2022 International Conference on Multimedia Analysis and Pattern Recognition (MAPR) | 978-1-6654-7410-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/MAPR56351.2022.9924933

Performance Evaluation of Optimizers for Deformable-DETR in Natural Disaster Damage Assessment

Minh Dinh, Vu L. Bui, Doanh C.Bui, Duong Phi Long, Nguyen D. Vo, Khang Nguyen
University of Information Technology, Ho Chi Minh City, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam
{20521597, 20520350, 19521366}@gm.uit.edu.vn, {longdp, nguyenvd, khangnttm}@uit.edu.vn

***Abstract*—Global natural disasters are becoming more frequent and severe as a result of climate change. Recent advances in computer vision, particularly deep learning-based techniques and unmanned aerial vehicle (UAV) remote sensing, can aid disaster response teams in assessing the damage. Prior methods appear to be ineffective or were designed with inductive biases, making them difficult to conduct during the disaster damage assessment. In this paper, we investigate deep-learning-based methods capable of rapidly assessing building damage that follows natural disasters. Furthermore, we examine Deformable DETR, which is an improvement upon DETR, an object detection method based on the Transformer architecture, in terms of efficiency and convergence time, while inheriting DETR's simple implementation and adaptable architecture, making it suitable for the task of damage detection. We also experimented and analyzed the performance of several optimizers to improve the performance of Deformable DETR.**

***Index Terms*—Object Detection, UAV, Natural Disaster Damage Assessment**

I. Introduction

Natural disasters devastate many vulnerable areas worldwide. Rapid response to natural disasters is of the utmost importance in saving lives and reducing heavy economic losses. By examining the visual components of an image, various computer vision techniques could help with damage assessment. Researchers have used DCNN (Deep Convolutional Neural Network) to assess these natural hazards [1], [2]. These researches were largely concerned with detecting and assessing damaged buildings (such as homes, schools, coastal structures and other institutions). The details of damage level, such as the probability of the structure's collapsing or the severity of area's damage, are crucial for facilitating rapid response and large-scale recovery.

Object detection is a long-standing study area in the field of computer vision, intending to recognize objects that meet a preset categorization. Aerial image data has long been implemented in the task [3], [4]. Images captured from a bird-eye perspective will provide a panoramic view of the regions, assisting in search and rescue missions. Several natural disaster datasets [1], [2], [5], [6] have been introduced to enable the development of detection models. However, high-quality research on comprehensive solutions with long-term stability is still scarce in the task of natural disaster damage assessment.

Fig. 1. **The input and output sample of the modified Deformable DETR on ISBDA [1] damage assessment benchmark. In conjunction with the GPS trajectory of the drones, damage detection could aid in the timely response to disasters.**

In this paper, we examined the performance of Deformable DETR [7], a recently published end-to-end object detection method, on the ISBDA dataset [1]. Deformable DETR aims to eliminate numerous hand-designed components in the detection process, while with the use of Transformer architecture in object detection task, [8] delivers competitive performance and fast convergence. We also conducted experiments to analyze the performance of adaptive optimization methods for training Deformable DETR and bring a novel survey and analysis of the performance of the adaptive optimizers on the Deformable DETR model. Furthermore, we modified the Deformable DETR network using several alternate state-of-the-art architectures as the backbone and empirically evaluate the modified models on the challenging ISBDA dataset. Experimental results show that the top model achieved higher results with two times fewer training epochs in comparison to the original Deformable DETR and MSNET - the former SOTA on the ISBDA dataset.

The main contributions of our work can be summed up as:

- This paper provides an overview of publicly available datasets for natural disaster assessment and building damage detection.
- We implemented and evaluated the performance of the Deformable DETR model on the ISBDA dataset. We modified the model to enhance the model performance and reduce training time. Finally, we verified its potential in assessing natural damage.

Fig. 2. **Sample images of ISBDA dataset [1].** ***We recommend zooming in at around 200-300% to see the detail of images.***

- We surveyed and analyzed the performance of several adaptive optimizers in the training of Deformable DETR [9] where AdaBelief was the optimal choice and achieved the highest results.

II. Related Work

A. Natural Disaster Damage Assessment Datasets

Natural Disaster Damage Assessment datasets are classified into two categories [5]: (1) non-image datasets (text, tweets, and social media posts) and (2) image datasets. Current image datasets of natural disasters can be put into three categories: ground-level images, satellite images, and aerial images. Several datasets for assessing the damage caused by natural disasters have been compiled in recent research. Table I summarizes the findings of a comparative study of natural disaster datasets.

Existing disaster datasets rely heavily on low-resolution satellite imagery. As a result, it is difficult to assess the damage accurately. Unmanned aerial vehicles (UAVs) [5] can easily access difficult locations during a disaster and collect high-resolution images, which can provide detailed observations of each structure from various non-top-down perspectives. These images can be used as data to aid in computer vision tasks, most notably object [1]. Among the existing datasets, ISBDA (the first dataset proposed for damage assessment tasks from drone-perspective videos) appears to be the most appropriate for our research because it provides an object detection benchmark that can be used to evaluate models in assessing building damage in real-world environments.

B. Previous Approaches

There are several models that have been used for detecting natural disaster damage, including PolarMask [15], TensorMask [16] and Mask-RCNN [17]. PolarMask is an anchor-free single-shot instance segmentation method using only damage masks as input. PolarMask is faster than TensorMask and Mask R-CNN thanks to its simple pipeline. However, PolarMask network uses the SGD [18] optimizer in training, which results in a slow convergence rate. On top of that, this method does not perform well on distinct building instances. Mask R-CNN, an improved version of Faster R-CNN [19] with masked output, used to be the state-of-the-art segmentation method. However, in the particular task of damage assessment, it shows low performance when predicting the damage masks of large buildings or predicts incorrectly when images contain a high level of noise.

MSNet supports detecting damage in buildings with aerial video. To train more robust representations, MSNet uses hierarchical relationships between buildings and images and inter-frame spatial consistency of multiple viewpoints. The main module of MSNet is the Hierarchical Region Proposal Network (HRPN), in which two Region Proposal Networks (RPNs), namely high-level RPN and low-level RPN, share one single backbone network. High-level RPN lets the model detect which region proposal is a damaged building and which is not. Low-level RPN takes the output from high-level RPN and uses it to generate anchor sampling. MSNet has two main drawbacks. First is the lack of datasets that have the annotation format required for MSNet. MSNet's architecture is a combination of the pyramid backbone, Score Refinement Network, HRPN and Mask R-CNN, which fits perfectly into the training on ISBDA as the dataset is annotated with both fine-grained building and damage bounding boxes that work with MSNet. Second, the architecture of MSNet's HRPN causes the model to have a long training time and low inference speed, even on the ISBDA dataset.

Most previous research focuses mainly on segmentation - the task of assigning semantic labels at the pixel level. However, disaster response teams only need to know the locations and severity of damage in order to allocate their resources and distribute relief aid appropriately, which could be bettered by detecting the bounding box and assessment of the damage. Thanks to recent advances in deep learning, object detectors can efficiently assist in search and rescue operations by making precise damage detection and assessment. Therefore, we implemented and evaluated Deformable DETR - a detection method that can assess the building damage directly with better accuracy while maintaining a reasonable computational cost.

III. Our Approach

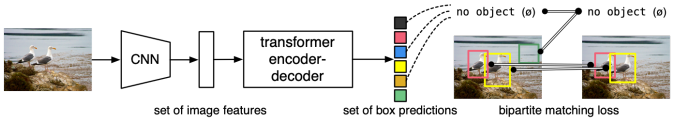


Fig. 3. **DETR [20] (CNN combined with Transformer), an end-to-end object detector, approaches object detection as a direct set prediction problem and thus eliminates the need for hand-crafted components (such as anchors, positive/negative samples selection and Non-maximum suppression (NMS)).**

TABLE I: A Brief Survey of Current Natural Disaster Databases.

Dataset	Size	Resolution	Image type	Task	# of Annotated Classes
ABCD [10]	22171	Varies	Satellite	Classification	2
SpaceNet + Deepglobe [11]	-	Varies	Satellite	Semantic Segmentation	2
fMoW [12]	1 million	Varies	Satellite	Classification	63
Rudner et al [13]	-	Varies	Satellite	Semantic Segmentation	2
xBD [6]	22068	1024x1024	Satellite	Instance Segmentation, Classification	4
ISBDA [1]	1030	-	Aerial (Social Media)	Object Detection	3
AIDER [14]	2545	-	UAV	Classification	4
FloodNet [5]	2343	3000x4000	UAV	Semantic Segmentation	9
RescueNet [2]	4494	3000x4000	UAV	Semantic Segmentation	11

A. *Deformable DETR*

Deformable DETR [7] is an end-to-end Transformer-based object detector improved upon the design of DETR [20] (depicted in Fig. 3). Deformable DETR is essentially identical to its predecessor in terms of architecture and processes, with a few modifications. Specifically, by combining the relation modeling capability of Transformer and the sparse spatial sampling of Deformable Convolution [21], some standard modules in the original DETR were upgraded to Deformable modules:

Deformable Attention Module: Unlike the Attention module in the original DETR, instead of looking at every possible spatial location on the feature maps, the Deformable Attention module only attends to a few key sampling points located around a reference point, without taking into account the size of the feature maps. This helps mitigate the convergence and spatial resolution issues that result in excessive resource consumption.

Multi-scale Deformable Attention Module: Due to the substantial benefit of multi-scale feature maps, many modern object detectors employ Feature Pyramid Networks (FPN) [22] for multi-scale feature extraction. Deformable DETR, on the other hand, employs the Multi-scale Deformable Attention modules, which are Deformable Attention modules extended for multi-scale feature maps, to aggregate multi-scale feature maps. Essentially, the Multi-scale Deformable Attention module works with the same mechanism as the single-scale version, except that it samples the key points on every scale of the multi-scale feature maps.

Deformable Transformer Encoder – Decoder: Follow the modification of the Deformable Multi-scale Attention module, the Transformer Encoder and Decoder have been enhanced to conform better to the tasks of Deformable DETR to work with multi-scale feature maps. The Encoder's Attention Modules that processes feature maps are replaced with Deformable Multi-scale Attention modules. Thus, both the input and output of the Encoder's are multi-scale feature maps with identical resolutions. A similar change goes for the Decoder, in which the Cross-Attention modules are replaced with Deformable Multi-scale Attention modules, whereas the Self-Attention modules are left unchanged.

Deformable DETR is both relatively more efficient and faster than its predecessor. As demonstrated in [7], Deformable DETR achieved superior performance with ten times fewer training epochs than DETR. Additionally, Deformable DETR has a FLOPs number close to that of Faster R-CNN and DETR-DC5 and has a training time lower by 25%, while achieving results comparable to many modern object detectors.

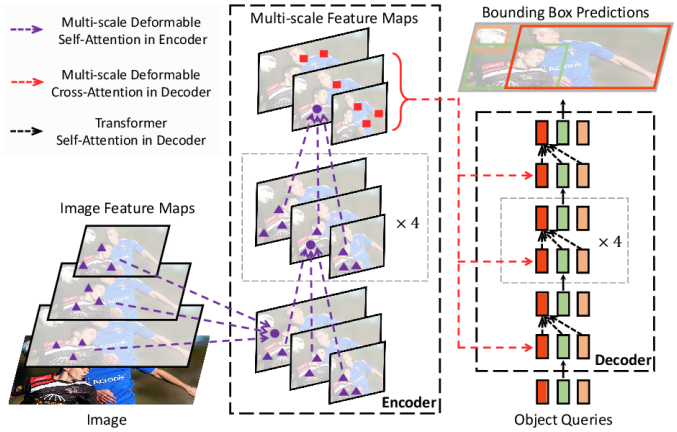


Fig. 4. **Illustration of the Deformable DETR [7] object detector with deformable modules.**

B. *Adaptive Optimization*

The most popular deep learning optimizers fall into two broad categories: adaptive methods (i.e. Adam [23], AdamW [7]) and accelerated schemes (i.e. stochastic gradient descent with momentum). Due to extremely lengthy training schedules and dropouts in Transformer, DETR and subsequently Deformable DETR have very slow convergence. To increase performance, several variants of adaptive optimizers were proposed and have shown improvement.

Adam [23] is the combination of RMSProp, an unpublished adaptive learning rate optimizer proposed by Geoff Hinton, and Momentum. Rather than adjusting the learning rates based on the average first moment as RMSprop does, Adam uses the gradient's average second moment. It calculates the exponential mean and square of the gradient. However, on a diverse set of deep learning tasks, Adam does not generalize as well as SGD with momentum.

AdamW [7] is the version of Adam that modifies the typical implementation of weight decay. Rather than L_2 regularization, it is implemented with the below modification where w_t is the rate of the weight decay at time t :

$$g_t = \nabla f\left(\theta_t\right) + w_t \theta_t$$

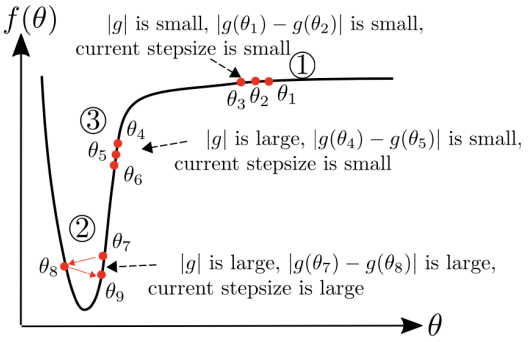


Fig. 5. **Intuitive idea of Adabelief: considering the curvature of the loss function, instead of taking a large (small) step following the gradient's size.**

AdamW decouples weight decay from the gradient update to exhibit significantly better generalization performance:

$$\theta_{t+1,i} = \theta_{t,i} - \eta \left(\frac{1}{\sqrt{\hat{v}_t + \epsilon}} \cdot \hat{m}_t + w_{t,i}\theta_{t,i} \right), \forall t$$

RAdam [24] stands for "Rectified Adam". The adaptive learning rate of Adam demonstrates a high degree of variance due to the training samples' limited model. To reduce such variance, RAdam uses smaller learning rates in the first few epochs of training - which justifies the warmup heuristic.

AdaBelief [9] is a version of Adam that adaptively caculates the variance value using the expected value of the gradient. In Adam, the update direction is $m_t/\sqrt{v_t}$, where m_t is the exponential moving average (EMA) of g_t, v_t is the EMA of g_t^2; in AdaBelief, the update direction is $m_t/\sqrt{s_t}$, where s_t is the EMA of $(g_t - m_t)^2$. As shown in Region 3 of Figure 5, the authors demonstrate AdaBelief's advantage over Adam in the "large gradient, small curvature" case. An ideal optimizer should consider the curvature of the loss function, rather than take a large (small) step where the gradient is large (small). In this case, $|g_t|$ and v_t are large, but $|g_t - g_{t-1}|$ and $|s_t|$ are small; this could happen because of a small learning rate α. In Adam, the denominator $\sqrt{v_t}$ is large, hence the stepsize is small, whereas AdaBelief takes a large step when observation g_t is close to prediction m_t (hence $\sqrt{s_t}$ is small) and a small step when the observation greatly deviates from the prediction.

EAdam [25] is the version of Adam that concentrates on the epsilon parameter. The modification in EAdam is adopting an adaptive ϵ rather than a constant.

Recent research in adaptive optimization functions treats the training of Transformer and object detection models as two separate tasks with different training strategies as well as hyperparameters tuning. In order to fill this gap, we worked on a performance analysis of multiple optimization functions. The result will help us determine which optimizer works best for training Deformable DETR and enhance the performance of Deforambale DETR on damage assessment.

IV. Experiment

This section provides detailed descriptions of how we implement Deformable DETR with different optimizers, including

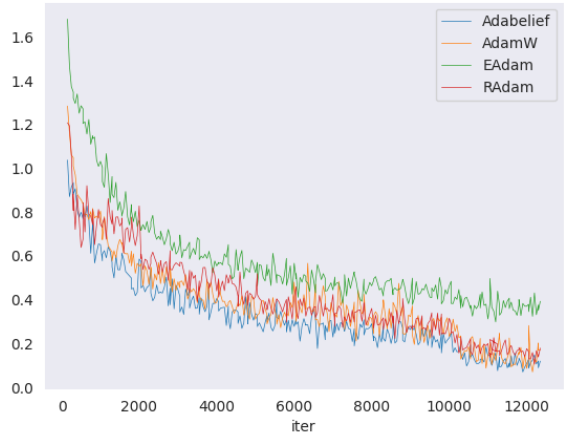


Fig. 6. **Training perplexity of optimizers on ISBDA, the lower is better.**

AdamW, RAdam, EAdam and Adabelief and thus achieve competitive results compared to MSNET and other state-of-the-art baselines, as well as quantitative evaluation on ISBDA. We provide code and pre-trained models to reproduce our experiments on our GitHub page [1].

A. Dataset

ISBDA [1] dataset is split as default into two subsets with three categories (Slight, Severe and Debris) and no overlapping scenes. Our models are trained on the train set (80% of the dataset) and tested on the validation set (20% of the dataset). Images are annotated with building-damage bounding boxes and masks. For training and evaluation, only bounding box annotations are used. If not specified otherwise, we report the AP as bounding box AP.

B. Evaluation Metrics

For evaluating detection on the ISBDA dataset and comparison between each trained model, we report the validation mAP (or simply called AP) at the last training epoch. The AP we used for evaluation is the modified COCO [26] metric reported in [1], with the pycocotools provided by the COCO API[2] modified following the official instructions on the author's GitHub repository.

We calculate the Average Precision scores of each category and take the average of them across all 6 IoU thresholds and all 3 categories as the last results (mAP). All the metrics we reported in this paper can be briefly explained as follow: AP denotes $AP^{IoU=0.25:0.05:0.5}$, AP@25 denotes $AP^{IoU=0.25}$ and AP@50 denotes $AP^{IOU=0.50}$.

C. Experimental Setting

We conducted our experiments on Google Colaboratory environment. MMDetection V2.23.0 framework is ultilized to implement (Deformable) DETR. We trained (Deformable)

[1]https://github.com/Banhkun/mmdetection
[2]https://github.com/cocodataset/cocoapi

TABLE II: Comparison of modified Deformable DETR with previous state-of-the-art detectors on the ISBDA dataset. In the Method column, **+Damage** denotes that the method takes only damage bounding boxes as input; **+Building+Damage** denotes that the method is co-trained with damaged buildings and damage bounding boxes, and thus significantly increases training cost. The bolded Methods indicate the ones with our proposed modification. The top results are highlighted in red font.

Method	Backbone	**AP**	**AP@25**	**AP@50**	**Total Epochs**
PolarMask+Damage	ResNet50	24.4	29.6	18.2	100
MaskR-CNN+Damage	ResNet50	35.9	40.9	29.4	100
MaskR-CNN+Building+Damage	ResNet50	34	40.3	25.7	100
MSNET+Building+Damage	ResNet50	38.7	44.4	31.5	100
DETR+Damage	ResNet50	29.1	32.2	25.9	150
Deformable DETR+Damage	ResNet50	32	35.4	28.5	50
Deformable DETR+Adabelief+Damage	**ResNet50**	39.2	44.7	33	50
Deformable DETR+Adabelief+Damage	**Res2Net50**	39.5	42.2	35.5	50
Deformable DETR+Adabelief+Damage	**ResNeSt50**	**44.65**	**43.6**	**36.7**	**50**

DETR with ResNet [27], Res2Net [28] and ResneSt [29] architecture for object feature extraction, and tested 4 different optimizers: AdamW, RAdam, EAdam and Adabelief. Settings and training strategies follow the official source code of DETR [20] and Deformable DETR. For optimizers, we referred to the work of Adabelief, EAdam, and RAdam to carefully perform hyperparameter tuning in our experiments, with the exception of AdamW being kept at default configuration.

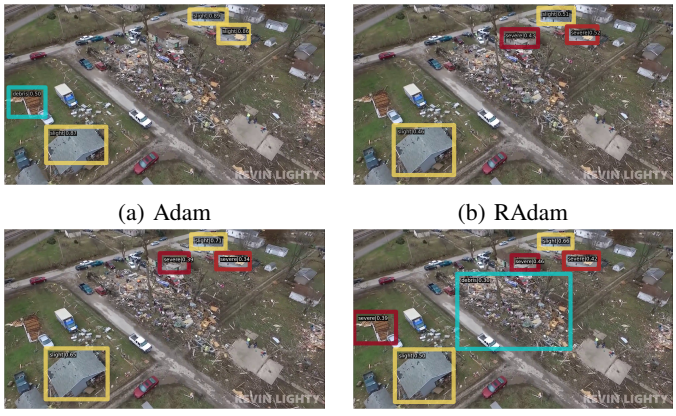
TABLE III: Comparison between four optimizers on (Deformable) DETR and various backbones. Adabelief outperforms other methods with the highest AP score (highlighted in **blue**) font. *Deformable DETR achieved a better AP score than DETR with the same backbone due to the introduction of deformable modules in Transformer.*

Model	Backbone	**AdamW**	**RAdam**	**EAdam**	**AdaBelief**
DETR	ResNet50	29.10	33.20	31.56	**36.70**
DETR	Res2Net50	30.48	33.21	31.60	**37.0**
DETR	ResNeSt50	32.54	35.28	32.56	**40.52**
Deformable DETR	ResNet50	31.80	35.61	32.06	**39.20**
Deformable DETR	Res2Net50	31.48	35.52	31.76	**39.50**
Deformable DETR	ResNeSt50	36.45	40.82	32.25	**44.65**

D. Results and Discussion

In this work, we compared the performance of Adabelief with other optimizers. The quantitive results in Table III of the modified (Deformable) DETR with various backbone architectures show that AdaBelief achieved the highest result. Furthermore, as shown in Figure 6, AdaBelief has a lower and more stable loss curve during training due to the mechanism of adaptively scaling the stepsize by the difference between the predicted gradient and the observed gradient, making it more efficient compared to other optimizers. Qualitative results shown in Figure 7 further explain quantitative results as in Fig. 7d., the model trained with AdaBelief has better generalization across all damage categories.

Moreover, we examined the performance of Deformable DETR with fine-tuned Adabelief compared to SOTA methods on the ISBDA dataset. Table II demonstrates the damage detection results. Deformable DETR with Adabelief outperformed previous SOTA methods, achieving better AP scores with only 50 training epochs and proved to be more cost-efficiency optimal, especially for DETR (150 epochs). On top of that, Deformable DETR generated significant improvement when utilizing the Resnest architecture for features extraction; the obtained results increased by 15.37% and 13.90% in comparison to MSNET and Deformable DETR with ResNet, respectively, indicating the improvement of the learned feature representations to boost the performance of ResNest.



(a) Adam (b) RAdam

(c) EAdam (d) Adabelief

Fig. 7. **Qualitative results of Deformable DETR with four optimizers. The yellow, red and green bounding boxes denote damages in Slight, Severe and Debris levels, respectively.**

V. Conclusion and Future Work

This paper discusses the issue of natural disaster relief following damage assessment using aerial images and deep learning methods. We survey several new optimizers and their uses in effective training of an end-to-end Transformer-based object detector - (Deformable) DETR. Finally, we evaluate the efficiency of various adaptive optimizers on (Deformable) DETR, in which Adabelief shows the highest degree of generalization of any optimizer. Our proposed modifications for Deformable DETR including the combination with Adabelief and the adoption of advanced models as the backbone (ResNet, Res2Net and ResNeSt) outperformed previous SOTA methods in the damage detection task. In future works, we aim at additional research into end-to-end object detection with transformers on more diverse aerial datasets and thus contribute to the development of deep learning models.

ACKNOWLEDGMENT

This research is funded by University of Information Technology-Vietnam National University Ho Chi Minh City under grant number D1-2022-28.

REFERENCES

[1] X. Zhu, J. Liang, and A. Hauptmann, "Msnet: A multilevel instance segmentation network for natural disaster damage assessment in aerial videos," *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 2022–2031, 2021.

[2] T. Chowdhury, R. Murphy, and M. Rahnemoonfar, "RescueNet : A High Resolution UAV Semantic Segmentation Benchmark Dataset for Natural Disaster Damage Assessment," vol. 14, no. 8, pp. 1–9, 2021. arXiv: arXiv:2202.12361v1.

[3] K. Nguyen, N. T. Huynh, P. C. Nguyen, K.-D. Nguyen, N. D. Vo, and T. V. Nguyen, "Detecting objects from space: An evaluation of deep-learning modern approaches," *Electronics*, vol. 9, no. 4, p. 583, 2020.

[4] T. V. Le, H. N. N. Van, D. C. Bui, P. Vo, N. D. Vo, and K. Nguyen, "Empirical study of reppoints representation for object detection in aerial images," in *2022 IEEE Ninth International Conference on Communications and Electronics (ICCE)*, 2022, pp. 337–342.

[5] M. Rahnemoonfar, T. Chowdhury, A. Sarkar, D. Varshney, M. Yari, and R. R. Murphy, "FloodNet : A High Resolution Aerial Imagery Dataset for Post Flood Scene Understanding," *IEEE Access*, vol. PP, p. 1, 2021. arXiv: 2012.02951.

[6] R. Gupta, B. Goodman, N. N. Patel, *et al.*, "Creating xbd: A dataset for assessing building damage from satellite imagery," *ArXiv*, vol. abs/1911.09296, 2019.

[7] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *ArXiv*, vol. abs/2010.04159, 2021.

[8] S. Khan, M. Naseer, M. Hayat, and S. W. Zamir, "Transformers in Vision : A Survey," pp. 1–30, 2021. arXiv: arXiv:2101.01169v5.

[9] J. Zhuang, T. Tang, Y. Ding, *et al.*, "Adabelief optimizer: Adapting stepsizes by the belief in observed gradients," *Adv. Neural Inf. Process. Syst.*, vol. abs/2010.07468, 2020.

[10] A. Fujita, K. Sakurada, T. Imaizumi, R. Ito, S. Hikosaka, and R. Nakamura, "Damage detection from aerial images via convolutional neural networks," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, 2017, pp. 5–8.

[11] S. Jegou, M. Drozdzal, D. Vázquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," Jul. 2017, pp. 1175–1183.

[12] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee, "Functional map of the world," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6172–6180.

[13] T. Rudner, M. Rußwurm, J. Fil, *et al.*, "Multi3net: Segmenting flooded buildings via fusion of multiresolution, multisensor, and multitemporal satellite imagery," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 702–709, Jul. 2019.

[14] C. Kyrkou and T. Theocharides, "Deep-learning-based aerial image classification for emergency response applications using unmanned aerial vehicles," Jun. 2019, pp. 517–525.

[15] E. Xie, P. Sun, X. Song, *et al.*, "Polarmask: Single shot instance segmentation with polar representation," Jun. 2020, pp. 12 190–12 199.

[16] X. Chen, R. Girshick, K. He, and P. Dollár, *Tensormask: A foundation for dense object segmentation*, Mar. 2019.

[17] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," Oct. 2017, pp. 2980–2988.

[18] S. Ruder, "An overview of gradient descent optimization algorithms," Sep. 2016.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, Jun. 2015.

[20] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. abs/2005.12872, 2020.

[21] J. Dai, H. Qi, Y. Xiong, *et al.*, "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 764–773.

[22] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," Jul. 2017, pp. 936–944.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[24] L. Liu, H. Jiang, P. He, *et al.*, "On the variance of the adaptive learning rate and beyond," *ArXiv*, vol. abs/1908.03265, 2020.

[25] W. Yuan and K.-X. Gao, "Eadam optimizer: How epsilon impact adam," *arXiv: Learning*, 2020.

[26] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," May 2014.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016. arXiv: 1512.03385.

[28] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 2, pp. 652–662, 2021.

[29] H. Zhang, C. Wu, Z. Zhang, *et al.*, "Resnest: Split-attention networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2022, pp. 2736–2746.