Robust FOD Detection using Frame Sequence-based DEtection TRansformer (DETR)

Xi Qin, Sirui Song, Jackson Brengman, Chris Bartone, Jundong Liu
School of Electrical Engineering and Computer Science
Ohio University
Athens, OH 45701, USA

***Abstract*—In this study, we develop a frame sequence-based transformer model for the automated detection of Foreign Object Debris (FOD) on airport runways. Our model integrates an LSTM network with a pre-trained DETR transformer to enhance detection robustness in terms of accuracy and consistence.**

Our approach captures short video sequences as input, using the encoder-decoder component of the DETR model to extract essential features. These features are then propagated through LSTM cells to incorporate temporal context. We explore various configurations of our proposed model and compare its performance with the baseline DETR. Experimental results demonstrate that our proposed model achieves significant improvements in both detection accuracy and consistency, showcasing its potential in enhancing safety on airport runways.

***Index Terms*—FOD detection, Transformer, LSTM, DETR**

I. Introduction

Foreign Object Debris (FOD) detection is essential for airport safety. FOD includes any foreign objects or materials on the airport surface or runway that have the potential to cause damage. Such objects vary from tools and aircraft parts to rocks, debris, and wildlife. Timely detection of FOD is critical as it helps prevent aircraft damage, preserves runway integrity, increase aviation safety, and reduces flight delays and cancellations, among other benefits.

Traditional solutions for FOD detection include manual inspections, patrol vehicles, sweeping machines, and magnetic bars. Some automated approaches include the use of radar systems and Closed-Circuit Television (CCTV) [1], [2]. However, some radar-based systems are challenged with detecting smaller items and require significant cost to install and maintain around a runway. CCTV commonly requires constant human monitoring, whose effectiveness is limited by visibility conditions.

Over the past decade, computer vision solutions, particularly those utilizing deep neural networks (DNNs), have been increasingly adopted for FOD detection [3]–[6]. Solutions based on Convolutional Neural Networks (CNN), such as *you only look once* (YOLO) and its variants [7]–[12], were among the initial approaches explored. In 2017, transformers, initially developed for Natural Language Processing (NLP) tasks, were introduced [13]. Subsequently, the use of transformers in computer vision became more prevalent, particularly in tasks like image classification [14] and object detection [4], [15], [16]. Transformers offer advantages such as global context understanding and an unlimited input size, making them more flexible and accurate compared to CNNs.

The existing DNN-based solutions for FOD detection commonly rely on single input images for their decision-making process. Such approaches tend to overlook contextual information over time, which can result in inconsistent and unstable detection outcomes. In this regard, FOD detection using 3D videos would be more advantageous, as it naturally captures all spatial contexts. However, this benefit comes with a heavy computational cost, especially for vision transformers.

In this paper, we develop a frame sequence-based transformer model for automatic FOD detection. Specifically, we combine a Long Short-Term Memory (LSTM) network with a pre-trained DEtection TRansformer (DETR) model [15], a state-of-the-art detection transformer, to enhance both the accuracy and frame-wise consistency of the network for robust detection performance. Our model captures short video sequences as the input and utilize the encoder-decoder part of the DETR model to extract features from the video frames. The LSTM is used to propagate these features over time, enabling the most critical features of each frame to be shared and distributed throughout the sequence of frames. To the best of our knowledge, this is the first work to utilize LSTM + DETR for FOD detection.

The data in this study were acquired at Ohio University Gordon K. Bush Airport (KUNI, https://www.ohio.edu/airport) in Sept. 2022. Fig. 1 shows the trajectory of our data acquisition van (in blue color), overlaid onto a geospatially matched Google Maps patch.

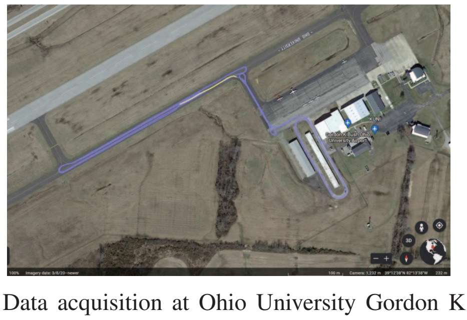


Fig. 1: Data acquisition at Ohio University Gordon K. Bush Airport (KUNI).

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00218

II. Background

A. RNN and LSTM

Recurrent Neural Networks (RNNs) are designed to process sequences of data, making them particularly well-suited for tasks that require understanding and integrating historical information. A RNN maintains a loop 'memory' cell, which allow the information from pervious inputs to be carried over from one step of the network to the next.

LSMT is a special type of RNN that is capable of learning long-term dependencies. Introduced to overcome the vanishing gradient problem found in the traditional RNNs, LSTMs have a more complex computational unit that includes three gates (input, output, and forget) and a cell state. This architecture enables them to not only process individual data points (such as words in a sentence) but also entire sequences of data.

B. Transformer

The *transformer*, introduced in a seminal paper [13], marked a significant shift in the approach to sequence-to-sequence tasks, commonly found in NLP applications like translation and text summarization. The transformer model is characterized by its use of self-attention mechanisms, which allows the model to weigh the relative importance of different parts of the input data.

The self-attention mechanism works by generating three vectors from each input in the sequence: Query (Q), Key (K), and Value (V). The query vector is used to determine the level of focus that should be put on other parts of the input sequence. The key vector is matched against the query in the attention mechanism, while the value vector represents the actual value that is used to construct the context. In the original paper, the authors use a *linear* layer and a *softmax* to predict the probability distribution over the vocabulary for next token, which can be described as follows:

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^\intercal}{\sqrt{d_k}})V \quad (1)$$

The transformer model has been extensively adapted for applications beyond NLP. Typically, a transformer consists of an encoder and a decoder module. The inputs to a transformer are embeddings, which are high-dimensional vector representations that encode the semantic information of the original input. In NLP tasks, these embeddings are often word embeddings, while in computer vision tasks, image embeddings are utilized. These embeddings are usually generated by other models. For example, Word2Vec [17] is a common model for word embeddings, and ResNet [18] is frequently used for image embeddings. In addition, positional encoding is applied to incorporate positional information into the input, enabling the model to understand the sequence order of the data.

III. Method

In this research, we began with a DETR model that was pretrained on the Microsoft *Common Objects in Context* (COCO) dataset [19]. We then fine-tuned this model with our own FOD dataset, using it as the baseline for our research. DETR, different from models like YOLO and its variants [7], [8], is a foundation detection model built on a transformer architecture. This design choice eliminates the need for predefined anchor boxes, directly predicting bounding boxes and class labels.

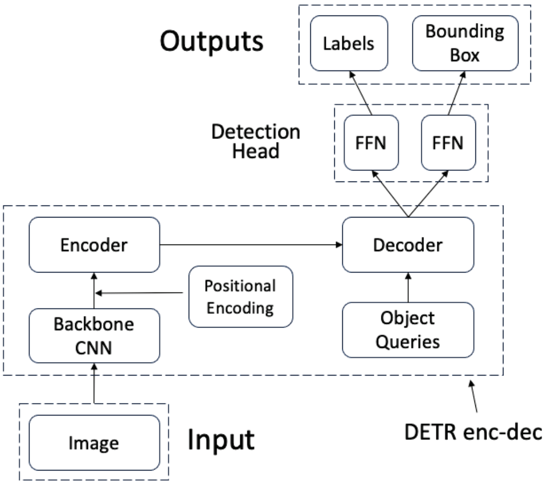


Fig. 2: An illustration of DETR's architecture. "DETR enc-dec" denotes the encoder-decoder component.

DETR follows an encoder-decoder architecture, similar to the original NLP transformer model, as illustrated in Fig. 2. Inputs to the network are images, which are first processed through a backbone CNN, usually ResNet-50 or ResNet-101, to extract features. The network's encoder and decoder comprise multiple layers of attention modules and Feedforward Networks (FFNs) to carry out the self-attention mechanism. DETR introduces the concept of *object queries*, which are learnable representations enabling the model to identify objects. A fixed number of learned positional embeddings, 100 as per the original design, inform the model about the spatial relationships between different parts of an image. The output from the decoder is then processed by the detection head, which includes two FFNs to produce the network outputs. One FFN is responsible for predicting class labels, while the other predicts bounding boxes.

In Fig. 2, we label the encoder-decoder component as ”DETR enc-dec.” This component forms the foundation of our proposed detection model, which will be explained in the next section.

A. Proposed sequence based DETR models

Our goal is to enhance the DETR detector by incorporating temporal information, enabling it to make detection decisions with context awareness. This addition is expected to improve the robustness and consistency of the model's output. To achieve this, we propose **seqDETR**, a detection model that connects multiple DETR enc-dec units using an LSTM network.

The integration of LSTM in our model is conducted by positioning LSTM cells directly after the decoder of DETR. In contrast to DETR, which processes individual 2D images, our seqDETR handles a sequence of video frames. Each frame, along with a small preceding set of frames, is input into to a DETR encoder-decoder (enc-dec) model. This model is essentially the DETR component minus the detection head, as shown in Fig. 3. The outputs from this frame sequence create a series of tensors, which are then fed into the LSTM cells. The LSTM is capable of retaining historical or temporal information from this sequence. To create a residual connection and ensure robustness, the output of the LSTM is combined with the output of the current frame, especially as a fallback if the LSTM does not effectively learn. This combined output is then fed into the detection head to produce the final result.

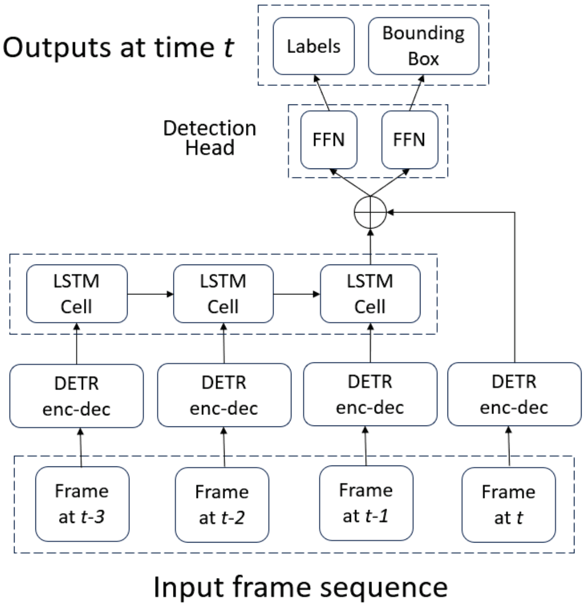


Fig. 3: An illustration of the architecture of the proposed seqDETR model.

In practice, we do not feed each frame and its corresponding mini sequence into the model every time. To simplifying the training process and computational complexity, we sequentially feed each frame but always store the outputs of the previous k frames from the decoder. We explored $k = 3$ and $k = 5$ in our implementations and experiments, leading to the development of the corresponding seqDETR models, named **seqDETR-3** and **seqDETR-5**, respectively. This approach allows us to use the results stored in memory for previous frames, rather than recalculating the LSTM input for each frame. This leads to significant savings in both GPU memory and computational power. As a result, the average inference time of **seqDETR-3** and **seqDETR-5** are 6.36 frames per second (fps) and 6.33 fps respectively, which are close to the 6.37 fps inference time of fine-tuned DETR.

In this work, our models are all trained and evaluated on a system equipped with an Intel Xeon W-2255 CPU (3.7 GHz, 10 cores), 128 GB of RAM, and two NVIDIA RTX A5000 GPUs.

IV. Experiments and Results

In this section, we present and evaluate the experimental results for the proposed seqDETR model. We used three different performance metrics commonly used on the Microsoft COCO dataset – mAP, mAP50, and mAR100 – to measure detection accuracy and consistency. mAP represents the mean Average Precision across all classes, calculated at 10 different *Intersection over Union* (IoU) thresholds, ranging from 0.50 to 0.95, and is also known as mAP50-95. mAP50 measures the average precision at a 50% IoU threshold across all classes. mAR100 denotes the mean Average Recall given 100 detections per image, reflecting the model's ability to identify all relevant instances. Generally, mAP scores indicate the accuracy of the model, while mAR scores show its comprehensiveness in detecting instances.

In addition, to evaluate the consistency of the detection results, we employ the average *standard deviation* (STD) of the bounding box widths and heights across various types of FOD. This metric assesses the variance of box dimensions between frames, with lower values indicating greater consistency.

A. Data used in our experiments

The data for our experiments was collected using a van equipped with a GoPro 10 camera for video capture and a Honeywell GPS/INS n380 to gather corresponding GPS data. We analyzed footage from two different trips, referred to as Trip A and Trip B. Both trips recorded multiple objects on an airport taxiway surface, providing a diverse dataset for our study. Trip A occurred around 6pm, while Trip B took place at approximately 7pm, with each covering different sections of the airport's taxiway.

Trip A consists of 638 image frames, while Trip B contains 493 images. Our experiments make use of eleven different types of FOD, including items such as a screwdriver, flashlight, cell phone, scissors, and washers, as illustrated in Fig. 4. Due to the relatively small size of our dataset, we have categorized all FOD types into a single class. To enhance our dataset, we applied simple data augmentation techniques including flipping, translation, and rotation.

B. Results of our experiments

To evaluate the models' performance, we carry out two different training and testing splits: *Setup* I) Using Trip A data for training and Trip B data for testing; and *Setup* II) Using Trip B data for training and Trip A data for testing. Three distinct experiments are conducted for each setup: 1) FOD detection using a fine-tuned DETR model on individual video frames, which is the baseline model in this work; 2) Detection using our proposed seqDETR with mini-sequences of three frames (seqDETR-3); and 3) Detection using seqDETR with mini-sequences of five frames (seqDETR-5).

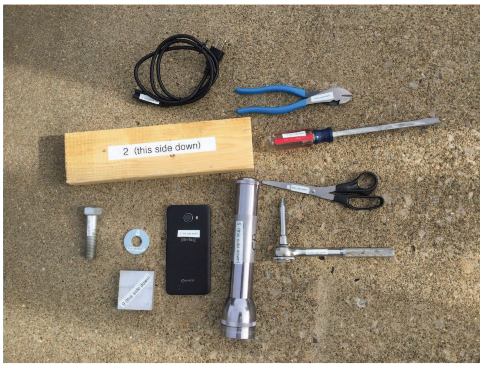


Fig. 4: FOD types used in our experiments.

TABLE I: FOD detection performance of different models.

Setup	Model	mAP	mAP50	mAR100
Setup I	Fine-tuned DETR	0.4287	0.7807	0.5854
	SeqDETR-3	**0.4553**	**0.8391**	0.6117
	SeqDETR-5	0.4432	0.7982	**0.6613**
Setup II	Fine-tuned DETR	0.3661	0.6686	0.6203
	SeqDETR-3	**0.3894**	**0.6848**	**0.6264**
	SeqDETR-5	0.3334	0.6263	0.5815

Table I presents the quantitative results of the competing models based on the selected COCO metrics. Our proposed seqDETR models outperform the baseline DETR in all experiments. In setup I, seqDETR-3 shows improvements of 6.20%, 7.48%, and 4.49% over the baseline model, while seqDETR-5 yields improvements of 3.87%, 2.24%, and 12.97%. In setup II, seqDETR-3 again outperforms the fine-tuned DETR, registering improvements of 6.36%, 2.42%, and 0.98%. However, for seqDETR-5, there is a deterioration of 8.93%, 6.33%, and 6.25% compared to the fine-tuned DETR. For both setups, seqDETR-3 performs the best on mAP and mAP50. SeqDETR-5 shows better performance on mAR100 in setup I while seqDETR-3 performs better on mAR100 in setup II. A possible explanation for 3-frame models generally outperforming 5-frame models is that the 5 frames may have overwhelmed the LSTMs, leading to decreased accuracy.

Table II shows the quantitative results based on the consistency metric. Our proposed seqDETR models demonstrate greater consistency and robustness than the baseline model, evident from the smaller variances observed over the detected bounding boxes. In setup I, seqDETR-3 and seqDETR-5 show improvements of 9.64% and 8.28%, respectively, over the baseline model. In setup II, improvements of 76.63% and 10.42% are observed for seqDETR-3 and seqDETR-5, respectively, compared to the fine-tuned DETR. seqDETR-3 shows smaller box variance than the ground truth in setup II, which may be attributed to two reasons: 1) the ground truth is labeled by humans, which inevitably introduces inconsistency, and 2) the test data in setup II, Trip A, may be less challenging than Trip B, as in setup I.

TABLE II: Bounding box variance comparisons: smaller box STD values indicate more stable detection performance.

Setup	Model	Box STD
Setup I	Ground Truth	12.2863
	Fine-tuned DETR model	14.8394
	SeqDETR-3	13.6100
	SeqDETR-5	**13.4093**
Setup II	Ground Truth	13.4672
	Fine-tuned DETR model	17.9285
	SeqDETR-3	**4.1897**
	SeqDETR-5	16.0606

Fig. 5 presents a visual comparison across 20 consecutive frames, which contain a screwdriver in a number of middle frames. A zoomed-in view of the screwdriver can be found in Fig. 6.(a). We stack these frames into a single image, highlighting the detected objects. Fig. 5.(a) displays the results from the baseline model, while (b) and (c) show the results of seqDETR-3 and seqDETR-5, respectively. The detected bounding boxes are shown as blue rectangles. It is evident that seqDETR-3 outperforms the others in terms of the number of detections and bounding box variance, aligning with the evaluation results. Notably, seqDETR-3 effectively detects FOD in various lighting conditions, including both light and shadow areas. In contrast, seqDETR-5 misses some FOD instances in certain frames, and the baseline model has an even higher miss rate, particularly when the FOD is in shadowed areas.

Fig. 6 provides a close-up view of two consecutive frames from Fig. 5, highlighting the impact of lighting changes due to shadows on detection. The baseline model in Fig. 6.(a) struggles with detecting the screwdriver in these conditions. In contrast, seqDETR-3 and seqDETR-5 successfully detect the FOD in both frames, demonstrating its robustness against variations in lighting and shadow. In summary, the experimental results demonstrate that the seqDETR models have successfully achieve our design goals.

V. Conclusions

In this research, we present a transformer based FOD detection model, seqDETR, for airport surfaces including runway and taxiway surfaces. This model processes daily inspection video footage as input. We modify the DETR model and incorporate an LSTM to aggregate temporal information from previous frames. Various training and testing scenarios, as well as model setups, were explored and analyzed. The modifications led to significant improvements in both mAP and bounding box variance. These results underscore the success of our modifications and demonstrate the potential of seqDETR in FOD detection. To improve seqDETR's adaptability, we plan to include more datasets covering various lighting and weather conditions in future work.

VI. Acknowledgment

This research is sponsored by the Federal Aviation Administration, Airport Technology Research and Development, ANG-E26.

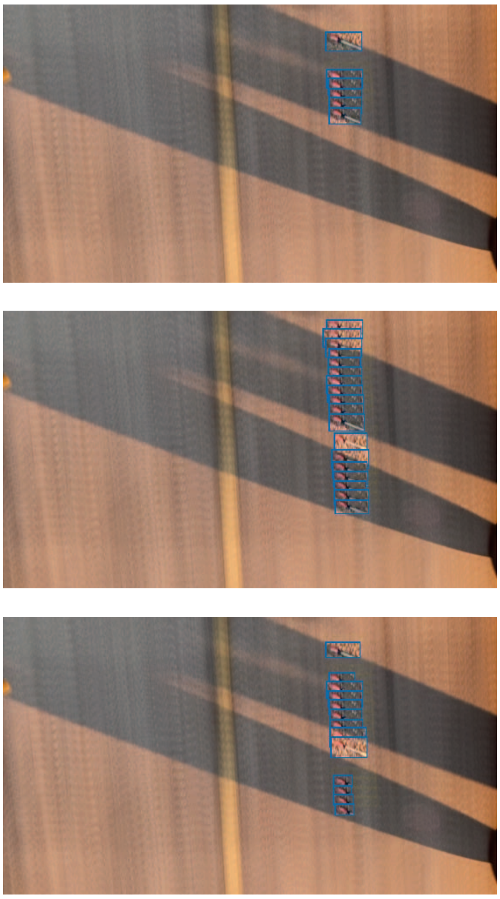
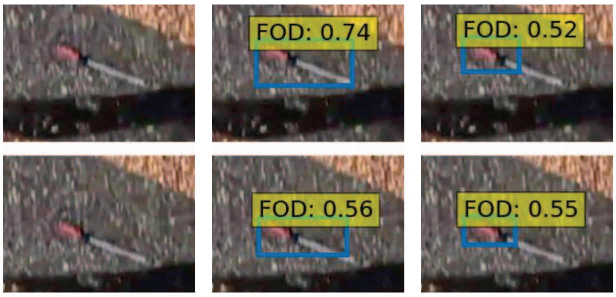


Fig. 5: Stacked detection results across 20 consecutive frames. a) first row: results from the baseline fine-tuned DETR model, b) second row: from seqDETR-3 and c) third row: from seqDETR-5.



(a) Baseline model (b) seqDETR-3 (c) seqDETR-5

Fig. 6: Zoomed-in view of the detection results of the three models for two consecutive frames.

References

[1] Jun Wang, Xueyin Geng, and Shaoming Wei, "Airport Runway FOD Detection System Based on 77GHz Millimeter Wave Radar Sensor," in *2019 IEEE ICTA*, Chengdu, China, Nov. 2019, pp. 140–143, IEEE.

[2] Saleh AlYahyai, Abid Khan, Mohamed Siyabi, Arshad Mehmood, and Tariq Hussain, "LiDAR Based Remote Sensing System for Foreign Object Debris Detection (FODD)," *Journal of Space Technology*, vol. 10, pp. 13–18, July 2020.

[3] Adam Parker, Felipe Gonzalez, and Peter Trotter, "Live Detection of Foreign Object Debris on Runways Detection using Drones and AI," in *2022 IEEE Aerospace Conference (AERO)*, Big Sky, MT, USA, Mar. 2022, pp. 1–13, IEEE.

[4] Travis Munyer, Daniel Brinkman, Xin Zhong, Chenyu Huang, and Iason Konstantzos, "Foreign object debris detection for airport pavement images based on self-supervised localization and vision transformer," *arXiv preprint arXiv:2210.16901*, 2022.

[5] Xiaoguang Cao, Peng Wang, Cai Meng, Xiangzhi Bai, Guoping Gong, Miaoming Liu, and Jun Qi, "Region Based CNN for Foreign Object Debris Detection on Airfield Pavement," *Sensors*, vol. 18, no. 3, pp. 737, Mar. 2018.

[6] Yunkai Liu, Yuanxiang Li, Jiawei Liu, Xishuai Peng, Yongiun Zhou, and Yi Lu Murphey, "FOD Detection using DenseNet with Focal Loss of Object Samples for Airport Runway," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, Bangalore, India, Nov. 2018, pp. 547–554, IEEE.

[7] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[8] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.

[9] Maiyu Ren, Weibing Wan, Zedong Yu, and Yuming Zhao, "Bidirectional yolo: improved yolo for foreign object debris detection on airport runways," *Journal of Electronic Imaging*, vol. 31, no. 6, pp. 063047–063047, 2022.

[10] Muhammad Reza Fairuzi and Fitri Yuli Zulkifli, "Performance analysis of yolov4 and ssd mobilenet v2 for foreign object debris (fod) detection at airport runway using custom dataset," in *2021 17th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering*. IEEE, 2021, pp. 11–16.

[11] Peng Li and Huajian Li, "Research on fod detection for airport runway based on yolov3," in *2020 39th Chinese Control Conference (CCC)*. IEEE, 2020, pp. 7096–7099.

[12] Sirui Song, Xi Qin, Jackson Brengman, Chris Bartone, and Jundong Liu, "Holistic fod detection via surface map and yolo networks," in *2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2023, pp. 1–6.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[15] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.

[16] Yuan Dai, Weiming Liu, Heng Wang, Wei Xie, and Kejun Long, "Yolo-former: Marrying yolo and transformer for foreign object detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–14, 2022.

[17] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014.