GMR Institute of Technology
An Autonomous Institute Affiliated to JNTUK, Kakinada

GMRIT
Training Tomorrow's
Engineers Today

Visual-Semantic Alignment using Zero Shot object detection

A project report submitted in partial fulfilment of the requirement

for the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted by

M. Subhashini	**(20341A05B2)**
K. Aswin Kumar	**(20341A0578)**
M. Madhan Sai	**(20341A05C5)**
M.V.S. Ajit	**(20341A05B9)**
K.V. Ranjith Varma	**(20341A05A7)**

Under the esteemed guidance of

Mr. V. Abishek Heyer
Assistant Professor, Dept. of CSE

GMR Institute of Technology

An Autonomous Institute Affiliated to JNTUK, Kakinada

(Accredited by NBA, NAAC with 'A' Grade & ISO 9001:2008 Certified Institution)

GMR Nagar, Rajam – 532127,
Andhra Pradesh, India
April 2020

GMR Institute of Technology
An Autonomous Institute Affiliated to JNTUK, Kakinada

GMRIT
Training Tomorrow's Engineers Today

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the thesis entitled **Visual-Semantic Alignment using Zero Shot object detection** submitted by **M. Subhashini(20341A05B2), K. Aswin Kumar(20341A0578), M.Madhan Sai(20341A05C5),M.V.S. Ajit(20341A05B9), K.V. Ranjith Varma (20341A05A7)** has been carried out in partial fulfilment of the requirement for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** of **GMRIT, Rajam** affiliated to **JNTUK, KAKINADA** is a record of bonafide work carried out by them under my guidance & supervision. The results embodied in this report have not been submitted to any other University or Institute for the award of any degree.

Signature of Supervisor
Mr. V. Abishek Heyer
Assistant Professor
Department of CSE
GMRIT, Rajam.

Signature of HOD
Dr. A. V. Ramana
Professor & Head
Department of CSE
GMRIT, Rajam.

The report is submitted for the viva-voce examination held on ………………..

Signature of Internal Examiner

Signature of External Examiner

ACKNOWLEDGEMENT

It gives us an immense pleasure to express deep sense of gratitude to my guide **Mr.V.Abishek Heyer,** Assistant/Associate/Professor, Department of Computer Science and Engineering for his whole hearted and invaluable guidance throughout the project work. Without his sustained and sincere effort, this project work would not have taken this shape. He encouraged and helped us to overcome various difficulties that we have faced at various stages of our project work.

We would like to sincerely thank our Head of the department **Dr. A. V. Ramana**, for providing all the necessary facilities that led to the successful completion of our project work.

We would like to take this opportunity to thank our beloved Principal **Dr.C.L.V.R.S.V.Prasad**, for providing all the necessary facilities and a great support to us in completing the project work.

We would like to thank all the faculty members and the non-teaching staff of the Department of Electronics and Communication Engineering for their direct or indirect support for helping us in completion of this project work.

Finally, we would like to thank all of our friends and family members for their continuous help and encouragement.

M. Subhashini (20341A05B2)
K. Aswin Kumar (20341A0578)
M. Madhan Sai (20341A05C5)
M.V.S. Ajit (20341A05B9)
K.V. Ranjith Varma (20341A05A7)

ABSTRACT

Object detection is a computer vision task that involves identifying and localizing objects in images or videos. Object detection models require a large amount of labelled data in order to be trained effectively. This can be time-consuming and expensive to obtain. The object detector will only be able to recognize objects that it has been trained on. It will not be able to recognize novel objects that it has not seen during training. This problem can be achieved by using Zero shot which is a Machine learning approach. The Zero-shot object detection has the potential to significantly reduce the amount of labelled training data required for object detection tasks. In this, we are going to detect an object using Zero shot learning. It typically works by learning a visual semantics of the objects, where semantics can be used to represent the characteristics and attributes of different objects, and it is used to identify the relationships between different classes of objects. These visual semantics can be done using Deep neural networks. This method typically involves training the model on a large dataset of images or videos. Finally, this method is used to classify objects based on their similarity to known classes.

Keywords: Object Detection, Zero-shot, Deep Neural Networks, Machine Learning, Visual Semantics

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS & ABBREVIATIONS

APN	:	Attribute Prototype network
CL	:	Computational linguistics
CLIP	:	Contrastive language image pre-training
FPN	:	Feature Pyramid Network
GCN	:	Graphical convolutional Network
ILSVRC	:	ImageNet large Scale visual recognition challenge
NAS	:	Neural architecture search
RN	:	Relation Network
VAN	:	Variational Auto Encoder
VIT	:	Vision Transformers
VMAN	:	Virtual mainstay alignment network
WGAN	:	Wasserstein Generative Adversarial network
ZSD	:	Zeroshot detection
ZSL	:	Zeroshot Learning

INTRODUCTION

Existing deep learning models typically perform well when trained on a significant amount of labeled data in a supervised learning environment. The study of learning under reduced supervision is a significant research challenge since large-scale supervision is expensive and difficult to achieve in many real-world circumstances. Since people can learn with relatively little supervision, human learning serves as an extraordinary motivator in this endeavour. Zero-shot learning (ZSL), which is inspired by human learning, tries to use only an object's semantics to reason about something it has never seen before and now we discuss that in detailed manner.

Visual semantic alignment is the process of linking visual concepts in images or videos with their corresponding semantic concepts, such as objects, scenes, or events. This can be useful for a variety of applications, including image and video classification, object recognition, and scene understanding. Zero shot object detection is a method of object detection that allows a model to detect novel object classes that were not present in the training data. This is achieved by training the model to recognize the visual features of each object class, as well as the semantic relationships between the object classes and their corresponding labels.

Using zero shot object detection for visual semantic alignment allows the model to detect and classify objects in images or videos, even if it has not seen those specific object classes during training. This can be useful in situations where the training dataset is limited or where the target domain contains a large number of novel object classes. To perform visual semantic alignment using zero shot object detection, a model is first trained on a large dataset of annotated images containing a diverse set of object classes.

The model is then fine-tuned to be able to detect novel object classes that were not present in the training dataset. The trained model can then be used to detect and classify objects in new images or videos, and the detected objects can be aligned with their corresponding semantic concepts. The performance of the visual semantic alignment method can be evaluated by comparing the aligned concepts with ground truth annotations.

RELATED WORK/ THEORETICAL STUDY

[1] Liu, J., Chen, Y., Liu, H., Zhang, H., & Zhang, Y. (2022). From Less to More: Progressive Generalized Zero-Shot Detection With Curriculum Learning. ***IEEE Transactions on Intelligent Transportation Systems*****.**

1) One of the main contributions is that they proposed a new generative model combining WGAN and CL of GZSD. CL is suitable for ZSD. Simulate multiple teachers to monitor each part of the generator to enhance its generalization ability and make the generated suggestions more realistic.

2) They presented the idea of meta-learning for ZSD. During training, some of the initially visible classes are used to simulate hidden classes. This allows the generator to achieve better generalization capabilities.

3) They proposed the idea of using curriculum learning to generate more accurate invisible visual features using the Faster-RCNN method.

4) His two datasets are used: the MSCOCO dataset and the traffic scene dataset KITTI. Those measured by several performance metrics such as recall, average precision, and HP scored 83% and 87% performance.

[2] Yang, Y., Zhao, L., & Liu, X. (2022). Iterative Zero-Shot Localization via Semantic-Assisted Location Network. ***IEEE Robotics and Automation Letters*****,** ***7*****(3), 5974-5981.**

1) The paper treats image-based zero-shot localization as a classification task and focuses mostly on this technique.

2) They created a model that can transfer semantic knowledge from classes that can be seen to classes that can't be seen.

3)In order to increase the consistency of the inter-class relationship between class embeddings and picture embeddings, they also created an iterative zero-shot training framework based on the EM method.

4) Compared to other cutting-edge zero shot localization techniques, the suggested iterative method outperforms them all.

5) Using the mean Average Precision (mAP) on the ICUBE dataset, the results of objection detection using YOLO v4 are 62.32%.

[3] Mao, Q., Wang, C., Yu, S., Zheng, Y., & Li, Y. (2020). Zero-shot object detection with attributes-based category similarity. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *67*(5), 921-925.

1) A new approach that uses attributes-based category similarity is briefly explained in this work.

2)To analyse and modify an attribute table and improve the synergy between the visual and semantic domains, an unsupervised learning approach is used.

3) The suggested strategy the goal of ACS-ZSD is to represent a picture semantically in the space of visual qualities. The zero shot object detector is built using the Retina Net architecture.

4) It has been accomplished that this algorithm's AP for the majority of classes is higher than that of the alternative techniques.

5) It demonstrates that Average Precision is somewhat low, but that the recall rate is very high (78.2%).

[4] Yan, C., Chang, X., Li, Z., Guan, W., Ge, Z., Zhu, L., & Zheng, Q. (2021). Zero nas: Differentiable generative adversarial networks search for zero-shot learning. IEEE transactions on pattern analysis and machine intelligence.

1) In this article, they turn to Neural Architecture Search (NAS) and make the first attempt to bring NAS technology into the ZSL realm.

2) Benchmark datasets show that Zero NAS can discover desirable architectures that perform well for the latest his ZSL and generalized zero-shot learning (GZSL) approaches.

3) I have a table showing the performance of ZSL and GZSL on the benchmark dataset.

4) They developed a model that focused on 1) the relevance and balance between generators and discriminators under search. 2) An expressive MLP search space for feature synthesis. 3) Continuous relaxation of the search space for efficient differentiable searches.

5) Compared with f-CLSGWAN with handcrafted architecture, the proposed method achieves up to 2.9%, 3.0%, 2.5%, and 3.5% improvement in CUB.

FLO, SUN, or AWA.

[5] Yan, C., Zheng, Q., Chang, X., Luo, M., Yeh, C. H., & Hauptman, A. G. (2020). Semantics-preserving graph propagation for zero-shot object detection. IEEE Transactions on Image Processing, 29, 8163-8176.

1) In this study, they investigated zero-shot object detection in the context of zero-shot learning (ZSD).

2) A cutting-edge Semantics Preserving Graph Propagation Model for ZSD Based on Graph Convolutional Networks (GCN).

3) It includes comparisons of different AP and MAP approaches on a range of datasets.

4) Because our model uses a multi-step graph propagation method, it is able to successfully reduce the visual-semantic gap, which is a major barrier to the effectiveness of ZSD systems based on the direct mapping-transfer strategy.

5) Our approach yielded a 24.8% mAP. The suggested model improves recall above IoU thresholds of 0.4, 0.5, and 0.6, respectively, by 1.3%, 3.2%, and 7.1% when compared to the other method.

[6] Wei, K., Deng, C., Yang, X., & Tao, D. (2021). Incremental zero-shot learning. IEEE Transactions on Cybernetics.

1) The first attempt to introduce and address IZSL, which more effectively connects real-world needs with computer vision building blocks, was this work.

2) To transform IZSL into traditional ZSL, a generative replay approach was used to amass historical knowledge of previously encountered classes.

3) On three benchmark datasets, experiments revealed that the technique performs significantly better than the competition.

4) To confirm that the two suggested strategies are crucial to achieving good performance, an ablation study was also carried out.

5) The model outperforms the competition on CUB by 7.72%, scoring 59.36%. The model outperforms the competition on FLO by 4.98%, scoring 79.49%. The model outperforms the competition on SUN by 1.63%, achieving 32.25%.

[7] Tian, Y., Zhang, Y., Huang, Y., Xu, W., & Ding, Z. (2022). Differential Refinement Network for Zero-Shot Learning. *IEEE Transactions on Neural Networks and Learning Systems*.

1) For the ZSL job, they suggested a unique DRNet in this study. To investigate effective semantic-visual relationships, a two-branch network with a basic network and a differential network was designed.

2) By comparing numerous prototypes, it will be possible to characterise interactions between distinct categories, which will aid in the development of real and discriminative visual centres.

3) They tested the effectiveness of the DRNet technique using four well-known datasets: OxfordFlowers (FLO) [35], Caltech-UCSD Birds-200-2011 (CUB) [47], Animals with Attributes

(AwA1) [22], and Animals with Attributes2 (AwA2) [48].

4) DRNet outperforms those existing meta-methods by 0.8% and 0.7% on the AWA2 and FLO datasets, respectively, and performs similarly on the AWA1 and CUB datasets.

[8] Lv, W., Shi, H., Tan, S., Song, B., & Tao, Y. (2022). A dynamic semantic knowledge graph for zero-shot object detection. *The Visual Computer*, 1-15.

1) In this article, they turn to Neural Architecture Search (NAS) and make the first attempt to bring NAS technology into the ZSL realm.

2) Benchmark datasets show that Zero NAS can discover desirable architectures that perform well for the latest his ZSL and generalized zero-shot learning (GZSL) approaches.

3) I have a table showing the performance of ZSL and GZSL on the benchmark dataset.

4) They developed a model that focused on 1) the relevance and balance between generators and discriminators under search. 2) An expressive MLP search space for feature synthesis. 3) Continuous relaxation of the search space for efficient differentiable searches.

5) Compared with f-CLSGWAN with handcrafted architecture, the proposed method achieves up to 2.9%, 3.0%, 2.5%, and 3.5% improvement in CUB.

FLO, SUN, or AWA.

[9] Yan, C., Chang, X., Luo, M., Liu, H., Zhang, X., & Zheng, Q. (2022). Semantics-guided contrastive network for zero-shot object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

1) They created a unique mapping contrastive method that is superior to the traditional mapping-transfer strategy and a semantics-guided contrastive network for ZSD. We believe that this is the first study to apply the contrastive learning mechanism for ZSD.

2) To better align the region features and the associated semantic descriptions, the proposed deep model integrates both region category and region-region contrastive learning.

3) Contrast ZSD, a detection framework that introduces contrastive learning mechanisms into the field of zero-shot detection, is a semantics-guided contrastive network for ZSD.

4) Extensive experiments are carried out on the two well-known ZSD benchmarks, PASCAL VOC and MS COCO.

5)The suggested Contrast ZSD performs significantly better than the second-best approach BLC, with an absolute HM performance gain of 6.9% Recall@100.

[10] Du, P., Zhang, H., & Lu, J. (2020). Learning Discriminative Projection With Visual Semantic Alignment for Generalized Zero Shot Learning. *IEEE Access*, *8*, 166273-166282.

1) proposed a new method to solve the domain shift problem by learning discriminative projections with visual semantic alignments in the latent space.

2) A linear discriminant analysis strategy is used to learn projections from the visual space to the latent space. This allows the projected features in the latent space to be more identifiable.

3) They assumed that each prototype is a linear sparse combination of other prototypes in all three spaces, including visual, latent and semantic, and that the sparsity coefficients are the same in all three spaces. This strategy creates links between visible and invisible classes, reduces the domain gap between them, and finally solves the domain shift problem.

4) Extensive experiments are performed on 5 common datasets

5) This method was performed using mAP and yielded results of 80% for SUN, 40% for CUB, 65% for AWA1, 57% for AWA2 and 52% for APY.

[11] Liu, Y., Dang, Y., Gao, X., Han, J., & Shao, L. (2022). Zero-Shot Learning With Attentive Region Embedding and Enhanced Semantics. *IEEE Transactions on Neural Networks and Learning Systems*.

1) They combined embedding and ZSL generative models to propose a novel autoencoder paradigm-based framework in which task-specific feature learning and invisible pattern generation are performed together.

2) The proposed model captures semantic-based visual features and enhanced semantics by AM and DS, respectively, effectively reducing the impact of the projective domain shift problem.

3) Finally, the proposed AREES model has been extensively evaluated against six well-known ZSL benchmarks, and encouraging results show that AREES is effective in processing both standard ZSL and GZSL tasks. indicates that there is

4) The proposed model significantly improves the state of the art for all segmentations. B. 657 4.00%/5.21%/4.82% improvement in accuracy compared to the strongest competitor CADA-VAE for 2-hop/M500/L5K split.

[12] Li, Y., Liu, Z., Yao, L., & Chang, X. (2021). Attribute-modulated generative meta learning for zero-shot learning. *IEEE Transactions on Multimedia*.

1)Zero-shot learning (ZSL) seeks to recognise unseen classes, that is, classes that are not present in the training process, by inferring a classification model from observed classes, that is, classes with labelled samples that are present in the training process.

2)This study uses an attribute-modulated generative meta-model (AMAZ) to create visual features for ZSL classes that haven't been seen before.

3)They suggest a unique Attribute-Modulated Generative Meta-Model for Zero-shot Learning that generates features dynamically (AMAZ).

4)In order to improve the generative adversarial network and meta-learning, AMAZ uses an attribute-aware modulation network. It combines the benefits of reducing biases against perceived class and supporting various tasks.

5)They added data quality to offer additional direction to a weighted classifier. Performance on all datasets is enhanced with the weighted classifier, particularly in SUN (3.6%).

[13] Xie, G. S., Zhang, X. Y., Yao, Y., Zhang, Z., Zhao, F., & Shao, L. (2021). Vman: A virtual mainstay alignment network for transductive zero-shot learning. *IEEE Transactions on Image Processing*, *30*, 4316-4329.

1)They proposed a simple and effective approach to generate a virtual main support (VM) sample to tackle the transductive ZSL (TZSL) problem.

2)In this document, there is only one parameter matrix to learn. Suppose H.W.W was learned by the WED/VMAN algorithm.

3)For a more realistic generalized he ZSL setting, we evaluated the performance of WED/VMAN on AWA, CUB, and SUN. Model training with his PS settings above only used part of the displayed image.

4) The proposed Virtual Mainstay Alignment Network (VMAN) can seamlessly solve the TZSL problem. Extensive evaluation shows that VMAN has reached new levels of technology in most of the standard benchmarks used.

5)The TZSL embedding method made a significant difference, with VMAN scoring the highest on AWA at 89.3%. VMAN performance is up to 69.3% on the difficult SUN dataset.

[14] Zhao, P., Zhang, S., Liu, J., & Liu, H. (2021). Zero-shot Learning via the fusion of generation and embedding for image recognition. *Information Sciences*, *578*, 831-847.

1)In this paper, the authors present a new ZSL model based on Double Latent Subspace Learning with Class Prototyping and Reconstruction (ZSL-CPLSR) that integrates generation and embedding into a unified framework.

2)We conducted a large-scale experiment to evaluate ZSL-CPLSR and adopted four benchmark datasets widely used in ZSL. Animals with attributes (AWA), CUB-200-2011 birds (CUB), Pascal -a Yahoo (aP & Y), and SUN attributes.

3)The authors elaborated on his proposed ZSL based on double latent subspace learning with class prototypes and reconstruction (ZSL-CPLSR).

4)Zero-shot learning (ZSL) has emerged to solve the above difficult task by mimicking the human ability to recognize objects from unseen classes using side information. increase.

5)ZSL has become an active research topic and can be used in many applications, such as: B. Rare species detection and novel virus detection.

[15]Xu, W., Xian, Y., Wang, J., Schiele, B., & Akata, Z. (2020). Attribute prototype network for zero-shot learning. *Advances in Neural Information Processing Systems*, *33*, 21969-21980.

1)In this work, they developed a zero-shot representation learning framework. H. Attribute Prototype Networks (APNs) that collectively learn global and local characteristics.

2)In this paper, they present an attribute prototype network that excels at predicting attributes with local features.

3)To measure the impact of each model component on the extracted image representation, they designed an ablation study. In this study, we train one basic mod with cross-entropy loss as a baseline, and with a proto mod and three loss functions he trains three variants of APN. Add gradually.

4)They compared the APN to his two sets of latest models. Non-generative models i. H. SGMA, AREN, LFGAA + hybrid and generative models, H. Lis GAN, CLSWGAN, and ABP, in ZSL and GZSL settings, for 86.8% in binary attributes of the displayed classes, It produced 86.4% with continuous attributes.

S.No	References	year	Objectives	Limitations	Advantages	Performance metrics	Gaps
1	Rahman, S., Khan, S., & Barnes, N. (2022). Polarity Loss: Improving Visual-Semantic Alignment for Zero-Shot Detection. IEEE Transactions on Neural Networks and Learning Systems.	2022	In this they proposed a novel loss function called polarity loss that promotes correct visual semantic alignment for an improved ZSD. They also proposed ZSD framework which is specially designed to work with single-stage detectors like Retina Net.	Fixed representations with a fixed embedding Words, the network cannot update the semantic representations and has limited flexibility to properly align visual and semantic domains.	1)The proposed PL promoted the correct alignment between visual and semantic domains. 2) They also demonstrate d that in the learned semantic embedding space, word vectors.	The methodology used the two datasets are MS-COCO and Pascal VOC 2007 datasets and evaluated using the Average precision, Recall and gives an accuracy of 93% and 76% respectively.	The approaches that are used here are not end-to-end trainable. Furthermore, they only investigate the recognition problem.
2	Yang, Y., Zhao, L., & Liu, X. (2022). Iterative Zero-Shot Localization via Semantic-Assisted Location Network. IEEE Robotics and Automation Letters, 7(3), 5974-5981.	2022	1)The paper mainly focuses on image based zero-shot localization and treat it as classification task. 2)They also designed a model which learn semantic information that can be transferred from seen classes to unseen classes.	It has attention mechanism in its message passing module, but its performance on WCP dataset is not significantly improved.	The proposed iterative method performs best among the state-of-the-art zero shot localization methods.	The results of objection detection by using YOLO v4 are using the mean Average Precision (m AP) on ICUBE dataset is 62.32%.	In the future, one should collect some datasets ourselves to evaluate the methods further
3	Mao, Q., Wang, C., Yu, S., Zheng, Y., & Li, Y. (2020). Zero-shot object detection with attributes-based	2020	1)This paper briefly explains a new algorithm using attributes based category similarity. 2) An unsupervised	In this methodology it is very difficult to detect the hair dryer from the trained model.	It have achieved that the AP of most of the classes in this algorithm which is higher than that of the	It shows that Average Precisionism a bit low, but with a very high recall rate (78.2%).	In the future, one should further improve the network structure to make it more adaptable to the zero

	category similarity. IEEE Transactions on Circuit and Systems II:Express Briefs, 67(5), 921-925.		learning method is utilized to evaluate and adjust an attribute table, which helps to establish a better synergy between visual and semantic domains.		other methods.		sample setting.
4	Liu, J., Chen, Y., Liu, H., Zhang, H., & Zhang, Y. (2022).From Less to More: Progressive Generalized Zero-Shot Detection With Curriculum Learning. IEEE Transactions on Intelligent Transportation Systems.	2022	They proposed an idea of using Curriculum learning to generate More precise unseen visual features using Faster-RCNN method to accomplish zero-shot object detection.	1)Existing methods did not enhance the prediction of the coordinates for the given proposals. 2)Its performance in traffic scenes relies too much on the diversity and accuracy of proposals extracted by Faster-RCNN.	During training, They utilized part of the original seen classes to simulate unseen classes. Through it, the generator achieved better generalization ability and also achieved state of-the-art performance with multiple splitting strategies.	In this, the two datasets are used MSCOCO dataset and the traffic scene dataset KITTI.Those evaluated using someperformance metrics like recall, Average precision, Hp etc. and achieved 83% and 87% performance.	Further they need to consider how to enhance the capability of the regressor in detector and how to obtain more refined semantics about the traffic objects.
5	Yan, C., Zheng, Q., Chang, X., Luo, M., Yeh, C. H., & Hauptman, A. G. (2020). Semantics-preserving graph propagation for zero-shot object detection. IEEE	2020	In this paper, They explored object detection in the context of zero-shot learning and A novel Semantics Preserving Graph Propagation model for ZSD based on	In this method it cannot recognises some of the images in the MSCOCO dataset.	By using the multi-step graph propagation process, our model can effectively mitigate the visual-semantic gap, which is a key factor impeding the performance	The mAP achieved by our method is 24.8%. Compared with the other method, the proposed model gains 1.3%, 3.2%and 7.1% improvement in recall over	In the future, we intend to further improve our model in order to relieve this problem.

	Transactions on Image Processing, 29, 8163-8176.		Graph Convolutional Networks (GCN) was proposed.		of ZSD approaches based on the direct mapping-transfer strategy.	IoU thresholds 0.4, 0.5and 0.6 respectively.	
6	Sun, X., Gu, J., & Sun, H. (2021). Research progress of zero-shot learning. Applied Intelligence, 51(6), 3600-3614.	2021	In this paper, the development of ZSL is reviewed comprehensively, including the evolution, key technologies, mainstream models, current research hotspots and future research directions.	There are some problems, such as semantic loss in ZSL and samples collapsing to a point through mapping.	The difficulty of ZSL is analyzed from the perspective of the evolution of network architectures . Second, the advantages of the visual feature extraction method based on deep learning are interpreted.	Eighteen typical ZSL models are listed, and it is found that the end-to-end networks have unique advantages.	Future work should concentrate on increasing the accuracy for a model by solving the different problems present in the ZSL
7	Yan, C., Chang, X., Li, Z., Guan, W., Ge, Z., Zhu, L., & Zheng, Q. (2021). Zero nas: Differentiable generative adversarial networks search for zero-shot learning. IEEE transactions on pattern analysis and machine intelligence.	2021	In this paper, They turn to the neural architecture search (NAS) and make the first attempt to bring NAS techniques into the ZSL realm.Zero NAS is capable of discovering desirable architectures that perform favorably against state-of-the-art ZSL and	The architectures used for transferred from the other three datasets decrease the performance of f-CLSWGAN ,it is one of the main disadvantage.	Experiments on four benchmark datasets demonstrate that the proposed ZeroNAS is capable of discovering state-of-the-art GAN architectures that enhance the performance of existing GAN frameworks for ZSL/GZSL.	Compared to f-CLSWGAN with han-crafted architecture, proposed method achieves up to 2.9%, 3.0%, 2.5%, and 3.5% improvements on CUB, FLO, SUN and AWA respectively.	Future work will concentrate on adapting the proposed architecture search method to other various kinds of architectures such as cyclic consistency or reconstruction regularizer based GAN models and fully MLP-based models.

			generalized zero-shot learning (GZSL) approaches.				
8	Wei, K., Deng, C., Yang, X., & Tao, D. (2021). Incremental zero-shot learning. IEEE Transactions on Cybernetics.	2021	This paper represented the first attempt to introduce and tackle IZSL, which better bridges the gap between real-world requirements and computer vision building blocks. A generative replay strategy was employed to accumulate historical knowledge of previously seen classes, which converts IZSL into traditional ZSL.	The results of the method had fluctuated, which do not conducive to a fair comparison. Especially, when the number becomes larger, the phenomenon of Catastrophic Forgetting will be more obvious and the performances will decrease.	Experiments showed that the method outperforms previous methods by a large margin on three benchmark datasets. An ablation study was also performed to verify that the proposed two strategies are both important to the achievement of good performance.	On CUB, the model achieves 59.36%, an improvement of 7.72% over the competitors. On FLO, the model achieves 79.49%, an improvement of 4.98% over the competitors. On SUN, the model achieves 32.25%, an improvement of 1.63% over the competitors.	An IZSL method with adaptive generative replay numbers should be proposed in future work to obtain better performance.
9	Tian, Y., Zhang, Y., Huang, Y., Xu, W., & Ding, Z. (2022). Differential Refinement Networkfor Zero-Shot Learning. IEEE Transactions on Neural Networks and Learning Systems.	2022	In this paper, they proposed a novel DRNet for the ZSL task. A two-branch network was proposed to explore effective semantic–visual relationships, which includes a basic network and a differential network.	The collected datasets do include noise samples and precise ones, which are to be treated differently during the training stage,So this takes more time.	The advantage is that, by comparing different prototypes, interactions between various categories will be characterized , benefiting the generation of authentic and discriminative visual center	DRNet yields 0.8% and 0.7% improvements over the current best results on AWA2 and FLO datasets, and showed competitive performance on AWA1 and CUB datasets.	In the future, one should continue thework in the from of two aspects. On the one hand, one should takedata uncertainty into consideration to further eliminate the domain shift problem.
10	Lv, W., Shi, H., Tan, S., Song,	2022	A novel DSKG is proposed in	The update strategy is	After DSKG is performed,	DSKG-ZSD outperforms	In future the model should

	B., & Tao, Y. (2022). A dynamic semantic knowledge graph for zero-shot object detection. The Visual Computer, 1-15.		this article, which can be flexibly embedded in any ZSD framework. The DSKG is employed to align between visual and semantic spaces, guaranteeing that unseen classes are inferred by semantic embedding.	detached from the constraints of visual features, resulting in class attributes that do not satisfy the visual space requirements.	the net work promotes semantic vectors to satisfy the diversity of visual pictures.	SB, DSES, CG-ZSD, RetinaNet, PL-ZSD, LSA-ZSD, and 0.73%, in terms of mAP performance for ZSD task.	be trained in a better way to give good accuracy and better features.
11	Rahman, S., Khan, S. H., & Porikli, F. (2020). Zero-shot object detection: Joint recognition and localization of novel concepts. International Journal of Computer Vision, 128(12), 2979-2999.	2020	They proposed a new experimental protocol for the ILSVRC-2017 dataset, specifying the seen–unseen, train-test split.They also developed an end-to-end trainable CNN model to solve the problem in object detection.	A ZSD method needs to correspond visual features with semantic word vectors which are generally noisy. This is used to degrades the overall confidence for ZSD.	The proposed approach employs a novel loss function to relate semantic and visual features of seen object classes with the unseen objects.	They performed experiments with VG dataset,Birds-200-2011 (CUB) dataset,on MS-COCO datasets based on average precision (AP) of individual unseen classes and mean average precision (mAP) and got 33.4%,76.8% .	The ZSD problem warrants further investigation . Instead of mapping image features to the semantic space, the reverse mapping should be done.
12	Yan, C., Chang, X., Luo, M., Liu, H., Zhang, X., & Zheng, Q. (2022).Semantics-guided contrastivenetwork for zero-shot object	2022	They developed a novel semantics-guided contrastive network for ZSD, underpinned by a new	The more challenging is zero-shot object detectionsetting,where the test samples may come from either seen or unseen	In this paper,The quantitative and qualitative experimental results confirm that the proposed framework	The proposed Contrast ZSD outperforms the second-best method BLC by a large margin, where the absolute HM performance	Future efforts should concentrate on the bias issue, which could improve accuracy.

	detection.IEEE Transactions on Pattern Analysis and Machine Intelligence.		mapping contrastive strategy superior to the conventional mapping-transfer strategy	classes, the bias problem degraded the performance significantly.	improves the performance of both the ZSD and GZSD task.	gain is 6.9% Recall@100 and 2.9% mAP for the 48/17 split and 6.8% Recall@100 and 4.2% mAP .	
13	Xie, Y., He, X., Zhang, J., & Luo, X. (2020). Zero-shot recognition with latent visual attributes learning. Multimedia Tools and Applications, 79(37), 27321-27335.	2020	They proposed a novel couple semantic dictionary learning framework to exploit the latent visual attributes for addressing the zero-shot recognition problem.	Here object categories is represented using the high-level semantic knowledge ,it is difficult to ensure the completeness and discriminative ness of human defined knowledge.	The experimental results on two benchmark datasets demonstrate that the proposed approach outperforms several state-of-the-art ZSL methods.	Zero-shot recognition results of different approaches on the AwA2 dataset. The best accuracy is shown in boldface is 62.8%	In the future one should conduct some further research lines that follow this framework. First,they should investigate the potential relationships between different visual attributes that can be used to improve the model.
14	Gao, R., Hou, X., Qin, J., Shen, Y., Long, Y., Liu, L., ... & Shao,L. (2022). Visual-Semantic Aligned Bidirectional Network for Zero-Shot Learning. IEEE Transactions on Multimedia.	2022	They proposed a visual-semantic aligned bidirectional network with cycle consistency to alleviate the gap between these two spaces, generating unseen features of high quality.	The experimental results showed that the framework does not gain the best accuracy on seen classes.	Extensive experiments demonstrate d that the framework achieved competitive performance with state-of-the-art methods in both conventional and generalized ZSL settings.	The proposed method outperforms cycle on AWA1 with 2.3% improvement in the harmonic mean. The method is competitive on SUN and aPY, and obtains significant improvement of 18.4% for harmonic mean .	Future work should concentrate on increasing the accuracy for a seen classes.

15	Du, P., Zhang, H., & Lu, J. (2020). Learning Discriminative Projection With Visual Semantic Alignment for Generalized Zero Shot Learning. IEEE Access, 8, 166273-166282.	2020	They proposed a novel method to solve the domain shift problem by learning discriminative projections with visual semantic alignment in latent space. A linear discriminative analysis strategy is employed to learn the projection from visual space to latent space.	The synthetic based methods achieved dull performance.	Using the prposed framework, they solved the domain shift problem by learning discriminative projections .	The method performed using mAP and gave results as 80% on SUN, 40% on CUB, 65% on AWA1, 57% on AWA2 and 52% on APY.	Future work should concentrate on increasing the accuracy for a synthetic methods.
16	Liu, Y., Dang, Y., Gao, X., Han, J., & Shao, L. (2022).Zero-Shot Learning With Attentive Region Embedding and Enhanced Semantics. IEEE Transactions on Neural Networks and Learning Systems.	2022	They proposed a variational autoencoder (VAE)-based framework, that is, joint Attentive Region Embedding with Enhanced Semantics, which is tailored9to advance the zero-shot recognition.	All results of methods are less than 10%, which indicates the difficulty of this problem where there is a large space for improvement of model.	The proposed model,AREES, carries out extensive evaluation and yields state-of the-art results for ZSL and GZSL tasks on six benchmarks	The proposed model significantly improves the state of the art for all splits, e.g., for 2-hops/M500/L5K splits, the accuracies increase by 657 4.00%/5.21%/4.82% compared to the strongest competitor CADA-VAE.	In the future, one should apply the semantic-guided visual features and enhanced semantics to other generative frame works for pursuing better classification accuracy under ZSL and GZSL tasks.
17	Li, Y., Liu, Z., Yao, L., & Chang, X. (2021). Attribute-	2021	In this paper, an attribute-modulated generative meta-model (AMAZ) to synthesize visual features	The synthetic data quality varies significantly across classes due to that The low-quality synthetic data	Extensive experiments on four widely-used benchmarks show that our model exceeds	The proposed attribute modulation network is effective on four benchmark datasets,	In the future, one should need to extend the model to address the insufficiency of labeled

	modulated generative meta learning for zero-shot learning. IEEE Transactions on Multimedia.		for unseen classes for ZSL. They generate features dynamically by proposing a novel Attribute-Modulated generative meta-model for Zero-shot learning (AMAZ).	largely misguided and impaired the training process of the final classifier.	state-of-the-art methods in both ZSL& GZSL settings. The qualitative and quantitative experiments in zeroshot image retrieval also show that AMAZ generates more discriminative features.	demonstrated by the improvements by up to 2.8%, 1.7%, 4.6%, and 3.8% on SUN, CUB, AWA1,and AWA2, respectively.	data for more complex retrieval tasks, such as fine-grained image retrieval and cross-media retrieval.
18	Xie, G. S., Zhang, X. Y., Yao, Y., Zhang, Z., Zhao, F., Shao, L. (2021). Vman: A virtual mainstay alignment network for transductive zero-shot learning. IEEE Transactions on Image Processing, 30, 4316-4329.	2021	They proposed a simple yet effective virtual mainstay (VM) sample generation approach for coping with the transductive ZSL (TZSL) problem. Throughout this paper, there is merely one parameter matrix to be learned, i.e., W. Supposethat W has been learned by the WED/VMAN algorithm.	The method don’t focus on the semantic space for the different attribute which reduces the performance.	The proposed virtual mainstay alignment network (VMAN) can seamlessly solve the TZSL problem. Extensive evaluations show that VMAN has achieved new state-of-the-arts on most of the utilized standard benchmarks.	TZSL embedding methods, by large margin,where VMAN achieves a new highest number of 89.3% on AWA; and For difficult SUN datasets, the performance of VMAN is up to 69.3%.	Further one should visualize the feature representations in the semantic/feature space for VMAN, where both seen/ unseen images are projected.
19	Zhao, P., Zhang, S., Liu, J., & Liu, H.(2021). Zero-shot Learning via the fusion of generation and	2021	In this paper, the authors presented a novel ZSL model based on class prototype and	If the dimension of the latent subspace is too small, the latent subspace cannot learn	The experimental results show that ZSL-CPLSR can effectively alleviate	Compared to ZSL-CPLSR-S, the classification accuracy of ZSL-CPLSR increases by	In future work, more semantic representations can be fused

	embedding for image recognition. Information Sciences, 578, 831-847.		dual latent subspace learning with reconstruction (ZSL-CPLSR), which integrates generation and embedding into a unified framework.	enough information. In the above case it degraded the recognition performance	the problems of domain shift and information loss and surpasses several state-of-the art ZSL methods.	21.9%, 13.2%, 27.0% and 2.7% on AwA, CUB, aP&Y and SUN.	in class prototype learning and latent subspace learning, such as word vectors extracted from text corpora and layered embeddings learned from WordNet
20	Xu, W., Xian, Y., Wang, J., Schiele, B., & Akata, Z. (2020). Attribute prototype network for zero-shot learning. Advances in Neural Information Processing Systems, 33, 21969-21980.	2020	In this work, they developed a zero-shot representation learning framework, attribute prototype network (APN), to jointly learn global and local features.	The model cannot give more accuracy compared to other models and cannot predict better features.	The model proved to the visual evidence of the attributes in an image, example for the CUB dataset, confirming the improved attribute localization ability of our image representation.	The method works equally well for both binary and continuous attributes in terms of attribute prediction accuracy of 86.4% with continuous attributes vs 86.8% with binary attributes onseen classes.	In future the model should be trained in a better way to give good accuracy and better features.

Fig 1.1 Literature survey table

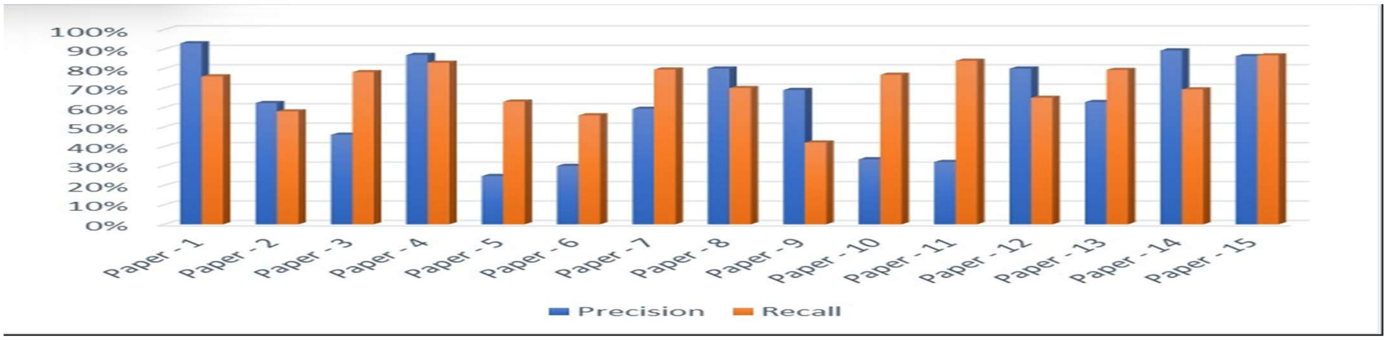


fig 2.1 **graphical representation for literature survey**

SYSTEM ANALYSIS AND DESIGN

We used zero shot object detection as part of our methodology for visual semantic alignment, and we proceed as follows:

1) Data Collection: As the initial phase in this project, we gathered and annotated the ImageNet dataset, a sizable collection of Images that contains a wide range of items. The object class labels and the bounding boxes for each item are both annotated in the dataset.
2) Pre-processing: After that, the data was pre-processed. The photographs are often resized, put into a uniform format, and have their pixel values normalised. This also includes pre-processing the textual descriptions, such as by transforming them into a numerical representation the model can use.
3) Model Implementation: Open AI built the huge language model CLIP (Contrastive Language-Image Pretraining), which was trained on a considerable amount of text and image data. After training CLIP on the dataset, the dataset is then fine-tuned for zero-shot item detection. An object detection method uses the Image features that are produced via CLIP as inputs. Then, using the image attributes produced by CLIP, we constructed an object detection algorithm using a Deep Neural Networks technique and occlusion algorithm. We used the training data to develop this object detection system, and the validation data to assess its performance.
4) Model Evaluation: Using the results produced for the provided image, we assessed the effectiveness of the object detection system. We compared the performance of a system that uses object annotations with the performance of a zero-shot object detection system.

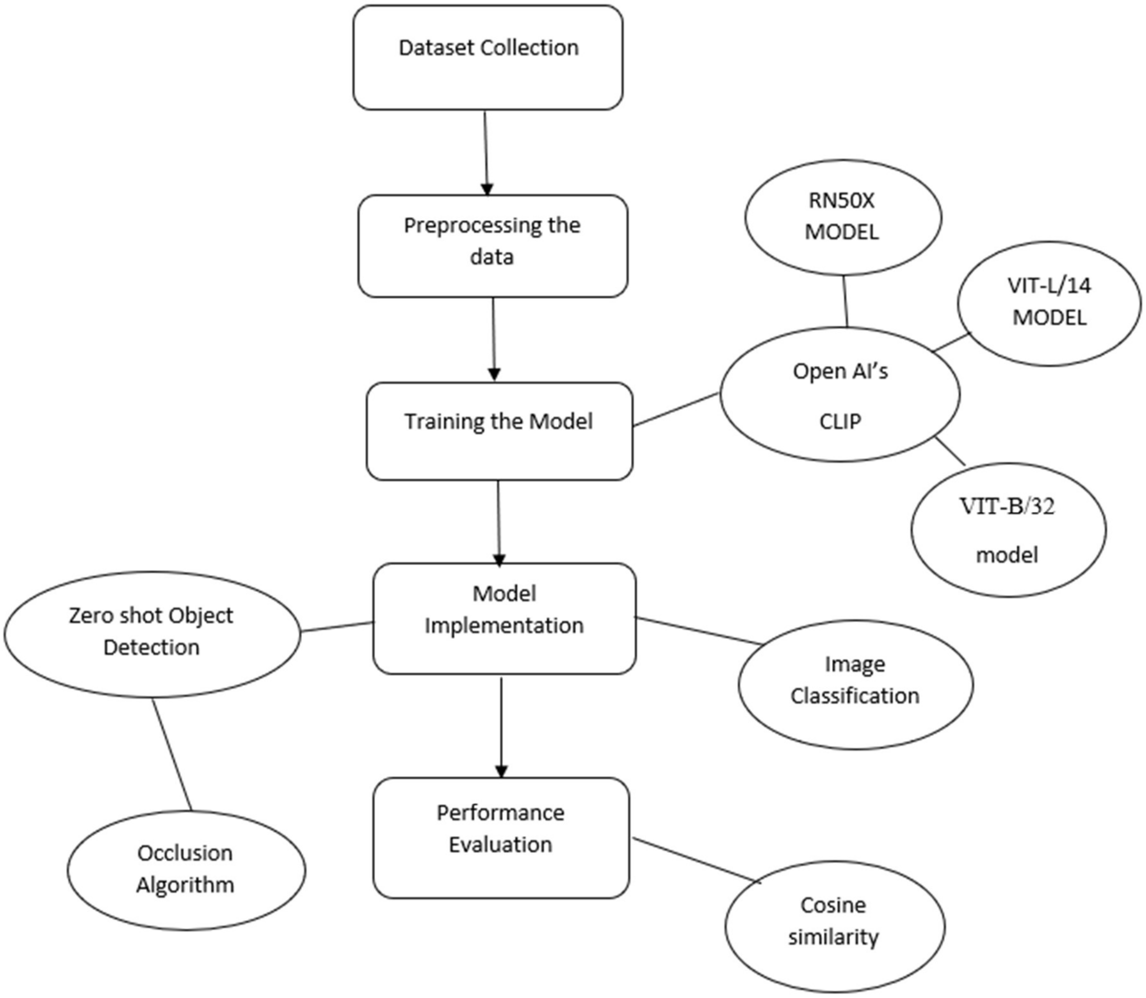


Fig 3.1 Flow chat

Algorithm:

Occlusion Algorithm: Zero-shot object detection is a type of object detection algorithm that can detect and classify objects that it has not seen during the training phase. An occlusion algorithm is used to accomplish this.

In a zero-shot object detection system, the occlusion algorithm uses the remaining visible information to determine the item's class after partially hiding a piece of the image. The steps are repeated for other areas of the image, and the aggregated results are used to create the final prediction.

According to this method, even if an item is not included in the training set, its occluded regions still contain details about its shape, texture, and other traits that can be used to identify it.

The algorithm can get a more thorough grasp of the object's attributes and create a more accurate forecast by excluding various portions of the image.

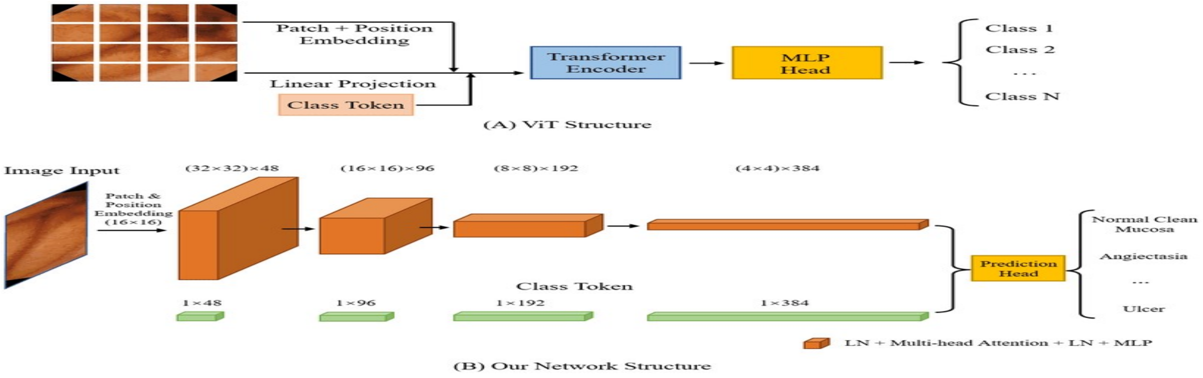
In general, the application of an occlusion technique in zero-shot object detection is a promising strategy that can enhance the precision and robustness of object detection systems, particularly when dealing with novel or unseen items.

Libraries & Models:

VIT-B/32 model:

Using the Transformer architecture initially created for natural language processing, ViT is a well-known computer vision model. The ViT-B/32 model's "B" denotes the quantity of Transformer blocks, and "32" denotes the patch's 32x32 pixel resolution. A series of patches taken from the image and linearly projected onto a high-dimensional embedding space serve as the input to the ViT model. The Transformer blocks then process the embedding sequence in order to learn the representation of the image. Twelve Transformer blocks make up the ViT-B/32 model, and each one has feedforward and multi-head self-attention layers. While the feedforward layers add nonlinearity to the model, the multi-head self-attention layers enable the model to capture long-range dependencies inside the series of embeddings.

The final embedding of the sequence is taken by a classification head at the end of the ViT-B/32 model, which translates it to the output classes. The model is trained using a cross-entropy loss function in a supervised fashion.



(A) ViT Structure

(B) Our Network Structure

Fig 3.2 VIT-B/32 model architecture

RN50 Model:

Microsoft Research Asia created the convolutional neural network architecture RN50 (ResNet-50) for image classification. It is a variation of the ResNet architecture, which was unveiled in 2015 and took first place in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in both those years.

Using 50 layers and skip connections, the RN50 architecture enables the model to learn residual functions with regard to the layer inputs. These skip connections aid in reducing the issue of disappearing gradients, which can happen when deep neural networks are being trained.

The architecture is built on the idea of segmenting the network into "residual blocks"—modules made up of two or three convolutional layers followed by a shortcut connection. The shortcut connection enables data to be sent straight from one block to the next, reducing the number of parameters and enhancing the performance of the model.

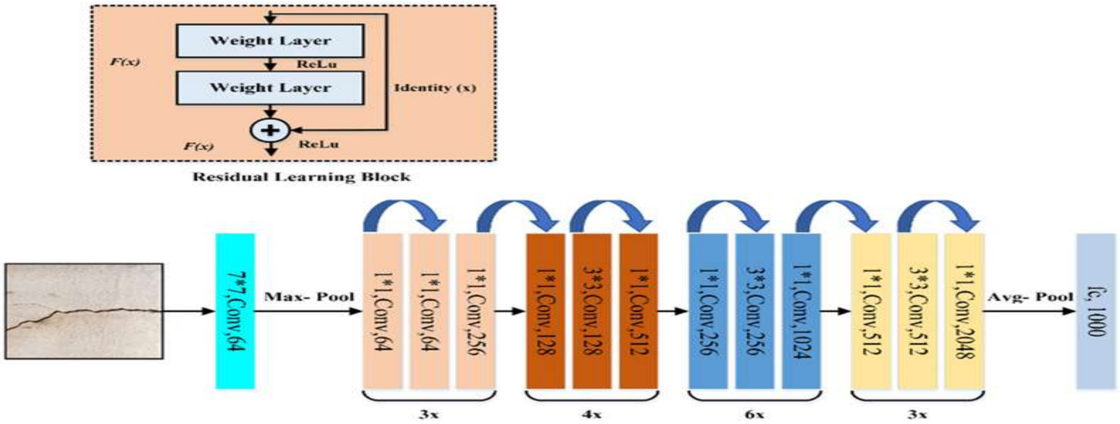


Fig 3.3 RN50 Model architecture

RN101MODEL:

A convolutional neural network architecture called RN101 (ResNet-101) is a variation of the ResNet (Residual Network) architecture. It was created for picture classification by Microsoft Research Asia and released in 2015. It triumphed in the 2015 and 2016 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), just like RN50.

In many aspects, RN101 and RN50 are comparable, but RN101 is deeper since it contains more levels. It has 101 layers and makes use of skip connections, which let the model pick up residual functions based on the inputs of the layers. These skip connections aid in reducing the issue of disappearing gradients, which can happen when deep neural networks are being trained.

The architecture is built on the idea of segmenting the network into "residual blocks"—modules made up of two or three convolutional layers followed by a shortcut connection. The shortcut connection enables data to be sent straight from one block to the next, reducing the number of parameters and enhancing the performance of the model.

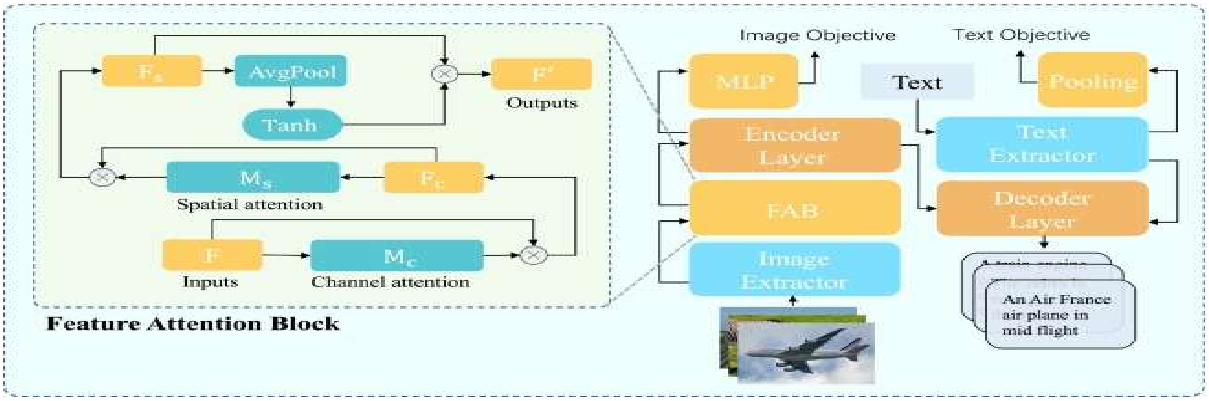


Fig 3.4 RN101 Model architecture

RN50X4 MODEL:

ResNet (Residual Network) is an architecture, and RN50x4 is a variation of it that was released by Facebook AI Research in 2020. It is a modified version of the original RN50 (ResNet-50) architecture with the goal of expanding the model's depth and width in order to enhance performance.

The original RN50 architecture is four times narrower and four times deeper than RN50x4. The model may learn residual functions with regard to the layer inputs thanks to the skip connections used in its 200 layers. These skip connections aid in reducing the issue of disappearing gradients, which can happen when deep neural networks are being trained.

The architecture is built on the idea of segmenting the network into "residual blocks"—modules made up of two or three convolutional layers followed by a shortcut connection. The shortcut connection enables data to be sent straight from one block to the next, reducing the number of parameters and enhancing the performance of the model.

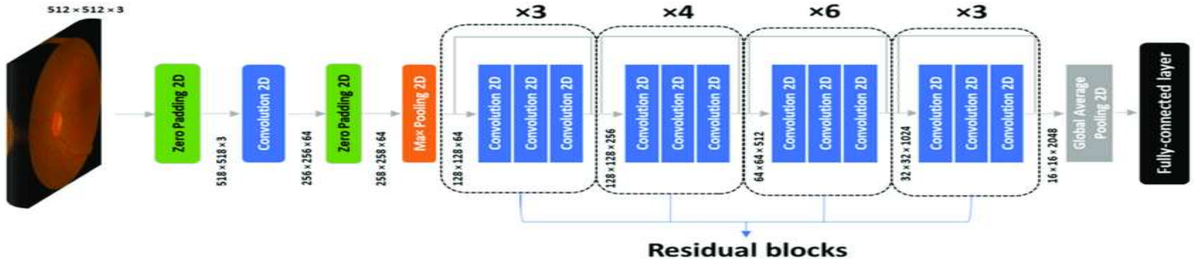


Fig 3.5 RN50X4 Model architecture

VIT-B/16 model:

The Vision Transformer is another name for the ViT-B/16 model, a kind of neural network architecture that Google researchers first described in a study in 2020. The transformer architecture, initially created for natural language processing, is one approach that the transformer architecture, one of numerous iterations of the ViT architecture, tries to adapt to image recognition challenges.

The transformer blocks that process image patches make up the ViT-B/16 model. It begins by dividing the input image into a grid of distinct, non-overlapping patches. After that, each patch is linearly embedded and sent through a number of transformer blocks, which enable the model to recognise inter-patch dependencies on a global scale. To create the final transformer output, the output of the last transformer block is sent via a classification head to produce final prediction. The ViT-B/16 model, which was pre-trained on massive datasets like ImageNet and JFT-300M, has produced state-of-the-art results on a variety of image classification benchmarks. It has also been applied in transfer learning contexts, where it is honed for particular image recognition tasks on smaller datasets.

The model's designation "ViT-B/16" refers to its particular configuration, where "B" denotes the quantity of transformer blocks and "/16" denotes the input patch size. Other configurations of transformer blocks, patch sizes, and other hyperparameters may be used in other iterations of the ViT architecture.

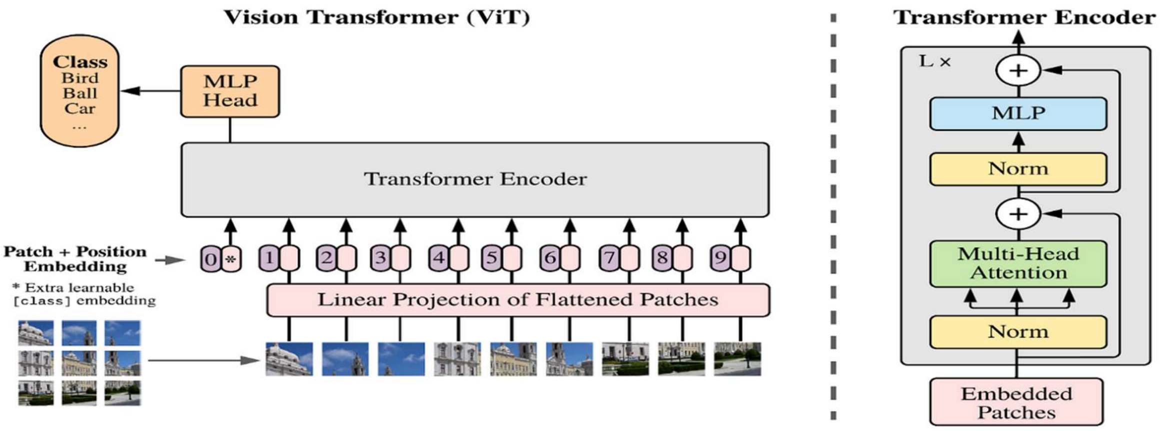


Fig 3.6 VIT-B/16 Model architecture

VIT-L/14 model:

The Vision Transformer is another name for the ViT-L/14 model, a category of neural network architecture that Google researchers first described in a study in 2020. The transformer architecture, originally created for natural language processing, can be applied to image identification applications in a number of ways, one of which being the ViT-L/14 model. Compared to the ViT-B/16 model, the ViT-L/14 model is deeper and wider, making it more powerful but also requiring more processing power to train. It is made up of several transformer blocks that handle picture patches, similar to the ViT-B/16 paradigm. It begins by dividing the input image into a grid of distinct, non-overlapping patches.

After then, each patch is linearly embedded and sent through a number of transformer blocks, which enable the model to recognize inter-patch dependencies on a global scale. The final prediction is generated by running the output of the final transformer block through a classification head. The ViT-L/14 model, which was pre-trained on massive datasets like ImageNet and JFT-300M, has also produced state-of-the-art results on a variety of image classification benchmarks. It has also been applied in transfer learning contexts, where it is honed for particular image recognition tasks on smaller datasets. The model's designation "ViT-L/14" alludes to its particular configuration, with "L" designating the quantity of transformer blocks and "/14" designating the input patch size.

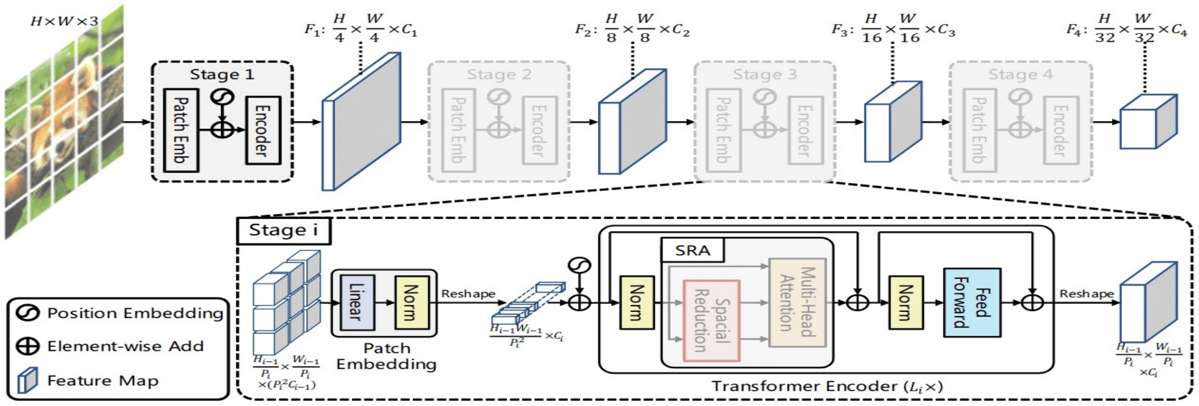

$H \times W \times 3$
Stage 1
Patch Emb
Encoder
$F_1: \frac{H}{4} \times \frac{W}{4} \times C_1$
Stage 2
Patch Emb
Encoder
$F_2: \frac{H}{8} \times \frac{W}{8} \times C_2$
Stage 3
Patch Emb
Encoder
$F_3: \frac{H}{16} \times \frac{W}{16} \times C_3$
Stage 4
Patch Emb
Encoder
$F_4: \frac{H}{32} \times \frac{W}{32} \times C_4$
Stage i
$\frac{H_{i-1}}{P_i} \times \frac{W_{i-1}}{P_i} \times (P_i^2 C_{i-1})$
Linear
Norm
Patch Embedding
Reshape
$\frac{H_{i-1}W_{i-1}}{P_i^2} \times C_i$
SRA
Norm
Spacial Reduction
Multi-Head Attention
Norm
Feed Forward
Transformer Encoder ($L_i \times$)
Reshape
$\frac{H_{i-1}}{P_i} \times \frac{W_{i-1}}{P_i} \times C_i$
Position Embedding
Element-wise Add
Feature Map

Fig 3.7 ViT-L/14 model architecture

VIT-L/14@336PX MODEL:

The ViT-L/14 model is a sort of neural network architecture that employs the transformer architecture for image identification tasks. The ViT-L/14@336px model is a version of the ViT-L/14 model.

Similar in architecture to the ViT-L/14 model, the ViT-L/14@336px model was developed using bigger input image sizes. In particular, the input images are scaled down from the 224x224 pixels of the original ViT-L/14 model to 336x336 pixels. This enables the model to collect finer features in the photos, perhaps enhancing its performance in some tasks. The ViT-L/14@336px model is made up of a number of transformer blocks that handle picture patches, just like the ViT-L/14 model. It begins by dividing the input image into a grid of distinct, non-overlapping patches.

After then, each patch is linearly embedded and sent through a number of transformer blocks, which enable the model to recognise inter-patch dependencies on a global scale. The final prediction is generated by running the output of the final transformer block through a classification head.

The ViT-L/14@336px model, which was pre-trained on massive datasets like ImageNet and JFT-300M, has produced state-of-the-art results on a number of image classification benchmarks. It has also been applied in transfer learning contexts, where it is honed for particular image recognition tasks on smaller datasets.

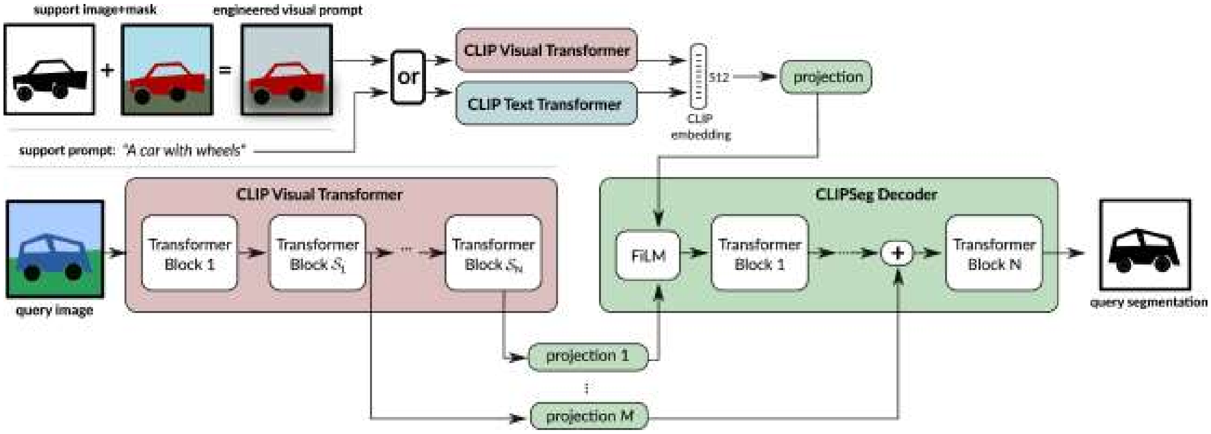


Fig 3.8 VIT-L/14@336P model architecture

EXISTING SYSTEM:

The paper "Polarity loss improving visual semantic alignment for zero shot detection" describes a system for zero-shot object detection that improves upon existing systems by incorporating a novel loss function called the polarity loss.

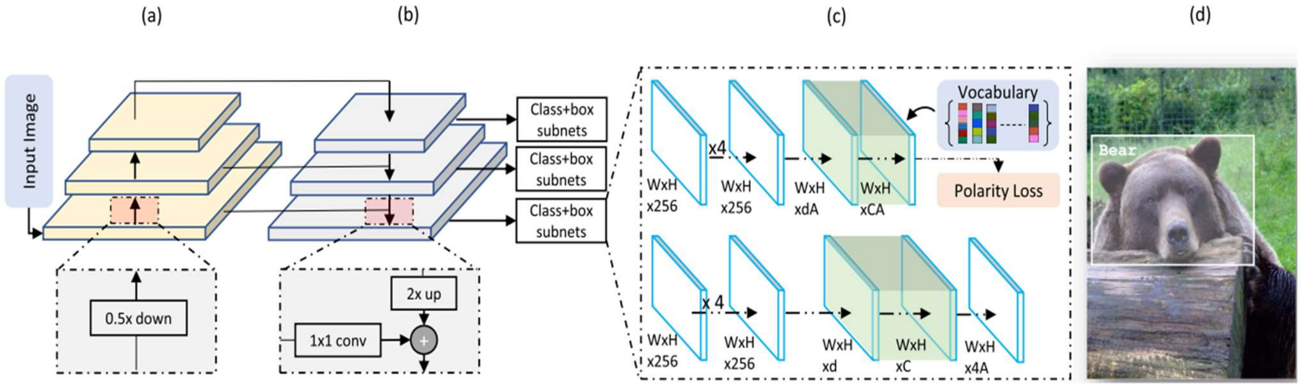


Fig. 7. Network architecture for ZSD. The green colored layer implements (9) (Our-PL-word) or (10) (Our-PL-vocab).(a) Bottom-up processing (ResNet). (b) Top-down processing (FPN). (c) Semantic alignment in classification (top) and box regression (bottom) subnets. (d) Output detections for unseen objects.

Fig 3.9 ZSD network architecture

In this, they proposed the ZSD framework, which was created specifically to operate with single-stage detectors. The main driving force is the direct relationship between localization and anchor categorization, which guarantees robust feedback for both tasks.

They decided to use Retina Net, a current unified single architecture, to put the suggested approach in this study into practise. Retina Net, which excels in both speed and accuracy, is the best detector. Two task-specific subnetworks for classification and box regression are branches of Retina Net's main backbone network, called FPN. In order to identify objects at various scales, FPN extracts rich, multiscale features for various anchor boxes from an image.

The feature extractor and the classifier are the two main parts of the system. The feature extractor is responsible for extracting features from input images, and the classifier is responsible for mapping these features to object classes based on their semantic representations.

Auxiliary data sources like text or image annotations are used to teach class attributes, which serve as the semantic representations of object classes. A visual-semantic alignment module, which is trained to reduce the polarity loss, is used by the system to align the visual characteristics of objects with their semantic representations.

By encouraging a more consistent and discriminative mapping, the polarity loss is an unique loss function that aids in better alignment between visual features and semantic representations.

PROPOSED SYSTEM:

Zero Shot Object Detection with Open AI's CLIP:

The Contrastive Language-Image Pretraining (CLIP) deep neural network architecture from OpenAI is a cutting-edge multimodal learning architecture. It was presented by OpenAI in 2020 and has since been applied to a variety of applications involving language and images.

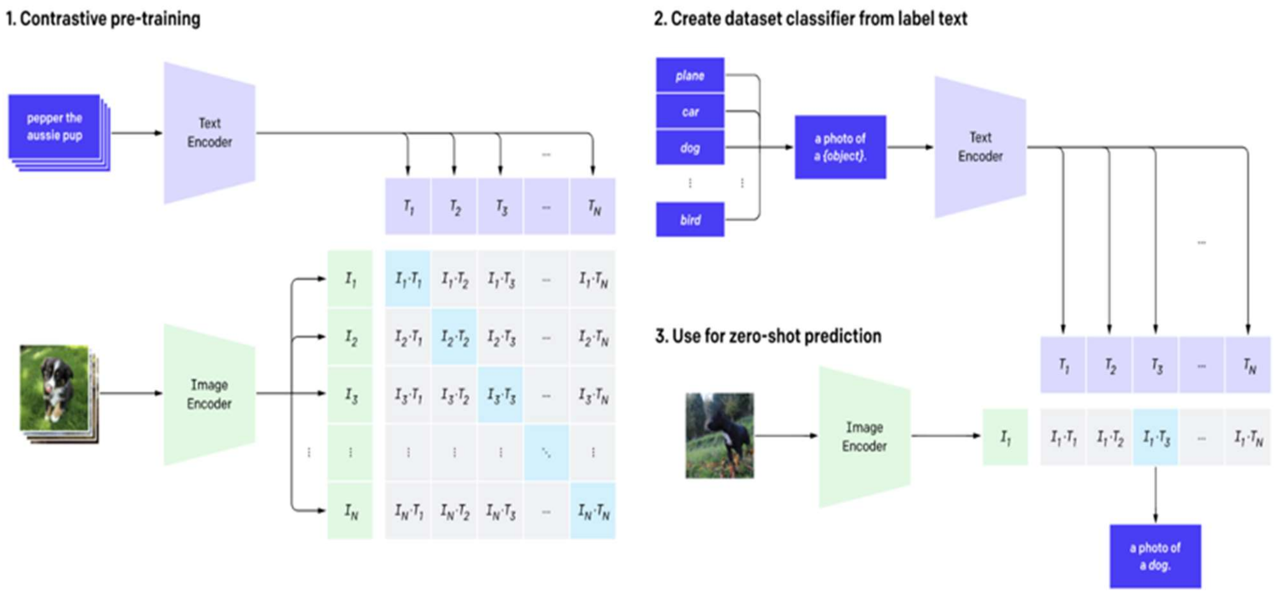

1. Contrastive pre-training
pepper the aussie pup
Text Encoder
T_1 T_2 T_3 ... T_N
Image Encoder
I_1 I_2 I_3 ... I_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
$I_2 \cdot T_1$ $I_2 \cdot T_2$ $I_2 \cdot T_3$ ... $I_2 \cdot T_N$
$I_3 \cdot T_1$ $I_3 \cdot T_2$ $I_3 \cdot T_3$ ... $I_3 \cdot T_N$
$I_N \cdot T_1$ $I_N \cdot T_2$ $I_N \cdot T_3$ ... $I_N \cdot T_N$
2. Create dataset classifier from label text
plane
car
dog
bird
a photo of a {object}.
Text Encoder
T_1 T_2 T_3 ... T_N
3. Use for zero-shot prediction
Image Encoder
I_1
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
a photo of a dog.

Fig 3.10 CLIP model architecture

A contrastive learning objective is used to pre-train a transformer-based architecture called CLIP on a substantial amount of text and image data. The contrastive learning objective encourages CLIP to generate high-level representations of images that are semantically and visually similar to the text representations. The pre-training process of CLIP involves training the model on pairs of text and image data, where the text can be a sentence, a caption, or a question, and the image can be a still image or a video frame. CLIP gains the ability to create high-level representations of images that are both aesthetically and semantically in line with the text representations during pre-training. After pre-training, CLIP can be fine-tuned for a particular job, such as object identification in zero-shots, image categorization, image captioning, or answering visual questions. The model's parameters are tuned to the task-specific data during fine-tuning, enabling the model to provide task-specific representations that are best suited for the given job.

In the case of CLIP, the model makes predictions using multimodal data that includes both an image and its accompanying textual description as input. Separate network branches process the text and image separately, and the results from these branches are then combined to produce a final prediction. The network's image branch, at its most basic level, is made up of a number of convolutional and pooling layers that take features from the image. The network's text branch handles the textual description using a transformer architecture. To create the final forecast, the outputs from these two branches are pooled and passed through a number of fully linked layers. By using the textual descriptions to infer details about the objects, the model is able to forecast new classes of items that it has not encountered during its training phase in the case of zero-shot object recognition.

IMPLEMENTATION

Implementation –1:

Open AI’s Clip model with Occulsion algorithm.

```
pip install datasets
# import dataset
from datasets import load_dataset
data = load_dataset(
    "jamescalam/image-text-demo",
    split="train",
    revision="180fdae",
)
data
type(data[2]['image'])
from PIL import Image
import requests
from io import BytesIO
import matplotlib.pyplot as plt
# Load the image from a URL or file path
img = data[2]['image']
# Plot the image using Matplotlib
```

```
plt.imshow(img)
plt.axis('off')
plt.show()
# Download the image
img.save('image.jpg')
from google.colab import files
files.download('image.jpg')
from torchvision import transforms
# transform the image into tensor
transt = transforms.ToTensor()
data = Image.open("image.jpg")

img = transt(data)
img.data.shape
# add batch dimension and shift color channels
patches = img.data.unfold(0,3,3)
patches.shape
# break the image into patches (in height dimension)
patch = 256
new_patches = patches.unfold(1, patch, patch)
new_patches.shape
# break the image into patches (in width dimension)
new_patches_1 = new_patches.unfold(2, patch, patch)
new_patches_1.shape
import torch
import matplotlib.pyplot as plt
import matplotlib.patches as patches
import os
os.environ["KMP_DUPLICATE_LIB_OK"]="TRUE"
window = 6
stride = 1
 print(0, new_patches_1.shape[1]-window+1, stride)
# window slides from top to bottom
for Y in range(0, new_patches_1.shape[1]-window+1, stride):
    # window slides from left to right
    for X in range(0, new_patches_1.shape[2]-window+1, stride):
        # initialize an empty big_patch array
        big_patch = torch.zeros(patch*window, patch*window, 3)
        # this gets the current batch of patches that will make big_batch
        patch_batch = new_patches_1[0, Y:Y+window, X:X+window]
        # loop through each patch in current batch
        for y in range(patch_batch.shape[1]):
            for x in range(patch_batch.shape[0]):
                # add patch to big_patch
                big_patch[
                    y*patch:(y+1)*patch, x*patch:(x+1)*patch, :
                ] = patch_batch[y, x].permute(1, 2, 0)
        # display current big_patch
        # plt.imshow(big_patch)
        # plt.show()
pip install transformers
```

```
from transformers import CLIPProcessor, CLIPModel
import torch
# define processor and model
model_id = "openai/clip-vit-base-patch32"
processor = CLIPProcessor.from_pretrained(model_id)
model = CLIPModel.from_pretrained(model_id)
# move model to device if possible
device = 'cuda' if torch.cuda.is_available() else 'cpu'
model.to(device)
window = 6
stride = 1
scores = torch.zeros(new_patches_1.shape[1], new_patches_1.shape[2])
runs = torch.ones(new_patches_1.shape[1], new_patches_1.shape[2])
for Y in range(0, new_patches_1.shape[1]-window+1, stride):
    for X in range(0, new_patches_1.shape[2]-window+1, stride):
        big_patch = torch.zeros(patch*window, patch*window, 3)
        patch_batch = new_patches_1[0, Y:Y+window, X:X+window]
        for y in range(window)
   for x in range(window):
                big_patch[
                    y*patch:(y+1)*patch, x*patch:(x+1)*patch, :
                ] = patch_batch[y, x].permute(1, 2, 0)
        # we preprocess the image and class label with the CLIP processor
        inputs = processor(
            images=big_patch,  # big patch image sent to CLIP
            return_tensors="pt",  # tell CLIP to return pytorch tensor
            text="an animal",  # class label sent to CLIP
                padding=True
        ).to(device) # move to device if possible
        # calculate and retrieve similarity score
        score = model(**inputs).logits_per_image.item()
        # sum up similarity scores from current and previous big patches
        # that were calculated for patches within the current window
        scores[Y:Y+window, X:X+window] += score
        # calculate the number of runs on each patch within the current window
        runs[Y:Y+window, X:X+window] += 1
scores /= runs
import numpy as np
# clip the scores
scores = np.clip(scores-scores.mean(), 0, np.inf)
# normalize scores
scores = (
    scores - scores.min()) / (scores.max() - scores.min()
)
scores.shape, new_patches_1.shape
# transform the patches tensor
adj_patches = new_patches_1.squeeze(0).permute(3, 4, 2, 0, 1)
adj_patches.shape
# multiply patches by scores
adj_patches = adj_patches * scores
# rotate patches to visualize
adj_patches = adj_patches.permute(3, 4, 2, 0, 1)
```

```
Y = adj_patches.shape[0]
X = adj_patches.shape[1]
fig, ax = plt.subplots(Y, X, figsize=(X*.5, Y*.5))
for y in range(Y):
    for x in range(X):
        ax[y, x].imshow(adj_patches[y, x].permute(1, 2, 0))
        ax[y, x].axis("off")
        ax[y, x].set_aspect('equal')
plt.subplots_adjust(wspace=0, hspace=0)
plt.show()
# scores higher than 0.5 are positive
detection = scores > 0.5
# non-zero positions
np.nonzero(detection)
y_min, y_max = (
    np.nonzero(detection)[:,0].min().item(),
    np.nonzero(detection)[:,0].max().item()+1
)
y_min, y_max

x_min, x_max = (
    np.nonzero(detection)[:,1].min().item(),
    np.nonzero(detection)[:,1].max().item()+1
)
x_min, x_max
y_min *= patch
```

```
y_max *= patch
x_min *= patch
x_max *= patch
x_min, y_min
height = y_max - y_min
width = x_max - x_min
height, width
# image shape
img.data.numpy().shape
# move color channel to final dim
image = np.moveaxis(img.data.numpy(), 0, -1)
image.shape
import matplotlib.patches as patches
fig, ax = plt.subplots(figsize=(Y*0.5, X*0.5))
ax.imshow(image)
rect = patches.Rectangle(
   (x_min, y_min), width, height,
   linewidth=3, edgecolor='#FAFF00', facecolor='none')
ax.add_patch(rect)
plt.show()
```

Implementation-2:

Open AI's clip integrated with 6 different models (RN50, RN101, RN50x4, RN50x16, ViT-B/32, ViT-B/16)

Importing Packages:

```
import numpy as np
import torch
```

```
from pkg_resources import packaging
print("Torch version:", torch.__version__)
Op: Torch version: 1.9.0+cu102
```

Loading the model:

```
import clip
clip.available_models()
op:['RN50', 'RN101', 'RN50x4', 'RN50x16', 'ViT-B/32', 'ViT-B/16']
model, preprocess = clip.load("ViT-B/32")
model’s().eval()
input_resolution = model.visual.input_resolution
context_length = model.context_length
vocab_size = model.vocab_size
print("Model parameters:", f"{np.sum([int(np.prod(p.shape)) for p in model.parameters()]):,}")
print("Input resolution:", input_resolution)
print("Context length:", context_length)
print("Vocab size:", vocab_size)

100%|████████████████████████████████████████| 338M/338M
[00:05<00:00, 63.0MiB/s]
Model parameters: 151,277,313
Input resolution: 224
Context length: 77
Vocab size: 49408
```

Image Preprocessing:

```
preprocess
compose( Resize(size=224, interpolation=bicubic, max_size=None, antialias=None)
CenterCrop(size=(224, 224)) <function _transform.<locals>.<lambda> at 0x7f3a24ffb440>
ToTensor() Normalize(mean=(0.48145466, 0.4578275, 0.40821073), std=(0.26862954,
0.26130258, 0.27577711)) )
```

Text Preprocessing:

```
clip.tokenize("Hello World!")
```

```
tensor([[49406, 3306, 1002, 256, 49407, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0]])
```

Setting up input images and texts:

```
import os
import skimage
import IPython.display
import matplotlib.pyplot as plt
from PIL import Image
import numpy as np

from collections import OrderedDict
import torch
%matplotlib inline
%config InlineBackend.figure_format = 'retina'

# images in skimage to use and their textual descriptions
descriptions = {
    "page": "a page of text about segmentation",
    "chelsea": "a facial photo of a tabby cat",
    "astronaut": "a portrait of an astronaut with the American flag",
    "rocket": "a rocket standing on a launchpad",
    "motorcycle right": "a red motorcycle standing in a garage",
    "camera": "a person looking at a camera on a tripod",
    "horse": "a black-and-white silhouette of a horse",
    "coffee": "a cup of coffee on a saucer"
}
original_images = []
images = []
texts = []
plt.figure(fig size=(16, 5))

for filename in [filename for filename in os.listdir(skimage.data_dir) if filename.endswith(".png"
) or filename.endswith(".jpg")]:
    name = os.path.splitext(filename)[0]
    if name not in descriptions:
        continue

    image = Image.open(os.path.join(skimage.data_dir, filename)).convert("RGB")

    plt.subplot(2, 4, len(images) + 1)
    plt.imshow(image)
    plt.title(f"{filename}\n{descriptions[name]}")
    plt.xticks([])
    plt.yticks([])

    original_images.append(image)
    images.append(preprocess(image))
    texts.append(descriptions[name])

plt.tight_layout()
```

Building features:

```
image_input = torch.tensor(np.stack(images)).cuda()
text_tokens = clip.tokenize(["This is " + desc for desc in texts]).cuda()
```

```
with torch.no_grad():
    image_features = model.encode_image(image_input).float()
    text_features = model.encode_text(text_tokens).float()
```

Calculating cosine similarity:

```
image_features /= image_features.norm(dim=-1, keepdim=True)
text_features /= text_features.norm(dim=-1, keepdim=True)
similarity = text_features.cpu().numpy() @ image_features.cpu().numpy().T
count = len(descriptions)

plt.figure(figsize=(20, 14))
plt.imshow(similarity, vmin=0.1, vmax=0.3)
# plt.colorbar()
plt.yticks(range(count), texts, fontsize=18)
plt.xticks([])
for i, image in enumerate(original_images):
    plt.imshow(image, extent=(i - 0.5, i + 0.5, -1.6, -0.6), origin="lower")
for x in range(similarity.shape[1]):
    for y in range(similarity.shape[0]):
        plt.text(x, y, f"{similarity[y, x]:.2f}", ha="center", va="center", size=12)

for side in ["left", "top", "right", "bottom"]:
  plt.gca().spines[side].set_visible(False)

plt.xlim([-0.5, count - 0.5])
plt.ylim([count + 0.5, -2])

plt.title("Cosine similarity between text and image features", size=20)
Text(0.5, 1.0, 'Cosine similarity between text and image features')
```

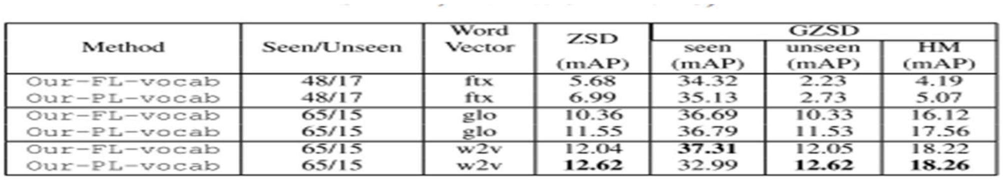
Zero-Shot Object detection:

```
from torchvision.datasets import CIFAR100
cifar100 = CIFAR100(os.path.expanduser("~/.cache"), transform=preprocess, download=True)
Downloading https://www.cs.toronto.edu/~kriz/cifar-100-python.tar.gz to /root/.cache/cifar-100-python.tar.gz
Extracting /root/.cache/cifar-100-python.tar.gz to /root/.cache
text_descriptions = [f"This is a photo of a {label}" for label in cifar100.classes]
text_tokens = clip.tokenize(text_descriptions).cuda()
with torch.no_grad():
    text_features = model.encode_text(text_tokens).float()
    text_features /= text_features.norm(dim=-1, keepdim=True)
text_probs = (100.0 * image_features @ text_features.T).softmax(dim=-1)
top_probs, top_labels = text_probs.cpu().topk(5, dim=-1)
plt.figure(figsize=(16, 16))
for i, image in enumerate(original_images):
    plt.subplot(4, 4, 2 * i + 1)
    plt.imshow(image)
    plt.axis("off")
    plt.subplot(4, 4, 2 * i + 2)
    y = np.arange(top_probs.shape[-1])
    plt.grid()
    plt.barh(y, top_probs[i])
    plt.gca().invert_yaxis()
    plt.gca().set_axisbelow(True)
    plt.yticks(y, [cifar100.classes[index] for index in top_labels[i].numpy()])
    plt.xlabel("probability")
plt.subplots_adjust(wspace=0.5)
plt.show()
```

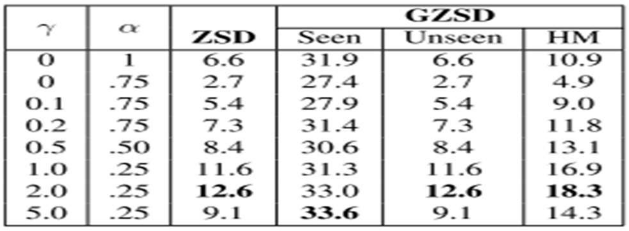
RESULTS AND DISCUSSIONS

Existing Methods:

The results using the zeroshot detection on polarity loss-word and polarity loss of vocabulary are as follows:



Method	Seen/Unseen	Word Vector	ZSD	GZSD		
			(mAP)	seen (mAP)	unseen (mAP)	HM (mAP)
Our-FL-vocab	48/17	ftx	5.68	34.32	2.23	4.19
Our-PL-vocab	48/17	ftx	6.99	35.13	2.73	5.07
Our-FL-vocab	65/15	glo	10.36	36.69	10.33	16.12
Our-PL-vocab	65/15	glo	11.55	36.79	11.53	17.56
Our-FL-vocab	65/15	w2v	12.04	**37.31**	12.05	18.22
Our-PL-vocab	65/15	w2v	**12.62**	32.99	**12.62**	**18.26**



γ	α	ZSD	GZSD		
			Seen	Unseen	HM
0	1	6.6	31.9	6.6	10.9
0	.75	2.7	27.4	2.7	4.9
0.1	.75	5.4	27.9	5.4	9.0
0.2	.75	7.3	31.4	7.3	11.8
0.5	.50	8.4	30.6	8.4	13.1
1.0	.25	11.6	31.3	11.6	16.9
2.0	.25	**12.6**	33.0	**12.6**	**18.3**
5.0	.25	9.1	**33.6**	9.1	14.3

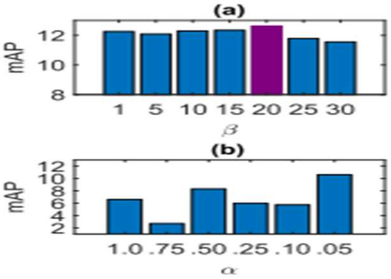
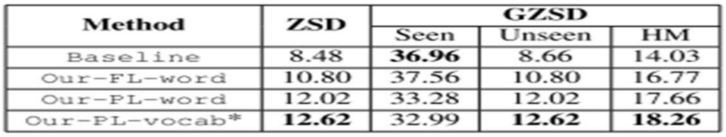


Fig. 10. Parameter sensitivity analysis on our proposed (65/15) MS-COCO split: *(Left)* Varying α and γ with a fixed $\beta = 20$. *(Right-a)* Impact of varying β, *(Right-a)* varying α with $\gamma = 0$ to see the behavior of our loss with only balanced CE. Note that actual hyperparameters choice is made on a validation set.



Method	ZSD	GZSD		
		Seen	Unseen	HM
Baseline	8.48	**36.96**	8.66	14.03
Our-FL-word	10.80	37.56	10.80	16.77
Our-PL-word	12.02	33.28	12.02	17.66
Our-PL-vocab*	**12.62**	32.99	**12.62**	**18.26**

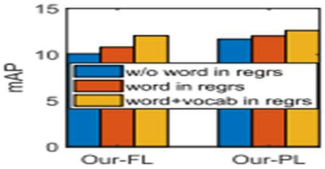


Fig 5.1 parameters analysis of existing method

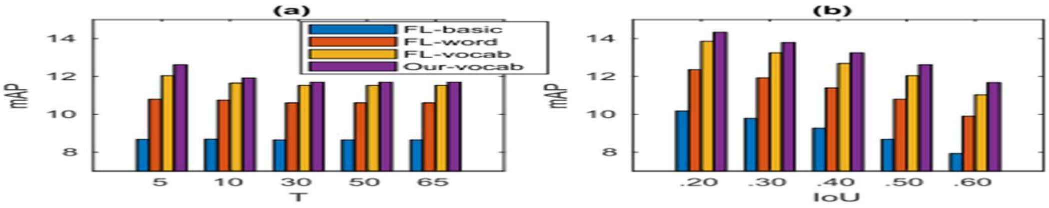


Fig. 12. (a) Impact of selecting close seen. (b) Impact of IoU.

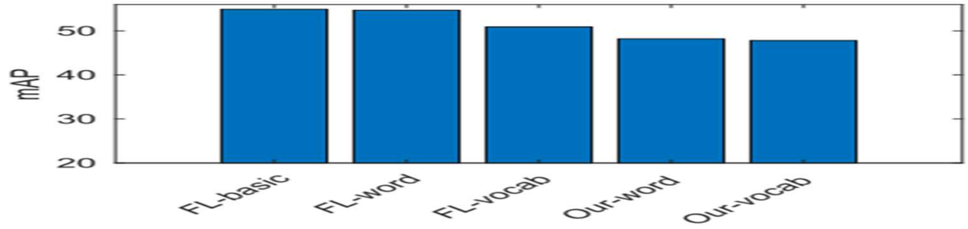


Fig 5.2 graphical representation

Proposed Methods:

cosine similarity: In the CLIP (Contrastive Language-Image Pre-Training) model, cosine similarity is used as a measure of similarity between two vectors, such as the vectors representing an image and its corresponding text description. Cosine similarity is calculated as follows:

cosine_similarity (u, v) = (u dot v) / (||u|| * ||v||)

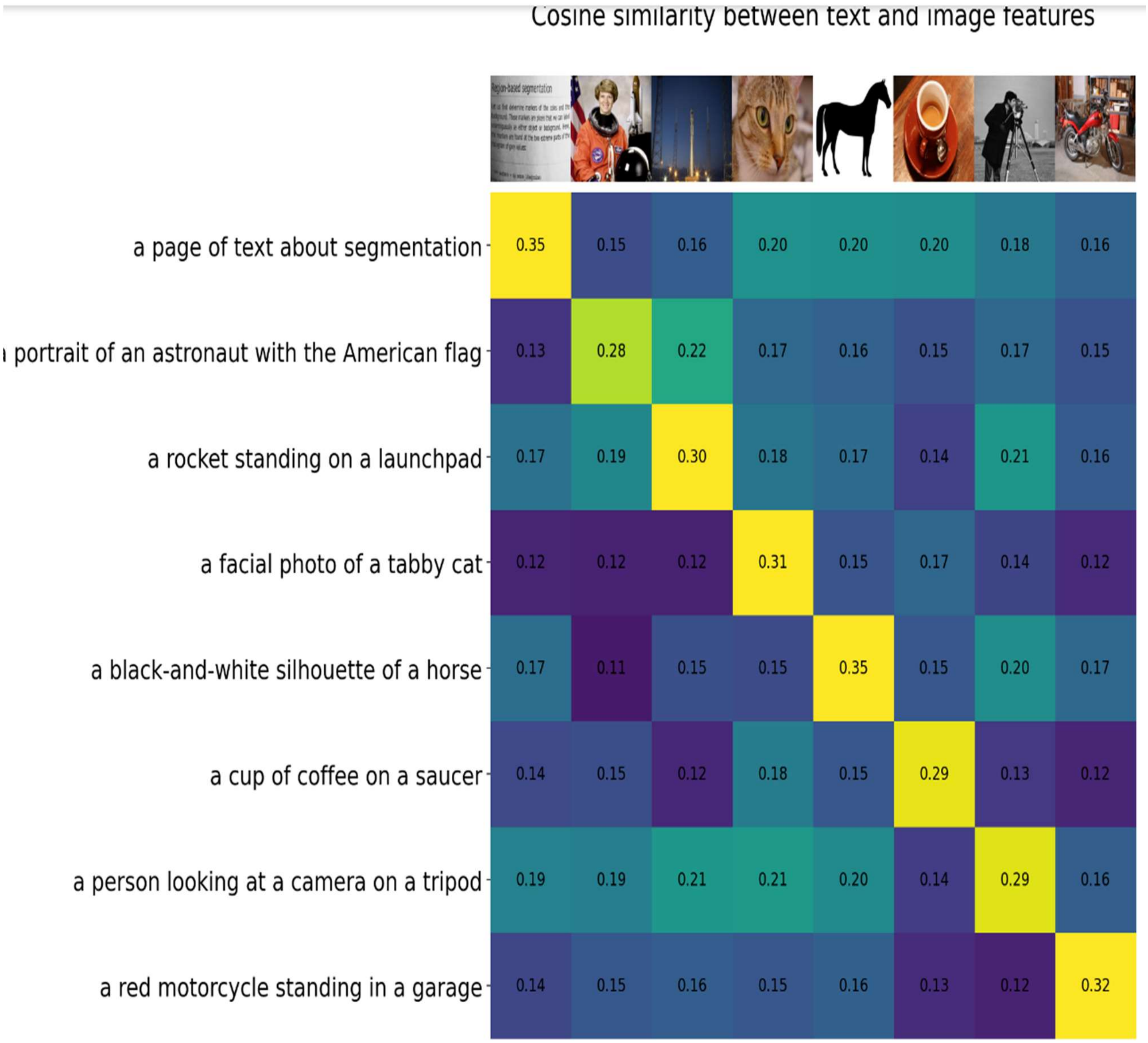


Fig 5.3 Cosine similarity between image and text

Zero shot object detection and image classification:

These are some of the results that are obtained using the vit-B/32 model which shows the probability of the given image and text.

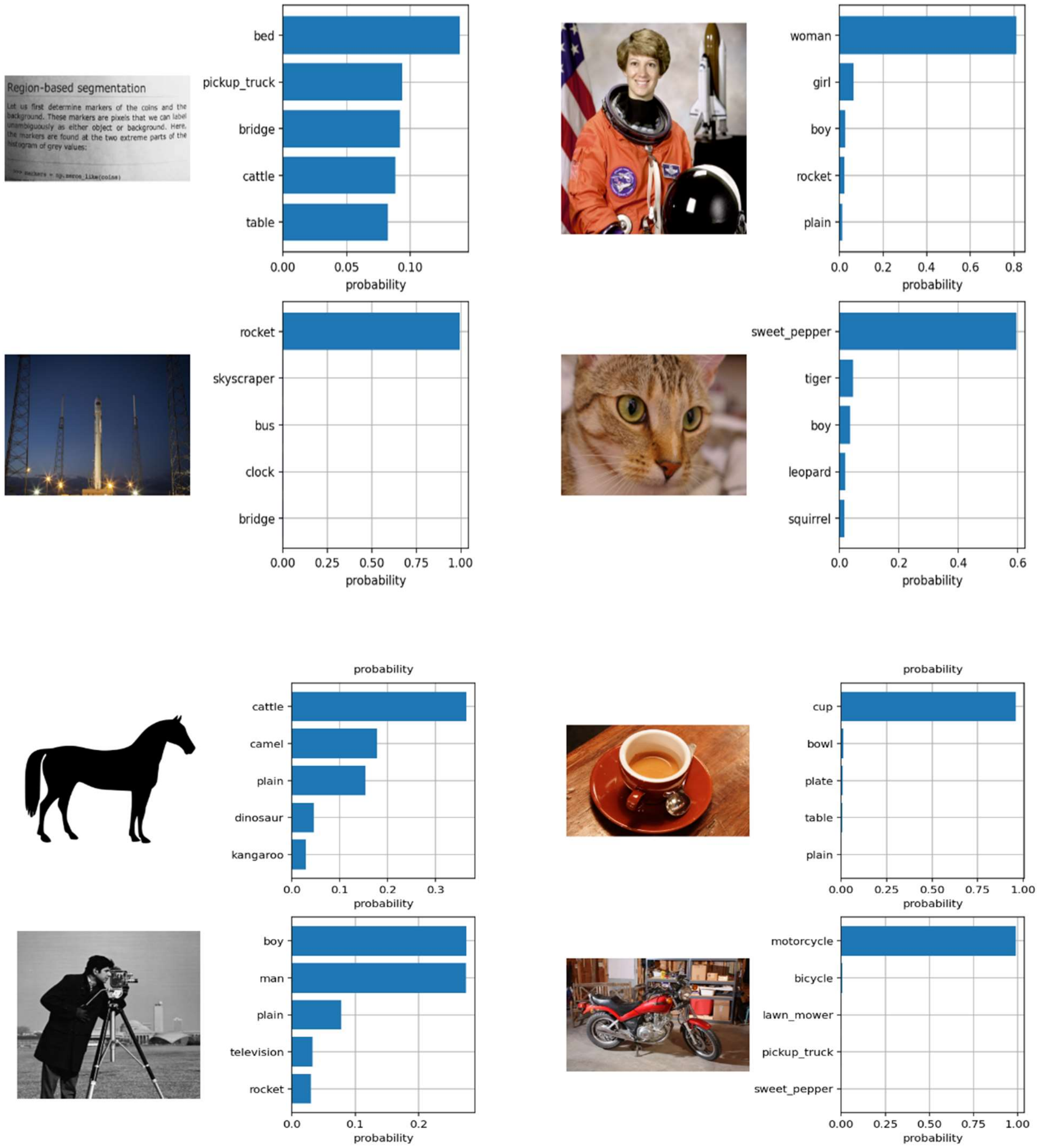


Fig 5.4 image classification and object detection probability

The Proposed system had shown a prominent result for the given dataset. It provided better results compared to the existing system, where the proposed system uses the occlusion algorithm and by using the PyTorch.

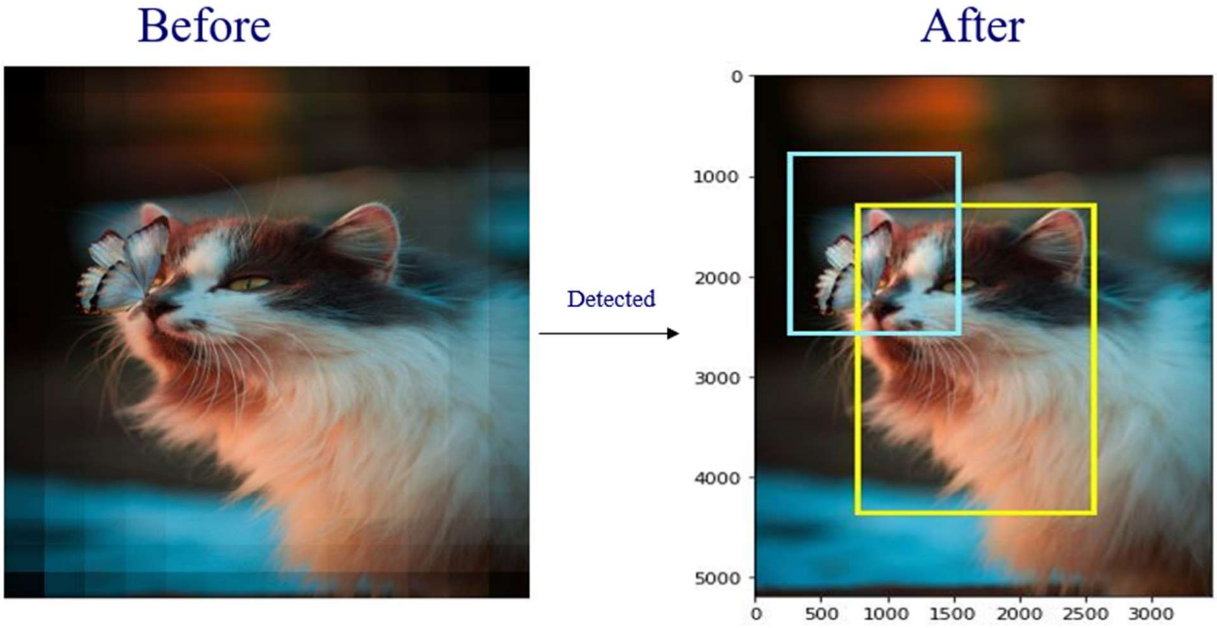


Fig 5.5 Object detection using occlusion algorithm

CONCLUSION AND FUTURE SCOPE

To sum up, research into visual semantic alignment for zero shot object detection tries to address the drawbacks of conventional object detection techniques. These algorithms are able to detect items that they have never seen before by adding semantic information from outside sources, which results in greater probability and a wider variety of detection.

In addition, the use of zero-shot object detection allows the model to generalise to unseen object categories, which is crucial for real-world applications. As a result, there is no longer a need for costly and time-consuming annotation efforts because the model can learn to identify new objects based on their semantic relationships to objects it has already seen.

There is still considerable research to be done in this area, including enhancing the zero-shot object detection models' reliability and accuracy as well as investigating how they may be used for more difficult tasks like image editing and visual storytelling. Additionally, using additional modalities like audio and text could improve the capabilities of these models even more.

REFERENCES

1)Rahman, S., Khan, S., & Barnes, N. (2022). Polarity Loss: Improving Visual-Semantic Alignment for Zero-Shot Detection. IEEE Transactions on Neural Networks and Learning Systems.

2) Yang, Y., Zhao, L., & Liu, X. (2022). Iterative Zero-Shot Localization via Semantic-Assisted Location Network. IEEE Robotics and Automation Letters, 7(3), 5974-5981.

3) Mao, Q., Wang, C., Yu, S., Zheng, Y., & Li, Y. (2020). Zero-shot object detection with attributes-based category similarity. IEEE Transactions on Circuit and Systems II: Express Briefs, 67(5), 921-925.

4) Liu, J., Chen, Y., Liu, H., Zhang, H., & Zhang, Y. (2022).From Less to More: Progressive Generalized Zero-Shot Detection With Curriculum Learning. IEEE Transactions on Intelligent Transportation Systems.

5) Yan, C., Zheng, Q., Chang, X., Luo, M., Yeh, C. H., & Hauptman, A. G. (2020). Semantics-preserving graph propagation for zero-shot object detection. IEEE Transactions on Image Processing, 29, 8163-8176.

6) Sun, X., Gu, J., & Sun, H. (2021). Research progress of zero-shot learning. Applied Intelligence, 51(6), 3600-3614.

7)Yan, C., Chang, X., Li, Z., Guan, W., Ge, Z., Zhu, L., & Zheng, Q. (2021). Zero nas: Differentiable generative adversarial networks search for zero-shot learning. IEEE transactions on pattern analysis and machine intelligence.

8)Wei, K., Deng, C., Yang, X., & Tao, D. (2021). Incremental zero-shot learning. IEEE Transactions on Cybernetics.

9) Tian, Y., Zhang, Y., Huang, Y., Xu, W., & Ding, Z. (2022).Differential Refinement Network for Zero-Shot Learning. IEEE Transactions on Neural Networks and Learning Systems.

10) Lv , W., Shi, H., Tan, S., Song, B., & Tao, Y. (2022). A dynamic semantic knowledge graph for zero-shot object detection. The Visual Computer, 1-15.

11) Rahman, S., Khan, S. H., & Porikli, F. (2020).Zero-shot object detection: Joint recognition and localization of novel concepts. International Journal of Computer Vision,128(12), 2979-2999.

12) Yan, C., Chang, X., Luo, M., Liu,H., Zhang, X., & Zheng, Q. (2022).Semantics-guided contrastive network for zero-shot object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence.

13) Xie, Y., He, X., Zhang, J., & Luo, X. (2020). Zero-shot recognition with latent visual attributes learning. Multimedia Tools and Applications, 79(37), 27321-27335.

14) Gao, R., Hou, X., Qin, J., Shen,Y., Long, Y., Liu, L., ... & Shao,L. (2022). Visual-Semantic Aligned Bidirectional Network for Zero-Shot Learning. IEEE Transactions on Multimedia.

15) Du, P., Zhang, H., & Lu, J. (2020). Learning Discriminative Projection With Visual Semantic Alignment for Generalized Zero Shot Learning. IEEE Access, 8, 166273-166282.

16) Liu, Y., Dang, Y., Gao, X., Han, J., & Shao, L. (2022).Zero-Shot Learning With Attentive Region Embedding and Enhanced Semantics. IEEE Transactions on Neural Networks and Learning Systems.

17) Li, Y., Liu, Z., Yao, L., & Chang, X. (2021). Attribute-modulated generative meta learning for zero-shot learning. IEEE Transactions on Multimedia.

18) Xie, G. S., Zhang, X. Y., Yao, Y., Zhang, Z., Zhao, F., Shao, L. (2021). Vman: A virtual mainstay alignment network for transductive zero-shot learning. IEEE Transactions on Image Processing, 30, 4316-4329.

19) Zhao, P., Zhang, S., Liu, J., & Liu, H.(2021). Zero-shot Learning via the fusion of generation and embedding for image recognition. Information Sciences,578, 831-847.

20) Xu, W., Xian, Y., Wang, J., Schiele, B., & Akata, Z. (2020). Attribute prototype network for zero-shot learning. Advances in Neural Information Processing Systems, 33, 21969-21980.