A Topological Data Analysis-Based Approach to Object Localization: A Comparison with ViT and Yolov7

Hong Cheng, Zeng Ze Ong

Department of Math and Computer Science

Southern Arkansas University

Magnolia, AR, USA

hcheng@saumag.edu

Abstract: In this paper, we present a modified algorithm based on topological data analysis (TDA) for object localization, and compare its performance with two well-known supervised models, Vision Transformer (ViT) and Yolov7, on two different datasets. Our TDA-based approach returns an IOU (intersection over union) score of 64% and 66% on the two datasets, respectively. We find that both ViT and Yolov7 outperform our unsupervised algorithm, with ViT achieving a better performance on one dataset at 90%, while Yolov7 performs better on the other dataset at 89%. Our results suggest that while TDA-based approaches show promise for object localization, there is still room for improvement in comparison with supervised models.
Keywords —object localization, topological image processing, topological data analysis, IOU, Vision Transformer, YOLOV7

1. Introduction

Object localization is a fundamental problem in computer vision that involves identifying the location of objects within an image or video. The purpose of doing research on object localization is to develop algorithms and techniques that can accurately and efficiently locate objects within an image or video.

There are several reasons why object localization is an important area of research. Firstly, accurate object localization is a crucial component of many computer vision applications such as autonomous vehicles, robotics, and surveillance systems. Secondly, object localization is a challenging task due to the complexity of real-world images and videos, making it an interesting and intellectually stimulating research problem. Additionally, advances in object localization can also lead to improvements in related areas such as object recognition and image segmentation.

Object localization is the process of determining the location of an object within an image, which is usually represented as a bounding box. This bounding box is defined by a set of four numbers, namely, the x and y coordinates of the top-left corner of the box, as well as its width and height. Object localization algorithms predict these four parameters in order to draw a bounding box around the object of interest in an image.

Two famous deep learning models for object recognition are R-CNN [1] and Fast R-CNN [2]. However, Yolov (You Only Look Once) has emerged as the State-of-the-Art Object Detection model [3] since its introduction. At the same time, Vision Transformer (ViT) has gained popularity in various machine learning applications. The ViT model [4] is built on the self-attention-based Transformer architecture [5], which has become the preferred model for natural language processing (NLP). In ViT, the Transformer architecture with self-attention is applied to sequences of image patches, making it powerful in image classification and other machine learning domains.

Yolov7 and ViT are supervised learning models that rely on annotated images for training. During training, the models learn from the annotated images to make predictions. The performance of the models is evaluated using the IOU (intersection over union) metric, which compares the predicted bounding boxes to the annotated ones.

Vandaele[6] introduced a Topological Data Analysis (TDA)-based method for object detection called Topological Image Modification (TIM) and Topological Image Processing (TIP). Our research aims to extend this method as an unsupervised algorithmfor object localization and evaluate its performance. In addition, we will compare the results with Yolov7 and ViT to determine which model performs better. We will assess the performance of Yolov7 and ViT and compare which one outperforms the other.

2. Background

1.1 Topological Image Processing

Topological image processing (TIP) is a method that uses topological data analysis (TDA)[7,8] to extract features from images. TDA relies on a tool called persistent homology to identify the number of connected components, cycles, and voids in an image, as well as their birth and death during an iterative process called a filtration. Fig. 1 shows a motocycle image with its persistent diagram. TIP works by iterating over all the lifetimes in the image, using persistent diagrams to select a threshold by averaging the two lifetimes with the largest difference. Any birth and death pairs with a lifetime above this threshold are processed, and components are merged using the elder rule. This rule determines that when constructing the persistence diagram, the youngest component or hole is considered dead when two are merged. The output of TIP is a binary image that marks objects from the original image with inferred components through its persistence diagram.

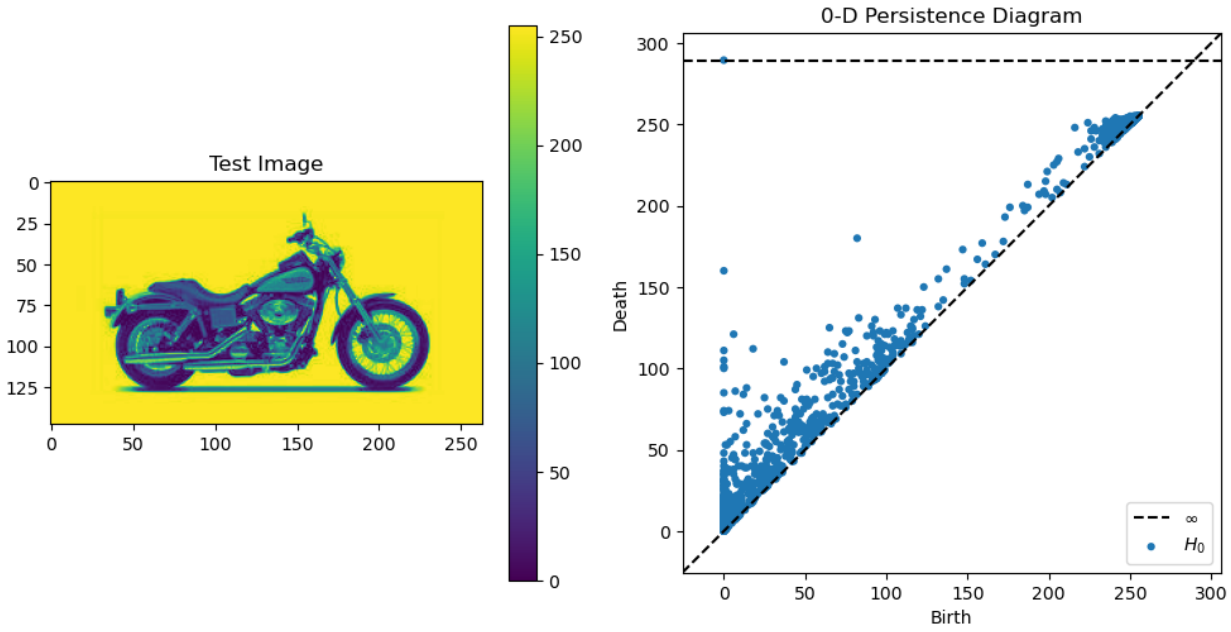

Fig. 1. Left: A motocyle image in grayscale Right: A 0-D Persistence Diagram of the image

First, the images are transformed into grayscale format. Then, the gray scale image data undergoes a filtration process, specifically the Low Star filtration [9], also known as sublevelset filtration. This filtration method identifies local minimums as birth times and saddle points as death times, resulting in a 0-dimensional persistence diagram. Next, a sparse distance matrix is generated, where each pixel in the image serves as a vertex, connected to its 8 spatial neighbors (except for those at the boundary). The edge weights are determined by selecting the maximum value between the two connected pixels. Finally, a threshold is chosen.

The filtration F for image M is defined as

$$K_i \coloneqq (\{\sigma \in K : f(K) \leq i\})$$

Where

$$f : K \to R : \sigma \to \max_{p \in \sigma} gray_M\,(p)$$

And $K_0 \subseteq K_1 \subseteq K_2 \ldots \subseteq K_T = K$. For each pixel in K_i, its corresponding pixel value in image I is set to 1 if it is a member of K_i, and 0 otherwise. Connected components of K_i, where i is an integer from 1 to T, represent clusters of pixels with the same value of 1 that are maximally connected. The elder rule governs the merging of components and states that when constructing the persistence diagram, if two components or holes are merged, the one with the highest birth-time is eliminated.

1.2 Vision Transformer

Although Transformers are relatively new to computer vision, recent studies have demonstrated their potential. Originally developed for language processing, Transformers analyze the relationships between words in a sentence to establish "context." This approach can be applied to images, considering each image as a "sentence" for the Transformer to interpret.

Alexey Dosovitskiy [4] introduced attention-based image processing in the Vision Transformer (ViT) models. ViT treats images as a list of "words" and divides them into patches with assigned positions. Fig. 2 shows this process. These patches are encoded through Transformer blocks that relate one patch to another via self-attention. A MultiHeadAttention layer is applied to the sequence of image patches. The encoded patches and self-attention layer outputs are normalized and then processed through a multilayer perceptron (MLP). The model's output is a set of four-dimensional coordinates representing the object's bounding box.

Fig. 2: Left: ViT Model, Right: A MotorCycle split into 49 patches

1.3 YOLO (You Only Look Once)

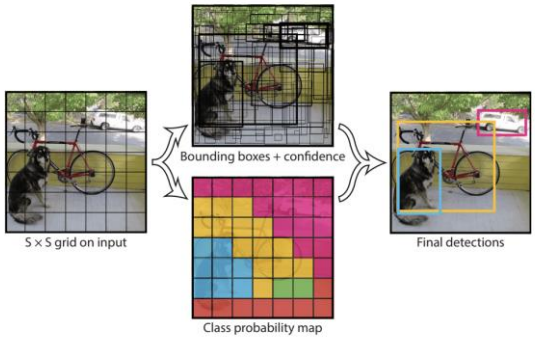

Fig. 3. Summary of Predictions made by YOLO Model [3]

YOLO stands for You Only Look Once, referring to its use of single-shot object detection [3]. Single-shot detection only processes the image once, which is less demanding computationally. YOLO predicts bounding boxes and classes at the same time, making it faster and more efficient than previous real-time detection algorithms. See Fig. 3.

YOLO first resizes the image to a set number of pixels and divides images into a grid of equal proportions. Each grid cell that contains the center of an object will attempt to predict the area of the bounding box and the class of the object. There have been multiple versions of YOLO, each faster and more accurate than the previous one. YOLO v7, the newest version, introduces “focal loss”, which focuses more on objects that are harder to detect. In turn, the function focuses less on objects that were already well-classified. It also detects smaller objects better due to its higher resolution at 608 by 608 pixels, an improvement over the previous 416 by 416 pixels. It is also faster than its previous versions, processing images at 155 frames a second. The first version of YOLO processed images at 45 frames a second. However, YOLO v7 still struggles to detect objects that are small and in crowded scenes

accurately. Even though it is not perfect, YOLO v7 still outperforms many other object detection algorithms in terms of speed and results.

1.4 IOU (Intersection over Union)

Intersection over Union (IoU), is the most popular evaluation measure for tasks like object localization. Localization, which is determining the location of an object in an image. IOU is defined as area of intersection over area of union.

A bounding box B is defined by its four components: x1,y1.x2, y2 which represent two points of upper left (x1, y1) and bottom right (x2, y2) on the diagonal. IOU can be calculated using predicted box P and annotated box A as following:

$$x1 = \max(P.x1, A.x1)$$
$$y1 = \max(P.y1, A.y1)$$
$$x2 = \min(P.x2, A, x2)$$
$$y2 = \min(P.y2, A.y2)$$
$$area_{intersect} = (\max(x2 - x1,0) + 1) * (\max(y2 - y1,0) + 1)$$
$$area_{union} = areaP + areaA - area_{intersect}$$
$$IOU = \frac{area_{intersect}}{area_{union}}$$

3. Dataset Description

We experiment with two sets of images from dataset Caltech-101 [10]. One set has 800 images of airplane and the other has 798 images of motorcycles. Fig. 4 shows some sample images from two sets of data. Each image has an annotation file in mat format. We could easily retrieve four numbers as x and y coordinates of the upper left corner and lower right corner of the bounding box from each annotation file.

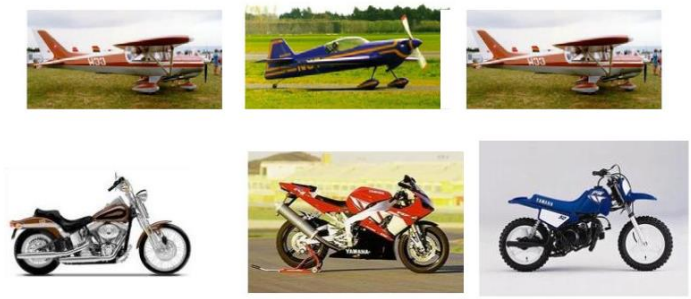

Fig. 4 Sample pictures from Caltech 101

4. Object Localization Based on Topological Data Analysis

We use the extend algorithm1 from [6] to generate a bounding box of interest in an image, and a segmented image as following:
Input: Image I
Result: Bounding Box and Processed Image, Segmented Image
G=Gray_Scale (I)
J=Zeros_like(I)
D= Persistent Diagram of G
Lifetimes = D.death-D.birth

Sort Lifetimes in descending order
Threshold=average of the two lifetimes with the largest difference
for birth,death as lifetimes>Threshhold and lifetime<inf
    C=All pixels connected between birth and death
    J[C]=1
Let col and row as two lists which holds maximum in each column and row of matrix J
We determine top bound using a loop:
 t=0
 while not row[t]:
     t+=1
We could bottom bound b, left bound l and right bound r in similar ways
B=[l, t, r, b]
$\bar{J}$=Invert(J)
G[$\bar{J}$]=0
S=I[$\bar{J}$]=0
End
Return B, G, S

TDA Algorithm: pseudocode to generate bounding box of an image based on its persistent diagram

An image is first converted in a GrayScale image, we then processed to generate a mask based homology diagram. We then use the mask to generate a box, a processed image , and a segmented the original image as shown in Fig. 5.

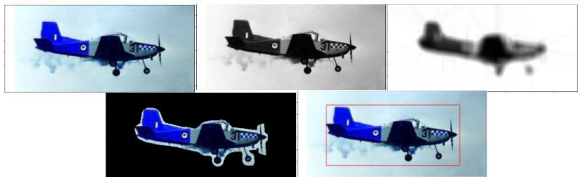

Fig. 5: The procedure of TDA algorithm

5. Experiments and Results

We conducted experiments using three different algorithms: TDA, Yolov7, and ViT. For each algorithm, we measured the average IOU for all images in the test set, and we also generated plots for a number of randomly selected images that showed the predicted box, annotated box, and IOU.

To experiment with Yolov7, we developed a Python program that generated an XML file similar to Fig. 6 for each annotation file in MAT format. We then loaded both the images and XML files into Roboflow [11] and trained them using Yolov7 [12] on Google Colab. We used the best.pt file from the training as the weights for our modified detection program, which calculated the IOU for each image and the average IOU for all images in the test set.

```
<?xml version="1.0"?>
- <annotaion>
    <folder>data/</folder>
    <filename>image_0005</filename>
  - <size>
      <width>395</width>
      <height>176</height>
      <depth>3</depth>
    </size>
    <segmented>0</segmented>
  - <object>
      <name>airplance</name>
      <pose>Unspecified</pose>
      <truncated>0</truncated>
      <difficult>0</difficult>
    - <bndbox>
        <xmin>48</xmin>
        <ymin>18</ymin>
        <xmax>339</xmax>
        <ymax>146</ymax>
      </bndbox>
    </object>
</annotaion>
```

Fig. 6: Sample of Annotation File in XML

Finally, we experimented with Vision Transformer [13]. We resized each image to 224 by 224 and used a patch size of 32 by 32. Each image was divided into 49 patches, and each patch had 3072 elements. We used the PatchEncoder layer to transform each patch into a vector with position embedding. The ViT model had multiple Transformer blocks, and we used the MultiHeadAttention layer for self-attention, which was applied to the sequence of image patches. The encoded patches and self-attention layer outputs were normalized and fed into a multilayer perceptron (MLP). The model outputted a vector representing the bounding box coordinates of an object with four coordinates.

We used the same training and testing sets for ViT as we did for Yolov7. The table 1 below shows the results of our experiments.

Table 1. Average IOU.

Data Set	TDA	ViT	Yolov7
Airplane	66%	90%	82%
Motorcycle	64%	83%	89%

Our TDA algorithm achieved similar IOU results in both datasets, with 64% and 66%. Some samples are shown in Fig. 7. These are decent results considering that our algorithm is unsupervised and does not require any training. Both Yolov7 and ViT also returned good results. ViT performed better on the Airplane dataset, while Yolov7 performed better on the Motorcycle dataset. However, it is important to note that both Yolov7 and ViT are supervised models that require annotated training sets.

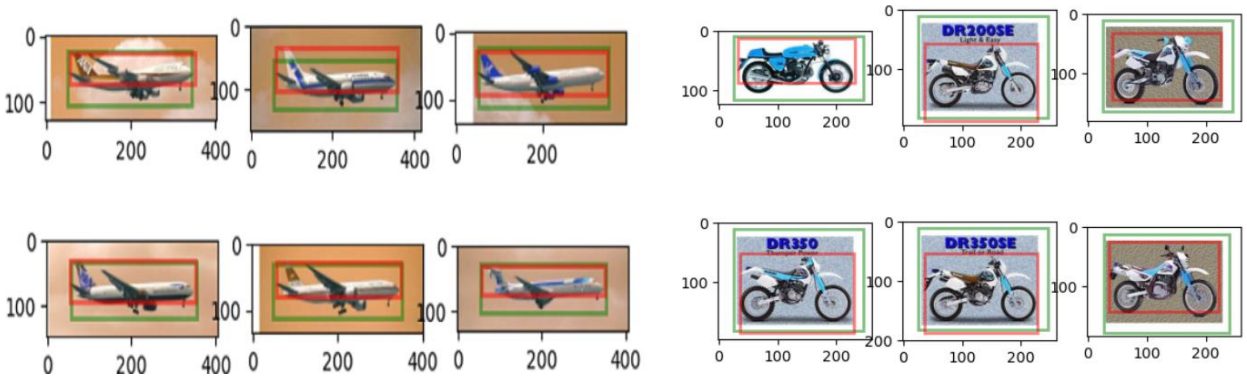

Fig. 7: Sample Results Using TDA (predicted boxes are in green lines and annotated boxes in red lines)

We have included 12 randomly selected examples from our test sets below in Fig. 8. The first row shows the results from Yolov7, where we have replaced the confidence number with an IOU number displayed in green on the top left. The second row displays the results from ViT, where we have printed the IOU in red on the bottom left.

Fig. 8: Samples Results with Yolov7 on the top and ViT on the bottom (predicted boxes are in green lines and annotated boxes in red lines)

6. Conclusion and Future Work

Our experiment has shown object localization based on topological data analysis has a good potential but its performance is still far below to results from supervised trained models such as YOLO or ViT. That is understandable, as supervised models are trained on a large amount of annotated data and are able to learn rich representations of the objects they are trained on. On the other hand, object localization based on topological data analysis is a relatively new approach that has not yet been widely used or researched, and its performance is still being improved.

We believe improving the TDA algorithm for object localization is a promising area of research, and there may be many ways to achieve this. For example, exploring different types of filtration could lead to new insights into the structure of the data and how it relates to object localization.

YOLOv7 is generally considered to be a faster and more efficient model for object detection, due to its single-shot design and anchor-based approach. However, ViT has a more powerful transformer-based architecture that allows it to learn rich representations of the objects it is detecting, and it has been shown to outperform YOLOv7 on one dataset in our experiments. We could not conclude which of Yolov7 and ViT is better in terms of object localization as each outperform the other in one set of images.

Ultimately, the best model for object localization depends on the specific requirements of our use case, such as the size of the objects we are detecting, the speed at which we need detections to be made, and the amount of computational resources we have available. If we have time and computational resources, it might be a good idea to try both models and compare their performance on your specific dataset, to see which one is best suited to our needs.

Similarly, experimenting with different variations of ViT is also a good idea. There have been many recent advancements in the field of transformer-based models, and these may provide new ideas for how to improve the performance of ViT for object localization. We could try using different types of self-attention mechanisms, or integrating other techniques such as convolutional neural networks or recurrent neural networks, to see if this leads to improved performance.

7. Acknowledgment and Disclaimer

This material is based upon work supported by the National Science Foundation (NSF) under Award
No. OIA-1946391. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
2. Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” https://arxiv.org/abs/1506.01497
3. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” https://arxiv.org/abs/1506.02640
4. Dosovitskiy, Alexey, et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale, Cornell University, 3 June 2021, https://arxiv.org/abs/2010.11929.
5. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In NIPS, 2017
6. Vandaele, R., Nervo, G.A. & Gevaert, O.,”Topological image modification for object detection and topological image processing of skin lesions,” Sci Rep 10, 21061 (2020). https://doi.org/10.1038/s41598-020-77933-y
7. Ghrist, R. Barcodes, “ the persistent topology of data. Bull. Am. Math. Soc,” 45, 61–75 (2008)
8. Carlsson, G.,”Topology and data. Bull. Am. Math,” Soc. 46, 255–308 (2009).
9. Ripser.py 0.6.4 documentation, “https://ripser.scikittda.org/en/latest/notebooks/Lower%20Star%20Image%20Filtrations.html”

10. Li, F.-F., Andreeto, M., Ranzato, M., & Perona, P. (2022). Caltech 101 (1.0) [Data set]. CaltechDATA. https://doi.org/10.22002/D1.20086
11. www.roboflow.com. Accessed Jan 10 2023.
12. Kin-Yiu, Wong, https://github.com/WongKinYiu/yolov7, Accessed Jan 10 2023.
13. Karan V. Dave. https://keras.io/examples/vision/object_detection_using_vision_transformer/ Accessed Jan 10 2023.