

DISASTER INITIAL RESPONSES MINING DAMAGES USING FEATURE EXTRACTION AND BAYESIAN OPTIMIZED SUPPORT VECTOR CLASSIFIERS

Yasuno Takato, Amakata Masazumi, Fujii Junichiro, and Shimamoto Yuri

Research Institute for Infrastructure Paradigm Shift (RIIPS),
Yachiyo Engineering, Co. Ltd., Tokyo, Japan

<mailto:{tk-yasuno,amakata,jn-fujii,yr-shimamoto}@yachiyo-eng.co.jp>

ABSTRACT

Whenever a natural disaster occurs, it is important to quickly evaluate the damage status in high-priority locations. Frequently, owing to the restrictions imposed by the availability of disaster management resources, spatial information is predicted where the infrastructure manager makes an initial response. It is critical that an initial response be effective to mitigate social losses. In recent years, Japan has experienced several great earthquakes with magnitudes of around 6, most notably the Great East Japan earthquake of March 2011 (M9), as well as those striking Kumamoto (April 2016 (M7)), Osaka (June 2018 (M6.1)), and Hokkaido (September 2018 (M6.7)). These huge earthquakes occur not only in Japan but around the world, with an earthquake and tsunami striking Indonesia as recently as October 2018. The initial response to future earthquakes is an important issue related to knowledge of natural disasters and to predict the degree of damage to infrastructure using multi-mode usable data sources. In Japan, approximately 5 million CCTV cameras are installed. The Ministry of Land, Infrastructure and Transportation uses 23,000 of these cameras to monitor the infrastructure in each region. This paper proposes a feature extraction damage classification model using disaster images with five classes of damage after the occurrence of a huge earthquake. We present a support vector damage classifier for which the inputs are the extracted damage features, such as tsunami, bridge collapses, and road damage leading to a risk of accidents, initial smoke and fire, and non-disaster damage. The total number of images is 1,117, which we collected from relevant websites that allow us to download records of huge earthquake damage that has occurred worldwide. Using ten pre-trained architectures, we have extracted the damage features and constructed a support vector classification model with a radial basis function, for which the hyper parameters optimize the results to minimize the loss function value with an accuracy of 97.50%, based on the DenseNet-201. This would provide us with further opportunities for disaster data mining and localized detection.

KEYWORDS

Disaster Response, Damage Mining, Feature Extraction, Support Vector classifier, Bayesian Optimization

1. INTRODUCTION

This section reviews the related papers and works related to natural disaster management and machine learning for disaster data resources. The authors highlight earthquake disasters and the mining of five classes of earthquake damage data sets.

1.1. Literatures related to disaster management

Manzhu et al. [1] reviewed the major big data sources and the associated achievements in disaster management phases to monitor and detect natural hazards, and to mitigate disaster damage, as well as the recovery and reconstruction processes. This paper focuses on the urgent response phase after an earthquake in which damage is monitored and detected to make the

decisions needed to address initial rapid actions regarding high priority infrastructures such as roads, intersections, bridges, river gates, and urban and rural areas. During 2014–2016, a variety of data sources could be observed in articles, when the topic of big data was popular in disaster management. These data sources are as follows: satellite, social media, crowd sourcing, sensor web and IoT, mobile GPS, simulation, unmanned aerial vehicles (UAV), Light Detection and Ranging (LiDAR), and spatial data. Among these digital data sources, satellite imagery [2][3] and social media [4][5] data serve as the most popular data for disaster management.

However, a satellite used for remote sensing always moves slowly, such that there is a delay between the times at which data is acquired. The resulting series of photographs is thus not useful for recognizing earthquake features. Therefore, disaster detection can be done based on social media: Twitter is used as a source of text mining, and spatial temporal analysis. However, social media users cannot always monitor disaster damage accurately. Also, users tend to be agitated and fearful for their safety after the occurrence of a huge earthquake. Messages sometimes lack essential details owing to noise and may start false rumors. Therefore, this study focuses on closed-circuit television (CCTV) data sources for monitoring damage to critical infrastructure in order to make decisions related to high-priority responses.

1.2. Works related to disaster images

CCTV cameras are being set up around real-world places such as houses for crime prevention, industrial processes to detect anomalies, banks for security, shopping stores, schools, rail stations for safety, traffic monitoring, sports events, and offices to monitor employees. CCTV, also known as video surveillance, involves the use of video cameras to transmit a signal to a specific place with a set of monitors. The first CCTV system was installed by Siemens AG at Test Stand VII in Nazi Germany in 1942 for observing rockets [6]. The earliest video surveillance systems involved constant monitoring because there was no means of recording and storing the information. A modern machine vision system enables the constant monitoring of infrastructures and determine whether earthquake disaster damage has occurred, with several cameras recording simultaneously, with features such as time lapse and motion-activated recording. The resulting savings in time and cost had resulted in an increase in the use of CCTV. Recently, CCTV technology has been enhanced with a shift towards Internet-based products. There were an estimated 350 million surveillance cameras worldwide as of 2016, compared with 160 million in 2012 [7]. Sixty-five percent of the CCTV cameras installed around the world are in Asia. There are currently five million CCTV cameras installed in Japan. This enables the creation of systems to support decisions related to immediate initial responses with respect to high-priority locations if a large earthquake were to occur.

As a low level approach, both the input and output are images, with several techniques for analyzing changes that are detected after a disaster. Supanee et al. [8] presented a building detection process that could detect damage to both small and large buildings with 75% accuracy. That method was applied to obtain data from the 2004 tsunami that struck the coast of Thailand. However, only one class of building was analyzed and the satellite images, which were limited to the coastal area, had a 1-meter resolution. Ranga et al. [9] presented a probabilistic detection system that provides information regarding changes in an area and which minimizes the post-detection threshold procedure often required in traditional change-detection algorithms. However, their method was intended for land use detection such as growth, loss, and no change. Maeda et al. [10] proposed a method that uses CCTV images with reduced background noise and subtracts the change between the ex ante and ex post when an earthquake occurs. A low-level application contains simple algorithms that may be unstable and not highly accurate. This method has certain disadvantages, making it important to optimize the thresholding parameter and the balancing trade-off between the damaged signal and the background noise detection.

On the other hand, as a high-level approach of a level equal to human vision, Kataoka et al.[11] surveyed 1,600 studies of computer vision and devised the concept of semantic change detection regardless of whether a building is damaged or not after an earthquake. However, that conceptual method requires hundreds of thousands or even millions of disaster-image datasets; further, there is no proof of the concept of semantic segmentation focusing on buildings and change detection regarding damaged buildings. Also, we cannot obtain a middle-level application to classify earthquake disaster images, owing to the lack of datasets addressing the instant at which an earthquake occurs. The present study addressed middle-level image analysis where the input is CCTV images in the order of thousands, and the output is disaster-class labels for decision support with respect to the initial response to be applied to high-priority locations.



Figure 1. A thumbnail of QuakeV datasets randomly chosen 100 images

1.3. Mining of earthquake damage datasets

With regard to disaster image datasets, NOAA [12] offers a natural-hazard image database with 1,163 photo images of 67 earthquake events that have occurred in the 100 years from 1915 to 2015. It enables the viewing of a gallery of images from each earthquake event. The database includes earthquake disaster images from around the world, including the USA, Mexico, Guatemala, Colombia, Nicaragua, Peru, Chile, Haiti, Ecuador, Russia, Iran, Turkey, Pakistan, Algeria, Romania, Italy, Papua New Guinea, Australia, New Zealand, Samoa, China, Indonesia, Taiwan, and Japan. However, the viewpoints differ, such as satellite images with low resolution, airplane downward views, views of damage captured outdoors and indoors, as well as of destroyed homes. In short, their focuses and viewpoints are wide ranging, while the privately captured historical photos were taken without any unified rules. For each event, there are only a

limited number of images at the half 30 earthquake events on that database. Almost all of the images were recorded after the earthquakes had occurred, while it took more than one week for academic surveys to obtain the relevant data. We attempted to collect open-source web pages from which earthquake damage images could be downloaded.

This paper highlights four earthquake disaster features such as tsunami damage, bridge collapses, road damage giving rise to accident risks, and initial smoke and fire. The total number of earthquake disaster feature datasets collected by the authors was 1,117. This paper focuses on earthquake images, which we used to build a dataset named QuakeV. Figure 1 shows thumbnails of the earthquake damage image dataset for which the validation data are randomly chosen with 100 images of each of the above classes.

2. DISASTER DAMAGE CLASSIFICATION

This section proposes a means of disaster damage classification using image data sets. The disaster damage data are extracted into features using a pre-trained deep network. The features output from a concatenated layer are used as an input to the support vector classification model with more than two classes. To minimize the prediction error, the authors applied a Bayesian optimization method for hyper parameters to enable a warm start based on the result of previous training runs.

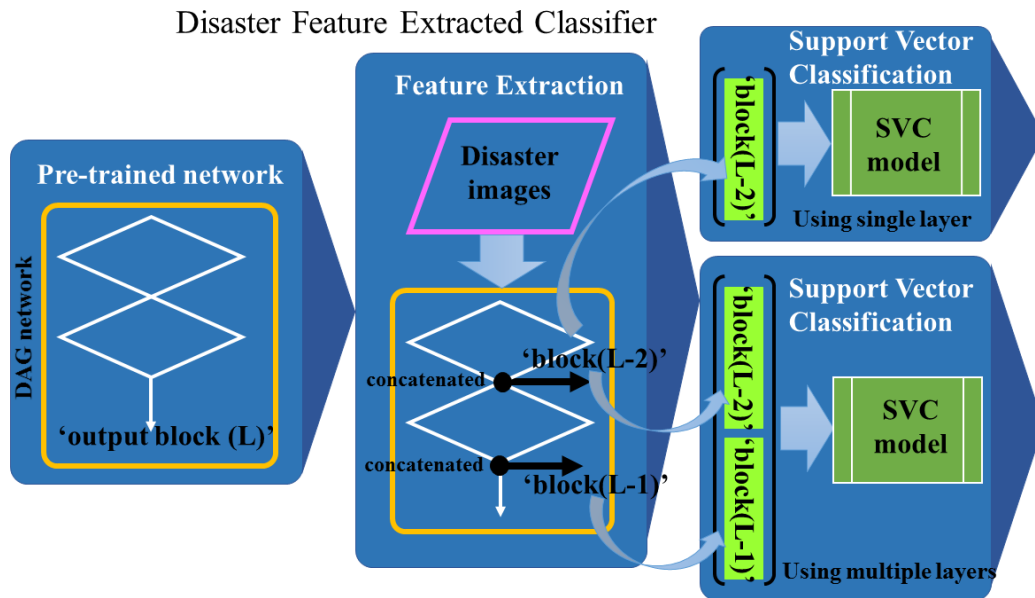


Figure 2. Disaster features are extracted using pre-trained network, whose features are used as an input for classifier incorporating a single layer, and multiple layers at concatenated points.

2.1. Feature extraction using pre-trained networks

Feature extraction is commonly used in machine-learning applications. We can consider a pre-trained network and use it as an input feature to learn a classification task. Image classification using feature extraction is usually much faster and less demanding of computing resources than a transfer learning process involving the tuning of weights and biases in the deep layers. We can rapidly construct a classifier for a new task using an extracted feature at the concatenated layer as a training column vector [13]. This paper proposes a support vector classification model using a single layer obtained from concatenated feature extraction. Furthermore, we provide a classification model using multiple layers from several extracted features at significant concatenation points. We are able to load the pre-trained network such as AlexNet [14],

GoogleNet [15], VGG16 and VGG19 [16], ResNet18, ResNet50 and ResNet101 [17], Inception v3 [18], Inception-ResNet-v2 [19], and DenseNet-201 [20]. The features output from a concatenated layer are used as the input to the support vector classification model with several disaster damage classes.

2.2. Support vector classifier with multiple damage classes

Assume that there are multiple disaster feature classes for a support vector classification model. In multiple classification models with more than two classes, we use a voting strategy [21][22]: each binary classification is considered to be voting where votes can be for all data points, a point is designated to be in a class with the maximum number of votes. The LIBSVM rapid library implements a one-to-one approach for multi-class classification. Many other methods are available for multi-class support vector classification [21][22]. The present study used a kernel with a radial basis function with a gamma parameter. For the image classification, the number of extracted features is always a large number. In the present study, there is a maximum number of instances for which the number of disaster-feature column vectors is 400,000. For the earthquake damage data set that we examined, the support vector classification method confirmed the preferred advantages of speed and accuracy compared with other classification methods such as k-nearest neighbor, decision tree, random forest, and boosting method. A support vector classifier was constructed using extracted disaster features based on the above pre-trained networks.

2.3. Hyper parameter optimization to minimize error

Automated machine-learning methods use Bayesian optimization to tune the hyper parameters of a machine-learning pipeline. We can implement libraries such as Auto-WEKA, auto-sklearn, Auto-Model, and so forth [23][24]. Grid search and randomized search do not use the information produced by previous training runs, which is disadvantageous to Bayesian-based optimization methods. Bayesian-based optimization methods leverage the information gained from previous training runs to determine the hyper-parameter values for the next training run and to navigate through the hyper parameter space in a smarter way. The basic idea of warm start is to use the information gained from previous training runs to identify better starting points for the next training run. When we are building machine learning models, a loss function helps us to minimize the prediction error during the training phase.

The authors propose a Bayesian optimization method for which the objective function is a loss function from five-fold cross validation to minimize the classification error using a support-vector classifier for which the input is extracted features based on a pre-trained network. As the standard setting, we propose that the support vector classification model be based on a radial-basis kernel function with two hyper parameters such as box constraint C and kernel scale gamma [21]. In the present study, the authors attempted to identify those hyper parameters that would minimize the cross-validation loss function in thirty iterations using the Bayesian optimization method.

3. APPLIED RESULTS

This section demonstrates case studies applied to earthquake damage data sets divided into five classes. Using ten pre-trained network architectures, the image data are extracted from a few or more concatenated layers next to the final output. The extracted features are imported as an input to the support-vector classification model and compared in terms of the accuracy for the pre-trained networks. Among the comparison studies, the most accurate classifier was obtained with the hyper-parameter optimization method.

3.1. Earthquake damage data sets

We attempted to collect open-source web pages from which earthquake damage images could be downloaded. The large earthquake disaster images were primarily collected from large Japanese earthquakes such as the Great Hanshin Earthquake (January 17, 1995) and the Great East Japan Earthquake (March 11, 2011). However, the areas were not limited to Japan, with images being acquired from around the world, provided they were usable. The present study highlights four earthquake disaster features such as tsunami damage, bridge collapses, road damage giving rise to a risk of accidents, and initial smoke and fire.

Table 1 shows that the number of each type of disaster image is 221, 222, 210, and 210, respectively. The number of non-disaster images is 254. The total number of earthquake disaster feature datasets is 1,117 with a size of 931 Mb. The sizes of these disaster images were not always the same, but the smallest was $268 \times 188 \times 3$, while the largest was $1920 \times 1080 \times 3$, with the median size being $720 \times 480 \times 3$. These disaster images were resized as the input of feature extraction using a pre-trained network, these are resized $224 \times 224 \times 3$, frequently.

Table 1. The number of each class for an earthquake damage images : QuakeV

Earthquake damage class	Number of data
Tsunami damage	221
Bridge collapse	222
Road damage with accident risk	210
Initial smoke and fire	210
Non-disaster	254
Total of dataset	1,117

3.2. Single- and two-layer extracted feature classifier applied results

Utilizing ten pre-trained network architectures, the QuakeV image data set, mentioned above, was extracted from one or two concatenated layers next to the final output. The extracted features were applied as the input of the support vector classification models. The settings required to compute them were as follows: 1) Using the preferred ten pre-trained architectures, 2) Constructing a support-vector classification model based on the training-feature matrix with 782 rows and as many columns as the number of elements in one or two extracted layers, and test features with 335 rows and the same column size, 3) The execution environment used GTX1070 8-GB GPUs with a computation capacity of 6.1.

Table 2 shows that the one and two layers of damage features are extracted as the inputs for which the feature matrix is applied to support the vector classifier trained results using the QuakeV dataset. On the three rows showing views such as Alex Net, VGG16, and VGG19, the first column classifier accuracy under the extracted single-layer neighbor final output is higher than that of the second column classifier under the one back-concatenated layer, respectively. Furthermore, the third column classifier under both extracted layers is the most accurate at each row, at around 92% accuracy. From the next view point on the three rows such as the Google Net, ResNet50 and Inception v3, there is the same relationship between the first classifier under the extracted layer neighbor of the final output and the second classifier under the one back-concatenated layer. However, the third classifier under both extracted layers is less accurate, with the accuracy of each row decreasing by 0.3% to 1.2%. In contrast, the ResNet50-based classifier produces the highest accuracy at the second column classifier under the extraction of the one back-concatenated layer, which is more than the third classifier under both extracted

layers. Therefore, it is not always true that an increase in the number of extracted features that are used as an input leads to a higher accuracy of the support-vector classifier. In the present study, the most promising classifier was found to be the ResNet50-based support vector classifier under the ‘add_15’ layer extracted features, with 100,352 elements.

Table 2. Single and two layers of damage feature extracted support vector classifier learning results using a QuakeV dataset.

Pre-trained network	Near final Extracted single layer (1)	One backed Extracted single layer (2)	Extracted both layers
Alex Net	‘fc7’: #4,096 91.94%	‘fc6’: #4,096 92.24%	Both ‘fc7’ and ‘fc6’ 92.84%
VGG16	‘fc7’: #4,096 91.34%	‘fc6’: #4,096 91.94%	Both ‘fc7’ and ‘fc6’ 92.24%
VGG19	‘fc7’: #4,096 90.75%	‘fc6’: #4,096 92.24%	Both ‘fc7’ and ‘fc6’ 92.24%
Google Net	‘inception_5b’: #40,768 91.94%	‘inception_5a’: #40,768 93.73%	Both ‘inception_5b’ and ‘inception_5a’ 92.84%
ResNet18	‘res5b’: #25,088 93.43%	‘res5a’: #25,088 92.54%	Both ‘res5b’ and ‘res5a’ 94.03%
ResNet50	‘add_16’:100,352 92.84%	‘add_15’:#100,352 94.93%	Both ‘add_16’ and ‘add_15’ 93.73%
Inception v3	‘concat_2’: #49,512 87.76%	‘concat_1’: #49,512 89.55%	Both ‘concat_2’ and ‘concat_1’ 89.25%

Here, #4,096 abbreviates that the number of elements is 4,096 contained at the extracted layer.

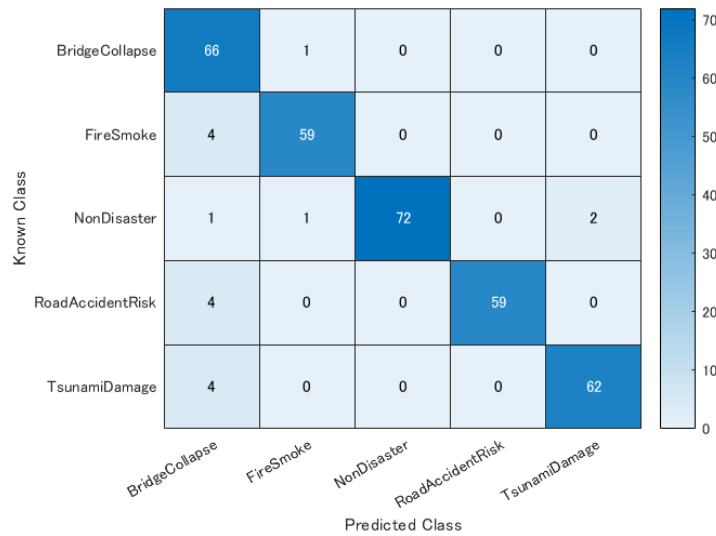


Figure 3. Confusion matrix of the QuakeV support vector classifier based on ResNet50 extracted single feature at the ‘add_15’ layer

Figure 3 shows the confusion matrix of the QuakeV support vector classifier based on the ResNet50 extracted single feature at the ‘add_15’ layer. Regarding the diagonal value of the confusion matrix, the predicted labels for each class match almost all the actual disaster feature classes. In the first row of bridge collapse, there is one false prediction regarding the initial smoke and fire. In the row for the initial smoke and fire, there are four false predictions related

to bridge collapse. In the row related to non-disaster damage, there are four false predictions related to bridge collapse, smoke and fire, and tsunami damage. In the row for road damage leading to a risk of accidents, there are four false predictions related to bridge collapse. Because bridges are linked to the road network, there are images that fall between bridge collapse and road damage leading to a risk of accidents, in that there is a view of the road surface in the background. In the row for tsunami damage, there are four false predictions regarding bridge collapse. Therefore, the precision of the bridge collapse prediction is lower than that of the other classes. There are certain cases in which bridges are damaged by a tsunami flow, as occurred in the Great East Japan Earthquake of 2011. Those predictions were based on one iteration of a five-fold cross validation classifier. Further hyper-parameter optimization would be required to minimize the prediction error. Figure 4 shows 15 randomly selected images with the predicted label of the test image based on the ResNet50 classifier extracted single feature for the ‘add_15’ layer. These predicted labels are in good agreement with the images of the earthquake damage.

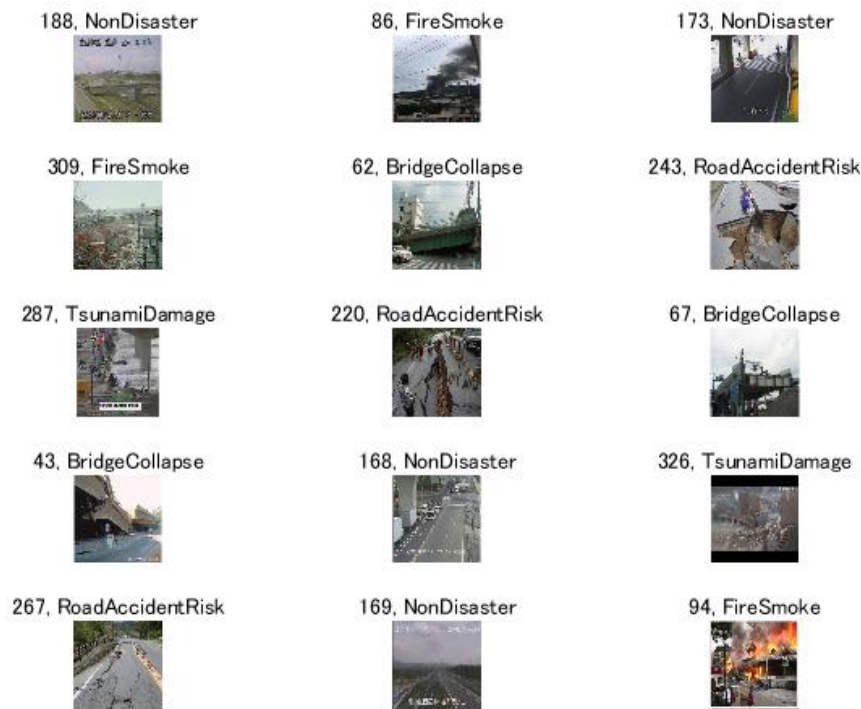


Figure 4. Predicted label of support vector classifier using ResNet50 extracted single feature at the ‘add_15’ layer

3.3. Three-layer extracted feature classifier applied results

Table 3 shows the triple deep layers of damage features extracted from the support vector classifier learning results using a QuakeV dataset. The highest accuracy is 95.82% for the first row of the triple feature extracted classifier, based on the ResNet101. However, there is a large number of input features, specifically, 30,000 elements. For this reason, the model incurs disadvantages in that it requires much more memory and a longer computing time. In the fourth row of Inception-ResNet-v2, which corresponds to a rare case, the extraction of five back layers gives the classifier with the highest accuracy, specifically, 94.63%. Also, previous triple-feature extraction studies have shown that an increase in the number of extracted features as an input does not always lead to a higher level of accuracy of the support vector classifier. A view of a single layer near the final output, such as that of ResNet101 under ‘res5a’ and that of DenseNet-201 under ‘conv5_block32,’ for which the extracted features are used as an input to the support vector classifier with the highest accuracy, 95.52%, for the single-layer extraction. The

extracted feature has 9,000 or 10,000 fewer elements than the triple layers extracted with ResNet101, as mentioned above. Next, we implement hyper parameter optimization for the triple feature classifier extracted using ResNet101 for three layers, such as ‘res5c,’ ‘res5b,’ or ‘res5a.’ Furthermore, we carried out single-feature classifier extraction based on ResNet101 ‘res5a’ and DenseNet-201 ‘conv_block32.’

Table 3. Triple deeper layers of damage feature extracted support vector classifier learning results using a QuakeV dataset.

Pre-trained network	Near final Extracted single layer (1)	One backed Extracted single layer (2)	Two backed Extracted single layer (3)	Triple Extracted layers
ResNet101	‘res5c’: #100,352 94.33%	‘res5b’: #100,352 94.63%	‘res5a’: #100,352 95.52%	Triple: ‘res5c’, ‘res5b’ and ‘res5a’ 95.82%
ResNet101	‘res5a’: #100,352 95.52%	‘res4b22’: #20,704 92.24%	‘res4b21’: #20,704 92.84%	Triple: ‘res5a’, ‘res4b22’ and ‘res4b21’ 94.03%
Inception-ResNet-v2	‘block8_10’: #133,120 93.43%	‘block8_9’: #133,120 91.34%	‘block8_8’: #133,120 93.73%	Triple: ‘block8_10’, ‘block8_9’, ‘block8_8’ 93.13%
Inception-ResNet-v2	‘block8_7’: #133,120 94.03%	‘block8_6’: #133,120 94.63%	‘block8_5’: #133,120 94.03%	Triple: ‘block8_7’, ‘block8_6’, ‘block8_5’ 94.03%
DenseNet-201	‘conv5_block32’: #89,736 95.52%	‘conv5_block31’: #89,736 95.22%	‘conv5_block30’: #89,736 95.22%	Triple: ‘conv5_block32’, ‘block31’, ‘block30’ 95.22%
DenseNet-201	‘conv5_block30’: #89,736 95.22%	‘conv5_block29’: #89,736 95.52%	‘conv5_block28’: #89,736 95.52%	Triple: ‘conv5_block30’, ‘block29’, ‘block28’ 95.22%

3.4. Hyper parameter optimized results

Table 4 lists the hyper parameter optimization results regarding the top three support vector classifiers for an input of extracted features using pre-trained ResNet101 and DenseNet-201. The objective function is the loss function of the five-fold cross validation. This evaluation process is iterated 30 times to minimize the loss. The first row of the table shows the result where the triple layer features are extracted using ResNet101 under ‘res5c,’ ‘res5b,’ ‘res5a,’ and the feature inputs are evaluated to optimize the hyper parameters for the support vector classifier, for which the accuracy is improved to 97.01%, whereas the previously trained value was 95.82%. The minimum objective function value is 0.0575. Given the large number of feature elements with three concatenated layers, the evaluation runs took 3.5 h. The second row of the table shows the results of extracting a single-layer feature using ResNet101 under ‘res5a,’ and for which the feature input is applied to optimize the hyper parameters for the support vector classifier, for which the accuracy is improved to 97.50%, whereas the previously trained value was 95.52%. The minimum objective function value is 0.0627. The training runs took 1.5 h to complete. The third row of the table shows those results for which the single-layer feature is extracted based on DenseNet-201 under ‘conv5_block32_concat,’ and the feature input are computed to optimize the hyper parameters for the support vector classifier, for which the accuracy is improved to 97.50%, where the previously trained value was 95.52%. This accuracy improvement is the same as that obtained for the extraction of a single layer with ResNet101. However, the minimum objective function value is 0.0588. The validation iterations took 1 h,

50 min. Thus, the DenseNet-201-based feature extraction and hyper parameter optimized support vector classifier are the most promising when using a QuakeV earthquake damage data.

Table 4. Hyper parameters optimized results of top-3 support vector classifier under an input of extracted feature using pre-trained ResNet101 and DenseNet-201.

Pre-trained network	Extracted layer trained classifier	Hyper parameter optimized classifier
ResNet101 Triple layers extraction	'res5c', 'res5b', 'res5a': #301,056 95.82%	Objective function : 0.0575 Box constraint C : 0.0061 Rbf kernel scale : 3.0131 Training run time : 215m39s 97.01%
ResNet101 Single layer extraction	'res5a': #100,352 95.52%	Objective function : 0.0627 Box constraint C : 808.2094 Rbf kernel scale : 625.0391 Training run time : 97m34s 97.50%
DenseNet-201 Single layer extraction	'conv5_block32_concat': #89,736 95.52%	Objective function : 0.0588 Box constraint C : 0.4305 Rbf kernel scale : 0.0649 Training run time : 108m26s 97.50%

Note) The objective function stands for the 5-fold cross validation function value.

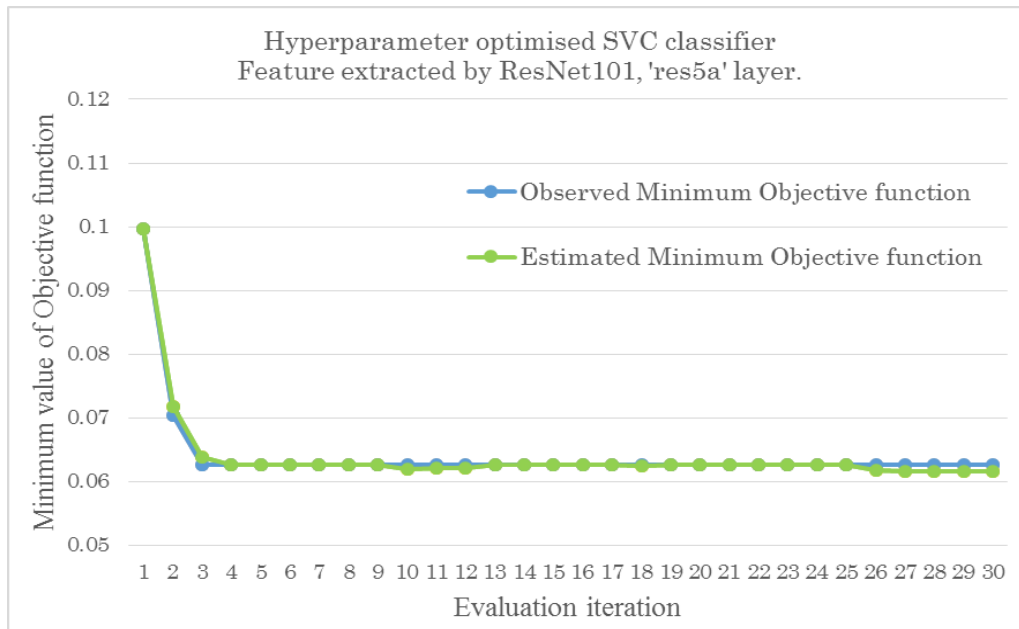


Figure 5. Hyper parameters optimization process of support vector classifier based on ResNet101 extracted single feature 'res5a' layer

Figure 5 shows the hyper-parameter optimization process for a support vector classifier based on the ResNet101 extracted single feature 'res5a' layer. After three iterations of five-fold cross validation, the loss function was minimized at a stable level of around 0.06 during the validation runs. Furthermore, Figure 6 shows the hyper-parameters optimization process for the support vector classifier based on the single feature 'conv5_block32_concat' layer extracted with DenseNet-201. After three iterations, the loss function was minimized at a stable level around 0.062. Furthermore, at the points corresponding to six and twelve iterations, the objective

function value was again improved at that point at which the evaluation process converged to a minimum of 0.058.

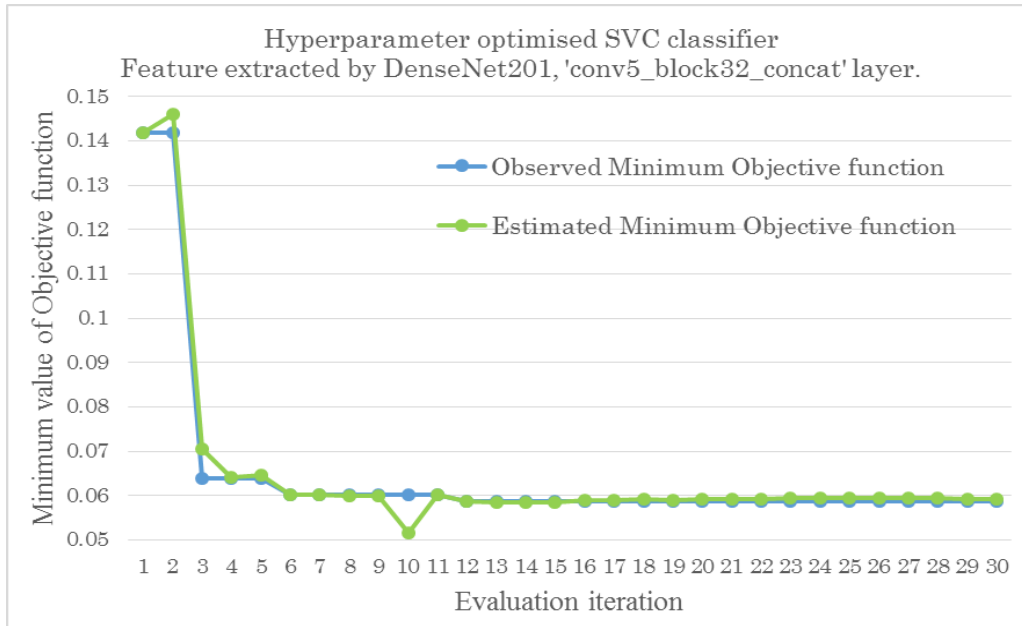


Figure 6. Hyper parameters optimization process of support vector classifier based on DenseNet-201 extracted single feature ‘conv5_block32_concat’ layer

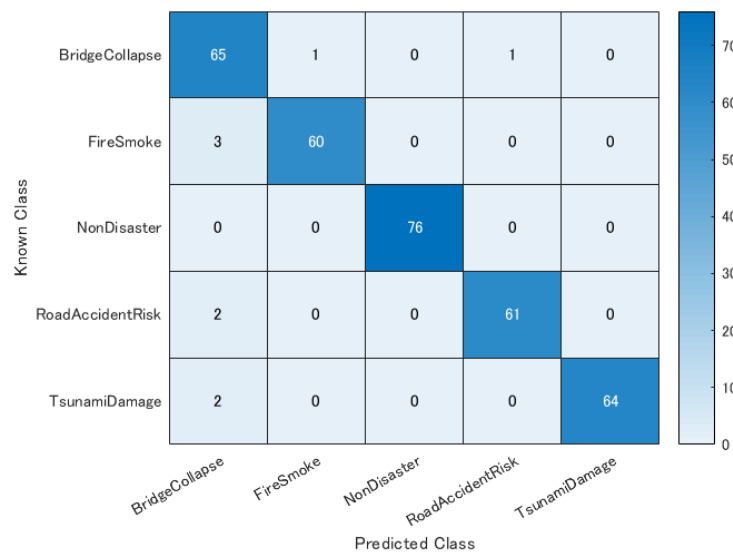


Figure 7. Confusion matrix of a QuakeV hyper parameter optimized support vector classifier extracted feature ‘conv5_block32_concat’ layer based on DenseNet-201

Figure 7 shows the confusion matrix for a QuakeV hyper-parameter optimized support vector classifier feature ‘conv5_block32_concat’ layer, extracted based on DenseNet-201. Regarding the diagonal value of the confusion matrix, the matching of the predicted labels for each class are improved for almost all the actual damage classes than those shown in Figure 3. In the first row, for bridge collapse, there is one false prediction regarding initial smoke and fire and road damage leading to a risk of accidents. In that row for initial smoke and fire, there are three false predictions regarding bridge collapse, which is less than the prediction shown in Figure 3. In the

row for the non-disaster damage, there is no improvement in the number of false predictions relative to Figure 3. In the row for the road damage leading to a risk of accidents, there are two false predictions regarding bridge collapse, which is less than in Figure 3. In the row for tsunami damage, there are two false predictions for bridge collapse. In the first column, there are still seven false bridge collapse predictions, although this is an improvement over the thirteen false predictions in Figure 3. Figure 8 shows fifteen randomly selected test images and the predicted labels for the QuakeV hyper parameter optimized support vector classifier extracted feature ‘conv5_block32_concat’ layer, based on DenseNet-201. These predicted labels are in good agreement with the actual images of earthquake damage, being similar to the prediction results shown in Figure 4.



Figure 8. Predicted label of a QuakeV hyper parameter optimized support vector classifier extracted feature ‘conv5_block32_concat’ layer based on DenseNet-201

4. CONCLUSIONS

To conclude this paper, we present the contribution of this work as demonstrated through several machine-learning case studies. We believe that disaster damage data mining offers the opportunity to discover further knowledge needed for disaster mitigation and social loss.

4.1. Contribution of this work

This paper proposes an application to classify disaster damage using feature-extracted support vector classifiers based on ten pre-trained architectures. These results were applied to a QuakeV earthquake damage data set. It was found that the disaster damage classifier based on DenseNet-201 under the single ‘conv5_block32_concat’ layer feature extraction is the most promising with approximately 97.50% accuracy. Although the ResNet101-based classifier produced the same level of accuracy, the minimum loss function value is larger than that of the DenseNet-201-based classifier. To support decision making with respect to the initial response and to mitigate the relevant loss after a large earthquake using CCTV images, this paper highlighted disaster damage features such as tsunami damage, bridge collapse, initial smoke and fire, and

road damage leading to a risk of accidents. We actually implemented the image classification method by applying it to a dataset containing 1,117 images. We drew on relevant open-source websites from which we could download digital image records of large earthquake damage. Using the ten pre-trained architectures, we constructed a support vector classification model based on 782 training data sets and 335 validation images. For one of the feature-extracted learning results, based on ResNet101 and DenseNet-201 using a single layer, we achieved an accuracy of 95.52%. Furthermore, the hyper parameters of these models could be optimized at an accuracy of 97.50% among our trial classifiers. After the classification model reads an input image, it can compute the indexes of the predicted label to determine whether the true disaster feature class belongs to the actual class or not. Thus, the proposed disaster damage classifier application and the QuakeV earthquake dataset can be used with datasets consisting of thousands of images.

4.2. Future work

It should be possible to apply disaster damage classification not only to earthquake damage, but also to other disasters such as building collapse [26], traffic signal failure, landslides, strong winds [27], and heavy rain and floods [28]. In our daily life, fire and flood disasters occur much more frequently than earthquakes. This proposed classifier could enable target disaster surveillance for each region using thousands of disaster feature images covering the target classes. It would take a long time and a considerable amount of work to collect newly obtained disaster damage features based on CCTV records and other multi-mode data resources which contain initial damage features. We will continue to collect video data after a large earthquake occurs. Disaster datasets are not always learned from overall images, but rather from localized detections for which the original images focus on clear disaster features. In contrast, the background disaster region of interest should be excluded for noise reduction. We would monitor the added variations in the disaster features that have not yet been experienced, such that disaster damage mining would enable the discovery of the knowledge needed to make decisions on initial responses with respect to high-priority locations with significantly damaged infrastructure.

ACKNOWLEDGEMENTS

We would like to thank Fukumoto Takuji and Kuramoto Shinichi (MathWorks Japan) for providing us with useful information on the MATLAB resources such as the Image Processing, Statistics Machine and Learning, and Deep Learning.

REFERENCES

- [1] Manzhu, Y., Chaowei, Y., and Yun, L, (2018) Big Data in Natural Disaster Management : A Review, *Geosciences* 8, 165.
- [2] Michel, U., Thunig, H., Reinartz, P. (2012) Rapid Change Detection Algorithm for Disaster Management, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol.1-4.
- [3] Singh, A. (1989) Review Article Digital Change Detection Techniques using Remotely-sensed data, *International Journal of Remote Sensing*, Vol.10, No.6, 989-1003.
- [4] Saptarsi, G., Sanjay, C., Sanhita, G. et al. (2016) A Review on Application on Data Mining Techniques to Combat Natural Disasters, *Ain Shams Engineering Journal*.
- [5] Sakaki, T., Okazaki, M., Matsuo, Y. (2010) Earthquake shakes Twitter users : Real-time Event Detection by Social Sensors, *Proceedings of 19th International Conference on World Wide Web*, ACM.
- [6] Walter, D. (1954) *Ballantine Books*, V-2, ASIN: B000P6L1ES, 14.

- [7] SDM (2016) Rise of Surveillance Camera Installed Base Slows, <https://www.sdmmag.com/articles/92407-rise-of-surveillance-camera-installed-base>.
- [8] Supanee, T., Kurt, T., Sally, E. (2009) Object Oriented Change Detection of Buildings after a Disaster *ASPRS Annual Conference* Baltimore, Maryland.
- [9] Ranga, R. Jordan, G. (2012) Probabilistic Change Detection Framework for Analyzing Settlement Dynamics using Very High-resolution Satellite Imagery, *Procedia Computer Science*, 9, 907-916.
- [10] Maeda, Y., Konno, A, Morita, K. et al. (2018) Infrastructure Disaster Damage Information Real-time Monitoring, Abstraction, Sharing Techniques Developments, SIP Disaster Prevention Research 2014-2018, (in Japanese), National Institute for Land and Infrastructure Management (NILIM) Report 2018, 96-97, <http://www.nilim.go.jp/english/eindex.htm>.
- [11] Kataoka, H., Shirakabe, S. et al. (2017) Futuristic Computer Vision through 1,600 papers survey, *CVpaper.challenge* in 2016.
- [12] National Oceanic and Atmospheric Administration (NOAA), accessed at 2018 Sept 17: *Natural Hazards Image Database*, Events contains Earthquake, Tsunami, Volcano, and Geology, <https://www.ngdc.noaa.gov/hazardimages/#/earthquake>.
- [13] Aurelien, G. (2017) *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*.
- [14] Krizhevsky, A., Ilya S., and G. E. Hinton. (2012) ImageNet Classification with Deep Convolutional Neural Networks. *Advances in neural information processing systems*.
- [15] Szegedy, C., Wei L., Yangqing J., et al. (2015) Going deeper with convolutions, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1-9.
- [16] VGG model, the Visual Geometry Group at University of Oxford, Simonyan, K. and Zisserman, A. (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition, *ICLR*.
- [17] ResNet model, Kaiming, H. Xiangyu, Z., Shaoqing, R. et al. (2015) Deep Residual Learning for Image Recognition, *arXiv:1512.03385v1*.
- [18] Inception v3 model, Szegedy, C., Vincent, V., Sergey, I. et al. (2015) Rethinking the Inception Architecture for Computer Vision, *CVPR*, 2818-2826.
- [19] Inception-ResNet-v2 model, Szegedy, C., Sergey, I., Vincent, V. et al. (2016) Inception-v4, Inception-ResNet and Impact of Residual Connections on Learning, *arXiv:1602.07261v2*.
- [20] DenseNet model, Huang, H., Liu, Z., Maaten, L. et al. (2017) Densely Connected Convolutional Networks, *CVPR*.
- [21] Hsu C-W., Chang C-C. and Lin C-J. (2003) A Practical Guide to Support Vector Classification, *Technical report*, Department of Computer Science, National Taiwan University.
- [22] Multi-class classification, pp.29-30, edited by Chan C-C., Lin C-J. (2013) *LIBSVM: A Library for Support Vector Machines*, Initial version 2001, Last updated March.
- [23] Feurer, M., Klein, A., Eggenberger, K. (2015) Efficient and Robust Automated Machine Learning, Neural Information Processing Systems Conference (*NIPS*).
- [24] Sibanjan, D., Cakmak, U. M. (2018) *Hands-On Automated Machine Learning*, Packt.
- [25] Gonzalez, R., Woods, R., Eddins, S. (2015) *Digital Image Processing Using MATLAB* second edition, McGrawHill Education.
- [26] Prince, S. (2017) *Computer Vision Models, Learning, and Inference*, Cambridge University Press.
- [27] Rajalingappaa, S. (2018) *Deep Learning for Computer Vision*, Packt.
- [28] Yasuno, T. (2012) Daily Interaction Behavior, Urgent Support Network on Tohoku Tsunami 2011, Tottori Quake 2000, *Social Capital and Development Trends in Rural Areas* Vol.8, Chapter12.

- [29] Yasuno, T. (2009) Estimating Occurrence Probability and Loss Index to Manage the Social Risk of Strong-Winds, *International Symposium Society for Social Management Systems(SSMS)*.
- [30] Yasuno, T. (2018) Dam Inflow Time Series Regression Models for Minimizing Loss of Hydro Power Opportunities, *PAKDD Proceedings, the Workshop of Data Mining for Energy Modeling and Optimisation (DaMEMO)* , Melbourne.

Authors

Yasuno Takato received his D.E. degree from Tottori University. He has over 18 years of consulting engineer experience in infrastructure planning. Now he works proof of concept (PoC), prototype consulting tool as a senior researcher at the institute (RIIPS) in Yachiyo Engineering Co., Ltd. His interest is Data Mining and Machine Learning, Predictive Infrastructure Management, and Time Series Regression. He is a member of JSAI, ORSJ.



Amakata Masazumi received his B.E degree from Kyoto University and D.E. degree from Kanazawa University. He has over 20 years of experience in flood control, water utilization , river environment and he is a specialist of the fluid analysis. Now he works as a manager of the research institute (RIIPS) in Yachiyo Engineering Co., Ltd. His research interest is to apply Machine Learning and Deep Learning to the field of civil engineering.



Fujii Junichiro received his B.E degree from Kyoto University and M.A.S. (Interdisciplinary Information Studies) degree from University of Tokyo. He has over 15 years of experience in information systems development, and presently works as a senior researcher at the institute (RIIPS) in Yachiyo Engineering Co., Ltd. His research interest is applying artificial intelligence to the field of civil engineering.



Shimamoto Yuri received her B.E and M.E from Ehime University. She has research experience in machine learning and image processing, when she is a student concerning concrete crack. Now she works as a researcher at the institute (RIIPS) in Yachiyo Engineering Co., Ltd. Her research interest includes Machine Learning and Image Analysis for civil engineering.

