A Tagging Model using Segmentation Proposal Network

Suha Dh. Athab[1*], Abdulamir A. Karim[2]

[1,2] Department of Computer Science, University of Technology, Bagdad, Iraq

Emails: suha.athab@gmail.com ; 110004@uotechnology.edu.iq

Abstract

This paper presents a tagging model used the Segmentation map as reference regions. The suggested model leverages an encoder-decoder architecture combined with a proposal layer and dense layers for accurate object tagging and segmentation. The proposed model utilizes a pre-trained VGG16 encoder to extract high-level features from input images, followed by a decoder network that reconstructs the image. A proposal layer generates a binary map indicating the presence or absence of objects at each location in the image. The proposal layer is integrated with the decoder output and further refined by a convolutional layer to produce the final segmentation. Two dense layers are employed to predict object classes and bounding box coordinates. The model is trained using a custom loss function that combines categorical cross-entropy loss and means squared error loss. Experimental results demonstrate the effectiveness of the proposed model in achieving accurate object tagging and segmentation.

Keywords: Tagging; Encoder decoder; Semantic segmentation; Object detection

1. Introduction

Accurate object tagging plays a vital role in computer vision and finds extensive applications in various domains, including autonomous driving, image understanding[1], and object recognition. Semantic image segmentation, a fundamental task in this field, seeks to assign a class label to every pixel in an image [2]. By harnessing the capabilities of convolutional neural networks (CNNs), [3]. deep learning-based methods have achieved remarkable progress in semantic segmentation, empowering a detailed comprehension of an image's contents. In recent years, deep learning-based approaches have achieved remarkable success in semantic segmentation by leveraging the power of convolutional neural networks (CNNs)[4]. This paper proposes a new tagging model using a Segmentation Proposal Network (SPN) for object detection. Our model is designed to address the challenges of accurately localizing and classifying objects in complex scenes. It integrates an encoder-decoder architecture with a proposal layer and two dense layers to achieve robust and precise segmentation and localization results. The experimental results validate the effectiveness of the proposed model and highlight its potential for accurate object tagging and segmentation in computer vision applications. The introduction of Fully Convolutional Networks (FCNs)[5] marked a significant milestone in semantic segmentation. FCNs replaced the fully connected layers of traditional CNNs with convolutional layers, allowing for pixel-wise predictions. FCNs achieved impressive results by leveraging skip connections to fuse information from different scales, enabling both local and global context understanding. However, FCNs often struggle with precise object localization and suffer from class imbalance issues. Region-based segmentation methods were suggested to overcome the limitations of FCNs by combining proposal-based object detection with semantic segmentation. Selective Search, proposed at [6], is a widely used region proposal method that generates potential object regions based on color, texture, and size cues. These region proposals are then used to refine the segmentation results, improving both accuracy and localization. However, these methods are computationally expensive and may suffer from inaccurate proposal generation.

Attention mechanisms have gained significant attention in semantic segmentation research. These mechanisms aim to focus on informative image regions while suppressing irrelevant or noisy regions. The Spatial Pyramid

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

Pooling (SPP) module was introduced [7], which selectively aggregates features from different regions of varying scales. This allows for capturing multi-scale context and improves the segmentation performance. Similarly, [8] was proposed that incorporates spatial and channel-wise attention modules to enhance feature representations and refine segmentation results. Encoder-decoder architectures have been widely adopted in semantic segmentation models. The encoder component extracts high-level feature representations from the input image, while the decoder component reconstructs the segmentation map by up sampling the learned features. The U-Net architecture [9] introduced, which utilizes skip connections between the encoder and decoder to preserve fine-grained details during up sampling. This architecture has shown promising results in medical image segmentation tasks. More recent works, such as DeepLabv3 [10] and PSPNet [11], have further improved the encoder-decoder architecture by incorporating dilated convolutions and pyramid pooling modules, respectively. Proposal-based segmentation networks aim to leverage object proposals to guide the segmentation process and improve localization accuracy. The Fully Convolutional Instance-aware Semantic Segmentation (FCIS) model [12] was introduced, which combines proposal generation, instance segmentation, and semantic segmentation into a unified framework. FCIS achieves impressive results by explicitly considering object boundaries and sharing computation between proposal generation and segmentation tasks.

2. Methods

A tagging model that leverages a Segmentation Proposal Network (SPN) was proposed. The model achieves highly accurate object segmentation. The model integrates an encoder-decoder architecture with a proposal layer and dense layer predictions, combining the advantages of deep feature extraction, accurate semantic segmentation, and precise object localization. The suggested model architecture requires multiple layers as follows:

The first set of layers required to initialize the encoder. A pre-trained convolutional neural network (VGG16) architecture was used, where the top fully connected layers were excluded, to retain the pre-trained weights while training the additional layers for segmentation and object detection tasks. The encoder layers were initialized with the weights from the 'imagenet' dataset.

Then decoder layers were constructed, comprising five blocks composed of convolutional, dropout, and up-sampling layers. In the decoder, to predict the probability of each pixel in the original input image, the probability map needed to be up sampled to match the size of the decoder output. This compensated for any size-reduction caused by pooling operations in the encoder. The probability map was up sampled and then concatenated with the decoder output.

The second step was creating the proposal layer. The layer utilizes a CNN for processing input images and generating a probability map indicating the likelihood of each pixel being part of an object boundary. This was achieved by passing the output of the final convolutional layer through a 1x1 convolutional layer with a single filter and a sigmoid activation function. The resulting probability map ranged between 0 and 1 for each pixel, allowing object boundary segmentation by thresholding the probability values.

The concatenated output underwent further processing through another convolution layer followed by a Dropout layer. This step aimed to extract relevant features from the data. The output of the Dropout layer was flattened and passed through two Dense layers to predict the object class and bounding box coordinates. The object class was predicted using a Dense layer with a softmax activation function, while the bounding box coordinates were predicted using a Dense layer with a linear activation function.

The model was compiled with the custom loss function and Adam optimizer. A custom loss function combining categorical cross-entropy (CCE) loss for the object class prediction and mean squared error (MSE) loss for the bounding box coordinates prediction was introduced. The custom loss function using (1) was carefully designed to balance the two tasks and provide appropriate gradients for both during backpropagation.

$$loss = CCE(y_{true} + y_{class}) + MSE(y_{trueBox} + y_{Box}) \quad (1)$$

Where the MSE[13] can be represented using (2):

$$MSE = \frac{1}{N} \times \sum (y_{true} - y_{pred})^2 \quad (2)$$

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

N is the total number of data points, y_{true} represents the true values, y_{pred} represents the predicted values. The goal is to minimize the MSE loss to make the predicted values as close as possible to the true values. By penalizes larger differences between the predicted and true values more heavily, as it squares the differences While the CCE compares the predicted probability distribution over classes to the true one-hot encoded labels. Mathematically, CCE[14] can be represented using (3):

$$CCE = -\sum y_{true} \times \log(y_{pred}) \quad (3)$$

y_{true} represents the true class labels (one-hot encoded), y_{pred} represents the predicted class probabilities. It measures the dissimilarity between the predicted class probabilities and the true class labels. The CCE loss function encourages the predicted probabilities to be close to 1 for the true class and close to 0 for the other classes. The goal is to minimize the CCE loss to improve the accuracy of the predicted class probabilities.

3. Results

We conducted two experiments. The first experiment focused on segmentation, aiming to separate the different components within an image. The results of this initial experiment were highly promising, showcasing effective segmentation techniques. Buoyed by the success of the segmentation experiment, we proceeded to undertake the second experiment, which involved developing a comprehensive model capable of obtaining tags for the segmented images. By leveraging the insights gained from the initial segmentation experiment, this second experiment aimed to identify and label the various segments within an image accurately. Together, these two experiments allowed us to build upon the foundation of segmentation and expand the capabilities of the model to encompass the entire image analysis process. The fruitful outcome of the first experiment paved the way for the successful completion of the second, ultimately enabling us to achieve accurate tagging for segmented images.

3-1 Experiment 1

The first experiment focused on the segmentation part, specifically the encoder-decoder architecture. For this experiment, we trained the segmentation model using Microsoft Common Objects in Context (COCO) dataset[15]. The COCO dataset consists of a large collection of images with corresponding object annotations and masks. Firstly, the images and their corresponding masks need to be resized to a consistent resolution to ensure compatibility during training. Furthermore, it is important to pre-process the dataset by normalizing the pixel values of the images. This normalization step ensures that the input data has zero mean and unit variance, which helps stabilize the training process and improves convergence. Once the dataset is appropriately prepared, the model can be trained using the encoder-decoder architecture with proposal-based segmentation. The model is trained on the resized images and their corresponding masks. Fig. 1 shows an example of the segmentation experiment. The results of the segmentation experiment demonstrated the superiority of the encoder-decoder architecture. The model achieved higher segmentation accuracy and showed its effectively captured fine-grained details and accurately delineated object boundaries, showcasing its robustness and effectiveness in semantic image segmentation. The Intersection over Union (IoU) was used to demonstrate the results of this experiment. IoU can be mathematically represented using (4)[16].

$$IOU = \frac{1}{N}\sum_{i=1}^{N} \frac{y_i \hat{y}_i}{y_i + \hat{y}_i - y_i \hat{y}_i} \quad (4)$$

The binary value (label) of each pixel y_i, along with the predicted probability for that pixel, represented as $\hat{y}_i$, are crucial in evaluating the accuracy of semantic segmentation. To assess the alignment between the predicted boundary and the actual object boundary (ground truth), the Intersection over Union (IoU) metric is commonly employed. The mIoU for this experiment was 0.82 %

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

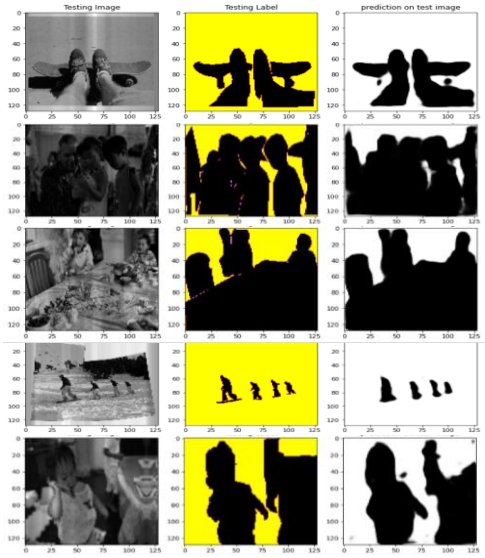


Figure 1: Segmentation Results for Randomly Chosen Samples: Input Image, Ground Truth, and Prediction

3-2 Experiment 2

Experiment 1 results indicate that the encoder-decoder architecture for semantic image segmentation was highly effective, providing strong evidence to use it as a region proposal method. The successful outcomes of this experiment served as a crucial encouragement to proceed with training the complete tagging model, we moved forward to the next phase of our research, which involved training the entire tagging model. By integrating the segmentation module with the rest of the architecture, we aimed to leverage the benefits of the region proposal approach and further enhance the model's overall performance. The model architecture was trained and tested using 8 k of the COCO dataset. The model trained for 10000 steps with SGD optimization algorithm and learning rate initial value 0.04. It is trained to detect objects from 80 different classes. The model was trained with a batch size of 4 with resized image of size (224×224×3). The training process involves iteratively feeding batches of training data through the model, calculating loss, and updating the model's weights through backpropagation. The model's performance in terms of training classification loss, localization loss, and total loss per 1000 steps are shown in Figures 2,3,4,5 respectively.

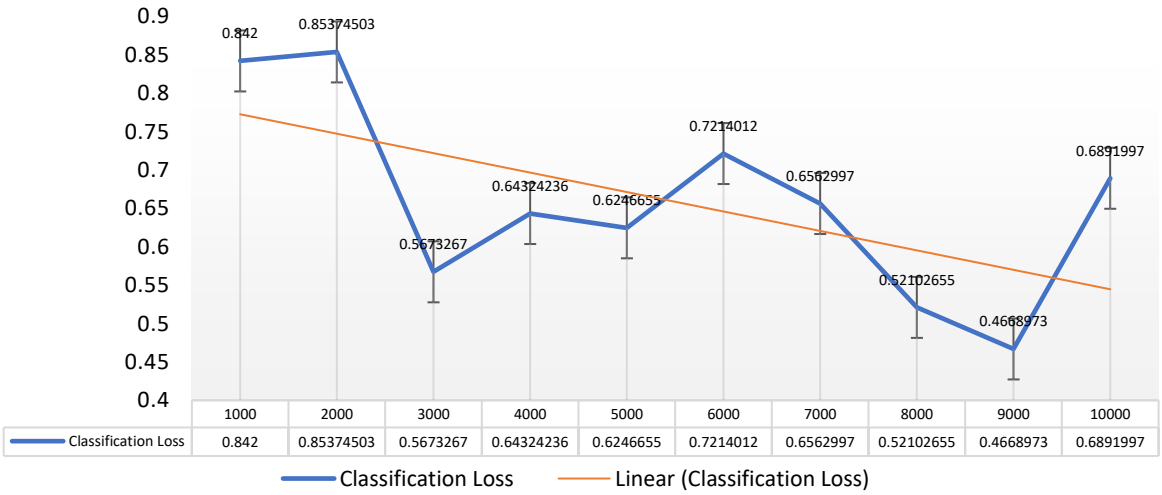


Figure 2: The classification loss per 1000 step for training tagging model using COCO dataset

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

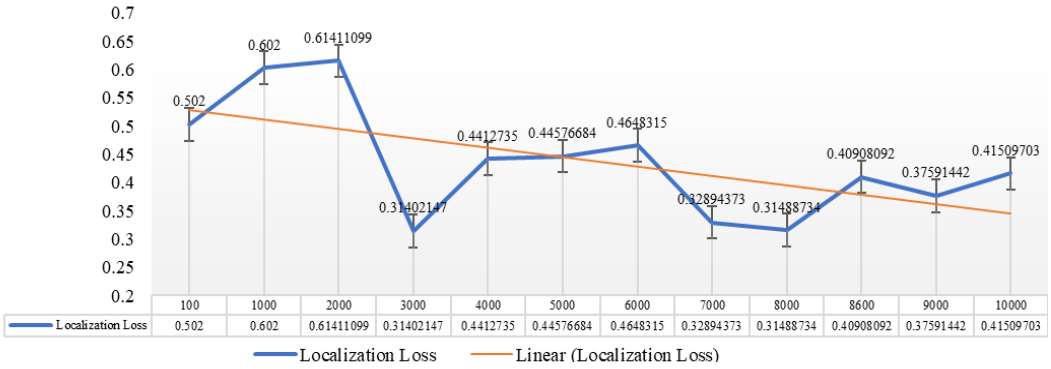


Figure 3: The localization loss per 1000 step for training tagging model using COCO dataset

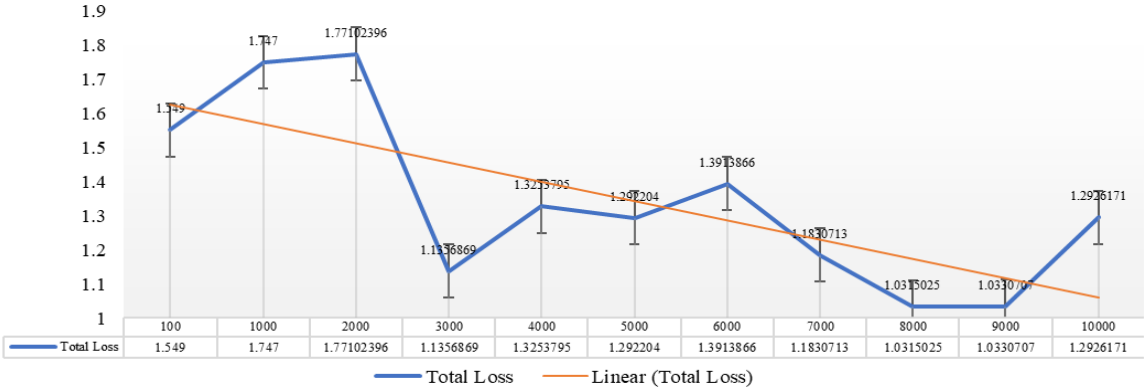


Figure 4: The total loss per 1000 step for training tagging using COCO dataset

The loss trends in the training process show fluctuations in classification loss throughout the training process. It generally decreases but with occasional increases. The minimum classification loss observed is 0.286746 (step 7800), and the maximum is 0.8538537 (step 2100).

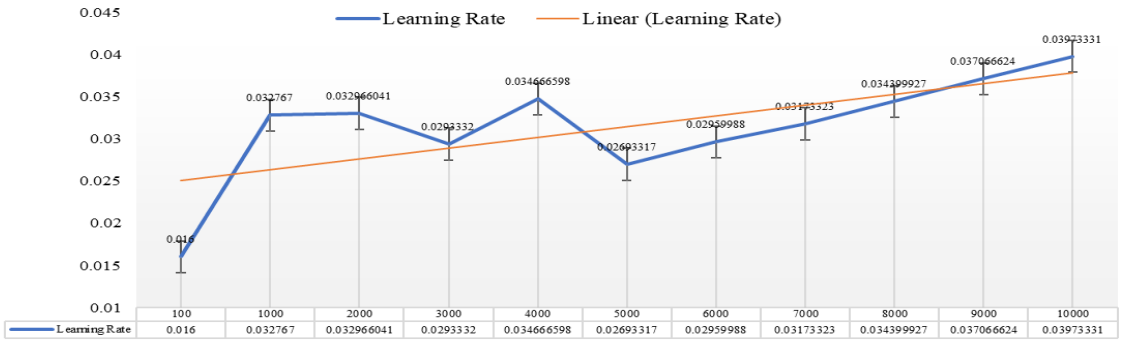


Figure 5: The learning rate per 1000 step for training tagging model using COCO dataset

Localization loss is like classification loss, it fluctuates but decreases overall. The minimum localization loss observed is 0.22438045 (step 5400), and the maximum is 0.6270897 (step 3800). Total loss is the sum of classification and localization loss. It follows a similar trend of fluctuation but generally decreases. The minimum

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

total loss observed is 0.7245022 (step 7800), and the maximum is 1.7712464 (step 2100). The learning rate increases initially and then decreases gradually as the training progresses. The optimal range of learning rate seems to be around 0.025 to 0.034, as it is within this range that the loss values are relatively lower. While there are fluctuations, the overall trend suggests that the model is improving since the loss values (both classification and localization) are generally decreasing over time. The final step at 10,000 shows a classification loss of 0.6892, localization loss of 0.4151, total loss of 1.2926, and a learning rate of 0.0397. This indicates that the model has improved significantly in terms of loss compared to earlier steps. The experimental results demonstrate the effectiveness of our model in achieving superior segmentation accuracy and localization performance as shown in Table (1)

Table 1: Comparative analysis for Tagging Model against the state-of-the-art methods using the COCO dataset

Method/year	**Description**	**Advantages**	**Disadvantages**	**Results (mAP)**
Mask R-CNN 2017[17]	It extends Faster R-CNN by adding a mask prediction branch alongside the existing bounding box regression and classification branches.	Accurate instance segmentation and object detection	Computationally expensive	0.64
FCOS: Fully Convolutional One-Stage Object Detection [18] 2019	FCOS is a one-stage object detection model that eliminates the need for anchor boxes. It directly predicts object bounding boxes and class scores using a fully convolutional approach.	End-to-end detection without anchor boxes	Challenging to handle small objects and dense scenes	0.37
PANet: Path Aggregation Network[19] 2018	PANet enhances feature representation in instance segmentation by employing a feature pyramid and path aggregation. It improves information flow across different network layers.	Better feature representation through feature pyramid and path aggregation	Complexity in design and training	0.76
HTC: Hybrid Task Cascade[20] 2019	HTC combines high-quality instance segmentation and object detection in a cascaded manner. It sequentially refines instance segmentation results for improved accuracy.	Accurate and high-quality instance segmentation and object detection	Increased complexity and memory requirements	0.69
Sparse R-CNN: End-to-End Object Detection with Learnable Proposals[21] 2020	Sparse R-CNN reduces computation and memory overhead through sparse convolution and adaptive sampling. It efficiently handles object detection with learnable proposals.	Reduced computation and memory overhead through sparse convolution and adaptive sampling	Lower recall rates for small objects and dense scenes	0.51
BorderDet: Border Feature for Dense Object Detection[22] 2020	BorderDet focuses on improving object boundary localization and detection performance. It enhances the detection of object boundaries.	Improved object boundary localization and detection performance	Increased complexity in design and training	0.53
AutoFocus: Efficient Multi-Scale Inference[23] 2023	AutoFocus achieves efficient multi-scale object detection and instance segmentation. It optimizes inference for objects at different scales.	Efficient multi-scale object detection and instance segmentation	Computational complexity	0.61
FoveaBox:Beyond Anchor-based Object Detector[24] 2019	FoveaBox focuses on accurate object detection with an emphasis on high-quality regions. It enhances detection in of the image.	Accurate object detection with a focus on high-quality regions	Increased computational complexity	0.53
RepPoints Point Set Representation for Object Detection[25] 2020	Important areas RepPoints represents objects efficiently using point-based representations. It focuses on point sets for object detection.	Efficient point-based object detection representation	Sensitivity to hyperparameter selection	0.62

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

Proposed method	Multi Object Tagging with dynamic anchor generation	a novel multi-object tagging model tailored for object detection,	Require careful parameter tuning and memory overhead	0.74

Table (1) results used as a reference point for assessing the performance of the suggested model. Each method is accompanied by a description of its advantages and disadvantages. Reported results in terms of mAP on the COCO dataset provided for each method. The table provides valuable insights into the landscape of computer vision methods applied to the COCO dataset. It reflects the continuous progress made in the field and the strengths and weaknesses of different approaches. The methods listed in the table span different years, highlighting the evolution of techniques. Figure (6) summarizes Table (1) Some methods, such as Mask R-CNN and PANet, achieve high accuracy but are computationally complex. Others, like YOLACT, prioritize real-time performance but may have slightly lower accuracy. Some methods leverage transfer learning from pre-trained models, such methods use backbone networks like ResNet. Additionally, architectural innovations, such as focal loss and feature pyramid networks (FPN), have been introduced to improve performance.

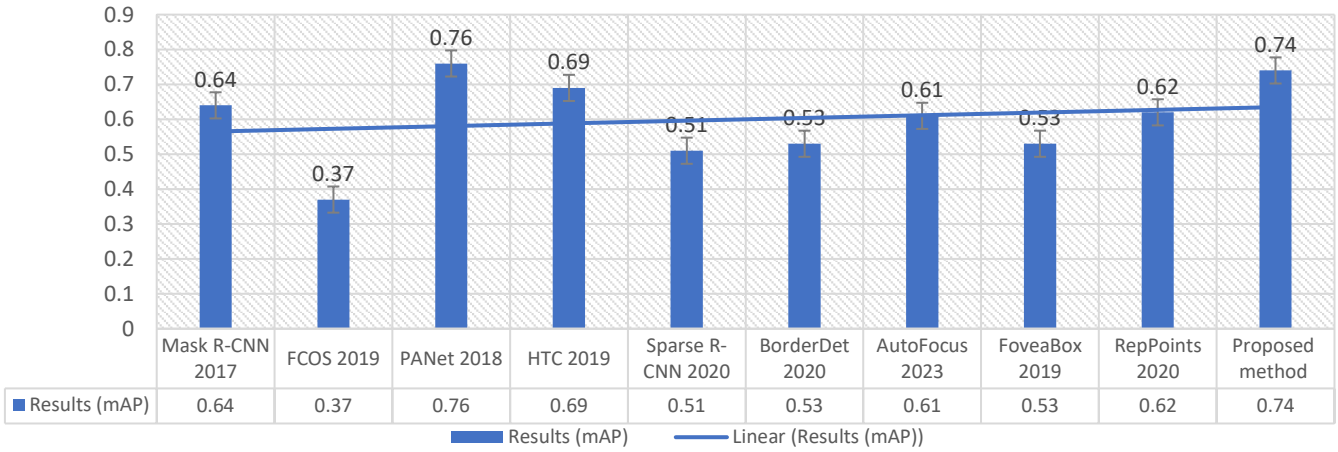
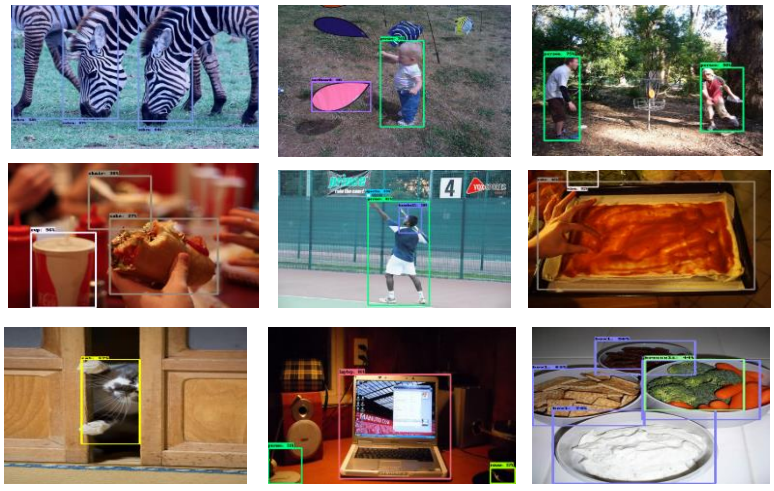


Figure 6: Comparative analysis between suggested method and state of arts methods using COCO dataset

After training the model for 10,000 steps, the performance of the trained model was evaluated on a test part of the COCO dataset. Figure (7) shows some random visual examples highlighting tagging results for the proposed Tagging Model



Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

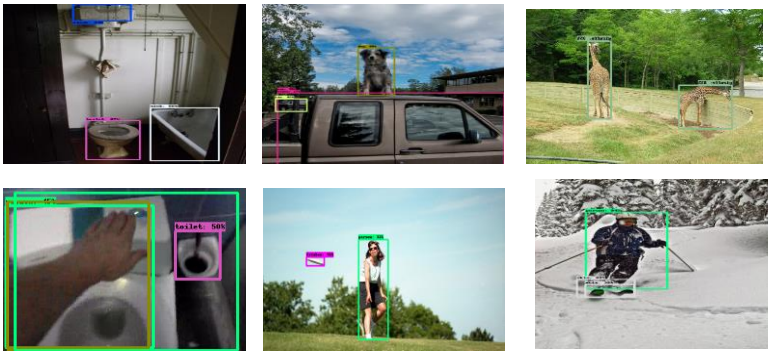


Figure 7: Tested result for the proposed tagging model using random samples from COCO dataset

4. Conclusion

The proposed tagging model in this paper leverage the power of deep learning, encoder-decoder architecture, proposal-based localization, and dense layer predictions. Through extensive experiments, we validate the effectiveness and robustness of our model, positioning it as a promising approach for various computer vision applications that require fine-grained segmentation and accurate object localization. The encoder component of our model extracts rich and discriminative features from the input image, while the decoder component reconstructs the image by up sampling these features, resulting in high-resolution segmentation maps with accurate object boundaries and intricate details. To address the challenge of object localization, we introduce a proposal layer that generates a binary map indicating the presence or absence of an object at each location in the image. This proposal map provides crucial spatial context and guides the segmentation process towards object regions. By fusing the proposal information with the decoder's output, our model enhances its ability to accurately segment objects while maintaining overall contextual understanding. The fusion is achieved by concatenating the proposal map with the decoder's features and passing them through another convolutional layer.

References

[1] Y. Li, L. Yuan, and N. Vasconcelos, "Deep Hierarchical Semantic Segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 1-10.

[2] S. Mehta and M. Rastegari, "Simple and Efficient Architectures for Semantic Segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022, pp. 1-10.

[3] F. Lateef and Y. J. N. Ruichek, "Survey on semantic segmentation using deep learning techniques," vol. 338, pp. 321-348, 2019.

[4] J. M. Stokes *et al.*, "A deep learning approach to antibiotic discovery," vol. 180, no. 4, pp. 688-702. e13, 2020.

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431-3440.

[6] T. Kong, A. Yao, Y. Chen, and F. Sun, "Hypernet: Towards accurate region proposal generation and joint object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 845-853.

[7] J. H. Giraldo et al., "Hypergraph Convolutional Networks for Weakly-Supervised Semantic Segmentation," *arXiv preprint arXiv*:2210.05564, 2022.

[8] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3146-3154.

[9] A. Aakerberg and M. Felsberg, "Semantic Segmentation Guided Real-World Super-Resolution," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2022, pp. 1-10.

[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510-4520.

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023

[11] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881-2890.
[12] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3150-3158.
[13] E. Temlioglu, I. Erer, and D. Kumlu, "A least mean square approach to buried object detection in ground penetrating radar," in *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2017, pp. 4833-4836: IEEE.
[14] Z. Zhang and M. J. A. i. n. i. p. s. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," vol. 31, 2018.
[15] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 2014, pp. 740-755: Springer.
[16] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proceedings of the AAAI conference on artificial intelligence*, 2020, vol. 34, no. 07, pp. 12993-13000.
[17] Z. Hao *et al.*, "Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (Mask R-CNN)," vol. 178, pp. 112-123, 2021.
[18] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as Points," in *arXiv preprint arXiv*: 1904.07850, 2019.
[19] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759-8768.
[20] K. Chen *et al.*, "Hybrid task cascade for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4974-4983.
[21] P. Sun *et al.*, "Sparse r-cnn: End-to-end object detection with learnable proposals," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14454-14463.
[22] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 3, pp. 1089-1102, 2020, doi: 10.1109/TPAMI.2019.2951682.
[23] Y. Chen et al., "YOLO-MS: Rethinking Multi-Scale Representation Learning for Real-time Object Detection," *arXiv preprint arXiv*:2308.05480, 2023.
[24] T. Kong, F. Sun, H. Liu, Y. Jiang, and J. J. a. p. a. Shi, "FoveaBox: Beyond anchor-based object detector. arXiv 2019," vol. 2, no. 5.
[25] F. Wei, X. Sun, H. Li, J. Wang, and S. Lin, "Point-Set Anchors for Object Detection, Instance Segmentation and Pose Estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 527–544.

Doi: https://doi.org/10.54216/FPA.130212
Received: April 15, 2023 Revised: July 26, 2023 Accepted: October 08, 2023