CYBER-PHYSICAL SYSTEMS
https://doi.org/10.1080/23335777.2022.2051210

Deep Learning Assist IoT Search Engine for Disaster Damage Assessment

Hengshuo Liang[a], Lauren Burgess[a], Weixian Liao[a], Erik Blasch[b] and Wei Yu[a]

[a]Department of Computer and Information Sciences, Towson University, Towson, MD, USA; [b]Air Force Research Laboratory, DC, USA

ABSTRACT

In this paper, we address the issue of disaster damage assessments using deep learning (DL) techniques. Specifically, we propose integrating DL techniques into the Internet of Things Search Engine (IoTSE) system to carry out disaster damage assessment. Our approach is to design two scenarios, Single and Complex Event Settings, to complete performance validation using four Convolutional Neural Network (CNN) models. These two scenarios are designed with three possible network services. Our experimental results confirm that all four CNN models can learn each label during the single event setting well. Whereas, with complex event settings, the CNN models have learning difficulty because multiple events have closely related labels.

ARTICLE HISTORY

Received 29 November 2021
Accepted 15 January 2022

KEYWORDS

Internet of things (IoT); IoT search engine (IoTSE); machine learning (ML); deep learning (DL)

1. Introduction

As technology advances, so does our ability to provide data and deep learning (DL)-driven analytics and predictions. ML techniques, especially DL, have received growing attention and applied to numerous areas, including image and video classification, natural language processing, robotics, networking, mobile computing and cybersecurity, among others [1–9]. DL is prominent in our daily lives; at home with our digital assistants (Apple's Siri, Amazon's Alexa, Google Home), within our businesses that forecast finance models, among social interactions affording collaboration, and around emerging in autonomous vehicles and smart homes with a variety of smart devices (NEST thermostats, Ring cameras) [3,10]. DL techniques work to secure the network with anomaly and intrusion detection systems using supervised and unsupervised techniques, provide the capability of identifying security breaches that can occur on a specific computer or on the network [11–15], and balance privacy, utility, safety and reliability [16,17].

CONTACT Wei Yu wyu@towson.edu Department of Computer and Information Sciences, Towson University, Towson, MD, USA

As a viable technique, DL can also positively impact the way emergency response teams and humanitarian organisations handle damage assessments after natural disasters. DL damage assessments provide these emergency teams a way to efficiently respond and effectively determine appropriate resources to manage the aftermath, such as getting the Federal Emergency Management Agency (FEMA) involved and others. Typically, these agency groups need to assess within the first 48 to 72 hours before any resources can be provided [18]. To support the rapid response, before a disaster, it is up to the local government to have accomplished preliminary damage assessments, acquired a plan of action, completed training and conducted exercises to know how to respond when a disaster takes place [19,20]. When looming danger is nearing, the local government needs to be able to provide a warning that tends to ensure public safety [19]. After a disaster, the government must work to provide needed services such as water, power, communications, transportation, shelter and any medical care [19]. Along with the local government, there are public, military, and private utility crews and emergency response teams, including fire and police units, medical personnel and rescue workers, who have the responsibility of providing aid and restoring essential services [19,20]. Communication is vital for recovery from a disaster, which is the reason why many times the local government relies on using media to publicise the various assistance available, advertise how everyone can access them, and then initialise the damage assessment [19,20].

Generally, there are three ways for damage assessments to be completed by these response teams. First, they can gauge the extent of the damage by determining how many damaged or flooded buildings there are by performing a ground survey. It is important to note that ground surveys are labour-intensive, can be time-consuming and are typically done by first-hand sources [18,21]. Second, overhead imagery can be used; which has limited knowledge due to the pixel resolution of the imagery. Third, distributed computing services such as crowd-sensing that engage a number of distributed workers to jointly complete sensing and computing tasks [22,23]. Nonetheless, crowd-sensing requires that there is a reliable network infrastructure so that the data can be updated and analysed [17].

The Internet of Things (IoT), along with the vision of cyber-physical systems (CPS), has created a novel ecosystem for sensing and actuation, enabling intelligently controlled autonomous systems and resource/data sharing to conserve energy, water, crops, manage factories and provide situational awareness on an unprecedented scale, leading to next-generation smart-world systems [4'13,24–29]. As IoT progresses, it calls for designing efficient and fast IoT search engines (IoTSE) to find IoT devices, retrieve IoT data and assess impacts [30,31]. While basic examples of IoTSE exist, considerable challenges prevent the full realisation of an efficient and intelligent IoTSE that provides universal data service, scalable data communication and retrieval, and efficient querying of massively distributed heterogeneous IoT devices and data using distributed computing architecture, algorithmic design and optimisation [30,32–34]. In this study, we focus on the intelligence aspect of IoTSE and utilise IoTSE with DL capability to improve and enhance the efficiency of completing damage assessments.

In this study, we make the following two contributions.

- **A Framework**: We propose a new framework that leverages IoTSE with the DL capability to carry out disaster damage assessments. To meet different requirements in the scenarios, the IoTSE framework is demonstrated in three examples: (i) SERVICE-STABLE: when the network status is still stable in the damaged area, the whole IoTSE system will collect all the related information from social media on the related topics; (ii) SERVICE-DEGRADED: when people have no access to the network in the damaged area, the IoTSE system will collect and filter related data from smart IoT devices (e.g. traffic cameras, cameras in smart cars located in the city), and those devices that are repeatedly recording images and uploading into remote server; as well as (iii) SERVICE-OUTAGE: when the damaged area is entirely out of service, dynamic networking infrastructure formed by drones as data collectors to transmit data or as remote data analysers, only sending results back to a data centre.
- **Extensive Evaluation**: We carry out extensive performance evaluation of IoTSE to validate the efficacy. We first preprocess the data by mechanisms such as randomly cropping and obtaining a validated dataset [35]. We then set two different scenarios for testing four pre-trained DL models: (i) *Single Event Test* (SET): uses only one single event in the validation. Our experimental results show that all of these models can learn well and perform better when increasing data quality and quantity. (ii) *Complex Event Test* (CET): In this setting, multiple events occur in different locations with a similar topic. In the CET testing, we find that all models have a hard time learning the same label across the different events; additionally, when more similar event topics are fed into the model, the learning performance becomes worse.

The remainder of this paper is organised as follows: Section 3 briefly introduces damage assessment issues, DL and IoTSE. Section 2 reviews existing research efforts that are relevant to the paper. Section 4 provides the framework design, scenarios and clarifies the reason why these scenarios are essential for validation. Section 5 overviews the methodologies as well as analyzes the DL performance in different scenarios. Section 6 discusses some remaining issues for future research and Section 7 concludes the paper, respectively.

2. Related works

In the following, we review various research efforts that are closely relevant to our study.

Recently, there are a number of existing research efforts with DL to carry out tasks such as classification in a variety of applications [5,36,37]. For example, Mikołajczyk et al. applied a new data augmentation in DL training in order to improve performance in image classification [36]. Cheng et al. designed a new neutral network structure with a feature map over the data set in order to obtain better performance in image classification [37]. Likewise, Liang et al. designed a CNN-based scheme to automatically recognise industrial components in industrial IoT system [5].

There are also some existing efforts on leveraging DL in IoT systems in order to enhance security and improve network performance [12,15,38,39]. For example, Li et al. applied DL within IoT-based edge computing to improve the performance of information extraction in a complex environment [38]. Jie et al. proposed a DL approach for the low power IoT products in order to gain better performance while using less power consumption [39]. Likewise, Xu et al. explored the evolutionary process of data integrity threats and defence in CPS and leverages deep neural networks to detect attacks [12].

In addition, there are some research efforts on IoTSE [24,30,40]. For example, Hatcher et al. applied Long Short-Term Memory (LSTM) machine learning scheme in IoTSE in order to predict incoming query volume, which leads to query efficiency [30]. Likewise, Cheng et al. designed an IoTSE platform based on Constrained Application Protocol (COAP) and conducted experiments on query optimisation algorithms [40].

Furthermore, there are some research efforts for public safety and related applications [17,42,43]. For example, Yu et al. proposed a user-side-based solution for enhancing public safety communications [17]. Jarwan et al. used multi narrow bands in Long Term Evolution (LTE) network to deal with different kinds of disaster scenarios [43]. Likewise, Wang et al. designed several resource allocation schemes for out-of-courage device-to-device group communication, which is important to public safety.

3. Preliminaries

The following Section discusses the issues related to damage assessment and provides an introduction to deep learning and IoTSE.

3.1. Damage assessment issues

There have been several research efforts on understanding and addressing various issues when it comes to completing damage assessments. Existing solutions such as utilising satellite images [20,44] or social media [18,21], while prior efforts include combing both for physical and human information fusion (PHIF) [34]. Generally speaking, damage assessments are carried out by conducting a field evaluation. Experts typically take pictures, complete interviews of people in the affected area [18,21] and consult prior imagery. The field assessments have some challenges, such as having limited resources at the site, time-consuming and severe living conditions.

There are a number of technologies available to complete damage assessments, including remote sensing, mobile surveys, remote validation, and others. For example, remote sensing (e.g. aerial imagery) would be used to identify damaged homes and infrastructure without having someone at the location [20,44]. Nonetheless, utilising remote sensing-based schemes can have some issues such as usability due to weather and limited network availability [18,20]. Mobile surveys have been used in conjunction with field assessments for immediate verification and feedback [20], while there are no images to review before the event to do a comparison [21]. In addition, *remote validation* is used when the work has been completed, when the damage can be validated easily through photography, or when onsite validation is not needed [20]. Utilising remote validation means that any privacy and information security during data collection needs to be considered, such as personally identifiable information (PII), data accuracy, timing and data storage [20]. These challenges delay the process of gathering data and completing the damage assessment, which can hinder the arrival of needed relief operations [18].

3.2. Deep learning (DL)

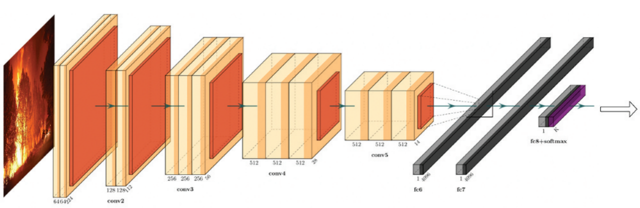
DL can be beneficial to addressing the damage assessment, as shown in the xVIEW2 challenge (https://xview2.org/). DL is a type of advanced ML, which is closely related to computational statistics, providing the ability of making predictions using computers and mathematical models [6]. These DL mathematical models provide ML a way to automatically learn and find patterns in large amounts of data [6,45]. ML primarily focuses on classification and regression from features that have been learned from training data [6,46].

Nonetheless, traditional ML techniques are limited as pattern-recognition required engineers to design a way to extract information and convert the raw data into feature vectors, allowing classifiers to identify patterns in the input [46]. Deep learning (DL), on the other hand, focuses on establishing a neural network for analytic learning allowing for the interpretation of data, such as images, audio and text [2,6]. For our research, we utilise image classification, assuming that a label is extracted from a feature set using an algorithm [47]. The goal is to determine whether DL could be an efficient way to conduct remote damage assessments instead of sending resources onsite to complete assessments. To begin the research, we leverage four of the most popular DL models (Vgg19 [48], ResNet50 [49], GoogleNet [50], and MobileNetV2 [51] and well-defined training sets), which will be discussed in detail in Section 4.3. These data sets are pre-processed and fine-tuned for training simulation as discussed in Section 5.1. There are a variety of training set sizes, but for DL algorithms, the larger the dataset, the more accurate the algorithms become [3].

An example of DL workflow for assessing damage to disaster is shown in Figure 1. As shown in the figure, an image related to California Wildfires is used as input and DL model (i.e. VGG16 CNN) is adopted to extract features from the given image. Via a set of layers in CNN, a list of probability for individual labels is computed as results, which are further used to determine that the input image belongs to serve damage scene.

There are basic types of paradigms in DL: supervised, semi-supervised, unsupervised, as well as related methods such as transfer, active and reinforcement learning [2,3,6,45,52]. Supervised learning requires labelled data. Unsupervised learning uses unlabelled (or un-categorised) data but has assumed structural properties [6,53]. Unsupervised learning provides the ability to find hidden patterns or data groupings, which can be beneficial when using DL [3,6]. Reinforcement learning (RL) is a combination of supervised learning as a goal has to be specified and unsupervised learning in the fact that a label is not needed for collected [3]. By interacting with the environment, RL uses autonomous agents in order to make determination of actions to take in an environment [3,54].

Supervised learning simply means that the data is made up of features and a label to attempt to correctly map features to the label. During the training phase, the machine is able to adjust parameters as it is learning, ultimately reducing the errors or the distance between the provided and the desired output [3,46]. Supervised learning is most commonly used during training in both ML and DL due to its performance on the available training data [3,45,46].



Damage Label	Probability
Severe Damage	**0.98955**
Mild Damage	0.87874
Little or No Damage	0.65222
Can not juggle	0.21285
Nan (Not related)	0.12456

Figure 1. Example of DL for recognising severe damage scene.

DL has proven to show incredible accuracy due to the computational power of the latest processors and the large amounts of existing data that can be used [3]. For the purpose of our research, image-based DL is combined with IoTSE.

3.3. Internet of things search engine (IoTSE)

Along with DL, IoTSE also provides solutions in damage assessments. There are various reasons why IoT data analytics can support damage assessment. Since IoT systems exchange more data between the smart devices versus between users and devices, high amounts of data are created. With IoT data sharing, from the large amount of data, analytics supports future decisions [24,32,55]. Currently, the large amounts of data, data management and service become a concern as each IoT system has its own data structure, data rate and performance requirements due to the heterogeneous data sources [32,55]. Sharing the data also raises a concern due to the different standards and settings, making it difficult to share with other devices and organisations [24,25,32,56]. Scalability and accuracy are other concerns with IoT devices as connecting to the cloud can be much costly for processing data, but there can also be time-critical applications that requires timely decisions [55].

IoTSE seeks to solve IoT systems' management and search issues between devices and data [32]. The idea behind IoTSE is to mimic a typical Internet browser; however, all the IoT data would become a Uniform Resource Locator (URL) link, allowing user to be able to view the data based on their queries [32]. Like web-based search engines, IoTSE is capable of providing functionalities: data collection, indexing and organisation [24,32]. Data collection in IoTSE can use a subscription/notification-based scheme to gather data from IoT devices instead of having to utilise web servers in a typical web-based search [32]. However, IoTSE has some of its own issues that it needs to resolve, including performance, multi-system integration, security and privacy [24].

With the capabilities of powerful data analysis tools like ML and DL, the demands to access IoT data have increased drastically [24]. DL should be able to provide efficiency in an IoTSE architecture as it is capable of improving performance computation, reducing workloads and improving decision-making [24,55]. IoTSE combined with DL should be able to create a standard format regardless of the heterogeneous IoT devices, operational domains and communication systems [32].

4. Our approach

This Section provides an overview of IoTSE that utilises DL for damage assessment. Particularly, the section outlines the framework problem space, presents the dataset, analyzes the data labels and evaluates the proposed scenarios.

4.1 Framework

Based on current damage assessment issues in Section 3.1 and the benefits of DL in Section 3.2, this section presents the framework of leveraging DL to support the damage assessment as an application of the IoTSE-based architecture. The IoTSE-based architecture consists of three major components: (i) Source of the data, (ii) IoTSE and (iii) DL applications, as shown in Figure 2. In the DL-IoTSE architecture, IoTSE will be considered to be hosted on edge or cloud data centre, where the related DL applications are running in the IoTSE at the application layer.

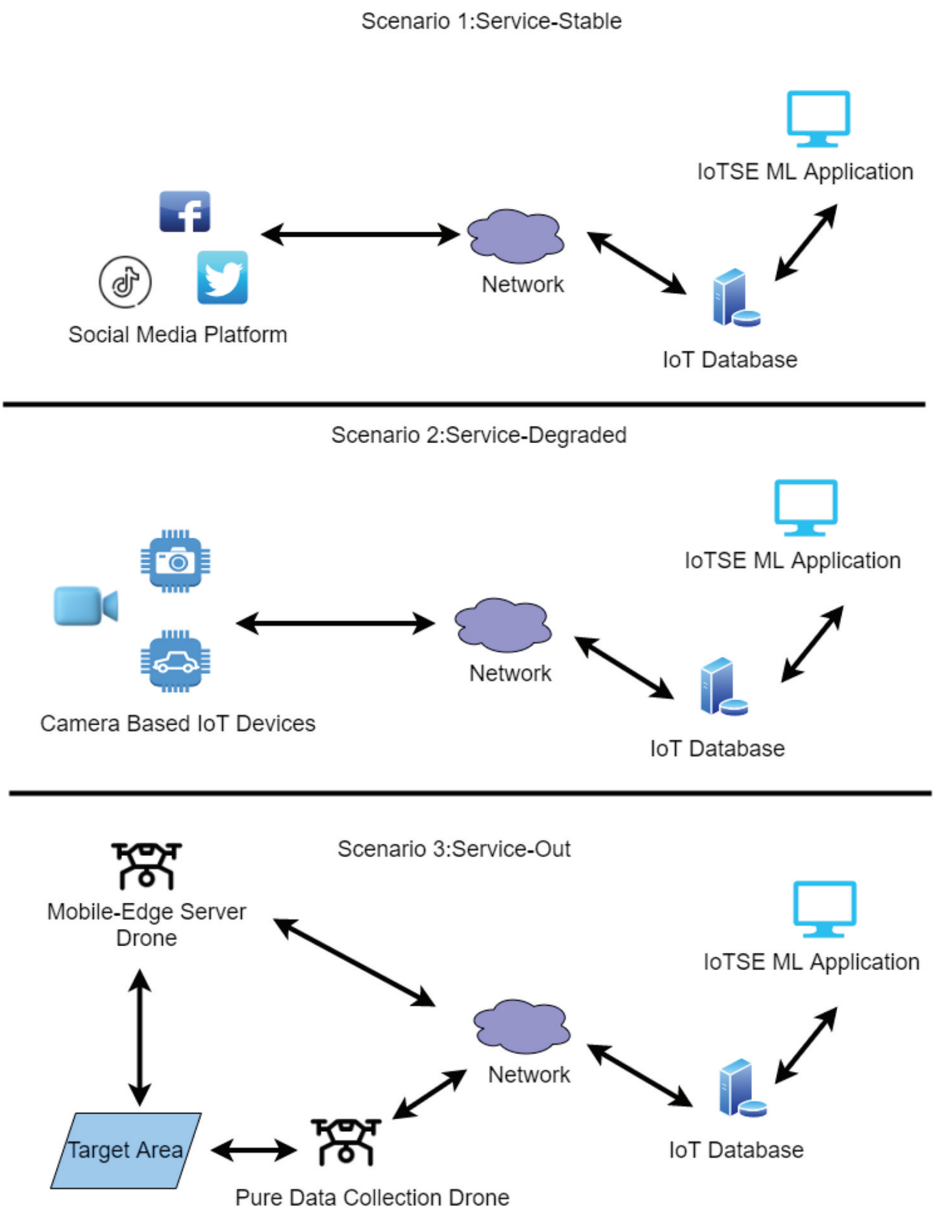


Figure 2. Scenarios of IoTSE for damage assessment.

In order to address the various network status updates at a specific location, the following three methods of data collection are used in the possible scenarios accordingly: (i) SERVICE-STABLE when the network environment of the requested area is operational and stable, IoTSE can utilise cross-crawling schemes in order to gather keyword-based information from social media platforms (Twitter, Facebook, etc.). (ii) SERVICE-DEGRADED, when people are located in a disaster zone with no access to the network, IoTSE can collect the related data from IoT devices in the city. This would be carried out by IoTSE querying data from servers, which are recently updated by smart cars, traffic cameras, and other cameras to find relevant data to complete the damage assessment. (iii) SERVICE-OUT, when the disaster zone is completely out of all services, which is designated as the worst case scenario, we can deploy dynamic network infrastructure deployment (e.g. drone-based wireless network) [17,57] to cover the area in order to gain the data and provide back to the IoTSE for further computing and analysis. Nonetheless, if the total damage is massive, meaning the data would not be gathered and sent back in a short time, drones in dynamic-based network infrastructure can be used as a mobile edge server to carry out local analysis and then provide the results back to the IoTSE.

4.2. Problem space

Based on the proposed IoTSE architecture (Figure 1), there are three aspects that could affect the efficacy of IoTSE to carry out damage assessments: (i) **Network Performance**: The design of the network topology, any limitations of bandwidth and the network protocol chosen could significantly affect the delivery latency of required data, the stability of the constructed network and the quality of the required data. (ii) **IoTSE Management**: When massive amounts of data streams are coming into the IoTSE, determining how to efficiently and swiftly store the data will be critical. When the upper layer of IoTSE requests a large amount of metadata from cross-data sets, how the IoTSE manages data access efficiently will also play an important role. Furthermore, there are a number of computing and network resources in IoTSE, how to efficiently conduct computing and network resource management jointly so that the overall performance of IoTSE can be maximised. (iii) **Machine Learning Performance**: The ML (especially DL) model size, training time and the power consumption, among others (hyper parameters of DL, the architecture of DL, etc.) will determine the ultimate performance of the learning model in certain damage assessment tasks.

Thus, the problem space of the IoTSE architecture can be generalised as a three-dimensional model, as represented in Figure 3. Each axis represents one of the three key indicators generalised from above. For this paper, the focus is on the performance of DL models.

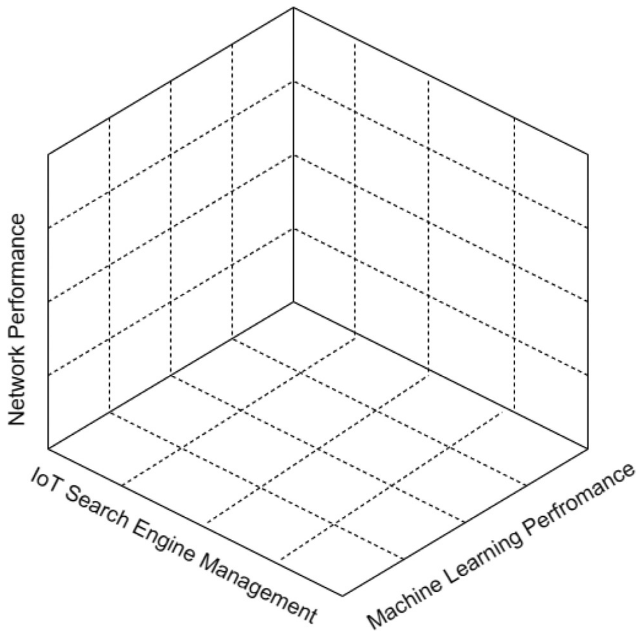


Figure 3. Problem space.

4.3. Data collection and human annotations

For our study, we utilise and leverage the labelled data of natural disaster from the Artificial Intelligence for Disaster Response (AIDR) [35] platform. AIDR completes web-crawls for images posted on social media during natural disasters and then manually marks these with labels of the damage level. We choose the following disaster events: California Wildfires, Hurricane Harvey [58], Hurricane Irma and Hurricane Maria [59]. Table 1 lists all the datasets and provides the total number of images that were initially collected in each.

The purpose of data curation is to be able to assess the severity of damage shown in an image. In these natural disaster cases, it is all about the physical destruction shown in the scene, including broken bridges, buildings that have smoke, cracked roads, etc. There are five levels of possible damage: (i) **Severe Damage**: The image shows major destruction of buildings or targets. (ii) **Mild Damage**: Only part of building is missing or damaged in that image but has some functionality. For example, only some levels of a building have lost electricity after an earthquake. (iii) **Little or No Damage**: In no-damage, the image shows

Table 1. Dataset for four disaster events.

Crisis Name	Number of Images
Hurricane Irma	4504
Hurricane Harvey	4434
Hurricane Maria	4556
California Wildfires	1589

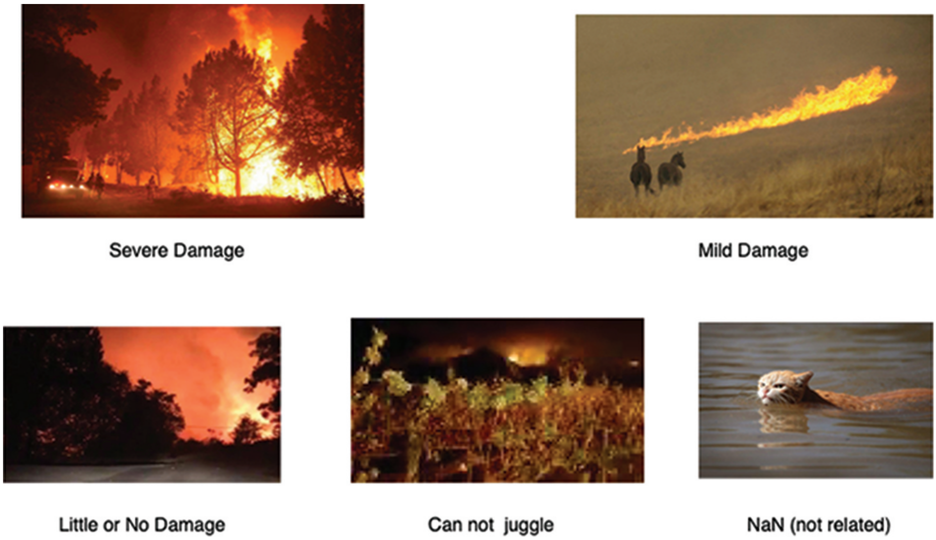


Figure 4. Examples of labelled images in California wildfires.

Table 2. Number of labelled images for each dataset.

Classes	California Wildfires	Hurricane Maria	Hurricane Harvey	Hurricane Irma
Severe Damage	465	509	556	316
Mild Damage	51	273	220	229
Little or No Damage	15	80	116	250
Can not judge	14	41	22	19
NaN (not related)	1044	3653	3520	3690
Total	1589	4556	4434	4504

buildings are targets that are damage-free. (iv) **Can not judge**: This case only presents some details of large target or area that does not reflect any meaningful information, such as taking a picture of a burnt branch that was near a building fire, but not of the actual fire. (v) **NaN (not related)**: The context in the image is not visually related to any kind of damage, such as the news reporting on damage in flooding. Figure 4 shows the examples of aforementioned damage levels.

Using the levels of damage defined above, the number of each level of damage that is contained in each dataset is presented in Table 2.

4.4. Damage assessment scenarios

In order to meet the requirement of damage assessments, we design a system with two aspects in mind, namely quality and similarity.

- *Quality of Image*: Since DL models rely on the information within an image, any images that have more pixels will provide the system with more useful information. In this case, the system should learn and adapt to the variety of image quality, while still providing results towards best efforts.

- *Resilience of Same Event but with Different Details in Context*: The system should be able to understand the same damage related information of similar events that have occurred in a different location. It is critical to be able to differentiate locations as most data sources come from social media and are not always clear about where and when the event occurred.

In the following, we design the two types of scenarios in order to evaluate the performance of the IoTSE system.

- **Single Event Setting**: For the Single Event scenario, two datasets are used from AIDR (Hurricane Harvey and California Wildfires) as shown in Table 3. Each dataset was run as separate test events in order for the DL model to train and test. With each event, the original pixel of the image is considered to be the baseline, and it is compared to the performance of the DL models that are run with different settings of pixels contained (75%, 50% and 25%).
- **Complex Event Setting**: In the Complex Event scenario, the labelled data that is provided is from a similar or related topic, but not the same event. Obtaining data of a labelled single-event disaster is not possible due to limited resources or budget. We carry out several complex-event experiments, in which the trained model is used in Hurricanes shown in Table 4. We consider the whole number of images for Hurricane Harvey as the baseline. We then use 60% of Hurricane Harvey and 60% of Hurricane Irma images as the first comparing group. Lastly, we combine 60% of Hurricane Harvey with 60% of Hurricane Irma and 60% of Hurricane Maria as the second comparing group. For each group, we provide the four different image quality pixels contained (100%, 75%, 50% and 25%).

5. Performance evaluation

We have conducted extensive performance evaluations to validate the efficacy of the IoTSE approach. In the following, we first introduce our evaluation methodology and then present evaluation results.

Table 3. Dataset in single event setting.

Event Combination	Name
Group 1	Hurricane Harvey
Group 2	California Wildfires

Table 4. Dataset in complex event setting.

Event Combination	Name
Group 1	100% Hurricane Harvey
Group 2	60% Hurricane Irma+ 60% Hurricane Harvey
Group 3	60% Hurricane Harvey + 60% Hurricane Maria + 60% Hurricane Irma

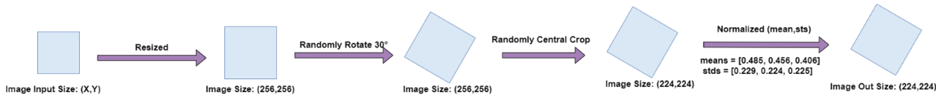


Figure 5. Data prepossessing.

5.1. Methodology

IoTSE uses Pytorch [60] as the platform to carry out the experiments. The following section introduces the dataset, set learning scheme and environments, and key performance indicators.

5.1.1. Data prepossessing

We use the following four steps shown in Figure 5 to preprocess the dataset by resetting image into 256*256, randomly rotating 30°, randomly central cropping into 224*224 and normalising image with certain mean value and standard deviation recommended by the Pytorch [61].

5.1.2. Learning schemes

Four pre-trained CNN-based deep learning models are used: Vgg19 [48], ResNet50 [49], GoogleNet [50] and MobileNetV2 [51]. These pre-trained models are well trained in ImageNet [62], which contain ten million images with over a thousand categories. For all these models, we re-train the last fully connected layer (noted as FC layer) with our own dataset.

5.1.3. Learning settings

For the fine-tuning training, we choose the backpropagation with mini-batch stochastic gradient descent with momentum [63]. The IoTSE experiments are set with a batch size of 64, momentum of 0.9 and fine tuning with a learning rate of 10^{-3}. For training each model, we only provide 25 epochs. The reason why we only provide 25 epochs is due to trying to simulate a training status that takes a limited time (such as disaster response), which is more fitting to the requirement of our framework for damage assessments. We then divide each dataset into two subsets: training (60%) and test (40%). We also use two Nvidia GeForce RTX 2080 Ti as our computing resource for training.

5.1.4. Key performance indicator

In order to evaluate the performance of the four defined scenarios, six metrics were considered: (i) **Training Time**: It records the training time after the 25 epochs for each model have been completed. (ii) **Best Training Accuracy**: Each training session keeps a recording of the best training accuracy, learning weight and replacing the current one. (iii) **Average Testing Accuracy**: The test session computes the average of whole results to measure the overall accuracy. (iv) **Average Precision and Average Recall in Testing**: The test session records each time test with precision and recall, computing an average result. (v) **Macro F1-Score**: Since this is a multi-classifying problem, macro F1 score would be a better metric to determine performance, based on the previous two metrics in testing. (vi) **Confusion Matrix**: which are the statistics to compute the precision-recall for each label learning perforce test, the true-positive rate and false-positive rate, as well as a matrix with the true label and prediction label as the axes.

5.2. Results

The following section analyzes the performance results in the single event setting, followed by performance results in the complex event setting.

5.2.1. Single event setting

California Wildfires Event Result: Table 5 shows the best training accuracy and time in the training of the models. The macro-averaged testing results show in terms of precision, recall, F1 score and the accuracy in the testing phase of the models. These results are for those DL models with different ratios of image quality in the data set. For the experiment, we use the least amount of data for the test event; which is the California Wildfires with almost 1589 images in Table 1. When the image quality increased, the accuracy of all models increased in training and testing. Nonetheless, these are slight improvements, as less than 2% each time, the image quality is increased, the accuracy increases. One possible reason is that there are not enough data for the training models.

Table 5. California wildfires event result.

Model	Training Part		Testing Part			
	Best Training Accuracy	Training Time	Precision	Recall	Accuracy	F1 Score
(a) 25% Original Image Pixel						
Vgg 19	0.767296	2 m 59s	0.380162	0.290064	0.745844	0.291375
ResNet 50	0.75	2 m 56s	0.29499	0.273354	0.734207	0.280902
Googlenet	0.734277	1 m 38s	0.287664	0.255341	0.715113	0.253755
MobileNet V2	0.764151	1 m 36s	0.286096	0.295903	0.750605	0.290608
(b) 50% Original Image Pixel						
Vgg 19	0.790881	3 m 46s	0.377322	0.319438	0.776549	0.318176
ResNet 50	0.795597	3 m 36s	0.296463	0.300933	0.768438	0.297597
Googlenet	0.781447	2 m 24s	0.287132	0.280202	0.743879	0.279209
MobileNet V2	0.789308	2 m 23s	0.357223	0.298582	0.763602	0.296436
(c) 75% Original Image Pixel						
Vgg 19	0.811321	5 m 12s	0.419531	0.309721	0.778615	0.306255
ResNet 50	0.790881	4 m 51s	0.314056	0.314682	0.776499	0.305996
Googlenet	0.792453	3 m 39s	0.289412	0.287729	0.751033	0.285976
MobileNet V2	0.789308	3 m 25s	0.306893	0.302338	0.768741	0.298715
(d) 100% Original Image Pixel						
Vgg 19	0.795597	6 m 21s	0.400919	0.31089	0.78005	0.304696
ResNet 50	0.801887	6 m 9s	0.402287	0.305873	0.772519	0.289224
Googlenet	0.812893	5 m 10s	0.292321	0.283006	0.752116	0.253283
MobileNet V2	0.794025	4 m 58s	0.402216	0.305187	0.770655	0.28926

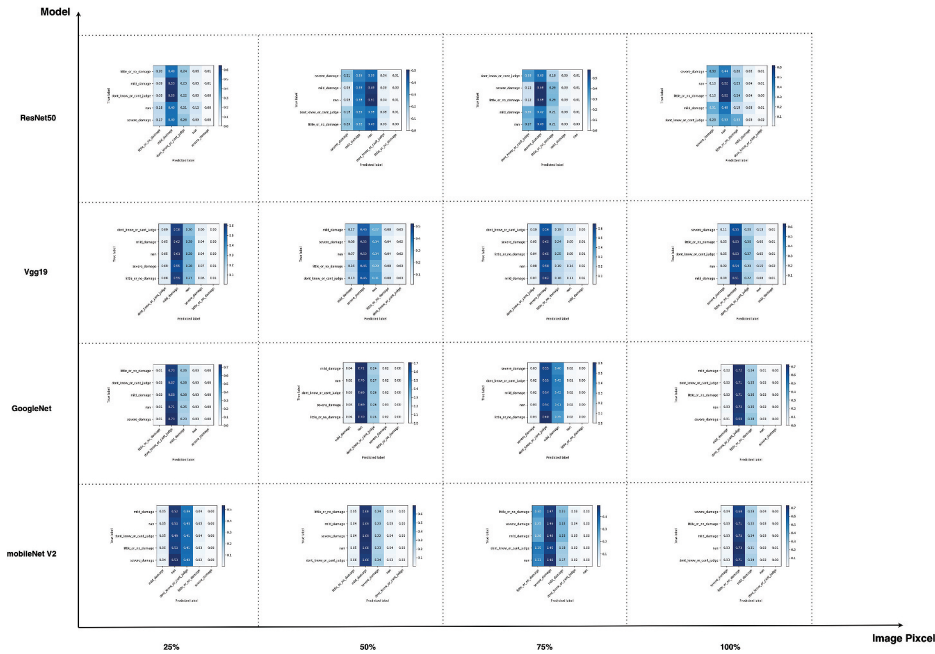


Figure 6. California wildfires event confusion matrix.

Figure 6 highlights that different models have different learning sensitivities for the labels in the testing as shown with one axis for the different models and the other axis is for the different ratios of image quality. Within each sub-block of the confusion matrix, the ‘y’ axis is the true label and the ‘x’ axis is the predicted label. The values then present how much the percentage of true label of this category is considered as the current category in the ‘x’ axis. In this dataset, the NaN (not related) category is the majority labelled; however, it is difficult for all the models to learn. Since most of the results for each model in each setting are less than 30%, we find that most models are able to mostly recognise the severe damage and the can-not-judge labels where the mild damage and the little-or-no damage are the most difficult to learn.

Hurricane Harvey Event Result: In this test event, we use a large data set, three times the size of the California Wildfires case. In Table 6, as compared to Table 5, the larger data set improves the accuracy of the training, showing all the models with over 80%. We found that the difference in training time is less for all models in each setting, even though the data size became three times larger than before. Nonetheless, we could find that the improvements in each model for different image quality settings are very limited. However, with limited epochs, those models could not learn the features well enough, even with a large data set.

Table 6. Hurricane Harvey event result.

Model	Training Part		Testing Part			
	Best Training Accuracy	Training Time	Precision	Recall	Accuracy	F1 Score
(a) 25% Original Image Pixel						
Vgg 19	0.828636	9 m 2s	0.420888	0.293119	0.820194	0.304696
ResNet 50	0.838219	6 m 12s	0.442552	0.278055	0.819436	0.289224
Googlenet	0.821871	5 m 11s	0.342001	0.250843	0.810972	0.253283
MobileNet V2	0.833145	5 m 7s	0.433586	0.280478	0.819499	0.28926
(b) 50% Original Image Pixel						
Vgg 19	0.844419	11 m 12s	0.445209	0.31392	0.831302	0.331795
ResNet 50	0.837655	8 m 58s	0.425043	0.315602	0.832384	0.326473
Googlenet	0.842728	7 m 48s	0.386692	0.267091	0.818624	0.273969
MobileNet V2	0.838782	7 m 32s	0.422264	0.304417	0.82633	0.312939
(c) 75% Original Image Pixel						
Vgg 19	0.843856	14 m 3s	0.457888	0.325421	0.834342	0.342907
ResNet 50	0.840474	12 m 23s	0.462851	0.319505	0.834036	0.332098
Googlenet	0.837655	11 m 10s	0.397578	0.282866	0.824516	0.287368
MobileNet V2	0.838219	11 m 18s	0.430023	0.309726	0.829632	0.320808
(d) 100% Original Image Pixel						
Vgg 19	0.851184	19 m 44s	0.443432	0.335784	0.835019	0.308308
ResNet 50	0.844983	16 m 25s	0.46543	0.324818	0.837401	0.302018
Googlenet	0.83991	16 m 27s	0.371912	0.278739	0.823659	0.282496
MobileNet V2	0.842728	15 m 49s	0.457641	0.312792	0.831852	0.302472

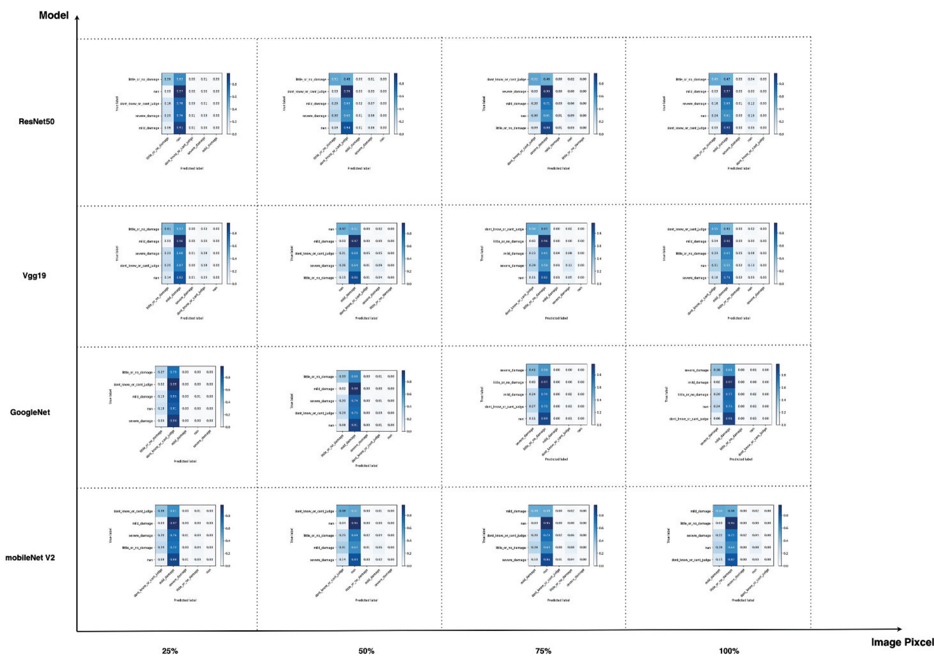


Figure 7. Hurricane Harvey event confusion matrix.

In Figure 7, the NaN label is the most difficult for VggNet19 and ResNet50 to learn. From the beginning, starting with 25% image quality setting, these two models are able to learn relatively well. When the image quality begins to increase, we find that the result decreased drastically. The performance of GoogleNet and MobileNet V2 both show that they are learning well and even when the image quality is increased to 50%, both models are able to maintain or improve the performance. Nonetheless, even though we continue to increase the image quality from 50% to 75% and from 75% to 100% the accuracy in these models continues to decrease. For this test event, we also discover that all the models are able to learn the labels of mild damage and severe damage well in all settings.

5.2.2. Complex event setting

60% Hurricane Irma and 60% Hurricane Harvey Event Result: In Table 7, all models except MobileNet V2 are able to recognise the same damage level for the two datasets and are getting better when the image quality increases. MobileNet V2, on the other hand, does not do well in the 25% setting, but began getting better when image quality increased to 50% and 75%. Nonetheless, MobileNet V2 begins getting worse in 100% setting. The reason for this shown in Figure 8, is that the majority label (NaN label) in dataset is not learned well. In detail, in the 25% setting, the result seen is 58% accurate, but increases to 88% in the 50% setting. In the 75% setting, the accuracy is at 84%, but drops to 65% when the image quality is running in the 100% setting.

In Figure 8, we find that the mild damage and severe damage label can be learned and are well recognised for all the models. The No-damage and can-not-judge label are hardest to learn, since these categories look less similar in different datasets.

Table 7. 60% Hurricane Irma and 60% Hurricane Harvey event result.

Model	Training Part		Testing Part			
	Best Training Accuracy	Training Time	Precision	Recall	Accuracy	F1 Score
(a) 25% Original Image Pixel						
Vgg 19	0.761194	25 m 59s	0.229338	0.230152	0.692657	0.229041
ResNet 50	0.791511	23 m 12s	0.222469	0.230764	0.701164	0.224603
Googlenet	0.816231	22 m 16s	0.192636	0.199168	0.768507	0.189456
MobileNet V2	0.616138	18 m 18s	0.200343	0.203582	0.495687	0.183711
(b) 50% Original Image Pixel						
Vgg 19	0.795243	28 m 59s	0.241613	0.238089	0.724448	0.23918
ResNet 50	0.716884	25 m 42s	0.243014	0.249308	0.674896	0.234426
Googlenet	0.806437	25 m 25s	0.199193	0.199783	0.761821	0.191519
MobileNet V2	0.810634	21 m 33s	0.204363	0.203974	0.721463	0.202552
(c) 75% Original Image Pixel						
Vgg 19	0.787313	30 m 40s	0.239876	0.246823	0.691119	0.24171
ResNet 50	0.817164	30 m 34s	0.234289	0.247481	0.700836	0.23503
Googlenet	0.79291	30 m 33s	0.211189	0.209797	0.747881	0.205734
MobileNet V2	0.812966	27 m 16s	0.223764	0.210058	0.694007	0.211615
(d) 100% Original Image Pixel						
Vgg 19	0.794776	37 m 27s	0.038539	0.234388	0.720754	0.235544
ResNet 50	0.807369	36 m 37s	0.242905	0.253674	0.727455	0.245896
Googlenet	0.821362	37 m 25s	0.194639	0.199275	0.776119	0.188762
MobileNet V2	0.55783	34 m 53s	0.207885	0.208551	0.553925	0.193469

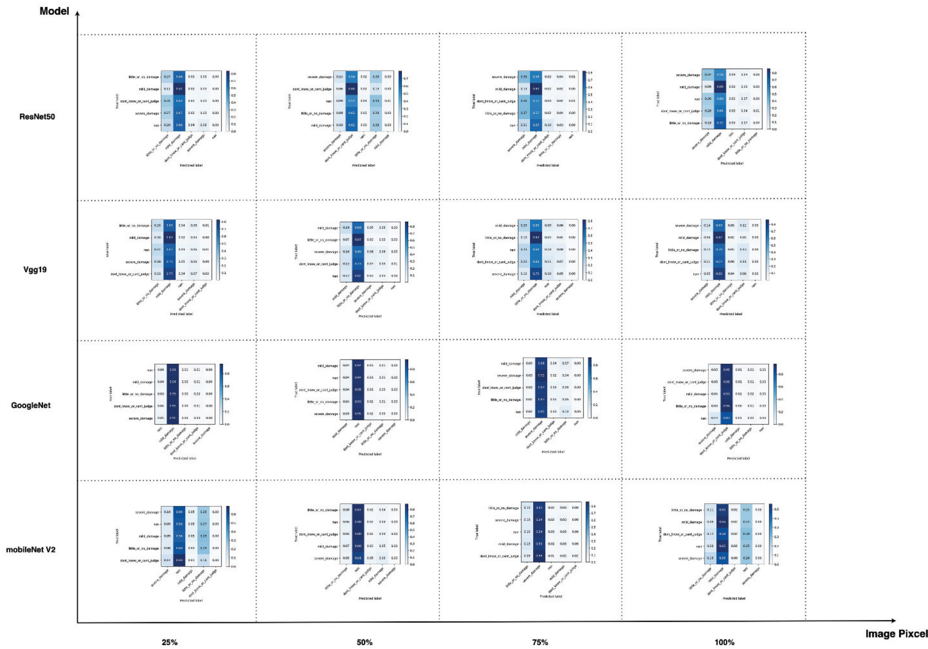


Figure 8. 60% Hurricane Irma and 60% Hurricane Harvey Event Confusion Matrix.

Table 8. Results on 60% Hurricane Harvey, 60% Hurricane Maria and 60% Hurricane Irma.

Model	Training Part		Testing Part			
	Best Training Accuracy	Training Time	Precision	Recall	Accuracy	F1 Score
		(a) 25% Original Image Pixel				
Vgg 19	0.546943	39 m 39s	0.220723	0.216382	0.448936	0.217579
ResNet 50	0.572884	34 m 29s	0.22699	0.235277	0.451037	0.228706
Googlenet	0.561458	35 m 22s	0.191525	0.197051	0.467649	0.190757
ModelNet V2	0.504324	26 m 29s	0.1953	0.197185	0.410335	0.193445
		(b) 50% Original Image Pixel				
Vgg 19	0.501235	44 m 28s	0.220723	0.216382	0.414679	0.224466
ResNet 50	0.367202	36 m 40s	0.22699	0.235277	0.392161	0.225728
Googlenet	0.575355	39 m 32s	0.191525	0.197051	0.4809	0.200014
ModelNet V2	0.581223	32 m 26s	0.1953	0.197185	0.465721	0.197299
		(c) 75% Original Image Pixel				
Vgg 19	0.55281	41 m 19s	0.234211	0.231856	0.459854	0.231559
ResNet 50	0.557134	45 m 20s	0.230168	0.247613	0.434632	0.231482
Googlenet	0.546634	54 m 9s	0.207774	0.202593	0.439778	0.195934
ModelNet V2	0.433601	40 m 41s	0.200774	0.206961	0.36734	0.187354
		(d) 100% Original Image Pixel				
Vgg 19	0.540766	53 m 27s	0.23401	0.233678	0.456889	0.232647
ResNet 50	0.562075	56 m 55s	0.240333	0.251584	0.448575	0.240366
Googlenet	0.553428	62 m 25s	0.204706	0.202685	0.487888	0.195857
ModelNet V2	0.575973	50 m 59s	0.202201	0.202386	0.479812	0.198143

60% Hurricane Harvey, 60% Hurricane Maria and 60% Hurricane Irma Event Result: In this result, three events are combined. The result at (25% setting) is very poor from Table 8 when compared with Figure 8 as all models at less than 60%. Even when the image quality is increased, most of the models remain the same. In some cases, they even become worse; such as ResNet 50 which drops from 46% in 25% setting to 39% in 50% setting during the testing portion. The training time also increases for all models in each setting.

From Figure 9, we find that all labels are learned poorly. It seems that it was difficult for the models to recognise the same damage level from three different datasets. We also find that the severe damage label is more recognisable than other labels, due to severe damage being easy to tell from context in each image. The mid-damage label is the hardest to learn for the models in this test.

Note that those experimental results are conducted under the following considerations. First, based on the existing pre-trained CNN models, we adopt the fine-tuning as our primary training strategy in all the scenarios and only retrain the last layer of all models. It provides a good base for further model re-training/tuning. Second, compared to text-based data, image is more robust to missing part of data as image itself has a lot of redundant information. This is the reason why the high compression ratio can be achieved when an image is compressed. Also, it is common practice that the context of an image can still be understood when missing even 75% of pixels. Because of this, we use 25% of image pixels contained as the threshold for the lowest image quality, ensuring that IoTSE can obtain

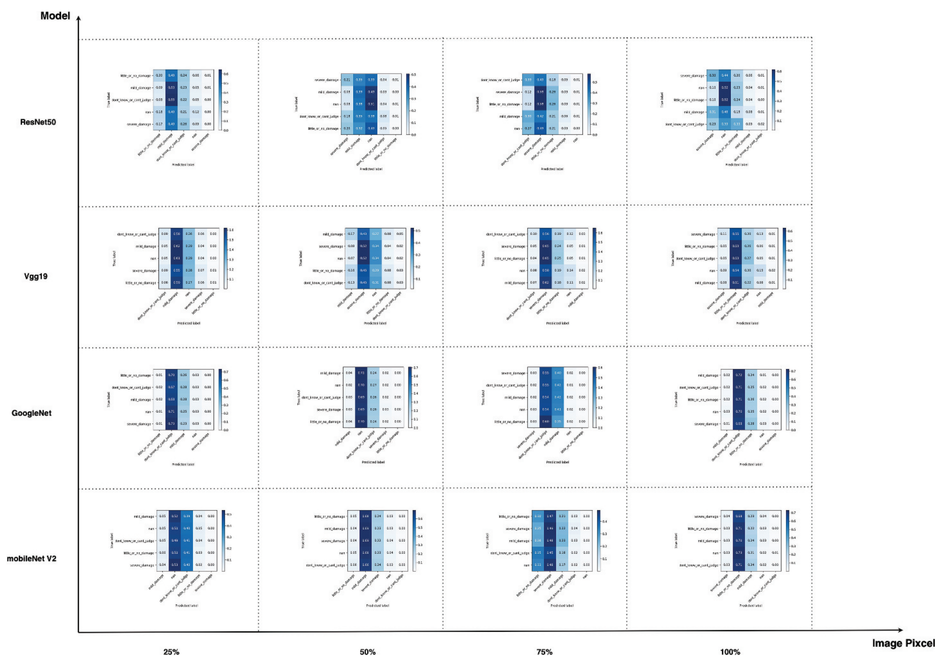


Figure 9. Confusion matrix with 60% Hurricane Harvey, 60% Hurricane Maria and 60% Hurricane Irma Event.

sufficient quality of image data for accuracy. We expect that when we further reduce the amounts of image pixels, the accuracy of learning model will be further declined.

6. Discussion

Potential future research directions for DL-based IoTSE for damage assessment concern performance and security.

- **Performance issue**: After running through the performance evaluation, we have determined that all of these models are less likely to recognise the same label in a different but same-topic event. There are two potential ways to improve this. One is utilising DL models that will be able to understand the same or similar context from different datasets. The other is designing a new data pre-processing scheme, which focuses on the similar topic dataset, making the same label data easier for the model to know whether they are the same thing. With respect to the network performance issue, since the whole IoTSE system may be operated over a constrained network environment, the more queries sent to IoTSE, more overhead occurs. To deal with a large number of queries (the cost for responding some queries could be high), we can not only develop the multi-class scheduling algorithms to handle queries with different priorities but also enhance the design of IoTSE architecture to reduce the time taken for processing queries [40 41]. In addition to use the traditional scheduling and optimisation techniques to efficiently manage resources in IoTSE, the new data-oriented network architectures such as Named Data Network (NDN) [64] could be considered.
- **Security issue**: IoTSE, as a critical infrastructure to carry out disaster assessment based on collected sensory data, it can be subjected to a variety of attacks. The adversary can launch attacks against the key components in IoTSE (e.g. data collection, data transmission and data analysis). For example, when the data collection process is compromised, the adversary could launch data poisoning attacks by injecting malicious image samples so that the accuracy of assessment decision of IoTSE can be affected. When the data transmission process is under attack, it will affect the availability of network transmission and the timely decision of IoTSE. Furthermore, the ML/DL as the essential part of data analysis component can be targeted by different attacks (e.g. model training/testing-based attacks, generative adversarial network (GAN)-based attacks, model function/performance-based attacks and others) [1,2,15,65]. Thus, it is critical to investigate the security of model training and testing process subject to different threats and

design corresponding countermeasures [1,66]. Also notice that since CNN models that we investigate in this paper did not understand the same label in different datasets well, there could be an attack, in which similar topic contexts are fed to the model so that the accuracy of learning model can be reduced. To deal with this issue, a Siamese network could be used as a data filter within the learning scheme 5.1.2. The Siamese network would be able to tell whether the incoming data is the required dataset for the training by comparing it with previous data.

7. Final remarks

This paper has addressed performance issues of damage assessment and proposed DL-based techniques to support an IoTSE-based solution. The paper has presented two types of scenarios for four popular DL models dealing with images that have different quality ratios. Our experimental results have confirmed that, in the single event setting, all the models were able to learn well for each label and with the increased dataset size and image quality, the better performance could be achieved. With the complex event setting, we have observed that the performance of the models has been reduced when learning the labels that are closely related topics. Future directions to enhance the performance and security issues with DL-based IoTSE to complete damage assessments include methods of domain adaptation and transfer learning to understand the same or similar context from different datasets.

Acknowledgments

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-20-1-0418. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the Air Force Office of Scientific Research [FA9550-20-1-0418].

References

[1] Liang F, Hatcher WG, Liao W, et al. Machine learning for security and the internet of things: the good, the bad, and the ugly. IEEE Access. 2019;7:158 126–158 147.

[2] Hatcher WG, Yu W. A survey of deep learning: platforms, applications and emerging research trends. IEEE Access. 2018;6:24 411–24 432.

[3] Capra M, Bussolino B, Marchisio A, et al. Hardware and software optimizations for accelerating deep neural networks: survey for current trends, challenges, and the road ahead. IEEE Access. 2020;8:225 134–225 180.

[4] Xu H, Yu W, Griffith D, et al. A survey on industrial internet of things: a cyber-physical systems perspective. IEEE Access. 2018;6:78 238–78 259.

[5] Liang F, Yu W, Liu X, et al. Toward edge-based deep learning in industrial internet of things. IEEE Int Things J. 2020;7(5):4329–4341.

[6] Yang X, Lingshuang K, Zhi L, et al. Machine learning and deep learning methods for cybersecurity. IEEE Access. 2018;6:35 365–35 381.

[7] Wang J, Varshney N, Zhang J, et al. Deep learning based link-level abstraction for mmwave communications. 2021 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI); 2021. p. 391–398.

[8] Liang Y, Cai Z, Yu J, et al. Deep learning based inference of private information using embedded sensors in smart devices. IEEE Network. 2018;32(4):8–14.

[9] Majumder U, Blasch E, Garren D. Deep learning for radar and communications automatic target recognition. IEEE. 2020. https://ieeexplore.ieee.org/document/9219995

[10] Brill TM, Munoz L, Miller RJ. Siri, Alexa, and other digital assistants: a study of customer satisfaction with artificial intelligence applications. J Marketing Manag. 2019;35(15/16):1401–1436.

[11] Usama M, Qadir J, Raza A, et al. Unsupervised machine learning for networking: techniques, applications and research challenges. IEEE Access. 2019;7:65 579–65 615.

[12] Xu H, Yu W, Liu X, et al. On data integrity attacks against industrial internet of things. 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech); 2020. p. 21–28.

[13] Liu X, Qian C, Hatcher WG, et al. Secure Internet of Things (IoT)-based smart-world critical infrastructures: survey, case study and research opportunities. IEEE Access. 2019;7:79 523–79 544.

[14] Yu W, Griffith D, Ge L, et al. An integrated detection system against false data injection attacks in the smart grid. Secur Commun Netw. 2015 Jan;8(2):91–109.

[15] Liu X, Yu W, Liang F, et al. On deep reinforcement learning security for industrial internet of things. Comput Commun. 2021;168:20–32. Available from: https://www.sciencedirect.com/science/article/pii/S0140366420320193

[16] Fitwi A, Chen Y, Zhu S, et al. Privacy-preserving surveillance as an edge service based on lightweight video protection schemes using face de-identification and window masking. Electronics. 2021;10(3):236. Available from: https://www.mdpi.com/2079-9292/10/3/236

[17] Yu W, Xu H, Nguyen J, et al. Survey of public safety communications: user-side and network-side solutions and future directions. IEEE Access. 2018;6:70 397–70 425.

[18] Imran M, Alam F, Qazi U, et al. Rapid damage assessment using social media images by combining human and machine intelligence. arXiv preprint arXiv:2004.06675. 2020.

[19] F. E. M. Agency. How communities and states deal with emergencies and disasters. p. 1–22.

[20] F. E. M. Agency. FEMA preliminary damage assessment guide. 2020. p. 1–127.
[21] Resch B, Usländer F, Havas C. Combining machine-learning topic models and spatio-temporal analysis of social media data for disaster footprint and damage assessment. Cartogr Geog Inf Sci. 2018;45(4):362–376.
[22] Barrington L, Ghosh S, Greene M, et al. Crowdsourcing earthquake damage assessment using remote sensing imagery. Ann Geophys. 2011;54:11.
[23] Kantarci B, Mouftah H. Trustworthy sensing for public safety in cloud-centric internet of things. IEEE Internet Things J. 2014 08;1:360–368.
[24] Liang F, Qian C, Hatcher WG, et al. Search engine for the internet of things: lessons from web search, vision, and opportunities. IEEE Access. 2019;7:104 673–104 691.
[25] Zheng X, Cai Z. Privacy-preserved data sharing towards multiple parties in industrial iots. IEEE J Sel Areas Commun. 2020;38(5):968–979.
[26] Mallapuram S, Ngwum N, Yuan F, et al. Smart city: the state of the art, datasets, and evaluation platforms. 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS); 2017; Wuhan, China. p. 447–452.
[27] Cai Z, Zheng X. A private and efficient mechanism for data uploading in smart cyber-physical systems. IEEE Trans Netw Sci Eng. 2020;7(2):766–775.
[28] Ekedebe N, Lu C, Yu W. Towards experimental evaluation of intelligent transportation system safety and traffic efficiency. 2015 IEEE International Conference on Communications (ICC); 2015; London UK. p. 3757–3762.
[29] Cai Z, He Z. Trading private range counting over big iot data. 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS); 2019; Dallas, Texas, USA. p. 144–153.
[30] Hatcher WG, Qian C, Gao W, et al. Towards efficient and intelligent internet of things search engine. IEEE Access. 2021;9:15778–15795.
[31] Blasch E, Ravela S, Aved A. Handbook of dynamic data driven applications systems. Springer; 2018.
[32] Liang H, Burgess L, Liao W, et al. Towards named data networking for internet of things search engine. 2021.
[33] Yu W, Le TN, Xuan D, et al. Query aggregation for providing efficient data services in sensor networks. 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No. 04EX975); 2004; Fort Lauderdale, Florida, USA. p. 31–40.
[34] Blasch E, Bossé E, Lambert D. High-level information fusion management and system design. 1st ed. USA: Artech House, Inc.; 2012.
[35] Artificial intelligence for disaster response (aidr). Available from: http://aidr.qcri.org/
[36] Mikołajczyk A, Grochowski M. Data augmentation for improving deep learning in image classification problem. 2018 international interdisciplinary PhD workshop (IIPhDW); IEEE; 2018. p. 117–122.
[37] Cheng G, Li Z, Han J, et al. Exploring hierarchical convolutional features for hyperspectral image classification. IEEE Trans Geosci Remote Sens. 2018;56(11):6712–6722.
[38] Li H, Ota K, Dong M. Learning iot in edge: deep learning for the internet of things with edge computing. IEEE Network. 2018;32(1):96–101.
[39] Tang J, Sun D, Liu S, et al. Enabling deep learning on iot devices. Computer. 2017;50(10):92–96.
[40] Qian C, Gao W, Hatcher WG, et al. Search engine for heterogeneous internet of things systems and optimization. 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech); IEEE; 2020. p. 475–482.

[41] Wang J, Rouil RA, Cintron FJ. Distributed resource allocation schemes for out-of-coverage d2d communications. 2019 IEEE Global Communications Conference (GLOBECOM); 2019; Hawaii, USA. p. 1–7.
[42] Bhattarai S, Wei S, Rook S, et al. Optimizing the location deployment of dynamic mobile base stations. 2015 International Conference on Computing, Networking and Communications (ICNC); 2015; Anaheim, California, USA. p. 579–583.
[43] Jarwan A, Sabbah A, Ibnkahla M, et al. LTE-based public safety networks: a survey. IEEE Commun Surv Tutorials. 2019;21(2):1165–1187.
[44] Tran DQ, Park M, Jung D, et al. Damage-map estimation using UAV images and deep learning algorithms for disaster management system. Remote Sens. 2020;12 (24):4169.
[45] Gu R, Niu C, Wu F, et al. From server-based to client-based machine learning: a comprehensive survey. ACM Comput Surv. 2021 Jan;54(1). DOI:10.1145/3424660.
[46] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436.
[47] Ryan L, Yulia A, Piotr S, et al. Image classification to support emergency situation awareness. Front Rob AI. 2016;3.
[48] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556. 2014.
[49] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition; 2016; Las Vegas, NV, USA. p. 770–778.
[50] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition; 2015; Boston, MA, USA. p. 1–9.
[51] Howard AG, Zhu M, Chen B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861. 2017.
[52] Xu H, Liu X, Yu W, et al. Reinforcement learning-based control and networking co-design for industrial internet of things. IEEE J Sel Areas Commun. 2020;38 (5):885–898.
[53] J. M. I. and M. T. M. Machine learning: trends, perspectives, and prospects. Science. 2015;349(6245):255–260.
[54] Liu X, Xu H, Liao W, et al. Reinforcement learning for cyber-physical systems. 2019 IEEE International Conference on Industrial Internet (ICII); 2019; Orlando, FL, USA. p. 318–327.
[55] Adi E, Anwar A, Baig Z, et al. Machine learning and data analytics for the iot. Neural Comput Appl. 2020;32(20):16 205–16 233.
[56] Munir A, Blasch E, Kwon J, et al. Artificial intelligence and data fusion at the edge. IEEE Aerosp Electron Syst Mag. 2021;36(7):62–78.
[57] Liang H, Gao W, Nguyen JH, et al. Internet of things data collection using unmanned aerial vehicles in infrastructure free environments. IEEE Access. 2020;8:3932–3944.
[58] Ofli F, Alam F, Imran M. Analysis of social media data using multimodal deep learning for disaster response. 17th International Conference on Information Systems for Crisis Response and Management, ISCRAM; 2020 May; Blacksburg, VA, USA. ISCRAM; 2020 .
[59] Alam F, Ofli F, Imran M. Crisismmd: multimodal twitter datasets from natural disasters. Proceedings of the 12th International AAAI Conference on Web and Social Media (ICWSM); 2018 June; Palo Alto, California, USA. 2018.
[60] Paszke A, Gross S, Massa F, et al. Pytorch: an imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, et al., editors. Advances in Neural Information Processing Systems 32. Curran Associates, Inc.; 2019. p. 8024–8035. Available from: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[61] Welcome to pytorch tutorials — pytorch tutorials 1.8.1+cu102 documentation. Available from: https://pytorch.org/tutorials/

[62] Deng J, Dong W, Socher R, et al. Imagenet: a large-scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition; Miami, FL, USA. Ieee; 2009. p. 248–255.

[63] Grinbaum A. Which fine-tuning arguments are fine? Found Phys. 2012;42(5):615–631.

[64] Zhang L, Afanasyev A, Burke J, et al. Named data networking. ACM SIGCOMM Comput Commun Rev. 2014;44(3):66–73.

[65] Cai Z, Xiong Z, Xu H, et al. Generative adversarial networks: a survey toward private and secure applications. ACM Comput Surv. 2021 jul;54(6):1–38.

[66] Chen Z, Tian P, Liao W, et al. Zero knowledge clustering based adversarial mitigation in heterogeneous federated learning. IEEE Trans Netw Sci Eng. 2021;8(2):1070–1083.