

1. Introduction

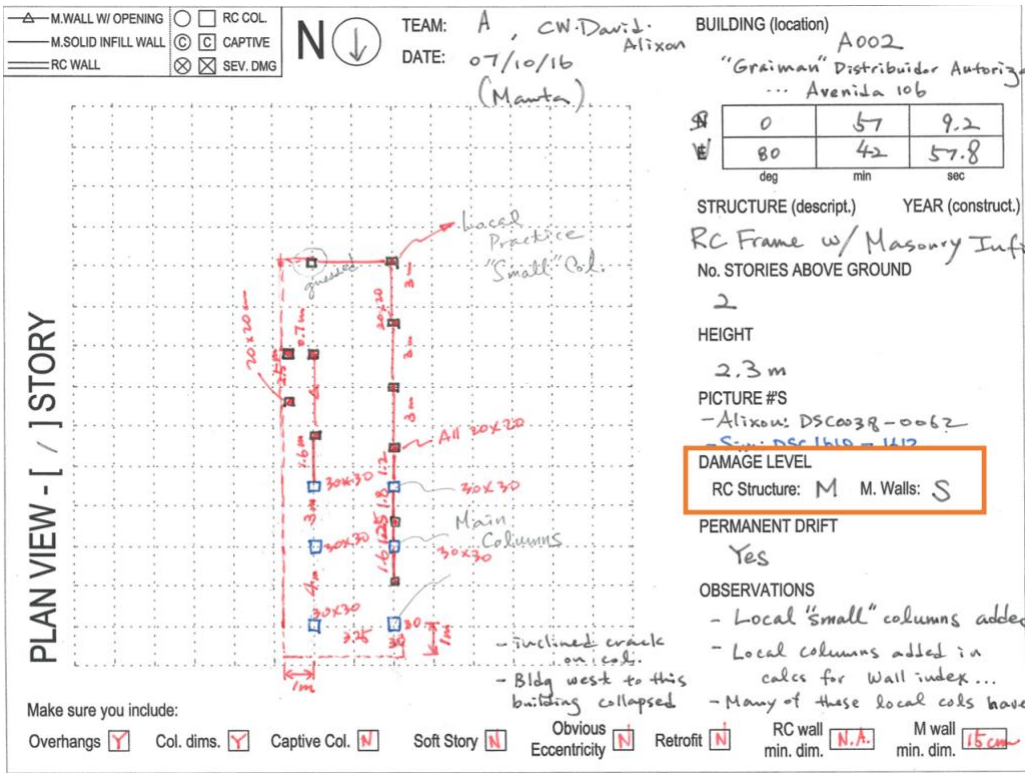
In the weeks after a natural disaster, reconnaissance teams dedicated to collecting perishable scientific data about the performance of the buildings are deployed. These teams are interested in collecting data, including a significant quantity of images, to synthesize lessons and identify new lines of inquiry by observing and classifying the damage in buildings in the region. Past examples of such reconnaissance missions include the teams deployed by ACI Committee 133, of which a summary of the lessons learned from these can be found in Laughery (2020) [3]. An example of the collected data during the missions are the 169 buildings, Villalobos et al. surveyed in 2016 [4]. For each building, the data collected included a number of photographs collected to document evidence of the post-event condition of the structure, and also other information like, the building coordinates, height, permanent drift, column dimensions, floor plan dimensions, and the overall damage state. These data are carefully documented for use by the researchers on the reconnaissance team, and are also organized and published in a public repository for other researchers to explore [5,6].

A sample of the type of form that is completed on-site during the building survey is provided in Fig. 1(a). Herein we will refer to this form as the *building survey form*. This form is accompanied by a large set of images collected by the engineers in the field to document their observations. Since good quality digital cameras have become widely available, such image sets have been growing in size, and in recent missions the teams typically gather about 100-200 images per building. Here we will refer to this visual data collected from a given building as the *set of images* (SOI, hereafter). The SOI contains images with scenes focused on structural components and nonstructural components, either exhibiting damage or showing undamaged views of damaged components, and also containing various undamaged components. The SOI also contains images of other objects, such as measurements and GPS devices, and other less

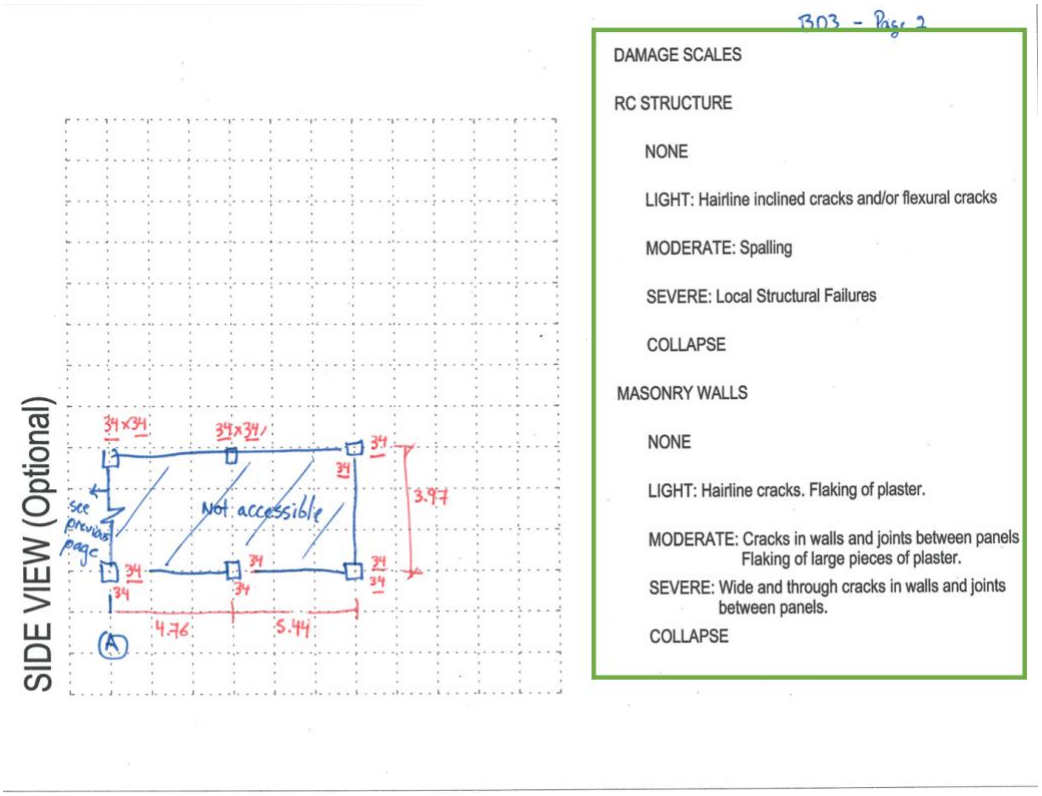
important objects. The SOI for a single building is certainly not comprehensive, and sometimes only representative damage to components is captured rather than collecting repetitive images.

Note that the sample building survey form here includes a sketch of the first floor of the building plan on the left, with an indication of where the structural columns are located. Information about the building is annotated, such as location, the number of stories and some basic observations. Some measurements are also collected to document the basic dimensions of the structure. In this particular mission, a key objective was to collect data to use in computing the Hassan index [7-9]. The Hassan index is a technique that has been used in many missions and by different teams to rapidly classify the vulnerability of a building based on the column and wall dimensions in each direction. An example of data collected using this approach can be found in Pujol et. al. [1]. Over the past two decades considerable effort has gone into obtaining data to support this technique. Reinforced concrete (RC) components (structural members) and masonry (M) components (non-structural members) contribute separately to the calculation of the index, and thus they are noted separately on the building survey form.

In the field, an important task for these reconnaissance teams is to classify the overall state of damage using general categories such as severe, moderate, etc. This damage classification task is performed separately for RC components and M components, as is evident from the information highlighted in the orange box in Fig. 1(a). These classes are assigned manually in the field following the guideline shown in the green box in Fig. 1(b). The guideline supports five states of damage each for both RC and M, including: none, light, moderate, severe, and collapse. Classifying the overall state of damage is just one example of the type of reconnaissance tasks that can be supported by automation and computer vision. Samples of images collected in the field are shown in Fig. 1(c).



(a)



(b)

(c)

Figure 1. Representative sample of the building survey forms used in the field: (a) the building survey form, (b) the guideline and (c) samples of images [10]

Assessing the post-event condition of buildings is complex and diverse, and in some cases, unsafe for reconnaissance teams. Villalobos (2016) showed that after the 2016 Ecuador earthquake, 45% of the buildings surveyed presented severe damage, 24% presented moderate damage and 31% light damage [4]. Image data is certainly collected from the exterior of the building as well. There is an interest in using drones to perform such data collection tasks in the future, although the tremendous number of images collected would require significant time and computational power to sort and analyze as well. Efforts have also been devoted to developing methods for post-event building condition assessment using such data. Computer vision techniques have been utilized to detect various types of damage in buildings such as cracks and spalling [11-13]. Yeum et al. [14] designed clear definitions and associated image classifiers to classify images of buildings into 'collapsed' or 'non-collapsed' based on images of the building overview (overview image). Satellite images also have been used to provide such information [15-17]. However, to date the research has focused on generating information from a single image. Techniques that can consider all of the images collected from a single building and produce a comprehensive output is lacking. Fusing the information from more than one images to support humans in making decisions has been developed for houses in hurricane surveys [18]. This work adopts a Bayesian-based method to fuse multiple overview

images and make a decision of the damage level of a house. This approach provides a basis for the method developed in this paper. A barrier to that approach when dealing with more complex structures is that the computational time increases considerably when the number of images grows, for instance when dozens of images are collected from a single building. This challenge is addressed in this paper.

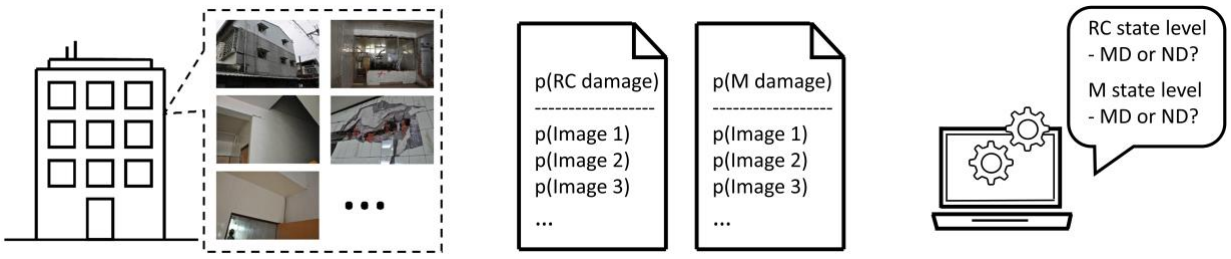
In this paper, we develop and validate an automated technique to process the visual data to classify the damage documented in the SOI for a building. The information collected during past reconnaissance missions and published in public repositories such as DataCenterhub [5] and Design-Safe [6] is used as the basis of the technique, and also as the ground truth for its evaluation. To establish this technique, the images are classified using convolutional neural network (CNN)-based image classifiers [19]. Two probability lists are formed, one for each category: for RC damage and M damage. Then, information fusion is performed to classify the overall RC damage state and the overall M damage state observed for the building. The main merit of this technique is that automation can assist survey teams by classifying the damage state of the building to support data organization and building-level classification. Such classification, into several broad categories based on damage state, will make useful data easier to search for in large reconnaissance data sets and serve the basis for a more targeted detailed assessment of particular structures.

The contents of the paper are organized as follows. Section 2 explains the methodology, including the schema designed for the image classifiers, and how to fuse the information to determine the overall damage states of the data set. Section 3 is the validation section, and describes the real-world dataset used for training and testing the image classifiers, and for validating the entire technique. A discussion of the results of this technique for earthquake induced structural damage, including pre-existing structural damage such as corrosion, is also included in this section. In Section 4, the conclusions of this work are provided along with a few recommendations for data collection and some of the existing techniques in the literature that can be used in conjunction with the technique developed in this paper.

2 Methodology

The overall workflow of the technique is shown in Fig. 2. The input to the technique is an SOI collected from a single building during a reconnaissance mission and stored in a digital format. The output of the fully automated technique is a classification of the overall RC and M damage state present in the building based entirely on the scenes in the images collected. Thus, the technique must make these predictions of the damage state based entirely on the available SOI.

To explain the technique, we divide it into three steps. Step 1 is to read the reconnaissance images that comprise one SOI. Based on our observations of building survey forms and datasets from past reconnaissance mission, these images can target building components with various types of damage, or they can contain no damage at all. Images of irrelevant objects can also be included in the SOI collected for a given building; they can be automatically filtered out with image classifiers. Metadata for the SOI are not needed, although sometimes information is available, including the time and date when the images were collected, GPS coordinates, etc. The approach requires a reasonable level of quality in the images, in terms of both visual content and standards. For such purposes, the images need to have a resolution larger than 299 by 299 pixels. Beyond that, the resolution of the images can vary in scale. The visual content of the images must be distinguishable, i.e., the damage should not take up of the entire image nor too small to be barely to be visible. Additionally, the images should not contain blur.



Step 1. Read reconnaissance images in an SOI

Step 2. Generate the damage state probability lists

Step 3. Use information fusion to predict the overall damage state for the building based on an SOI

Figure 2. Overall workflow of the approach

Step 2 is to generate values for the damage state in each image for populating two probability lists, one for RC damage and another for M damage. Taking RC damage as an example, each value in the list is a scalar probability between 0 and 1, and representing the probability of the corresponding image exhibiting RC damage state. This damage state probability is the raw prediction assigned by the respective image classifier to each image in the SOI. This classifier is applied after any irrelevant images are first filtered out automatically, which can be done using image classifiers. Irrelevant images are defined here as those for which the image classifier cannot generate a decision about the existence of RC damage, or for which no RC damage is present in the image. Detailed definitions of each of the classes used in the technique will be discussed in Section 2.1. A similar process is used for the M damage probability list. The generation of the two lists takes place in parallel, but they are entirely independent.

Then, in step 3, we use information fusion to determine the overall damage state for the SOI. The information fusion process is based on the naïve Bayesian method. For either RC damage or M damage classification, the process takes a probability list from step 2 as an input and generates a single probability value as the output. After performing information fusion, the output probability value is utilized to yield a damage state decision for the SOI corresponding to a particular building. A decision is made for the SOIs corresponding to each of the two types of damage, RC damage and M damage, respectively. For each type of damage, the decision will be determined as one of two states, either **moderate-to-collapse damage** (MD) or **none-or-light damage** (ND), indicating the overall damage state as determined from the SOI. The definitions for MD and ND will descripted in detail in Section 2.1. The decisions for RC damage and M damage are derived independently throughout the entire process.

Note that although this technique is developed based on a selection of data from past reconnaissance missions, the data used here are from many locations around the world and are quite broad. Thus, we anticipate that our classifiers will be robust to variations in architecture and construction; they can be applied without any retraining. If architectural styles and construction were to vary significantly in some location or future mission from those used herein to develop the technique, the classifiers could readily be updated.

2.1 Schema for the image classifiers

To support the technique, four independent image classifiers are designed for use in step 2 in the overall workflow shown in Fig. 2. All image classifiers used in this step are binary classifiers. The schema for the classifiers is shown in Fig. 3. To make the classification result consistent and to avoid ambiguity, it is important to ensure that each classifier has a clear definition and a distinguishable boundary between positive and negative results. The definitions are provided here, and then used for labelling a training and testing dataset later. These definitions are built based on the guideline as described in Section 1.

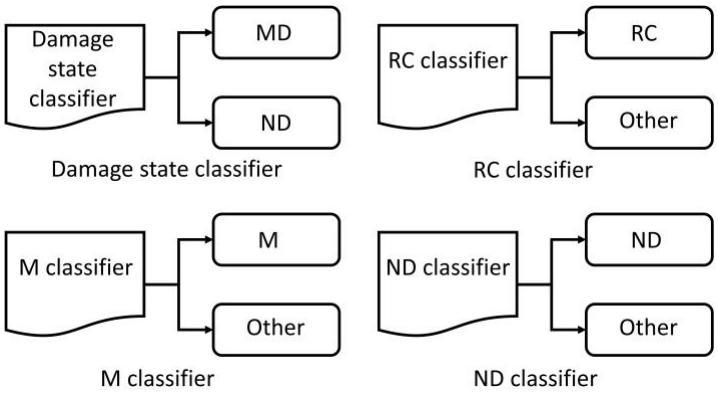


Figure 3. Schema designed for the image classifiers

The classifiers and the schema are as follows:

First, the **damage state classifier** classifies an image into either MD, or ND. Here, a single classifier applies to both RC damage and M damage. This approach takes into account the fact that RC damage and M damage are likely to be correlated with each other in post-event buildings, that is, if RC and M structural components both exist in that building, i.e., when the building is a reinforced concrete building that contains nonstructural masonry walls. And, more importantly, RC damage and M damage may frequently be present in a single image. Thus, this classifier is not meant to capture all types of damage but to focus on RC damage and/or M damage.

- **Moderate-to-collapse damage (MD)**: Image that contains building components having considerable damage. To be specific, the damage includes damage scales ranging from moderate, to severe, to collapse, as defined in the guideline. Moreover, it should be noted that damage in an image should be easily observed and identified, i.e., a significant part of the scene in the image should include the damage. Based on our past experience with similar classifiers, if the damage is extremely small in size as compared to the size of the image contents, it would be inappropriate to classify that image as an MD image. To quantify this relationship, we estimate that, to be classified as an MD image, the damaged region should take at least 30% of the entire area of that image without cropping.
- **None-or-light damage (ND)**: Image that contains building components having minor damage or no damage at all. This class includes damage scales from none to light, as defined in the guideline. This class is determined based entirely on the visual contents in the image, not the actual state of the building component. Thus, if the component is seriously damaged, an image capturing a healthy side of the component would also be considered as an ND image. Furthermore, an image with MD damage only in the background or damage that is hard for a human to distinguish would also be a valid ND image. To quantify this relationship, if the region with MD damage takes up no more than about a few percent (less than 5%) of the entire area of a single image without cropping, we still expect that image to be classified as a ND image.

Second, the **RC classifier** classifies an image into either RC damage, or other.

- **RC damage (RC)**: Image that contains RC damage. This class includes damage scales of moderate, severe, and collapse with respect to the RC components as defined in the guideline. The damage should be visible on the RC structural components. The RC component should be easily recognized from the image, with visible concrete, rebar, etc. Images classified as RC should be a subset of the images classified as MD.
- **Other**: Image that is irrelevant to the condition classification of the building. Two types of images are included in this class. The first type is an image that contains no visible signs of MD damage to the building or the components. The image should not contain either RC damage or M damage, as defined above. Furthermore, the damage scale of moderate, severe and collapse are the target images that should

be excluded from this class. The second type is the image that does not have the evidence to classify the building as ND, as defined in the above. An image belonging to ND can show no signs of damage, but it suggests that the building component captured in the image is in ND condition, therefore, it contributes to the decision of overall damage state to the building based on the SOI in the later process. Thus, ND images should be excluded from 'Other' class. Specifically, this class includes images about everyday objects, e.g., we have observed GPS, watches, people, vehicle, natural scenes, scenery other than infrastructure, etc. Some images with damage are also included in this class, if the scene includes irrelevant subjects such as people, papers, vehicles, etc. that represent at least 2/3 of the area of the damaged region in the image, making the damage hard to identify from the image.

Third, **M classifier** classifies images into either M damage, or other.

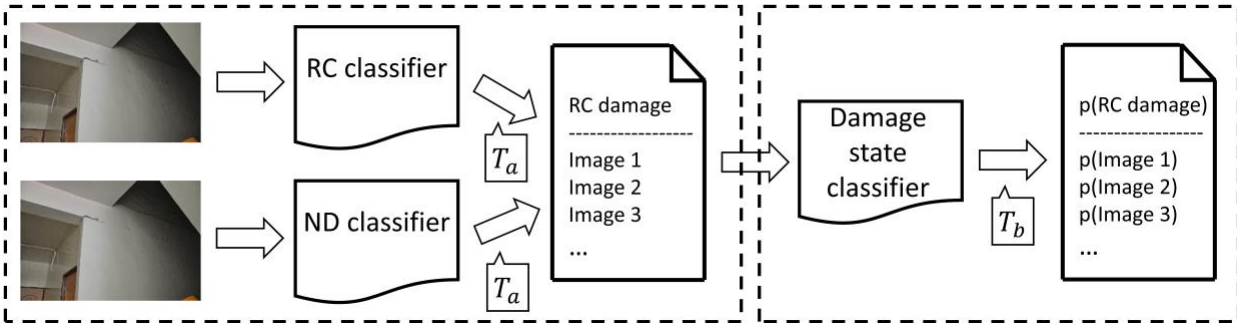
- **M damage (M)**: Image that contains M damage. This class includes damage scale of moderate, severe, and collapse with respect to the masonry components, as defined in the guideline. Similar to the definition of RC, the damage should be visible in the M components of the structure. The M component should be easily recognized from the scenery, with visible bricks, mortar, stones, etc. Images classified as M should be a subset of images classified as MD.
- **Other**: this class is defined in the same way as 'Other' in the RC classifier.

Fourth, the **ND classifier** classifies an image into either ND, or other.

- **ND**: this class is defined in the same way as 'ND' is defined in the damage state classifier.
- **Other**: this class is defined in the same way as 'Other' in the RC classifier.

2.2 Use image classifiers to generate probability lists

In this section, the details of step 2 in the overall workflow, as in Fig. 2, are explained. Using the schema for the image classifiers defined in section 2.1, we developed a process to generate two probability lists, one for RC damage and one for M damage. The process takes each image in the SOI as the input, and loops through each image in the SOI until it finishes.



Step 1a. Form the RC image list Step 1b. Form the RC probability list

Figure 4. Detailed process to form the RC probability list

Fig. 4 illustrates the process for predicting RC damage. We divide the process into two steps. Step 1a is to form the RC image list. First, each input image will be put through the RC classifier. The classification result determines whether or not the current input image should be included in the RC image list, i.e., if it contains RC damage with a sufficiently high probability. The decision is made by comparing the raw probability to a threshold, T_a. This approach is taken because the raw probability represents the confidence that the classifier should assign the corresponding label to that image. The closer the value is to 0 (or to 1), the more confident the classifier will be. Specifically, if the raw probability is larger than $1 - T_a$, we consider it to be valid to classify the image as RC damage, and it will be appended to the RC image list. The reason to use $1 - T_a$ is to have the threshold parameter is a region easy to visualize in the later steps.

Simultaneously, we implement the ND classifier on the input image, and follow the same procedure. The image is added to the RC image list if the probability exceeds the corresponding threshold. To simplify the method, we use the same threshold parameter for each case, and it will take the same value in the process. In this way, we identify all of the images in the SOI that can contribute to derive a decision about the condition of the building components. These include images that are highly likely to focus on RC components, and thus add evidence that the building's SOI is to be classified as a given MD state, and similarly for the ND classification.

Step 1b is to form the corresponding RC probability list. For each entry in the RC image list, each image will pass through the damage state classifier. This classifier assigns a probability to the image representing its likelihood of being either ND or MD. After comparing that value with a chosen threshold, T_b, the probability value will be appended to the RC probability list. It should be noted that we include images with a probability larger than $1 - T_b$ which is inclined toward MD, and images with a probability smaller than T_b which corresponds to ND. The RC probability list serves as part of the inputs to step 3 (from Fig. 2) for generating the overall RC damage state for the SOI.

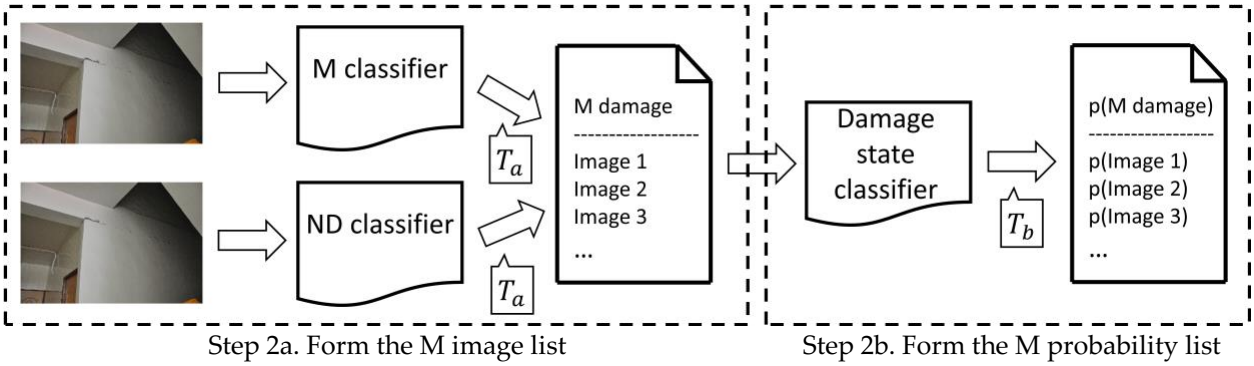


Figure 5. Detailed process to form the M probability list

A similar process is adopted for predicting M damage, as shown in Fig. 5. We use the M classifier and the ND classifier to select images that should be appended to the M image list in step 2a, then use damage state classifier to generate the M probability list in step 2b. The thresholds in the process, T_a and T_b, are chosen to be the same parameter in the process for RC damage, as in Fig. 4. They will be tuned simultaneously in Section 3.3. Also, we should point out that neither the RC image list and M image list, nor the RC probability list and M probability list, are mutually exclusive because an image can contain both RC damage and M damage at the same time. In that case, the image will be included in both lists, and measured by the damage state classifier in two separate processes.

2.3 Information fusion

After acquiring the RC probability list and the M probability list, information fusion is used to fuse each of the probability lists into a single probability value, as in Step 3 in Fig. 2. A single probability value is used to represent the damage state of either the RC components or M components of the building based on the SOI. In the following sections, we will explain the details of the information fusion algorithm. Subsequently, we will introduce a concern regarding the computational time of the algorithm. To address this concern, we integrate a sampling method to speed up the computations and the entire procedure will be explained.

2.3.1 Details of the information fusion algorithm

We use naïve Bayesian fusion to fuse each probability list with the goal to arrive at the fused probability indicating the damage state of the building based on an SOI [2,18]. This procedure is applied separately to generate a damage state for both RC and M components. Let $x_1, \ldots, x_n$ represent each image associated with

the probability list, and $p_1(x_1), \dots, p_n(x_n)$ represent the damage state probability of each image. Using these values, the probability of the building based on an SOI is written as $p(D = d|\, x_1, \dots, x_n)$. And it is expressed as

$$
\begin{aligned}
& p(D = d|\, x_1, \dots, x_n) \\
&= \sum\nolimits_{d_1,\dots,d_n \in \mathcal{D}} p(D = d|D_1 = d_1, \dots, D_n = d_n, x_1, \dots, x_n) p(D_1 = d_1, \dots, D_n = d_n|x_1, \dots, x_n) \\
&= \sum\nolimits_{d_1,\dots,d_n \in \mathcal{D}} p(D = d|D_1 = d_1, \dots, D_n = d_n) p(D_1 = d_1, \dots, D_n = d_n|x_1, \dots, x_n) \\
&= \sum\nolimits_{d_1,\dots,d_n \in \mathcal{D}} p(D = d|D_1 = d_1, \dots, D_n = d_n) \prod\nolimits_{i=1}^{n} p(D_i = d_i|x_1, \dots, x_n) \\
&= \sum\nolimits_{d_1,\dots,d_n \in \mathcal{D}} p(D = d|D_1 = d_1, \dots, D_n = d_n) \prod\nolimits_{i=1}^{n} p(D_i = d_i|x_i)
\end{aligned}
$$

To start with, $x_1, \dots, x_n$ are treated as the prior for the fused probability, since $p(x_i)$ only relates to x_i. n is the total number of the images in the probability list. Then, we define D as the random variable indicating the damage state of the building based on an SOI. And d will be a realization of the numerical value, as either 0 or 1, 0 denotes the damage state ND, and 1 denotes MD. Following that, we use the sum rule of probability to expand $p(D = d|x_1, \dots, x_n)$ to all the possible scenarios that each x has a chance being classified as ND or MD. Similar to the damage state of the building based on an SOI, D_i is the random variable for x_i, and d_i is its numerical value. Then, $p(D_1 = d_1, \dots, D_n = d_n|x_1, \dots, x_n)$ is written as the product of the probability for each x, this is because we consider the chance for each x being classified as ND or MD are independent from each other. In the end, $p(D_i = d_i\,|x_i)$ is the probability for x_i being classified as d_i. And $\mathcal{D}$ is the set consisting of all the possible combinations of $d_i = \{0, 1\}$.

The conditional probability is defined as,

$$
p(D = d|D_1 = d_1, \dots, D_n = d_n) = \begin{cases} \dfrac{\sum_{i=1}^{n} d_i}{n}, \forall\, d_i = 1, p(D_i = d_i|x_i) < 0.5 \\ \left\lceil \dfrac{\sum_{i=1}^{n} d_i}{n} \right\rceil, \exists\, d_i = 1, p(D_i = d_i|x_i) \geq 0.5 \end{cases}
$$

where $\forall\, d_i = 1, p(D_i = d_i|x_i) < 0.5$ means for all $d_i = 1$, $p(D_i = d_i|x_i) < 0.5$ or all x are classified as more likely to ND over MD. In such case, we use the ratio of sum of d_i to n as the conditional probability. On the second case, $\exists\, d_i = 1, p(D_i = d_i|x_i) \geq 0.5$ means there exists $d_i = 1$, $p(D_i = d_i|x_i) \geq 0.5$ or at least one of x is classified as more likely to MD over ND. In such case, we use $\lceil \cdot \rceil$ of the ratio to compute the conditional probability. $\lceil \cdot \rceil$ is the mathematical ceiling of the argument. This indicates if at least one x is classified as MD or $d_i = 1$, then the conditional probability is 1.

2.3.2 Use of sampling to speed up the process of information fusion

There are two characteristics we are looking for in an information fusion algorithm. Without a doubt, the first one is 'accuracy'. The algorithm should be designed to reflect the damage state of the image set as much as possible. An evaluation of accuracy will be carried out in Section 3. Aside from accuracy, we are also interested in computational efficiency to get the fused result. To illustrate why this is important, an example of how to perform information fusion is provided in Table 1. The input, the probability list, is chosen to have four elements, with values $[0.0115, 0.1635, 0.6988, 0.1226]$. It should be noted that this hypothetical example pertains to step 3 in Fig. 2, which means we are explaining what happens after all the image classification and filtering. The resulting list of probabilities is put through the information fusion algorithm. As explained in Section 2.3.1, the fused probability is formed by the sum rule. Thus, the

algorithm must consider all possible combinations of the input list to compute the associated products and add them together. That will yield the fused probability. However, the total number of combinations is $C_{total} = \binom{N}{1} + \binom{N}{2} + \cdots + \binom{N}{N}$, where N is the number of elements in the input list. Using the example in Table 1, $C_{total} = 16$, and it consumes a computation time of 0.9975 milliseconds in total. While this computation time is acceptable for a four-element list, C_{total}, will grow drastically as N increases. When N is 10, C_{total} will be 1,024. When N reaches 20, C_{total} will be 1,048,576. And when N approaches 25, C_{total} will be a whopping 33,554,432. Since the computational time for each combination varies with the number of elements in the input list, consider that a single combination requires 0.06234 milliseconds (roughly the average time taken in the example), then, N of 25 will be about 34.86 minutes. This value is a comparatively long time to endure for our technique to assess one building. Given the fact that an SOI will easily contain tens or hundreds of images, potential large computational times will inevitably limit the value of our technique for larger image sets. This remark is based on the assumption that N will increase as the total number of images in an SOI increases.

Table 1. Example of the time required for the conventional information fusion algorithm

	Combinations	Product
1	[]	0.000000
2	[1]	0.000636
3	[2]	0.010678
4	[3]	0.506983
5	[4]	0.007634
6	[1, 2]	0.000248
7	[1, 3]	0.005898
8	[1, 4]	0.000178
9	[2, 3]	0.099093
10	[2, 4]	0.002984
11	[3, 4]	0.070841
12	[1, 2, 3]	0.001153
13	[1, 2, 4]	0.000052
14	[1, 3, 4]	0.000824
15	[2, 3, 4]	0.013846
16	[1, 2, 3, 4]	0.000161
Input and output		
Input: probability list, [0.0115, 0.1635, 0.6988, 0.1226]		
Output: fused probability, 0.721209		

To address this issue, we adopt a sampling method to speed up the fusion process. The basic idea is to sample a smaller number of elements from the input list, and iteratively perform the information fusion using the sampled list. Then, the process is repeated until the result converges.

The entire implementation is shown in Algorithm 1. To start with, we have the input probability list, A, and we define an empty list $p\_history$ for keeping track of the temporary fused probability, p_temp, which is the fused probability computed at each iteration, an empty list $e\_history$ for holding all the e, which are the errors. Before the iteration, we first check whether or not $length(A)$, the number of the elements in the input list, is larger than 5; if not, we simply compute the fused probability with A and return the fused probability as the output. The function fuse_probability() applies the original fusion algorithm as introduced in Section 2.3.1. However, if the answer is yes, the process moves to the iteration steps. We define two stop conditions, either of which will stop the iterations: one is e reaching $e\_threshold$ which is

set to 0.01, since the maximum possible value of a probability is 1, $e\_threshold$ can be regarded as 1% of the maximum value, $e\_threshold$ is chosen for practical reasons so that the algorithm will reach a relatively accurate result in a reasonable iterations; and the other is reaching the $iteration\_limit$ which we set to 1,000. Based on experience developed during the present study, we define the number of the elements in the sampled list, N_sample, as 5. Inside each iteration, the process moves to the sampling steps.

To fairly represent A with the sampled list, B, we adopt the proportional stratified sampling strategy [20]. This strategy is typically used when the sampling group (here, A) can be divided into several subgroups. This strategy samples from each of the subgroups independently. If we consider the procedure in step 2 from Fig. 2, the RC probability list and M probability list are filtered by $threshold\_b$ to select candidates that fall into their respective classes with high confidence. This approach offers the chance to cluster A into two subgroups, one associated with probability values smaller than $threshold\_b$ which is defined in Section 2.2 and its value will be discussed and assigned in Section 3.3.2, and the other associated with probability values larger than $1 - threshold\_b$. We use proportional allocation to determine the number of elements to sample from each subgroup. Simply, the two sampled lists, denoted A_low_sample and A_up_sample, are sized to be proportional to the ratio between the size of the two subgroups, and they must add up to N_sample. Then, A_low_sample and A_up_sample form B. This sampling process is shown in lines 8 to 14 in Algorithm 1.

After sampling, p_temp is computed from B with the fusion algorithm. After appending p_temp to $p\_history$, we calculate e which is defined as the absolute difference between the current p_temp and the mean of $p\_history$. When either e is less than or equal to $e\_threshold$ or the process reaches 1,000 iterations, the iteration stops. When the iterations stop, if the total number of iterations is smaller than $iteration\_limit$, we take the last p_temp as the fused probability, p. Otherwise, we take the mean of $p\_history$ as p. This case applies in the rare cases in which the process does not converge early and the maximum iterations is reached. In our experience, this case has a very small chance of occurring. When it does happen, the modified process using sampling is still able to fulfill the goal of capturing the damage state of the image set, assuming that the input probability list is correctly provided. This approach works because we design the entire technique to predict a building based on an SOI as either MD or ND, rather than aiming to provide an exact probability value.

Algorithm 1. Implementation of the modified information fusion algorithm

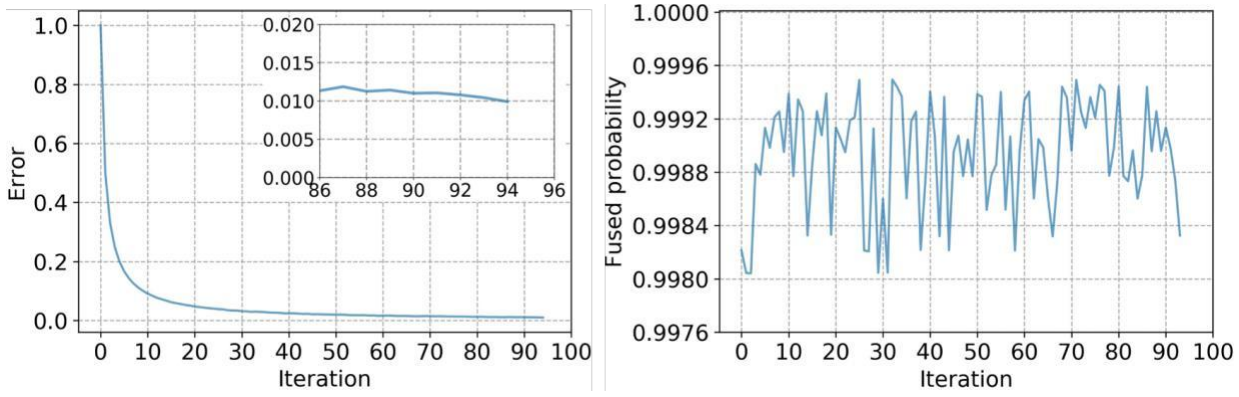
Algorithm 1:

Input: probability list, A
Output: fused probability, p

```
1   p_history = [], e_history = [], e_threshold = 0.01, iteration_limit = 1000, N_sample = 5
2   if length(A) <= N_sample
3       return p = fuse_probability(A)
4   else
5       e = 1, iteration =0, e_history = [1]
6       while e > e_threshold and iteration < iteration_limit
7           B = []
8           A_low = [element for element in A if element < threshold_b]
9           A_up = [element for element in A if element > 1-threshold_b]
10          Number_low = round(N_sample*length(A_low)/length(A))
11          Number_up = N_sample – Number_low
12          A_low_sample = random(A_low, Number_low) # randomly sampling
            Number_low of elements from A_low
```

```
            A_up_sample = random(A_up, Number_up) # randomly sampling Number_up of elements from A_up
            append elements of A_low_sample, A_up_sample to B
            p_temp = fuse_probability(B)
            append p_temp to p_history
            e = abs(p_temp – mean(p_history))
            append e to e_history
            iteration = iteration+1
    if iteration < iteration_limit
            return p = p_history[end]
    else
            return p = mean(p_history)
```

We examine the modified information fusion method with sample data consisting of a probability list with 27 elements, as $[0.9630, 0.9594, \ldots, 0.03351]$. The process stops at the 94th iteration where it reaches the stopping criterion when e meets $e\_threshold$ which is set to 0.01. The results are shown in Fig. 6. The error history is plotted in Fig. 6(a). Clearly, e decreases as the process proceeds. For a detailed view of the 94th iteration when $e\_threshold$ is reached, the error history from iterations 86 to 96 is shown in the upper-right corner of Fig. 6(a). The history of fused probability, $p\_history$, is shown in Fig. 6(b). The final outcome of the modified algorithm is 0.9983. As a comparison, the fused probability for the original fusion algorithm is 0.9999, and the number of combinations for the original algorithm would be 134,217,728. Meanwhile, the modified process drops this number to $94 * C_{total}(N = 5) = 3{,}008$. The actual computation time for the modified process is 0.0728 seconds, while the original algorithm requires 3.15 hours.



(a) (b)

Figure 6. Results of the modified information fusion algorithm: (a) error history, (b) fused probability history

3 Validation of the Technique

3.1 Validation dataset

We validated the technique with real world datasets from reconnaissance missions. The datasets were collected from the reconnaissance missions after several earthquakes, including Bingöl, Turkey in 2003; Haiti in 2010; Nepal in 2015; Taiwan in 2016; Ecuador in 2016; and Mexico City, Mexico in 2017 [10,21-25]. In these missions, 33,248 reconnaissance images were collected from 800 buildings. The images cover a various of structural components with different health conditions. And they are taken from both inside and outside of buildings. Some sample images are shown in Fig. 7.

During the missions, the reconnaissance teams walk through each of the buildings and manually collect each of the images in the datasets. For this work, we organized the datasets according to the building

that they were collected from. We do not specifically make use of the event itself. For each building, the datasets tend to include a number of reconnaissance images and a building survey form, as shown in the sample in Fig. 1. It should be noted that the images and the building survey forms in the original datasets do not exactly correspond to each other perfectly. Some buildings have images but lack of building survey forms, while some lack the images instead. Also, some building survey forms are empty or not legible for various reasons. As our technique aims to evaluate a building based on an SOI instead of single images, thus, we only use the data that has both an SOI and a valid building survey form for the same building. After examination of the data, there are 29,543 images and 720 buildings left for use in the following validation.

Haiti earthquake in 2007 (129 buildings, 3,129 images)

Ecuador earthquake in 2016 (171 buildings, 6,442 images)

Taiwan earthquake in 2016 (119 buildings, 9,164 images)

Mexico City earthquake in 2017 (81 buildings, 3,245 images)

Figure 7. Sample images from the reconnaissance image database [10,21-25]

To fully test the technique, we divide the full dataset mentioned above into two parts, validation dataset 1 and validation dataset 2. The detailed assignments for the dataset are shown in Table 2. From validation dataset 1, we select some of the images to form the training set and the testing set for each of the classifiers used in this technique. The total number of images in validation dataset 1 is 26,298, and we select 5,119 of them for this purpose. Next, validation dataset 1 will be used to tune the thresholds. In the end, validation dataset 2 will only be used for validating the technique. Since the process to develop the technique has not seen any of the data from validation dataset 2, using it for validation of the method is intended to represent an assessment of the performance of the technique on newly collected data. To form the two validation datasets, the events are randomly split as 90% (as 5) for validation dataset 1 and 10% (as 1) for validation dataset 2.

Table 2. Details of validation datasets

		Bingöl	Ecuador	Haiti	Nepal	Taiwan	Total
Validation dataset 1	RC: MD – ND	36 – 19	118 - 53	76 – 53	83 – 82	32 – 87	345 - 294
	M: MD – ND	49 – 6	133 - 38	91 - 38	119 – 46	33 – 86	425 - 214
	Total	55	171	129	165	119	639
		Mexico City					
Validation dataset 2	RC: MD – ND	33 – 48					
	M: MD – ND	46 - 35					
	Total	81					(unit: SOIs)

Also, as explained in Section 1, reconnaissance teams manually evaluate the RC damage state and M damage state of each target building and document them in the building survey form. The RC damage state or M damage state is given in the building survey form as one of five possible states, based on the following set of options: {none, low, moderate, severe, collapse}. The guidelines used by the reconnaissance teams were consistent across all the different datasets used in this validation section. As discussed in Section 2.1, we merge the five states specified in the guidelines into two states, designated MD and ND. The number of building SOIs that include the corresponding ground truth are also listed in Table 2.

3.2 Classifier design

To train the classifiers, we manually select images from validation dataset 1 and label them based on the guidelines used by the reconnaissance teams. Several sample images from each class are shown in Fig. 8. In general, we label four categories of images, including RC damage, M damage, ND and other. For training and testing the classifiers, RC and other form the dataset for the RC classifier, M and other form the dataset for the M classifier, ND and other form the dataset for the ND classifier, and RC and M form MD, together with ND, they form the dataset for the overall damage state classifier. Note that RC and M images are not mutually exclusive, as we discussed in Section 2.1. Also, RC and M damage can occur simultaneously and be captured in one image. The detailed number of images labelled and used are also listed in Fig. 8. The number of images in each class are not uniformly selected. Instead, we select images with ambiguous visual contents, and manually label them strictly by the definitions formed in Section 2.1. This approach results in a more robust classifier that can correctly classify the more challenging scenes. In total, 5,119 images are used here, as compared to the total number of images in validation dataset 1.

Image samples labelled as RC damage (843 images)

Image samples labelled as M damage (887 images)

Image samples labelled as ND (2363 images)

Image samples labelled as other (1026 images)

Figure 8. Labelled image samples for each class [10,21-24]

We use the same model for building all the classifiers. VGG16 is selected to be the base model of the classifiers, as its performance is one of the best in the ImageNet competition in 2014 [26]. The 5 main convolutional blocks are kept, and a new top block is attached to replace the original top block. The new top block generates a probability from 0 to 1 for each image, representing one of the binary categories of each classifier. During the training process, the pre-trained VGG16 weights, trained with ImageNet dataset, is used. The weights of the first two convolutional blocks in VGG16 are fixed, and the latter three blocks are allowed to be tuned. Together with the top block, the weights of the last three blocks are the only ones that are trained on the datasets. Since the training datasets for each classifier are not balanced, we set class weights to compensate for the imbalanced dataset in the training process.

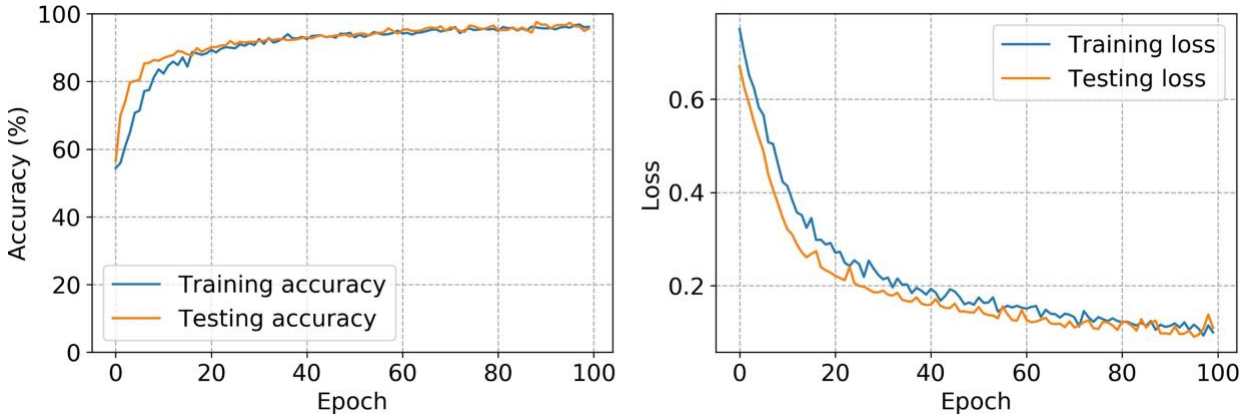


Figure 9. Training and testing of the RC classifier

Each classifier is trained for 100 epochs, and we use the final weights as the ones in the following test. As an example, the training and testing history for the RC classifier is shown in Fig. 9. In the first 20 epochs, the loss drops quickly while the accuracy rises, then both histories change gradually. The training accuracy and testing accuracy for each classifier that occur in the final epoch are listed in Table 3. The overall performance of the classifiers is acceptable. We observe the scenes in damage state classifier are more complex as compared to the scenes for the other three classifiers, thus attribute its slightly lower accuracy to this fact.

Table 3 Final metrics of all the classifiers

	Trained epochs	Final training accuracy	Final testing accuracy
Damage state classifier	100	94.28%	92.88%
RC classifier	100	96.17%	95.70%
M classifier	100	98.29%	98.97%
ND classifier	100	95.86%	94.38%

3.3 Threshold tuning

In this section, we tune T_a and T_b to find the values that yield the best performance of the overall technique. As mentioned in Section 2.2, the results of the RC classifier, M classifier and damage state classifier are filtered using the corresponding thresholds before moving to the next step in the process. Conceptually, the filters remove the portion of the results based on the confidence with which the categories are assigned to the images. To carry out the tuning, we implement the technique with validation dataset 1 using a range of values of T_a and T_b. After generating the RC probability list and the M probability list, we fuse each probability list using the method explained in Section 2.3 to form the two overall probability values, RC fused probability (RCFP) and M fused probability (MFP). Then, we simply use a threshold of 0.5 to decide whether the building based on an SOI should be classified as ND (< 0.5) or as MD (> 0.5). The result is evaluated by the metrics of recall and precision on the entirety of validation dataset 1.

3.3.1 Metrics for evaluating the technique and on an imbalanced dataset

First, we explain the metrics used for evaluating the results [27]. Then for demonstrating how to interpret the metrics, we generate hypothetical data, and the associated results are shown with the confusion matrix in Table 4. The main items in the confusion matrix are denoted as follows: for the predicted damage state classification, the damage type (RC or M) followed by a "--", the prediction (MD or ND), followed by a "-

-", and true (or false) of the prediction, e.g., RC-MD-True means the RCFP prediction is RC MD and it is True. Similarly, RC-MD-False means the RCFP prediction is RC MD and it is False. The latter indicates that the ground truth for the classification is RC ND; for the column of total, the damage type (RC or M)-the ground truth (MD or ND)-Total; for metrics, the damage type (RC or M)-the ground truth (MD or ND)-recall, and the damage type (RC or M)-the prediction (MD or ND)-precision. After introducing the main items, the metrics are defined as follows:

$$\text{RC-MD-recall} = \frac{\text{RC-MD-True}}{\text{RC-MD-True}+\text{RC-ND-False}}$$

$$\text{RC-MD-precision} = \frac{\text{RC-MD-True}}{\text{RC-MD-True}+\text{RC-MD-False}}$$

The ND and M related confusion matrix and metrics follow this same pattern. Nevertheless, when the dataset is imbalanced, there is an issue regarding the metrics shown in Table 4. The recall values for RC-MD and RC-ND are both pretty high, and this means the technique is quite successful in retrieving overall damage classification, both those classified as MD and ND. However, the precision values vary considerably; RC-MD-precision is 100% and RC-ND-precision is merely 1%. This outcome indicates that in the results that are predicted as ND, only 1% of them is True. The reason for this biased indication brought by the metrics is the imbalanced dataset. Because the total number of samples with a ground truth of RC MD is 101,000 while the number with RC ND is only 10, no matter how well the technique performs, RC-ND-precision will always struggle and have a relatively low value [28].

Table 4. Hypothetical data and results of the demonstration

RC (Hypothetical data and results)				
Ground truth\prediction	**MD**	**ND**	**Total**	
MD	RC-MD-True: 100,000	RC-ND-False: 1,000	RC-MD-Total: 101,000	RC-MD-Recall: 99%
ND	RC-MD-False: 0	RC-ND-True: 10	RC-ND-Total: 10	RC-ND-Recall: 100%
	RC-MD-Precision: 100%	RC-ND-Precision: 1%		

To compensate for this imbalance, we use a similar idea to the one adopted in Section 2.3.2 for accelerating the information fusion process. For the imbalanced dataset, we use a sampling method and sample from the categories with a larger number of SOIs. With these sampled results we compute the metrics, and then repeat the process until the metrics converge. The number of samples used is chosen as the number of SOIs in the smaller category, e.g., for the hypothetical data in Table 4, we simply sample 10 SOIs, which is the total number of RC-ND from the 101,000 SOIs as RC-MD, and use the 10 samples from RC-MD together with all of those in RC-ND to compute the metrics. We define the error to be

$$\text{Error} = \sum_{i \in metrics} |\text{i-mean(i\_history)}|$$

where i is one of the metrics, including: RC-MD-recall, RC-MD-precision, RC-ND-recall, RC-ND-precision, M-MD-recall, M-MD-precision, M-ND-recall, or M-ND-precision. $|\cdot|$ is the absolute value, and $i\_history$ is

the history of the metrics, and *mean()* is its expected value. Similar to Algorithm 1, we define the stopping criteria of the iteration as 0.01, as the maximum possible value of each metric is 1, or 100%. When Error $\leq$ 0.01 or the iteration exceeds a predefined number (here, we set it to be 10,000), the iterations stop. If the number of iterations is smaller than the pre-defined limit, we use the last computed metrics; if the limit of iterations is reached, the mean of the history is used.

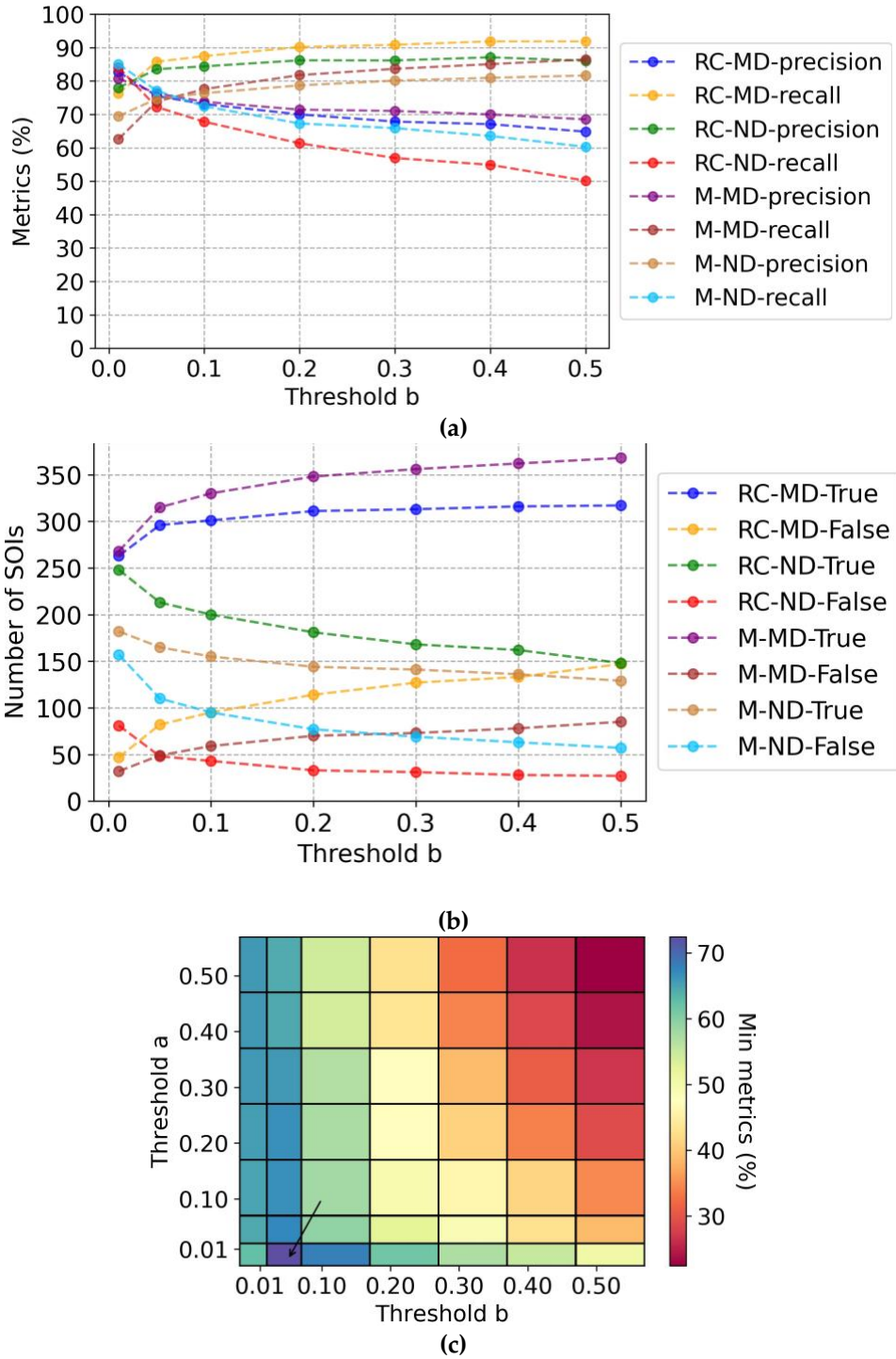
It is worth noting that even though similar sampling methods are utilized in both Section 2.3.2 and this section, the reasons for choosing to use them are fundamentally different. For the information fusion algorithm in Section 2.3.2, the sampling method is used to reduce the computation time that would be needed for the conventional method as much as possible. However, the goal for introducing the sampling method in this section is to overcome the issue caused by the imbalanced dataset. As the metrics are only computed one time after implementing the technique on the entire dataset, and, furthermore it will not be computed when the technique is actually implemented to classify the SOIs, the computation time is not of concern here.

3.3.2 Detailed procedure for threshold tuning

For tuning the thresholds, we begin by proposing candidate values of T_a as 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5 and T_b as 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5. The technique is run with each combination of these candidate values. Thus, with these candidates we perform 49 trials of the technique. For each trial, we classify all buildings in validation dataset 1 based on their respective SOI. The metrics are generated by the method introduced in section 3.3.1, and for each run 8 metrics are generated.

To illustrate the tuning, we plot a portion of the results for when T_a is fixed as 0.01 and T_b cycles through all its candidate values. In Figure 10(a), we plot the resulting metrics as a function of the values of T_b. As mentioned earlier, there are 8 metrics in total. As shown in the plot, these metrics drastically change with T_b. For example, RC-ND-recall drops from 84.07% to 50.17%, as T_b changes from 0.01 to 0.5. Meanwhile, M-MD-recall increases from 62.62% to 86.45%. To understand the reason behind why some metrics are larger when T_b is large, while other metrics have the opposite behavior, we plot the number of SOIs being predicted in Fig. 10(b). As shown in the plot, all the values related to MD are increasing as T_b increases, and all the terms related to ND are decreasing. This trend occurs because when T_b is getting larger, or $1 - T_b$ is getting smaller, the filters allowing images to be classified as MD and ND are getting less strict, allowing more images to be passed to the next stage of the process as MD and ND images. Because the MD images tend to dominate in the information fusion process to classify the damage state of the SOI as MD, this outcome results in an increasing number of SOIs being classified as MD, or equivalently, a reduced number of SOIs that are classified as ND. The consequence of this behavior is the significant changes in the values of the metrics.

To select an optimal combination of thresholds, we simply use the minimum metrics in each case as the indicator. For instance, we use 64.95% to identify the case in which T_a is 0.1 and T_b is 0.01. From all the indicators, we select the highest, which represents the thresholds that yield the approach with the best performance. The results are shown in Fig. 10(c). In this figure, we show the minimum metrics for each combination of different values. The most appropriate one is selected to be T_a as 0.01 and T_b as 0.05 corresponding to an indicator of 72.45%, which is pointed out by the arrow in Fig. 10(c).



564 **Figure 10. Thresholds tuning results: (a) results of metrics, (b) results of number of SOIs, (c) overall**
565 **results**

566

567 **3.4 Validation results**

568 **3.4.1 Validation results on an SOI example**

569 In this section, we demonstrate the technique using an SOI. Several sample images from the SOI are shown
570 in Fig. 11 [24]. The SOI is from the Taiwan dataset, and contains 129 images. First, we walk through the

workflow to demonstrate how the RC list is updated for one image. Updating the M list will be similar. The details relating to step 1a in Figure 4 are provided in Algorithm 2. The input is an image "$\boldsymbol{m}$" and the output is the updated RC list, RC_list. To begin, we obtain p_RC by applying the RC classifier and p_ND by applying the ND classifier on $\boldsymbol{m}$, respectively. Then, the following decision is made: p_RC or p_ND is larger than $1 - threshold\_a$, the RC list is updated by appending the image $\boldsymbol{m}$ to the RC list; otherwise, the RC list stays the same. This terminates the process of step 1a in Figure 4 for $\boldsymbol{m}$.

By going through this process, the algorithm avoids the extreme case of having an image classified as both RC and ND at the same time (a high probability from both the RC classifier and the ND classifier is an indication of misclassification). Take the forth sample image in Figure 11 as $p\_RC = 0.8831$, $p\_ND = 0.9778$, $1 - threshold\_a = 0.99$, this means $p\_RC < 1 - threshold\_a$ and $p\_ND < 1 - threshold\_a$, thus, this image will not be put in, thus, this image will not be appended to the RC list. For the second sample image in Figure 11, as $p\_ND > 1 - threshold\_a$, then, the image will be appended to the RC list.

Algorithm 2. Updating of the RC list with one image

Algorithm 2:	
	Input: m # image **Output**: RC_list # the updated RC list
1	p_RC = RC_classifier(m)
2	p_ND = ND_classifier(m)
3	**if** p_RC > 1-threshold_a **or** p_ND > 1-threshold_a
4	append m to RC_list

For the SOI example, the ground truth is MD for RC and MD for M. The resulting probabilities are 0.9999 for RC and 0.9999 for M. Thus, the prediction is MD for both RC and M, which agrees with the ground truth. Notice that our design approach, with a separate damage state classifier and category classifier, increases the robustness of the method to correctly predict the damage state for each image. Evidence of this robustness is found here with the forth sample image where the RC classifier assigns 0.8331 to the image indicating the image can be classified as RC-MD while the damage state classifier assigns 0.1365 indicating low damage state as the true condition of the image. The time required to generate this decision is 9.27 seconds, including the time for both image classification and information fusion.

Figure 11. Sample images from a Taiwan mission SOI including testing results [24]

3.4.2 Validation results on the validation datasets

As demonstrated in Section 3.3.2, the technique achieves good performance using the pre-determined thresholds with all metrics being above 72%. The detailed results corresponding to validation dataset 1 are shown in Table 5(a). We provide the results as a confusion matrix grouped by the buildings surveyed

during each event, and also provide the results over all events. For this work and similar implementations of classification methods, recall plays a more important role than precision. In particular, in this application it is critical to successfully identify as many buildings as possible in each class, without neglecting classes that happen to contain a smaller number of buildings [14]. Thus, we only calculate and show the recall for each category. Recall values are calculated directly, since the imbalance in the dataset only significantly affects the precision values, as shown in Section 3.3.1.

In general, the performance is good. The results do vary somewhat with the specific event. In most cases, the performance is above or close to the overall metrics, for instance, the Bingöl, Ecuador, and Haiti datasets. However, in a couple of cases the performance is noticeably lower, including the M-MD for the Bingöl dataset, and M-MD of the Taiwan dataset, etc. We believe this outcome is mainly because the misclassification of images occurs more frequently in certain datasets. A possible solution is to collect more images containing a variety of damage conditions and architectural styles to add to the overall dataset. The variety of the training dataset is generally a strong indicator of the robustness of the classifiers trained. Also, adding more SOIs to the datasets will also reduce the likelihood of outliers in the metrics. For instance, the M-ND-recall of the Bingöl dataset is 100%. Here the M-ND-Total is only 6, and thus the high recall value does not necessarily reflect the technique. It is reasonable to expect that datasets containing more SOIs in M-ND, the recall will drop to a level closer to the overall performance.

The results for validation dataset 2 are shown in Table 5(b). Here it is clear that the approach also achieves good performance, especially considering the technique has not seen any images in validation dataset 2 before this test. Note that M-ND-recall is higher here than in the results for validation dataset 1. The reason for this outcome is possibly the limited number of SOIs. Increasing the number of SOIs in validation dataset 2 can lead to a more representative result.

Table 5(a). Results for validation dataset 1

Ground truth\prediction		MD	ND	Total	Recall
Bingöl	RC	MD	RC-MD-True: 27	RC-ND-False: 9	RC-MD-Total: 36	75.00%
		ND	RC-MD-False: 3	RC-ND-True: 16	RC-ND-Total: 19	84.21%
	M	MD	M-MD-True: 30	M-ND-False: 19	M-MD-Total: 49	61.22%
		ND	M-MD-False: 0	M-ND-True: 6	M-ND-Total: 6	100%
Ecuador	RC	MD	RC-MD-True: 102	RC-ND-False: 16	RC-MD-Total: 118	86.44%
		ND	RC-MD-False: 17	RC-ND-True: 36	RC-ND-Total: 53	67.92%
	M	MD	M-MD-True: 107	M-ND-False: 26	M-MD-Total: 133	80.45%
		ND	M-MD-False: 13	M-ND-True: 25	M-ND-Total: 38	65.79%
Haiti	RC	MD	RC-MD-True: 69	RC-ND-False: 7	RC-MD-Total: 76	90.79%
		ND	RC-MD-False: 18	RC-ND-True: 35	RC-ND-Total: 53	66.04%
	M	MD	M-MD-True: 72	M-ND-False: 19	M-MD-Total: 91	79.12%
		ND	M-MD-False: 9	M-ND-True: 29	M-ND-Total: 38	76.32%
Nepal	RC	MD	RC-MD-True: 73	RC-ND-False: 10	RC-MD-Total: 83	87.95%
		ND	RC-MD-False: 25	RC-ND-True: 57	RC-ND-Total: 82	69.51%
	M	MD	M-MD-True: 86	M-ND-False: 33	M-MD-Total: 119	72.27%
		ND	M-MD-False: 12	M-ND-True: 34	M-ND-Total: 46	73.91%
Taiwan	RC	MD	RC-MD-True: 26	RC-ND-False: 6	RC-MD-Total: 32	81.25%
		ND	RC-MD-False: 18	RC-ND-True: 69	RC-ND-Total: 87	79.31%
	M	MD	M-MD-True: 20	M-ND-False: 13	M-MD-Total: 33	60.61%

		ND	M-MD-False: 15	M-ND-True: 71	M-ND-Total: 86	82.56%
Total	RC	MD	RC-MD-True: 297	RC-ND-False: 48	RC-MD-Total: 345	86.09%
		ND	RC-MD-False: 81	RC-ND-True: 213	RC-ND-Total: 294	72.45%
	M	MD	M-MD-True: 315	M-ND-False: 110	M-MD-Total: 425	74.12%
		ND	M-MD-False: 49	M-ND-True: 165	M-ND-Total: 214	77.10%

Table 5(b). Results for validation dataset 2

Ground truth\prediction		MD	ND	Total	Recall
Mexico City	RC	MD	RC-MD-True: 26	RC-ND-False: 7	RC-MD-Total: 33	78.79%
		ND	RC-MD-False: 16	RC-ND-True: 32	RC-ND-Total: 48	66.67%
	M	MD	M-MD-True: 30	M-ND-False: 16	M-MD-Total: 46	65.22%
		ND	M-MD-False: 6	M-ND-True: 29	M-ND-Total: 35	82.86%

3.3.3 Influence of corrosion and other types of nonstructural damage

As we have mentioned before, the data collection procedures do play a major role in the success of this technique. For instance, note that some of the damage visible in the images collected during the reconnaissance missions already existed prior to the seismic event. Additionally, some of the damage to concrete components was to nonstructural components. The presence of these images does bias the performance of the technique and can yield false predictions. To explore these as possible reasons for false predictions, we consider the influence of these images on the overall results. We manually remove two types of images, those with: pre-existing damage, which is evident by the level of corrosion visible, and nonstructural damage, for instance to components such as balconies or parapets.

During a reconnaissance mission, such evidence of distress in the building does not participate in the decision process because the human engineer is able to disregard this information. However, the computer is not yet able to distinguish between such cases. The design of new classifiers to filter out such data would be a viable option, however, we first must understand the role these images play in the overall success of the technique. We noticed that these situations are especially evident in the Ecuador dataset [10]. Thus, to examine the influence of these images, we manually remove such images (those with corrosion, indicating pre-existing damage; and with major nonstructural damage) from the Ecuador dataset. Then we re-run the technique on the reduced dataset and compare the results.

Several sample images that were removed because they contain corrosion are shown in Fig. 12. In total, 16 images from 5 SOIs are removed to examine their influence on the technique. As shown in the figure, they would be classified as MD images with varying probabilities. However, when the SOIs include these images, the predictions are likely to be MD, which does not match the ground truth and thus will reduce the associated metrics. The results of the Ecuador dataset without these images are shown in Table 6. It is obvious that the performance in RC-ND and M-ND improves, while RC-MD stay the same and a decrease happens in M-MD. One additional SOI is falsely evaluated as compared with the original predictions shown in Table 5(a). It is likely, with the tuned thresholds, that the removed images contribute to the MD prediction in this particular SOI. The improvement in the metrics agrees with the number of SOIs being altered. Because images with pre-existing damage are in 4 SOIs, RC-ND-True and M-ND-True increase by 2 and 2, respectively. Removing these images from the SOI, or not collecting them in the first place, would improve the results of the technique. This observation will be important for improving the data collection procedures.

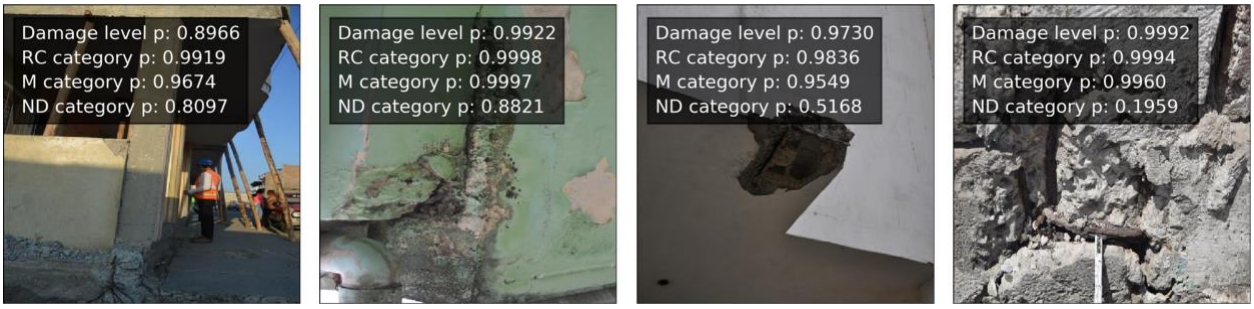


Figure 12. Sample images with corrosion as evidence of pre-existing damage from the Ecuador dataset [10]

Table 6. Results for the Ecuador dataset without corrosion images

Ground truth\prediction		MD	ND	Total	Recall	Original recall
Ecuador w/ corrosion	RC	MD	RC-MD-True: 102	RC-ND-False: 16	RC-MD-Total: 118	86.44%	86.44%
		ND	RC-MD-False: 15	RC-ND-True: 38	RC-ND-Total: 53	71.70%	67.92%
	M	MD	M-MD-True: 106	M-ND-False: 27	M-MD-Total: 133	79.70%	80.45%
		ND	M-MD-False: 11	M-ND-True: 27	M-ND-Total: 38	71.05%	65.79%

A similar situation is considered for images with purely nonstructural damage. Several sample images of this case are shown in Fig. 13. In total, 49 images from 10 SOIs are removed and the predictions are repeated. The results for the Ecuador dataset without these images are shown in Table 7. Two categories see improved metrics, raising the number of true predictions, while RC-MD stay the same and M-MD-True decrease by 1 likely due to the same reason in the corrosion case. Based on the sample here, it is clear that the data collection process does bias the results of the technique. These images, containing corroded components with pre-existing damage and damage to nonstructural components, contribute to the number of false predictions made by the technique. This sample case motivates the need for either new classifiers that can automatically filter out these images, or guidelines that discourage teams in the field from taking such images. The performance of such techniques will be improved with awareness about the overall process.

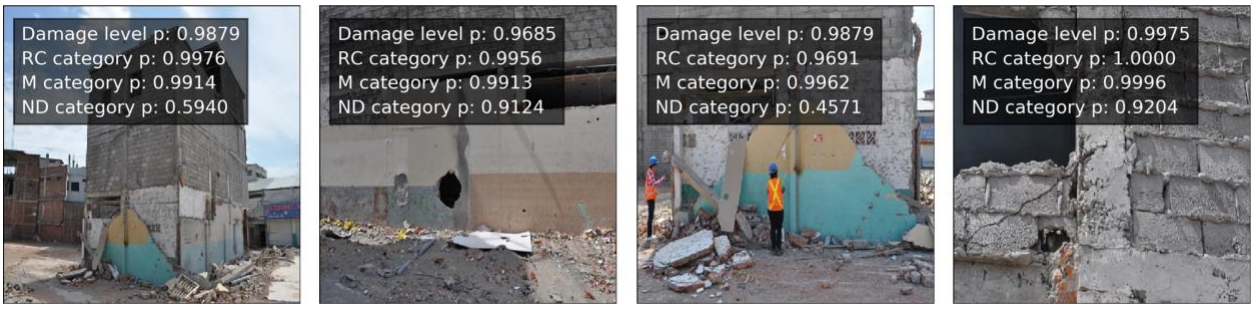


Figure 13. Sample images with nonstructural damage from the Ecuador dataset [10]

Table 7. Results for the Ecuador dataset without images of nonstructural damage

Ground truth\prediction		MD	ND	Total	Recall	Original recall
Ecuador w/ nonstructural damage	RC	MD	RC-MD-True: 102	RC-ND-False: 16	RC-MD-Total: 118	86.44%	86.44%
		ND	RC-MD-False: 11	RC-ND-True: 42	RC-ND-Total: 53	79.25%	67.92%
	M	MD	M-MD-True: 106	M-ND-False: 27	M-MD-Total: 133	79.70%	80.45%
		ND	M-MD-False: 9	M-ND-True: 29	M-ND-Total: 38	76.32%	65.79%

4. Conclusion

Post-event reconnaissance teams collect perishable data that can be studied, leading to research and new knowledge about the performance of our infrastructure. In such missions, an important task is to develop methods to classify the damage state of buildings after an event. For instance, the Hassan index is an example of a convenient and expeditious method that can help with regional vulnerability models for the built environment. However, the process of collecting the data can be both exhausting and dangerous for the reconnaissance teams, and efforts to increase the reuse of those data collected under such difficult conditions should be energetically pursued. Automating the steps will facilitate an increase in the amount of data collected and used to develop new knowledge, while also building greater confidence in vulnerability models developed from the reuse of such data.

The automated technique developed and validated herein is intended to classify the overall damage state of a building based on a set of images collected in the field. We target damage to both the RC structural components and masonry components of a building, based on a set of tasks that a reconnaissance team has actually performed in the field. For RC damage or M damage, the technique will process a set of input images and generate a list of probabilities, each corresponding to the probability that the corresponding image is either MD or ND. Then, we apply an information fusion algorithm to each probability list and yield the fused probability which is used to predict the overall damage state of the building as either MD or ND. The technique is demonstrated and validated with real world datasets from past earthquakes in different locations around the world. After building the classifiers and tuning the thresholds, the technique is able to predict the damage state of an SOI in less than one minute. Thus, this technique provides the ability to analyze a vast volume of reconnaissance images to predict the damage state of buildings. This technique also has potential to support the use of drones or robots for field data collection, which in turn, reduces life-threatening situations for reconnaissance teams. We also anticipate that it will promote the collection and reuse of more reconnaissance data to inform building design procedures.

The collection of the data collection can influence the outcomes of such automated techniques, and thus there is value in considering best practices for collecting data that will yield robust results from this and possibly other techniques. First, reconnaissance teams are encouraged to collect more images from each target building. As the number of images gathered from a given building grows, better damage classifications can be made. Second, the set of images should cover as many building components as possible. This technique is meant to anticipate an image is taken for every visible building region, whether or not those components contain damaged or undamaged building components, structural or non-

structural components, relevant or irrelevant to the damage state of buildings, etc. The technique can make more robust predictions when buildings are sufficiently covered by the reconnaissance images. Third, images should not be taken from so close that the context of the scene is not clear. A close in view of a crack can be useful, but does not provide information about whether the damage is to structural or nonstructural components, nor does it provide any sort of scale information. And finally, as mentioned earlier, when damage is intended to be captured with an image, the damage should consume a reasonable portion of the image area (we estimate 30%).

Limitations of the technique do exist, offering challenging directions for interested researchers. It is important to keep in mind that the technique can only base the outcome on the images that were collected. Note that when the images collected are not sufficient to cover every component of the building, the prediction yielded by this technique may not reflect the true state of the building. And as the image sets become larger, for instance with drones or other methods of gathering large volumes of images, more challenges do exist.

Author Contributions: Conceptualization, Xiaoyu Liu, Lissette Iturburu, Shirley J. Dyke and Julio Ramirez; Methodology, Xiaoyu Liu, Ali Lenjani and Xin Zhang; Validation, Xiaoyu Liu, Lissette Iturburu, Shirley J. Dyke and Xin Zhang; Writing and Paper Preparation, Xiaoyu Liu, Lissette Iturburu, and Shirley J. Dyke; Supervision, Shirley J. Dyke and Julio Ramirez.

Funding: We would like to acknowledge support from National Science Foundation under Grant No. NSF-OAC-1835473.

References:

1. Pujol, S., Laughery, L., Puranam, A., Hesam, P., Cheng, L. H., Lund, A., & Irfanoglu, A. (2020). Evaluation of seismic vulnerability indices for low-rise reinforced concrete buildings including data from the 6 February 2016 Taiwan Earthquake. *Journal of Disaster Research*, *15*(1), 9-19.

2. Altınçay, H. (2005). On naive Bayesian fusion of dependent classifiers. *Pattern Recognition Letters*, *26*(15), 2463-2473.

3. Laughery, L. A., Puranam, A. Y., Segura Jr, C. L., & Behrouzi, A. A. (2020). The Institute's Team for Damage Investigations. Concrete International, 32.

4. Villalobos E, Sim C, Smith-Pardo JP, Rojas P, Pujol S, Kreger ME. The 16 April 2016 Ecuador Earthquake Damage Assessment Survey. Earthquake Spectra. 2018;34(3):1201-1217. doi:10.1193/060217EQS106M

5. Catlin, A. C., HewaNadungodage, C., Pujol, S., Laughery, L., Sim, C., Puranam, A., & Bejarano, A. (2018). A cyberplatform for sharing scientific research data at DataCenterHub. *Computing in Science & Engineering*, *20*(3), 49-70.

6. Rathje, E. M., Dawson, C., Padgett, J. E., Pinelli, J. P., Stanzione, D., Arduino, P., ... & Mosqueda, G. (2020). Enhancing Research in Natural Hazards Engineering through the DesignSafe Cyberinfrastructure. *Frontiers in Built Environment*, *6*, 213.

7. Hassan, A. F., & Sozen, M. A. (1997). Seismic vulnerability assessment of low-rise buildings in regions with infrequent earthquakes. *ACI Structural Journal*, *94*(1), 31-39.

8. Brzev, S., Pandey, B., Maharjan, D. K., & Ventura, C. (2017). Seismic vulnerability assessment of low-rise reinforced concrete buildings affected by the 2015 Gorkha, Nepal, earthquake. *Earthquake Spectra*, *33*(1_suppl), 275-298.

9. Johnson, M. G., & Fick, D. R. (2018). Generalized Trends of Severe Damage Observed from Building Surveys of Seven Different Earthquakes. *11th U.S. National Conference on Earthquake Engineering, 25-29 June, 2018,* Los Angeles, California, USA.

10. Chungwook Sim, Enrique Villalobos, Jhon Paul Smith, Pedro Rojas, Aishwarya Y Puranam, Lucas Laughery, Santiago Pujol (2016), "2016 Ecuador Earthquake," https://datacenterhub.org/deedsdv/publications/view/535.
11. Jahanshahi, M. R., Masri, S. F., Padgett, C. W., & Sukhatme, G. S. (2013). An innovative methodology for detection and quantification of cracks through incorporation of depth perception. *Machine vision and applications*, *24*(2), 227-241.
12. Kim, H., Ahn, E., Shin, M., & Sim, S. H. (2019). Crack and noncrack classification from concrete surface images using machine learning. *Structural Health Monitoring*, *18*(3), 725-738.
13. Hoskere, V., Narazaki, Y., Hoang, T., & Spencer Jr, B. (2018). Vision-based structural inspection using multiscale deep convolutional neural networks. *arXiv preprint arXiv:1805.01055*.
14. Yeum, C. M., Dyke, S. J., Benes, B., Hacker, T., Ramirez, J., Lund, A., & Pujol, S. (2019). Postevent reconnaissance image documentation using automated classification. *Journal of Performance of Constructed Facilities*, *33*(1), 04018103.
15. Xu, J. Z., Lu, W., Li, Z., Khaitan, P., & Zaytseva, V. (2019). Building damage detection in satellite imagery using convolutional neural networks. *arXiv preprint arXiv:1910.06444*.
16. Gueguen, L., & Hamid, R. (2015). Large-scale damage detection using satellite imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1321-1328).
17. Gupta, R., Hosfelt, R., Sajeev, S., Patel, N., Goodman, B., Doshi, J., ... & Gaston, M. (2019). xbd: A dataset for assessing building damage from satellite imagery. *arXiv preprint arXiv:1911.09296*.
18. Lenjani, A., Dyke, S. J., Bilionis, I., Yeum, C. M., Kamiya, K., Choi, J., ... & Chowdhury, A. G. (2020). Towards fully automated post-event data collection and analysis: Pre-event and post-event information fusion. *Engineering Structures*, *208*, 109884.
19. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems* 25 (2012): 1097-1105.
20. Hirzel, A., & Guisan, A. (2002). Which is the optimal sampling strategy for habitat suitability modelling. *Ecological modelling*, *157*(2-3), 331-341.
21. Chungwook Sim, Nick Skok, Ayhan Irfanoglu, Santiago Pujol, Mete Sozen, Cheng Song (2016), "Database of low-rise reinforced concrete buildings with earthquake damage," https://datacenterhub.org/deedsdv/publications/view/454.
22. Prateek Shah, Santiago Pujol, Aishwarya Puranam (2016), "Database on Performance of High-Rise Reinforced Concrete Buildings in the 2015 Nepal Earthquake," https://datacenterhub.org/deedsdv/publications/view/409.
23. Prateek Shah, Santiago Pujol, Aishwarya Puranam, Lucas Laughery (2016), "2015 Nepal Earthquake Building Performance Database," https://datacenterhub.org/deedsdv/publications/view/537.
24. NCREE, Purdue University (2016), "2016 Taiwan (Meinong) Earthquake," https://datacenterhub.org/deedsdv/publications/view/534.
25. Purdue University (2018), "Buildings Surveyed after the 2017 Mexico City Earthquakes," https://datacenterhub.org/deedsdv/publications/view/536.
26. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
27. Ting K.M. (2011) Precision and Recall. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8_652
28. Davis, J., & Goadrich, M. (2006, June). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning* (pp. 233-240).