Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023
DOI : https://doi.org/10.33395/sinkron.v8i2.12378
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Implementation of ResNet-50 on End-to-End Object Detection (DETR) on Objects

Endang Suherman[1)], Ben Rahman[2)*], Djarot Hindarto[3)], Handri Santoso[4)]
[1,4)] Universitas Pradita, Serpong, Banten, Indonesia
[2,3)] Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional, Indonesia
[1)]endang.suherman@student.pradita.ac.id, [2)*]ben.rahman@civitas.unas.ac.id,
[3)]djarot.hindarto@civitas.unas.ac.id, [4)]handri.santoso@pradita.ac.id

Submitted : Apr 24, 2023 | **Accepted** : Apr 25, 2023 | **Published** : Apr 25, 2023

Abstract: Object recognition in images is one of the problems that continues to be faced in the world of computer vision. Various approaches have been developed to address this problem, and end-to-end object detection is one relatively new approach. End-to-end object detection involves using the CNN and Transformer architectures to learn object information directly from the image and can produce very good results in object detection. In this research, we implemented ResNet-50 in an End-to-End Object Detection system to improve object detection performance in images. ResNet-50 is a CNN architecture that is well-known for its effectiveness in image recognition tasks, while DETR utilizes Transformers to study object representations directly from images. We tested our system performance on the COCO dataset and demonstrated that ResNet-50 + DETR achieves a better level of accuracy than DETR models that do not use ResNet-50. In addition, we also show that ResNet-50 + DETR can detect objects more quickly than similar traditional CNN models. The results of our research show that the use of ResNet-50 in the DETR system can improve object detection performance in images by about 90%. We also show that using ResNet-50 in DETR systems can improve object detection speed, which is a huge advantage in real-time applications. We hope that the results of this research can contribute to the development of object detection technology in images in the world of computer vision.

Keywords: CNN, Computer Vision COCO Dataset; End-to-End Object Detection; Object Detection; ResNet-50

INTRODUCTION

Object recognition in images is one of the main problems in the world of computer vision. The main purpose of object detection is to identify and determine the location of objects contained in the image. Object detection has become a key focus in many applications such as autonomous vehicles, traffic monitoring, security monitoring and facial recognition. Several approaches have been developed to solve the problem of object detection in images. The traditional approach uses rule-based image processing techniques, but this approach has limitations in dealing with variations in objects, lighting, and complex image displays. Modern approaches to object detection use artificial neural networks, the Convolutional Neural Network (CNN) (Paymode & Malode, 2022), (Sze et al., 2022) architecture, which have proven very effective in image recognition tasks. End-to-end object detection is a relatively new approach in object detection, which involves using CNN and Transformer architectures to study object information directly from images. End-to-end object detection has yielded excellent results in object detection and has become the focus of recent research in the world of computer vision. In this research, we implemented ResNet-50 (Li & Lima, 2021) in an End-to-End Object Detection (DETR)

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.12378 p-ISSN : 2541-044X

system to improve object detection performance in images. ResNet-50 (de Zarzà et al., 2022) is a CNN architecture that is well-known for its effectiveness in image recognition tasks, while DETR utilizes Transformers to study object representations directly from images. We tested our system performance on the COCO dataset and demonstrated that ResNet-50 + DETR achieves a better level of accuracy than DETR models that do not use ResNet-50. In addition, we also show that ResNet-50 + DETR can detect objects more quickly than similar traditional CNN models. The purpose of this study is to contribute to the development of object detection technology in images in the world of computer vision. In the following chapters, we will go into more detail about object detection, CNN and Transformer architectures, and end-to-end object detection approaches using DETR and ResNet-50. We will also discuss our methodology and research results and discuss the implications of our research results.

End-to-End Object Detection (DETR) and You Only Look Once (YOLO) (Kong et al., 2022) are two popular approaches for object detection in images. Although both have the same goal, which is to identify and locate objects in the image, they differ in how they work. DETR is an end-to-end approach that uses the Transformer architecture to learn object information directly from images, while YOLO uses the CNN architecture to detect objects in images and identify object attributes such as class and object frame coordinates. The main difference between DETR (Carion et al., 2020) and YOLO (Xue et al., 2023) is in the way they process images. DETR takes the image as input and directly predicts the objects contained in the image. Meanwhile YOLO processes images in layers and divides them into several grids to detect objects in each grid. DETR is more suitable for object detection on a large scale, while YOLO is more suitable for object detection on a small to medium scale. DETR tends to be slower in object detection because it uses a complex Transformer architecture, while YOLO is faster because it uses a simpler CNN architecture. However, DETR has an advantage in generalizability. In DETR, image processing is not done in layers, so it is better at handling variations in object size and object displacement. DETR also does not require setting parameters such as grid size and overlap between grids that YOLO requires. Overall, both DETR and YOLO are effective approaches for object detection in images, and the choice between the two depends on the objectives and the specific circumstances of the object detection problem at hand.

Much research on object detection in images use the YOLO (Prabhakaran & Debebe, 2023) approach as the main detection algorithm. However, DETR is also a promising approach because it can learn object information directly from images and produce end-to-end object predictions without using supporting components such as feature extraction or complex parameter settings. Although DETR is relatively new and still not widely used in object detection in images, several studies have shown that DETR has advantages in solving complex object detection problems, such as dealing with object displacement and wide object size variations in images. In addition, DETR can also help reduce the time and effort required in preparing data and setting parameters because DETR does not require grids and overlap between grids as required by YOLO. This can make it easier for users to implement object detection in images without the need to set parameters manually. **State-of-the-art** research testing object detection using DETR is becoming increasingly interesting to do. By comparing the performance of DETR with other detection algorithms such as YOLO, it can help clarify the advantages and disadvantages of each approach and assist researchers in choosing the right algorithm to solve object detection problems in specific images. The problem with object detection is the use of very large objects, where Yolo detects small and medium images. So that DETR becomes a solution for very large images in detecting objects. In this research there are research questions related to research:

How to get image dataset for object detection research? (RQ1)

How to make a model from End-to-End Object Detection? (RQ2)

Research questions will be discussed later in the discussion section.

LITERATURE REVIEW

This literature review discusses several previous studies where object detection has made this research a trending research topic. Following below are some researches related to object detection:

End-to-End Object Detection with Transformers (Carion et al., 2020). The new model is conceptually simple and does not require a special library, unlike many modern detectors such as Yolo, Fast RCNN. DETR demonstrates accuracy and run-time performance on par with the established and highly

*name of corresponding author

optimized Faster RCNN baseline on object detection datasets, using the COCO dataset as the training process.

Object tracking and detection techniques under GANN threats: A systemic review (Al Jaberi et al., 2023). CNN is considered to be effective in addressing the viewpoint variation problem where handcrafted feature techniques fail to produce the desired results. Image classification, local objects, and detection techniques such as SVM, Haar cascade, CNN, Adaboost, HOG, YOLO, and GANN. GANN and CNN produce better results than other real-time object detection and tracking models. This paper has the disadvantage of not doing a comparison of all the presented algorithms, which causes the strength of the algorithm to be discussed.

Tiny object detection model based on competitive multi-layer neural network (TOD-CMLNN) (Chirgaiya & Rajavat, 2023). The main objective of this research is to increase the average accuracy and average memory of maps. Since a higher recovery indicates how well the detector is finding all positive cases, it is reasonable to assume that a better detector will find all positive cases. Increasing the FPS speeds up detection because the detector can process input more quickly. The average AP defined for all classes forms a map for object detection. The value of mAP@0.5 indicates that it is calculated with an IOU threshold of 0.5. FPS is influenced by several variables such as: input size, processing power and image quality. This paper has a weakness, namely the object detection still has a small image size.

Automated labeling of training data for improved object detection in traffic videos by fine-tuned deep convolutional neural networks (García-Aguilar et al., 2023). This paper presents a new procedure, detecting small-scale objects in a traffic sequence. Vehicle patterns detected from a set of frames are automatically generated via an offline process, using super-resolution techniques and a trained object detection network. Next, the object detection model is retrained with previously obtained data, adapting it to the scene being analyzed. This paper has a weakness, namely the object detection still has a small image size.

The above research detects objects with and the results can improve and use a simple Convolutional Neural Network technique without adding other methods or techniques, the average detection results have a high speed. The gap from this study, the detected object has a small image. Therefore, this paper proposes to detect large images using a simple technique, namely the Convolutional Neural Network plus the RESNET-50 method as a backbone, even though the detection speed is slow.

METHOD

Based on the research title "Implementation of ResNet-50 on End-to-End Object Detection (DETR) on Objects", this study may use an experimental method with an end-to-end object detection approach using DETR and compare it with other object detection algorithms such as YOLO or Faster R-CNN. This experimental method usually involves several stages such as data collection, data processing, model training, model testing, and analysis of the results. At the data collection stage, the researcher must collect a dataset that includes images of the object to be detected. At the data processing stage, these images must be processed and prepared so that they can be used for model training. Once the data is ready, researchers can conduct model training using specific object detection algorithms such as DETR or YOLO. At this stage, the parameters and hyperparameters of the model will be adjusted and optimized to achieve the best results. After the model is trained, the researcher can test the model using new images that the model has never seen before. The results of these tests are then analyzed and evaluated to determine the performance of the model and compare it with other object detection algorithms. This research may also use evaluation methods such as precision, recall, and mAP (mean Average Precision) to determine object detection performance. Apart from the experimental method, there are also other methods such as the survey method which can be carried out by collecting information from trusted sources such as scientific journals, books and websites to gain a better understanding of the research topic. Data analysis methods can also be used by collecting data from existing sources and analyzing them to gain new insights about the research topic.

Convolutional Neural Network (CNN) is a type of Machine Learning algorithm that is most often used in image recognition tasks, including in the field of Computer Vision such as classification, object detection, segmentation, and others. The CNN method is performed by processing the input image data gradually and repeatedly through a series of convolution, pooling, and other layers. In the convolution

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.12378 p-ISSN : 2541-044X

layer, the convolution filter will be used to retrieve important features from the image and produce output in the form of a feature map. After the convolution layer, it is usually followed by an activation layer such as ReLU (Rectified Linear Unit) to accelerate convergence in the training process. Furthermore, at the pooling layer, a down sampling operation will be carried out to reduce data dimensions by taking the maximum or average value of each data group. This is useful for reducing data size and increasing computational efficiency. After several convolution and pooling layers, the resulting data will be plated into a vector and passed to the Fully Connected (FC) layers or classification layers to produce predictions on the output. In the training process, the CNN method will be carried out by adjusting the model parameters to optimize predictions at the output. Usually, this method uses the Backpropagation method to determine the gradient of the error function on the output and optimizes the model parameters with optimization algorithms such as Gradient Descent. In general, the CNN method has proven to be very effective in image recognition tasks and has been used in many applications such as face detection, license plate recognition, and medical image classification.

ReLU (Rectified Linear Unit) is one of the activation functions that is often used in Convolutional Neural Networks (CNN) to accelerate convergence in the training process. This activation function will produce a positive output value equal to the input, and a negative output value equal to 0.
Mathematically, ReLU is defined as:

$$f(x) = \max(0, x) \quad \text{..... (1)}$$

where x is the input value and f(x) is the output value generated by the activation function. One of the advantages of using ReLU is computational efficiency, because the ReLU operation only involves one comparison operation and one checking operation whether the input value is greater than 0 or not. In addition, ReLU also overcomes a problem called the vanishing gradient problem, where the gradient value gets smaller and eventually disappears in the deeper layers of the CNN. By using ReLU, the gradient that is passed on to the next layers will not be lost and can increase the effectiveness of the training process. However, ReLU also has a weakness, namely having an output of 0 for a negative input value, which can cause loss of information. Therefore, there are variants of ReLU such as Leaky ReLU and ELU which overcome this weakness by providing a small output value instead of 0 for negative input values.

Fully Connected (FC) is a type of layer in a neural network consisting of several units or neurons that are fully connected to all units in the previous layer. In a neural network consisting of multiple layers, the FC layer is usually located at the end of the network and acts as a classification or regression output. In the FC layer, each unit or neuron will have a weight and bias that is applied to the input value received from the previous layer. Then, the output value of each unit or neuron is calculated by adding up the multiplication between the weight and the input value, plus the bias value. The output from the FC layer will then be fed to an activation function such as sigmoid or softmax, depending on the type of task being executed. In general, the FC layer is very useful in classification and regression tasks, where the resulting output is a discrete or continuous value. Examples of using the FC layer can be found in image recognition, where the input values generated by the previous layers will be converted into feature vectors and passed to the FC layer to produce output values that represent image classes. However, the use of the FC layer also has disadvantages, namely memory usage and high computation time, especially for data that has high dimensions. In addition, the FC layer is also prone to overfitting, where the model becomes too specific to the training data and cannot generalize well to new data. Therefore, in some of the latest neural network architectures such as the Convolutional Neural Network (CNN), the FC layer has been replaced with a more efficient layer such as Global Average Pooling or a regression layer.

YOLO (You Only Look Once) is an object detection algorithm developed by Joseph Redmon and colleagues in 2016. YOLO differs from traditional object detection algorithms in that it requires several steps such as region proposals and feature calculations first. In contrast, YOLO performs object detection on an end-to-end basis by evaluating the entire image at the same time. In YOLO, the input image is divided into grids and each grid is considered an anchor box. Then, YOLO will predict the location and class of objects in each anchor box using a pre-trained convolutional neural network (CNN)

*name of corresponding author

model. YOLO also uses a non-maximum suppression technique to remove redundant predictions and choose the best prediction for each object. The advantages of YOLO are its high detection speed and good accuracy. YOLO can perform object detection at high speed because it only needs to process the image once and uses a simple CNN architecture. In addition, YOLO can also detect overlapping objects and small objects with good accuracy. However, the downside of YOLO is its sensitivity to small objects located far from the center of the grid. YOLO also cannot detect objects that have extreme aspect ratios such as very long or very wide objects. Therefore, several studies have tried to improve YOLO performance by combining it with other techniques such as Region Proposal Network (RPN) and Feature Pyramid Network (FPN) to overcome these weaknesses.

Faster R-CNN (Rajeshkumar et al., 2023), (Zhu et al., 2020) (Region-based Convolutional Neural Network) is an object detection algorithm developed by Shaoqing Ren, et al. in 2015. This algorithm uses the CNN architecture to process the input image and derive important features from the image. Then, Faster R-CNN uses the Region Proposal Network (RPN) module to generate regional proposals from images which are then used to carry out object detection. In Faster R-CNN (Wahjuni & Nurarifah, 2023), an RPN is a network consisting of a convolution layer mounted on top of a feature convolution layer. RPN uses a sliding window to generate area proposals with various sizes and aspect ratios at each location on the image. Then, the regional proposals will be calculated with the score function to determine which regional proposals contain objects. RPN (Region Proposal Network) is the main module in the Faster R-CNN architecture for generating object area proposals in images. RPN is a convolutional neural network (CNN) that is used to map input images into several region proposals or regional proposals. The RPN is placed on the same convolution layer as the convolution layer on the CNN, so that the features generated from the CNN can be used as input to the RPN. RPN uses a sliding window with a certain size at each location on the feature map to produce several regional proposals. Sliding windows in RPNs usually have several different scales and aspect ratios to allow detection of objects of various sizes and shapes. Furthermore, each regional proposal generated by the RPN will be assessed with a score indicating the probability that the area contains the object. This score is calculated using a mapping function that generates a score for each regional proposal. Regional proposals that have scores above a threshold will be selected as regional object proposals which will then be included in the Region of Interest Pooling. RoI Pooling is used to extract features from each object area proposal and this feature will be used to perform classification and regression to determine object labels and object locations in the image. RPN is one of the key components in Faster R-CNN which allows for more accurate and fast object detection. Compared to the previous method which required detection of regional proposals separately before entering CNN, RPN can generate regional proposals in an end-to-end manner and is more efficient because it uses the features that have been generated from CNN as input. After getting regional proposals, Faster R-CNN uses the RoI (Region of Interest) Pooling module to change the size of regional proposals to a fixed size and then sends them to the CNN network to get features from regional proposals. These features are then used to classify objects and determine the exact location of these objects. The advantage of the Faster R-CNN is its accuracy in object detection and the ability to detect objects with different aspect ratios. However, the weakness of Faster R-CNN is the relatively slow processing speed compared to other object detection algorithms such as YOLO because it needs to go through several processing stages.

End-to-End Object Detection (DETR) is an object detection method that implements an end-to-end learning approach, namely learning that is carried out directly from raw images to get the final object detection output in one stage or one neural network. DETR combines new techniques, such as Attention Mechanism, with the Transformer architecture, a neural network architecture originally developed for Natural Language Processing (NLP). DETR transforms the object detection problem from classifying image pixels to solving global tasks across images by using a neural network that generates a proposed list of objects with their labels and locations. DETR uses a transformer architecture to map images globally into feature representations and uses attention mechanisms to consider the interactions between each object proposal and the features produced from the image. DETR also uses the Sinkhorn Attention technique which enables the calculation of the attention matrix to be more efficient by applying a Sinkhorn transformation to the matrix. This enables DETR to address scalability issues in object detection methods that require a large number of object proposals to achieve accurate object detection.

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.12378 p-ISSN : 2541-044X

The main advantages of DETR are being able to perform end-to-end object detection, reducing errors from overlapping object proposals, and speeding up training time and object detection. However, DETR still has weaknesses in detecting small objects and very close or overlapping objects.

ResNet-50 (Sarwinda et al., 2021), (Santos-Bustos et al., 2022) is a convolutional neural network (CNN) architecture developed by Microsoft Research Asia. This ResNet architecture has 50 layers, and is one of the deep learning models most widely used in various computer vision tasks such as image classification, object detection, and segmentation. ResNet-50 uses a residual learning approach, which enables the neural network to learn more complex features and avoid the so-called "degradation problem" as the number of layers increases. The degradation problem occurs when adding layers to a deeper neural network architecture causes poorer performance and lowered accuracy, which shouldn't happen if deeper neural networks have the ability to learn better representations. The residual learning approach in ResNet-50 allows information from the previous layer to be passed directly to the next layer, using "skip connections" or shortcut connections. In this way, the neural network can learn a better representation while retaining information from the previous layer and reducing the performance degradation that occurs when adding layers. ResNet-50 has an architecture consisting of several blocks consisting of several layers. The main block used in ResNet-50 is the Residual Block which consists of two convolution layers with a shortcut connection at the end of the block. In addition, ResNet-50 also uses global average pooling techniques and a fully-connected layer at the end of the architecture to produce the required output. ResNet-50 (Chen et al., 2022) has proven successful in various computer vision tasks, including winning several image processing competitions and being one of the most widely used deep learning models by researchers and practitioners.

Evaluation of the DETR model. AP (average precision) model is a performance evaluation metric of machine learning models in classification or object detection tasks in images. This metric measures how well the model can classify objects accurately and how well the model reduces the number of false positives (objects that are incorrectly detected). In general, AP measures the area under a precision-recall curve. The precision-recall curve is a curve that describes the relationship between precision and recall at various different threshold values. Precision is the number of true positives divided by the number of true positives and false positives, while recall is the number of true positives divided by the number of true positives and false negatives. Models that have a high AP value indicate that the model is able to classify objects properly and produces few false positives. Whereas a model with a low AP value indicates that the model is less able to classify objects properly and produces many false positives. In the context of object detection in images, AP is often used in conjunction with other evaluation metrics such as mean average precision (mAP) to evaluate overall model performance.

DETR (Detection Transformer) is an object detection model that uses a transformer approach in processing image information. Evaluation of the DETR model can be done using several metrics such as the following:

1. Mean Average Precision (mAP), mAP measures the accuracy of the model in detecting objects in the image. mAP is calculated by calculating the average precision at various recall values. Precision is calculated by comparing the number of true positives with the number of false positives and true positives. The DETR model that has a high mAP indicates that the model is able to detect objects well in various situations.
2. Average Precision (AP) in each class, AP in each class measures the accuracy of the model in detecting objects in each class. AP is calculated by taking into account the true positive, false positive, and false negative in each class. The DETR model that has a high AP in each class shows that the model is able to detect objects in each class well.
3. The F1-score measures the accuracy of the model in detecting objects taking into account precision and recall. F1-score is calculated by calculating the harmonic average between precision and recall. The DETR model which has a high F1-score indicates that the model is capable of detecting objects well in various situations.
4. Precision and recall measure the accuracy of the model in detecting objects by taking into account true positives, false positives and false negatives. Precision is calculated by comparing the number of true positives with the number of false positives and true positives, while recall is calculated by comparing the number of true positives with the number of false negatives and true positives.

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.12378 p-ISSN : 2541-044X

The DETR model which has high precision and recall shows that the model is able to detect objects well in various situations.

5. IOU measures the extent to which the bounding boxes generated by the DETR model overlap with the actual bounding boxes. The IOU is calculated by dividing the overlapping area by the combined area of the two bounding boxes. The DETR model that has a high IOU shows that the model is able to make an accurate and precise bounding box for the detected object.

RESULT

The results of the experiment using Google Colab are as follows:

```
model = torch.hub.load('facebookresearch/detr', 'detr_resnet50', pretrained=True)
model.eval();
```

Figure 1. Model using a torch

Figure 1, Development of deep learning models uses the PyTorch library referred to as "torch". PyTorch is an open-source library used to build and train machine learning models, especially deep learning models. Deep learning models usually consist of many layers (layers) that are interconnected and process data in stages to produce the desired output. In PyTorch, it is easy to create and define these layers using the provided modules, such as linear, convolution, recursive modules. PyTorch also provides a variety of functions that are very helpful in training models, such as activation, optimization, and loss functions. In PyTorch, it is easy to organize and manage data using "tensor" objects. At its core, PyTorch provides a very powerful and flexible framework for building deep learning models. This makes PyTorch a popular choice for researchers and practitioners in the fields of machine learning and artificial intelligence.

```
Downloading: "https://github.com/facebookresearch/detr/zipball/main" to /root/.cache/torch/hub/main.zip
/usr/local/lib/python3.9/dist-packages/torchvision/models/_utils.py:208: UserWarning: The parameter 'pretrained' is deprecated since
  warnings.warn(
/usr/local/lib/python3.9/dist-packages/torchvision/models/_utils.py:223: UserWarning: Arguments other than a weight enum or `None` fo
  warnings.warn(msg)
Downloading: "https://download.pytorch.org/models/resnet50-0676ba61.pth" to /root/.cache/torch/hub/checkpoints/resnet50-0676ba61.pth
100%|██████████| 97.8M/97.8M [00:00<00:00, 162MB/s]
Downloading: "https://dl.fbaipublicfiles.com/detr/detr-r50-e632da11.pth" to /root/.cache/torch/hub/checkpoints/detr-r50-e632da11.pth
100%|██████████| 159M/159M [00:04<00:00, 35.3MB/s]
```

Figure 2. The result of the DETR process

Figure 2 explains DETR (Detection Transformer) is a deep learning model used to detect objects in images. DETR incorporates Transformer technology in image processing to produce end-to-end object detection, enabling models to perform object detection and segmentation in a single-stage. This process downloads the Pytorch model from the FacebookResearch GitHub site.

PyTorch is one of the popular open-source deep learning frameworks developed by Facebook AI Research (FAIR) and the open-source community. PyTorch is developed based on the dynamic computational graph concept, which enables users to build deep learning models flexibly and intuitively. Facebook Research is a research team at Facebook that focuses on developing AI and machine learning technologies. The team consists of AI experts, data scientists and engineers working on various projects, including the development of PyTorch. Facebook Research also actively contributes to the development of AI and machine learning technologies by publishing related research papers and participating in related. conferences.

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023
DOI : https://doi.org/10.33395/sinkron.v8i2.12378
e-ISSN : 2541-2019
p-ISSN : 2541-044X

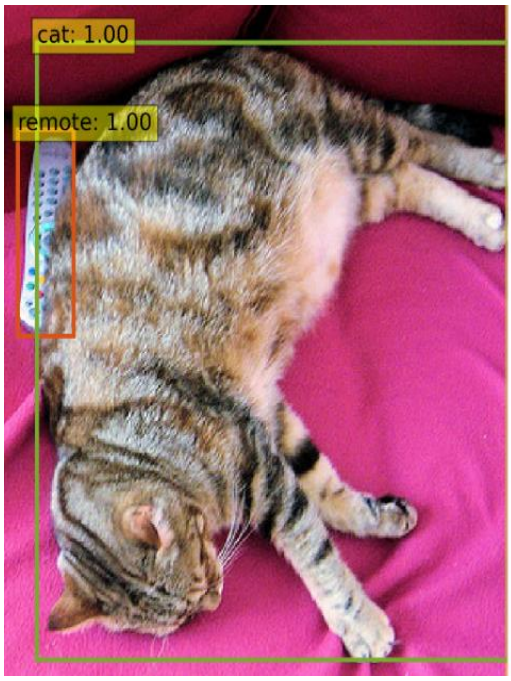

Figure 3. Remote and Cat Detection

Figure 3, Pretraining DETR using the RESNET-50 backbone is a technique commonly used to improve object detection performance in models. RESNET-50 is a very popular CNN architecture in image processing and has good performance in recognizing objects in images. After the pretraining process, the DETR model will be trained to recognize objects in the image. In this case, the model is trained to recognize two types of objects, namely remote and cat. After the model is trained, the model is then tested by providing remote images and paint, and the results are very accurate, which means that the model is able to recognize both types of objects well. The results of very accurate detection on the remote and paint show that the DETR model with the RESNET-50 backbone has a good ability to recognize objects in images. This is especially important in object detection applications, where high detection accuracy is required to avoid object recognition errors that can adversely affect system security and performance. In the context of remote object detection and cat, pretraining results with DETR and RESNET-50 backbone can be a good solution to improve object detection performance in security surveillance systems or other applications that require accurate object detection.

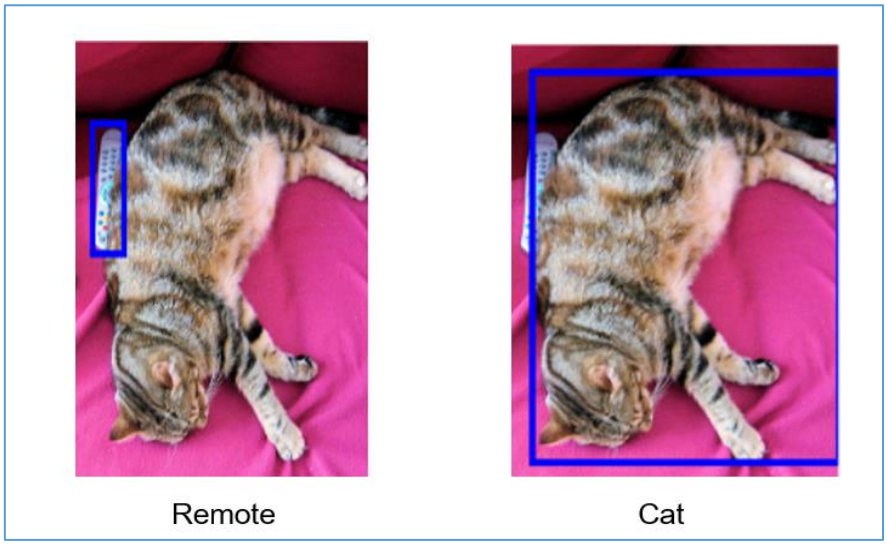

Figure 4. Remote and Cat

*name of corresponding author

Sinkron: Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 2, April 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.12378 p-ISSN : 2541-044X

Figure 4, describes the object, namely remote and paint separately. The image shows two different objects separately, namely remote and paint. Each object is marked with a bounding box indicating the object's location in the image, as well as a class label indicating the type of detected object (i.e., "remote" and "cat"). The results of the detection of separate objects show that the model has a good ability to distinguish and recognize different objects in the image. This is an important capability in object detection applications, where the system must be able to recognize many different types of objects with high accuracy.

Table 1. Performance model DETR

	Precision	Recall	F1-score	Support
Cat	0.89	0.73	0.80	287
Remote	0.90	0.89	0.92	285
Micro avg	0.83	0.63	0.72	470
Macro avg	0.69	0.69	0.48	470
Weighted avg	0.82	0.63	0.71	470

DETR (Detection Transformer) is an object detection model that uses a Transformer to generate object predictions in images. DETR model performance depends on various factors, such as model architecture, training dataset, and training parameters. Table 1 shows the level of performance from training that has shown good results. In general, DETR has been proven to achieve very good performance in object detection tasks on various datasets. In the COCO 2019 competition, DETR has shown very good results, even outperforming several other popular object detection methods. Performance Cat reaches 89% precision and remote reaches 90% precision. Performance detection on cat and remote objects is already good.

DISCUSSIONS

The discussion section discusses questions from research, which will be discussed in more depth so that more detailed answers can be explained. In this research there are two questions.

How to get image dataset for object detection research? (RQ1)

There are different ways to obtain an image dataset for object detection research, and the approach you choose will depend on your specific research goals and the type of objects you want to detect. Here are some options to consider:

1. Public datasets: There are many publicly available image datasets that can be used for object detection research, such as COCO, ImageNet, Pascal VOC, Open Images, and more. These datasets typically come with annotations that indicate the locations of objects in the images. Public datasets are collections of image data available for use freely by the public for research and development purposes, usually in the context of developing computer vision and machine learning systems. These datasets usually consist of thousands or even millions of images with annotations indicating the location and labels of objects in the image. This public dataset can be used as a data source for training object detection models, as well as a tool for comparing the performance of different models. Public datasets can be downloaded and used free of charge, and because they have been drawn from a variety of sources, they are usually diverse and cover many types of objects in different scenarios and lighting conditions. However, keep in mind that public datasets can also have flaws, such as undetected bias or imprecise labels, therefore, careful evaluation is required before such datasets are used for research and development purposes.
2. Data scraping: You can use web scraping tools to collect images from the web that contain the objects you are interested in. However, be aware of copyright issues and ensure that you have the right to use the images. Data scraping is the process of taking data from a website or other online

*name of corresponding author

source and storing it in an accessible and reusable format, such as a spreadsheet or database. Its purpose is to automatically and quickly gather information from the internet for analysis, market research, price tracking, and more. This process is usually carried out using special programs or scripts that can extract data automatically from online sources. However, please note that some websites may prohibit automatic data collection and prohibit this practice as part of their privacy policies. Therefore, before carrying out data scraping, it is important to check the rules and terms of use of the website from which the data will be extracted. Data scraping is often used by companies and organizations to gather data from competitors, discover market opportunities and gain insight into customer behavior. However, this technique is also often controversial because it can be misused for unethical purposes, such as extracting personal or confidential data from other people.

3. Data collection: You can collect your own images by taking photos or videos of objects in different settings and conditions. This approach can be time-consuming and resource-intensive but can give you greater control over the data and the ability to tailor it to your research needs. Image data collection for object detection is the collection of image data that is used to train machine learning models in recognizing certain objects in images. The goal of data collection image collection is to collect a large number of images showing the target object from various angles, at various sizes, and under various lighting conditions and backgrounds. The process of collecting image data for object detection involves several steps. Determining the target object is the first step is to determine the object to be detected in the image. These objects must be chosen carefully and must include many variations in appearance and placement. Creates a list of images used in training a machine learning model to create. These images must be chosen carefully to ensure they represent different conditions and situations. Marking objects, each selected image must be marked with the target object you want to detect. This enables machine learning models to learn how the target object looks in various situations. Collect tagged images from different sources. This can involve taking pictures with a camera, downloading pictures from the internet, or using an existing image dataset. Image collection data for object detection must include a large enough number of images to ensure the machine learning model has a lot of variety to learn from. The process of collecting image collection data is very important to improve the accuracy of machine learning models and ensure that the model can work well in recognizing target objects in new images.
4. Synthetic data: Generate synthetic images using computer graphics software or game engines. This approach allows you to control the object's appearance, pose, and environment and can be useful when you have limited access to real-world data. Synthetic data is data generated by computers through simulations or mathematical models, not from direct data collection from original sources such as humans, hardware, or software. This data is created to mimic or represent real data and is used in applications such as machine learning model training, software testing and development, and risk analysis. Synthetic data is usually created using algorithms or mathematical models to produce data that closely resembles real data in a controllable and reproducible way. Using synthetic data can help overcome some of the problems associated with original data collection, such as difficulty collecting large enough data, incomplete data, or insufficient data for a particular purpose. With synthetic data, users can control the type of data and the variations that are generated, making it easier to experiment and test. Synthetic data can be used to extend existing datasets, helping to increase the accuracy and reliability of machine learning models. The use of synthetic data can help protect individual privacy in sensitive data. However, the use of synthetic data also has some drawbacks. May not perfectly represent the original data: Although the synthetic data tries to mimic the original data, it can't capture the variability or diversity in the original data. Synthetic data may not be a substitute for real data for some types of analysis, such as studies that require very specific data. Creating synthetic data requires significant costs for model development and testing.
5. Data augmentation: Augment existing datasets by applying transformations to the images, such as rotation, scaling, and flipping. This can help you increase the size of the dataset and improve the model's robustness to variations in the data.

*name of corresponding author

Regardless of the approach you choose, it's important to ensure that the dataset is diverse, representative of the objects you want to detect, and annotated accurately and consistently.

How to make a model from End-to-End Object Detection? (RQ2)

Creating a model from End-to-End Object Detection involves several steps.

1. Collect data: Collect a large amount of image data with a variety of objects and backgrounds. Good data should include images with different resolutions, rotations, and scales. These images should be annotated with the desired object information.
2. Prepare data: Prepare data by dividing dataset into training set and test set. Typically, the training set covers around 70-80% of the dataset and the test set covers 20-30%. In addition, do data pre-processing such as resizing, normalizing, cropping, and image augmentation.
3. Building models: There are several types of models for End-to-End Object Detection, such as YOLO, Faster R-CNN, and SSD. Choose a suitable model for your needs and build the model using a deep learning framework such as TensorFlow or PyTorch. The model must go through a training process using a training data set.
4. Train the model: The training process involves fitting the model weights to the training data set. Training should be done in several epochs or iterations to increase accuracy and minimize errors. During training, the model must be optimized using the appropriate optimizer and loss function.
5. Model evaluation: After the model has been trained, it must be evaluated using a test set to measure the accuracy and performance of the model. Evaluation results can help determine whether the model needs further optimization or is good enough.
6. Inference: After the model is tested and found to be good, use the model to make inferences on new images. In inference, the model will detect objects in new images and provide bounding box coordinates and object labels.

Those are the steps in creating a model from End-to-End Object Detection. This process requires experience and expertise in deep learning and computer vision, as well as the appropriate hardware and software.

CONCLUSION

. Based on the previous explanation, it can be concluded that the implementation of ResNet-50 in end-to-end object detection using DETR (Detection Transformer) on certain objects such as remotes and cats, is able to provide accurate detection results. DETR is one of the newest object detection techniques that uses transformers to generate object predictions in images, which has proven to achieve very good performance in object detection tasks on various datasets. DETR model performance depends on various factors, such as model architecture, training dataset, and training parameters. Choosing the right model architecture, training dataset that is large enough and representative, optimal training parameters, and using data augmentation techniques can improve the performance of the DETR model. In practice, DETR can be a good choice for object detection in images with accurate and efficient results. Implementation of ResNet-50 in end-to-end object detection using DETR can be an attractive alternative in the development of object detection systems for various applications such as security surveillance, environmental monitoring, facial recognition, and others.

REFERENCES

Al Jaberi, S. M., Patel, A., & AL-Masri, A. N. (2023). Object tracking and detection techniques under GANN threats: A systemic review. *Applied Soft Computing*, *139*, 110224. https://doi.org/10.1016/j.asoc.2023.110224

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-End Object Detection with Transformers. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12346 LNCS*, 213–229. https://doi.org/10.1007/978-3-030-58452-8_13

Chen, Y., Lin, Y., Xu, X., Ding, J., Li, C., Zeng, Y., Liu, W., Xie, W., & Huang, J. (2022). Classification of lungs infected COVID-19 images based on inception-ResNet. *Computer Methods and Programs in Biomedicine*, *225*, 107053. https://doi.org/10.1016/j.cmpb.2022.107053

Chirgaiya, S., & Rajavat, A. (2023). Tiny object detection model based on competitive multi-layer

*name of corresponding author

neural network (TOD-CMLNN). *Intelligent Systems with Applications*, *18*(September 2022), 200217. https://doi.org/10.1016/j.iswa.2023.200217

de Zarzà, I., de Curtò, J., & Calafate, C. T. (2022). Detection of glaucoma using three-stage training with EfficientNet. *Intelligent Systems with Applications*, *16*(September), 1–10. https://doi.org/10.1016/j.iswa.2022.200140

García-Aguilar, I., García-González, J., Luque-Baena, R. M., & López-Rubio, E. (2023). Automated labeling of training data for improved object detection in traffic videos by fine-tuned deep convolutional neural networks. *Pattern Recognition Letters*, *167*, 45–52. https://doi.org/10.1016/j.patrec.2023.01.015

Kong, L., Wang, J., & Zhao, P. (2022). YOLO-G: A Lightweight Network Model for Improving the Performance of Military Targets Detection. *IEEE Access*, *10*, 55546–55564. https://doi.org/10.1109/ACCESS.2022.3177628

Li, B., & Lima, D. (2021). Facial expression recognition via ResNet-50. *International Journal of Cognitive Computing in Engineering*, *2*(January), 57–64. https://doi.org/10.1016/j.ijcce.2021.02.002

Paymode, A. S., & Malode, V. B. (2022). Transfer Learning for Multi-Crop Leaf Disease Image Classification using Convolutional Neural Network VGG. *Artificial Intelligence in Agriculture*, *6*, 23–33. https://doi.org/10.1016/j.aiia.2021.12.002

Prabhakaran, K., & Debebe, T. (2023). Skin Cancer Cancer diagnosis diagnosis with with Yolo Yolo Deep Deep Neural Neural Network Network. *Procedia Computer Science*, *220*, 651–658. https://doi.org/10.1016/j.procs.2023.03.083

Rajeshkumar, G., Braveen, M., Venkatesh, R., Josephin Shermila, P., Ganesh Prabu, B., Veerasamy, B., Bharathi, B., & Jeyam, A. (2023). Smart office automation via faster R-CNN based face recognition and internet of things. *Measurement: Sensors*, *27*(February), 100719. https://doi.org/10.1016/j.measen.2023.100719

Santos-Bustos, D. F., Nguyen, B. M., & Espitia, H. E. (2022). Towards automated eye cancer classification via VGG and ResNet networks using transfer learning. *Engineering Science and Technology, an International Journal*, *35*, 101214. https://doi.org/10.1016/j.jestch.2022.101214

Sarwinda, D., Paradisa, R. H., Bustamam, A., & Anggia, P. (2021). Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer. *Procedia Computer Science*, *179*(2019), 423–431. https://doi.org/10.1016/j.procs.2021.01.025

Sze, E., Santoso, H., & Hindarto, D. (2022). *Review Star Hotels Using Convolutional Neural Network*. *7*(1), 2469–2477.

Wahjuni, S., & Nurarifah, H. (2023). Faster RCNN based leaf segmentation using stereo images. *Journal of Agriculture and Food Research*, *11*(November 2022), 100514. https://doi.org/10.1016/j.jafr.2023.100514

Xue, G., Li, S., Hou, P., Gao, S., & Tan, R. (2023). Research on lightweight Yolo coal gangue detection algorithm based on resnet18 backbone feature network. *Internet of Things (Netherlands)*, *22*(March), 100762. https://doi.org/10.1016/j.iot.2023.100762

Zhu, X., Blanco, E., Bhatti, M., & Borrion, A. (2020). Leguminous seeds detection based on convolutional neural networks: Comparison of faster R-CNN and YOLOv4 on a small custom dataset. *Science of the Total Environment*, 143747. https://doi.org/10.1016/j.aiia.2023.03.002

*name of corresponding author