Master's thesis

Embedded and Intelligent Systems, 120 credits

A Rule-based approach for detection of spatial object relations in images

Computer Science and Engineering

Master's thesis (DT7001), 30 credits

Halmstad, January 16th, 2023
Wahaj Afzal

HALMSTAD UNIVERSITY

A Rule-based approach for detection of spatial object relations in images

Master Thesis

2022

Author: Wahaj Afzal

Supervisor: Eren Erdal Aksoy

Examiners: Slawomir Nowaczyk and Jens Lundström

School of Information Science, Computer and Electrical Engineering
Halmstad University
PO Box 823, SE-301 18 HALMSTAD
Sweden

A Rule-based approach for detection of spatial object relations in images

Wahaj Afzal

Master thesis report IDE 12XX

School of Information Science, Computer and Electrical Engineering

Halmstad University

ISSN XXXXX

Abstract

Deep learning and Computer vision are becoming a part of everyday objects and machines. Involvement of artificial intelligence in human's daily life open doors to new opportunities and research. This involvement provides the idea of improving upon the in-hand research of spatial relations and coming up with a more generic and robust algorithm that provides us with 2-D and 3-D spatial relations and uses RGB and RGB-D images which can help us with few complex relations such as 'on' or 'in' as well. Suggested methods are tested on the dataset with animated and real objects, where the number of objects varies in every image from at least 4 to at most 10 objects. The size and orientation of objects are also different in every image.

Keywords: Spatial relations, Deep learning, Computer vision, Artificial intelligence.

Acknowledgments

I would like to thank the dataset and source code providers which helped in initializing and studying the initial techniques used in deep learning and computer vision in the determination of spatial relations. I would like to thank and acknowledge the time and help provided by Andreea Bobu who provided help and guidance along with source code and dataset until the last step. During the thesis, I would like to thank Tucker Hermans and Weiyu Liu for providing the dataset.

Last but not least I would like to thank my supervisor from Halmstad University Eren Erdal Aksoy for providing all the help and feedback during the thesis work which helped in achieving the goal and completing the thesis.

Wahaj Afzal.

School of Information Science, Computer and Electrical Engineering
Halmstad University
PO Box 823, SE-301 18 HALMSTAD
Sweden

List of Figures

Acronyms

CNN: Convolution Neural Network

GNN: Graph Neural Network

MLPs: MultiLayer Perceptron

NER: Named Entity Recognition

VGG: Visual Geometry Group

R-CNN: Regions with Convolution Neural Network

FasterRCNN: Faster - Regions with Convolution Neural Network

NMS: Non-Maximum Suppression

IOU: Intersection Over Union.

IOUa: Intersection Over Union using Area of boxes.

IOUc: intersection Over Union using Center point of boxes.

School of Information Science, Computer and Electrical Engineering
Halmstad University
PO Box 823, SE-301 18 HALMSTAD
Sweden

Contents

1 Introduction

This chapter contains some key topics such as background, problem formulation, novelty and contribution, and datasets related to this thesis.

1.1 Background

A lot of research has been done and going on spatial knowledge and reasoning as human beings are relying highly on automation assistance provided by robots in real-world applications. Many representations and reasoning techniques have been proposed and implemented that aim at representing the spatial knowledge underlying the relations of the objects present in the environment or an image. Techniques used in finding the solution to spatial relations are taken from neural networks such as CNN [15][18][17], GNN [27], Transformers [38][24] , MLPs [10], and NER [11]. Computer vision plays an important role in the detection of relations among the objects present in the image. Computer vision presents several tasks and techniques to recognize objects in an image such as object detection, semantic segmentation, and semantic scene understanding.

1.1.1 Object detection

Object detection is a computer vision technique for detecting objects in an image. Object detection provides us with information related to an object, such as classes (animal, human, or any other object), and the position of an object in the image. In computer vision, many techniques and methods are available which can be used as object detectors. Along with computer vision, deep neural networks (DNN) also provide approaches that can be used as object detectors. Some of the common object detectors from both computer vision and DNN are support vector machine (SVM), CNN, and Faster R-CNN [35].

1.1.2 Semantic segmentation

Semantic segmentation is an approach to deal with multiple objects which belong to the same class in the image. In semantic segmentation, every pixel in the image is assigned to a different class [4]. To detect different objects in those classes instance semantic segmentation is used which identifies different objects present in the classes which are identified by semantic segmentation. Thus, semantic segmentation identifies different classes, but the instance semantic segmentation identifies objects within these classes. [5]

1.1.3 Semantic scene understanding

An attempt to understand and analyse the image is semantic scene understanding. In the analysis of the image under semantic scene understanding, every aspect is noted [31]. Every aspect can be described as what objects are present in the image, the relationship of objects, and the layout of the image [6].

1.1.4 Spatial relations

Spatial relations define the location of the object in the image following the reference object [7] . In an image, if two objects are present, in that case, object relation is defined as how the position of an object (child object) is in the image environment relative to another object (parent object). The spatial relation refers to the association between each child object and all the parent objects found in the picture.

1.1.5 High-level angle

A high-level angle shot is taken when the camera is looking down at the subject or scene from an elevated perspective. To achieve this camera should be placed higher and then angled down on the subject. The level of height between the subject and the camera depends on the user. [8]

1.2 Problem Formulation

The main objective of this thesis is to implement a spatial reasoning method that can take RGB images as input and provide as output pairwise object relations such as on, left, right, in front, etc as output. The input RGB images will be an image which is taken from a high-level angle and has a different number of objects. A high-level angle is selected so that in the future the developed algorithm can easily be deployed on robots. The number of objects in every picture can vary from four to eight objects. The number of objects will be different in every image; thus, the output will take one object as a reference or parent object and will provide spatial relation relative to all other children objects.

Every input RGB image will be of a different dimension and no pre-set value of dimension is used for input RGB images. Input images can be 240x240x3 or 7000x7000x3.

The ability to recognize spatial relations in RGB images taken from the dataset is of utmost importance. To achieve the objective an algorithm is going to be designed using deep neural networks and computer vision techniques and algorithms which will be implemented and performed on real-world examples. Recent work done mainly focus on RGB images only, not having real-world objects or having a limited number of spatial relations. However, this thesis is not going to be limited to these constraints.

The developed algorithm will be divided into different steps, described as follows:

1. In step one RGB image will be given as input to the algorithm to detect the objects. The input RGB images will be in .png format. Figure 1:1 shows the first step the input and output of the first step. Equation 1.1 shows how the step will move forward and the stages involved in step 1. Initially, faster RCNN will be tested as an object detector to get the desired result. If faster RCNN provides desired results, we will move on to the next step.

   Image (m x n x 3) ➔ Object detector (Faster R-CNN) ➔ Image with Bounding boxes (1.1)

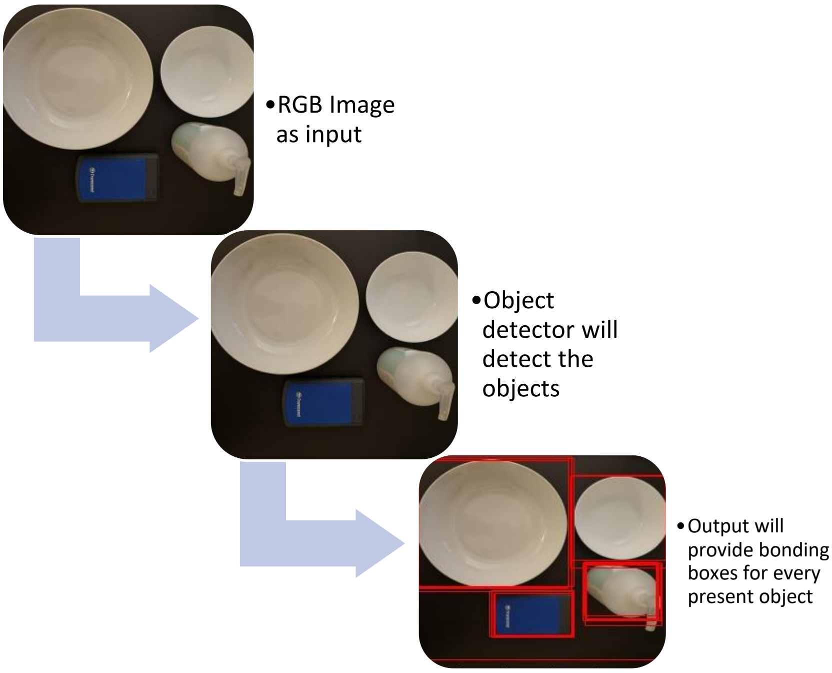

•RGB Image as input
•Object detector will detect the objects
•Output will provide bonding boxes for every present object

Figure 1:1 First step of the algorithm

2. The second step consists of the image with bounding boxes. So, the output of step 1 will be converted into the input of step 2. Equation 1.2 shows how the output of step 1 will be used in step 2. Different computer vision techniques will be tested here to get the accurate number of bounding boxes, the output image of step 2 must have one box for each object present in the image. So, we will have n number of bonding boxes for n number of objects. Figure 1:2 shows step two with the input and output.

   Image with Bounding boxes ➔ Reduction in number of bounding boxes ➔ Final image with bounding boxes (1.2)

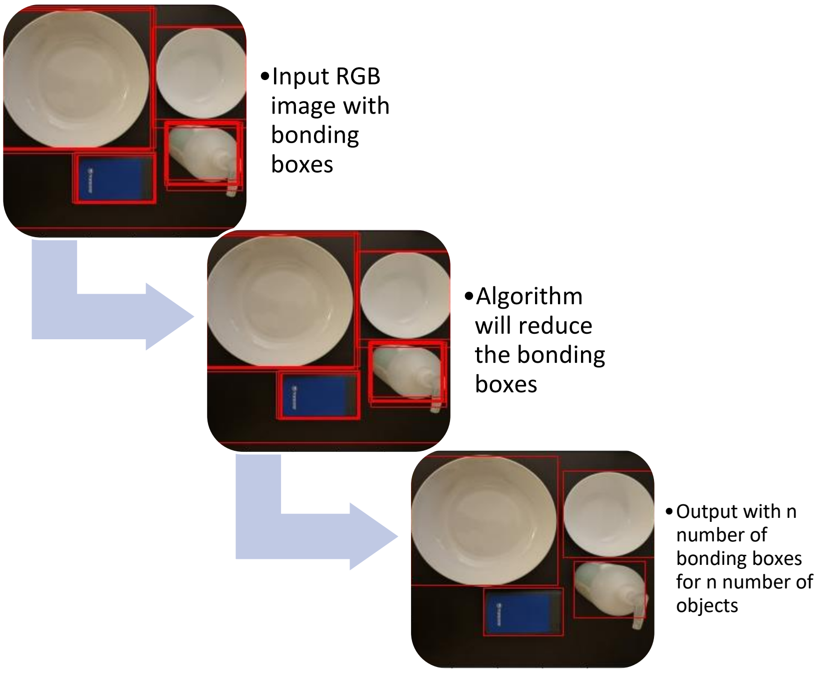

•Input RGB image with bonding boxes
•Algorithm will reduce the bonding boxes
•Output with n number of bonding boxes for n number of objects

Figure 1:2 Second step of the algorithm

3. The third step will provide the ability to recognize the spatial relationships among the objects present in the RGB image. The input for the third step will be an image which is taken as an output from step two, and the output of the third step will be spatial relations in form of a table, as shown in equation 1.15. Equation 1.15 to equation 1.23 represents how relations among objects will be determined. Figure 1:4 provides information about how step three will work.

   The spatial relationships are defined as in front (F), in back (B), left (L), right (R), front left (FL), back left (BL), front right (FR) and back right (BR). The spatial relationship between two objects can be defined as λ. The relation between object 1 and object 2 can be expressed as:
   λ(O_1,O_2) ϵ [F,B,L,R,FL,BL,FR,BR]

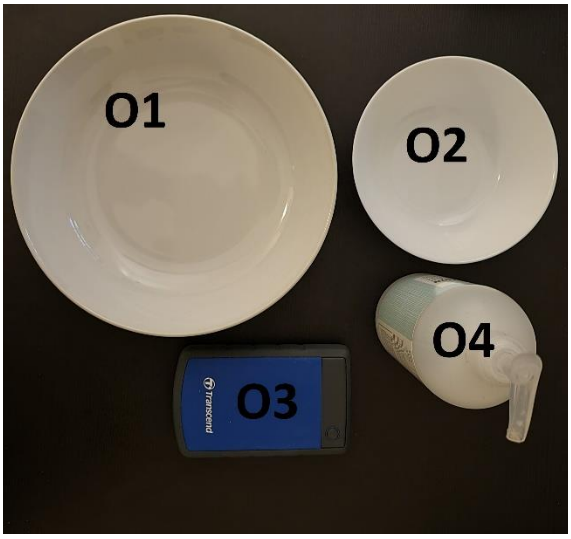


Figure 1:3: A labelled image to use as an example.

Taking Figure 1:3 as an example, the λ (spatial relations) will be determined as follow from equation 1.3 to 1.14:

λ (O_1, O_2) = L (1.3)

λ (O_1, O_3) = F (1.4)

λ (O_1, O_4) = FL (1.5)

λ (O_2, O_1) = R (1.6)

λ (O_2, O_3) = FR (1.7)

λ (O_2, O_4) = F (1.8)

λ (O_3, O_1) = B (1.9)

λ (O_3, O_2) = BL (1.10)

λ (O_3, O_4) = L (1.11)

λ (O_4, O_1) = BR (1.12)

λ (O_4, O_2) = B (1.13)

λ (O_4, O_3) = R (1.14)

Final Image with bounding boxes ➔ Extraction of spatial relations ➔ Table with all relations. (1.15)

In-front ➔ Angle between center points is $\frac{\pi}{2}$ (1.16)

In-back ➔ Angle between center points is > $\frac{\pi}{2}$ (1.17)

Front left ➔ $\frac{\pi}{2}$ > Angle between center points is > 0 (1.18)

Back left ➔ 0 > Angle between center points is > -$\frac{\pi}{2}$ (1.19)

Front right ➔ 0 < Angle between center points is > $\frac{\pi}{2}$ (1.20)

Back right ➔ 0 > Angle between center points is < $\frac{\pi}{2}$ (1.21)

Left ➔ Angle between center points is 0 (1.22)

Right ➔ π > Angle between center points is > -π (1.23)

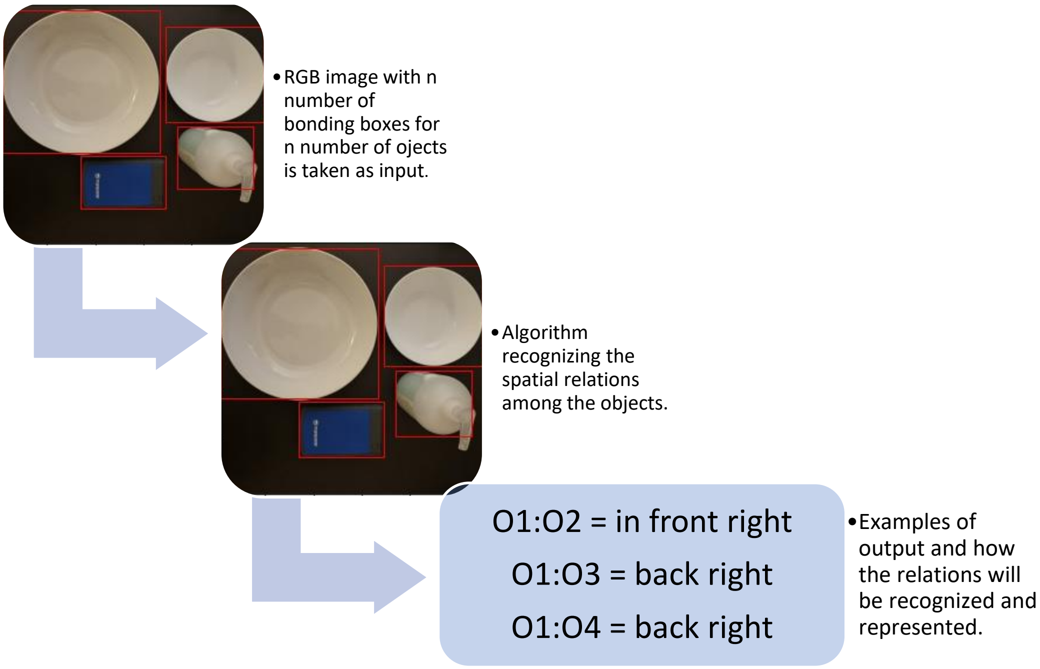

•RGB image with n number of bonding boxes for n number of ojects is taken as input.
•Algorithm recognizing the spatial relations among the objects.
O1:O2 = in front right
O1:O3 = back right
O1:O4 = back right
•Examples of output and how the relations will be recognized and represented.

Figure 1:4 Third step of the algorithm

The above-described three steps show how the problem will be solved and the algorithm will be designed to achieve the objective. RGB image shown in the steps as an example, is part of the dataset. How all three steps will solve the problem will be further described in the methodology section of the report.

1.2.1 Contribution and novelty

In this project, the contribution will be to bring up an algorithm that can recognize and deal with the spatial relationships among objects in the RGB images. The novelty will be to implement a robust and generic spatial relation detection algorithm from RGB images using advanced deep neural network architectures and computer vision techniques. The lack of an open RGB-D dataset with real-world objects is an obstacle in this implementation. The implementation of a generic and robust algorithm using advanced deep neural network architectures from RGB images that contain real-world objects has not been done previously and published.

1.2.2 Research Questions

To successfully fulfil the aim and purpose of the thesis, the following research questions will be answered:

- Is it possible to detect 3-D relations from given RGB-D images?
- How many complex relations (e.g an object inside other) can we detect using RGB images?
- Do the depth channel help in detecting the relations?

1.2.3 Dataset

The dataset with the required specifications was a challenge and hard to find and creates two datasets from past studies [38][24] are studied and examined. Both datasets have images with RGB and depth information, both datasets have 3-D images. The Dataset used in the study of [38] has two objects as shown in figure 1:5, and frames for each spatial relation depend on environments and samples while creating the dataset. The term Environments is a variable that controls the number of tabletop environments in parallel that are being used. If a dataset is created with ten environments and ten samples, then the frames present in the dataset will be one hundred.

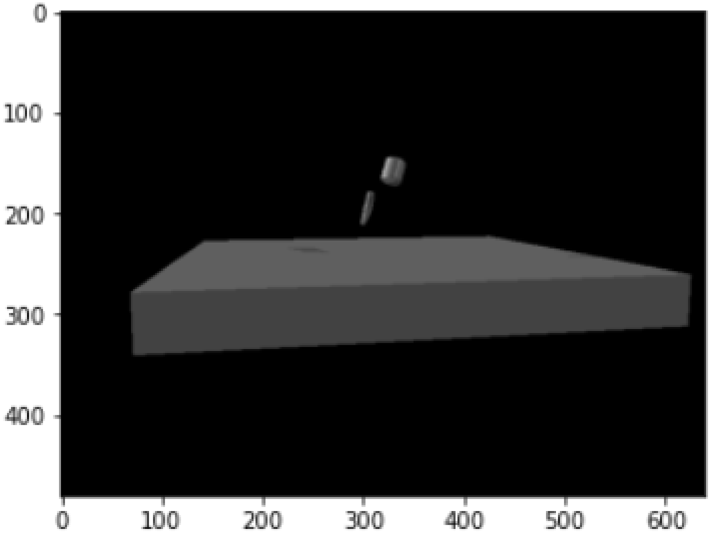


Figure 1:5: Dataset created with 10 environments and 10 samples showing two objects.

The Dataset used in the study of [24] has variable objects in every scene and the objects number varies between six to nine objects as shown in Figure 1:6 and Figure 1:7, two images with RGB and depth information per scene are available and overall, almost 23,044 frames are available.

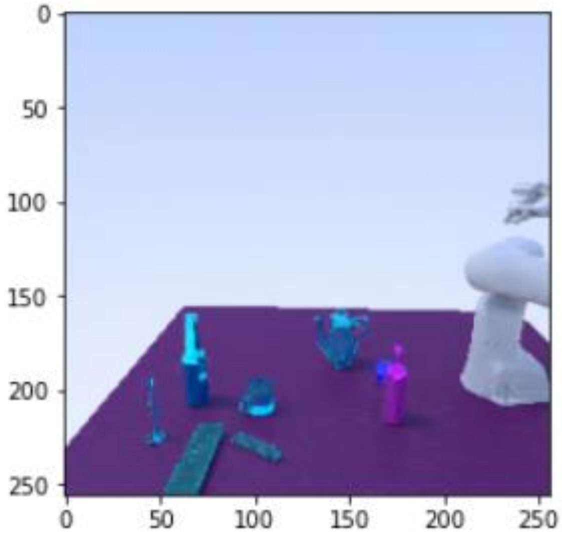


Figure 1:6: Scene from side view showing eight objects.

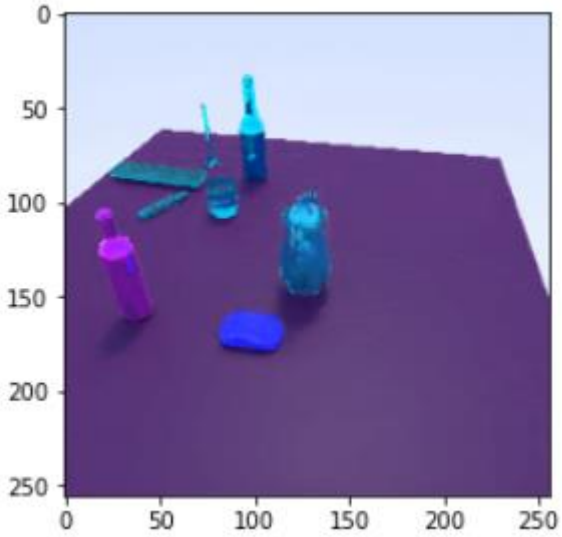


Figure 1:7: Scene from the side where the robot is placed showing eight objects.

The main spatial relations visualized in the initial few scenes of the datasets are right, left, front, back, above, and below in both datasets. Figure 1:8 visualizes the scenes in 3-D. The object used in this dataset is cutlery items, dishes, mugs, bottles, or items normally used in the kitchen. Objects are from 35 different classes and are 335 in number which produces arrangement sequences of more than 100,000.

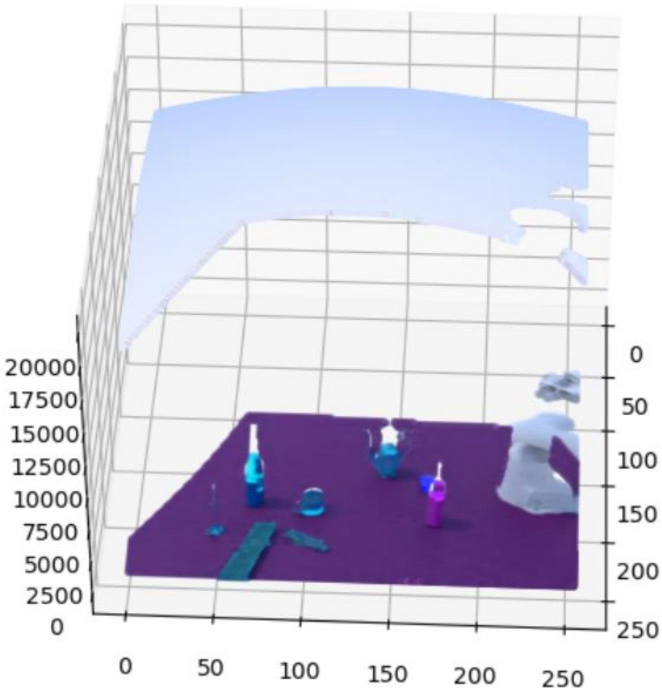


Figure 1:8: 3-D representation of the scene from the dataset used in [24]

Along with these two datasets, a small dataset is created which contains RGB images with real objects as shown in Figure 1:9 and Google drive link in the appendix, and as in both available datasets it is visible that objects presented are not real and obtaining point clouds is also not possible at this stage which leads to creating problems in obtaining few 3-D spatial relations such as "in" or "under".

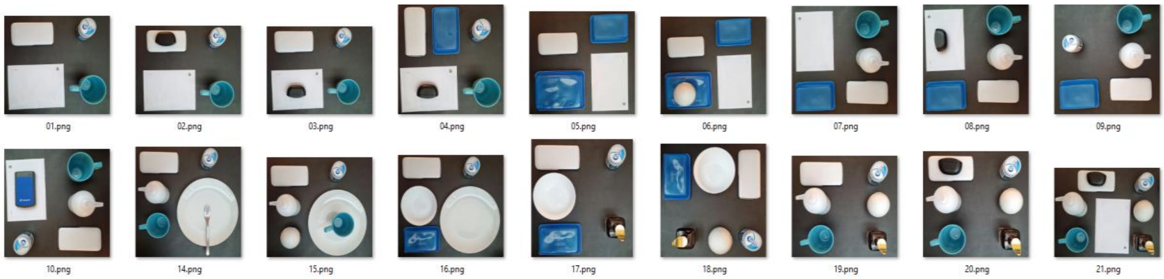


Figure 1:9: Small dataset created to test the algorithm.

2 Literature review

As mentioned in the introduction many researchers are working on learning spatial relations and coming up with new techniques which can make human life easier through the usage of robots and automation. Commonly used techniques in spatial relation reasoning are CNN, GNN, Transformers, NER, and MLPs. Most past researchers used spatial knowledge to represent spatial relations among objects which are usually restricted to one particular aspect of the space. The research done and related to the objective of the project presented in the problem formulation in chapter 1 is as follows:

1. Graph-based Visual Manipulation Relationship in Object Stacking Scenes by [17], this technique provides us with RESNET 101 and VGG16 as base models to train the object detection and feature extractor. Graph convolution network is used to determine the relationship between objects. This study focused on three different types of relations on, under, and no-relation. Images used to determine the relations were RGB images taken from the data set VMRD (visual manipulation relationship dataset). As CNN is used in this study the results taken out from the experiments have high accuracy, but that's only possible in the case of determining a smaller number of relations as the number of detected relations gets higher and the accuracy of the designed network drops.
2. StructFormer: Learning Spatial Structure for Language-Guided Semantic Rearrangement of Novel Objects by [24] worked on StructFormer a novel transformer-based technique and a model that enables the robot to rearrange the unknown objects present in the environment into meaningful spatial structures based on high-level language instructions. In this study total of 335 real-world objects from 35 different classes were used to create more than 100,000 rearrangement sequences. Dataset created in this study is divided into four different semantically meaningful spatial structures: Circles, lines, towers, and table settings (prepare the table for dinner) and these spatial relations can be done in three different sizes: small, medium, and large. Franka Panda Robot is used to test the algorithm, the robot had an RGB-D camera to evaluate the selected real-world object manipulation, but on the other hand, if a large portion of the object is hidden from the camera or is covered by any other object the algorithm will detect the object as a different object. When an object is unreachable, and the robot has to move to reach the object motion planning fails sometimes.
3. Robotic understanding of Spatial Relationships Using Neural-Logic Learning [20] research work mainly focused on data acquired by RGB-D sensors consisting of pair-wise point clouds of objects in the scene. This study provides us with neural logic networks, feedforward neural networks, and Faster R-

CNN to determine the relations among the objects present in the scene. Spatial relation finding is divided into three blocks: Grounding block, Spatial logic block, and inference block. Grounding blocks and spatial logic blocks are created by using a feedforward network. In grounding block for object detection faster, R-CNN is used and k-means clustering is done to select the dominant color of the object present in the bonding box. The inference block takes input from the spatial logic block to infer the complex relations in the objects. Relations determined in this study are Left, Right, Front, Below, Behind, and Front. From inference block by using logic rules in form of first-order logic few more complex relations are also determined in this study which are On, Beneath, Front left, and Between. In this study, the RGB-D sensor is used to learn the spatial relations which have been done rarely in this area of study but this also came with the limitation of determining the spatial relations, if the point of view of the RGB-D sensor is at a certain point and angle it will detect precise spatial relationships but it will miss the blurred out spatial relations.

4. Approximation-based technique determined by using CNN is proposed in the Approximation of dilation-based spatial relations to add structure constraints in a neural network [15] according to the authors of this study CNN provides faster calculations with low usage of memory. To encode the relative location of the target from the source as a set of fuzzy mathematical morphology elements a priori is proposed. Also, a comparative study of convolutional-based approximation of the mathematical morphology dilation is proposed. In this study, two classes of relations are defined one is based on closeness and the other is based on directional relative. In closeness relations determined are close to, far from, and inside of. Directional relative relations determined are similar to, “to the left of” and “right”. This study is created without the use of a dataset and is based on an approximation of source and target positions. CNN is used to learn the spatial relations between source and target which provided highly accurate results even in the approximation-based experiments, but this algorithm is not tested on a dataset of images or RGB images.
5. Tasks like Sweeping Objects in Line, Food Cutting, and Block Unstacking can also be performed by determining spatial relations. Similar tasks were provided by the research of Relational Learning for Skill Preconditions by Mohit Sharma and Oliver Kroemer [27]. Pairwise object interaction is used to learn object relations, and ResNet is used as the base model to process the input model. Two different networks are used for the precondition learning model, used networks are Relational Networks (RNs) and Graph Neural Networks (GNNs). In this study, simulations provided access to a large set of ground truth data such as contacts and the proposed approach can be used for precondition learning in a sample efficient manner. However, the method is evaluated by only using the mean of the 3D position and bounding box of the blocks as input. Precondition data can be fed to the network which can produce

accurate results but on the other hand, it will be taken from overfitted trained dataset. Also, only a 3D dataset is used, and it is not tested how the method will behave with the 2D dataset.

6. RGB image dataset is used in many studies to determine spatial relations. SORNet: Spatial Object-Centric Representations for Sequential Manipulation by [10] provided us with the approach of SORNet. SORNet is simpler and can solve spatial reasoning tasks for unseen and unknown object instances without retraining the network. The proposed SORNet method is based on the visual transformer. Two layers of MLPs are used for predicting relations, the first layer is used for predicting unary relations and the second layer is used to determine binary relations. Four different relations are determined here which are left, right, behind, and front. SORNet can be used in the manipulation of novel objects without retraining the network but the data provided to the network is already processed with large-scale pre-training because of which proposed methods works with an arbitrary number of unseen objects.
7. Learning perceptual concepts by bootstrapping from human queries by [38] proposed a new approach through which a robot learns a low-dimensional variant of a concept, and generates a larger data set from which learns the concept in the high-dimensional space. This allowed the robot to arrange the objects in different positions and relationships defined are: Above, near, aligned (vertical, horizontal), upright, forward, front, and top. 7-DoF Franka robot is used to demonstrate the learned approach. It has been tested on this method how a human can teach a robot by providing or manipulating with very little data which allows the robot to quickly learn the concept in lower dimensional space. The study still needs to investigate how concepts taught by real people would fare. The noise can also affect the results but not too much.
8. Visual Manipulation Relationship Network is a CNN-based approach that is also used in determining relations among objects present in RGB images. Visual manipulation relationship recognition in object-stacking scenes by Hanbo Zhang et al [18]. The same approach is used to find relations among objects VGG-16 and ResNet-101 is used as a base model for feature extraction and Faster-RCNN and SSD are used for object detection. RGB dataset Visual Manipulation Relationship Dataset (VMRD) is used from which 4683 images are used. For simplification network was divided into three different steps: Feature extractor, Object detector, and VMR predictor. The method works well for known objects and can detect objects from edge to edge, and it is possible to use the method to detect unseen objects as well but for that strong object detector is needed to perform the task. Also, to increase the performance better GPU is needed.
9. Graph R-CNN for Scene Graph Generation by [26] described the relations of subject and object for this task a relation proposal network (RePN) is introduced which models the relationships between the objects and learns to efficiently estimate the relatedness of two objects. Two MLPs of the same structure but different hyperparameters are used to obtain projection functions

for subjects and objects. Attentional GCN is introduced to obtain contextual information from the graph structure. Attentional is an extended form of conventional GCN, in which authors used 2-layer MLPs over linked node features and computed a softmax over the resulting scores. Defined relations vary according to the picture and type of contact between objects that depend on the complexity of the relations. Real-world images are used to learn spatial relations between subject and object, spatial relations are not predefined, and a tree is being generated which can visualize the spatial relation with less accuracy with respect to other CNN methods also results provided by the tree are at times somewhat difficult for the reader to interpret.

10. Named entity recognition (NER) is a speech recognition network that is proposed in Semantic Scene Manipulation Based on 3D Spatial Object Relations and Language Instructions by Kartmann et al [11]. In this study determined relations are right, left, front, behind, close to, inside, on top, above, under, closer, farther, another side, between, and among. The humanoid robot is used to demonstrate the learning outcomes from the proposed method in which pick, and place were performed. The proposed method can manipulate 2D spatial relations by taking command in natural language, implementation of the proposed method on a robot also works smoothly. At this time method only understands a few sets of commands and a few simple models are being learned by the method and more complex manipulation actions are missing.
11. Obtaining results from the robot which are precise after giving commands through speaking in natural language is difficult to achieve. Composing pick and place tasks by grounding language by Oier and Wolfarm [1] provided us with a method with two parts first a grounding model that identifies the referred object and second part a neural network that predicts and determines the final location of the object conditioned on a set of learned spatial relations. The grounding network model uses Mask-RCNN to detect the object after which the neural network using spatial RelNet and pixel-wise probability maps per relation place the object at the right spot. In this study, the proposed method is the first of its kind which allows new and non-expert users to instruct the robot which can perform tabletop manipulations and it can be done effectively and realistic environment, but on the other hand, the method came with limitations of failure in case of placing large objects because methods work on 2D images and input. Also, GPD often fails to find and grasp object which is not openly visualized, are covered by other objects, or when method face noise in the measurements. Pick and place actions in tabletop tasks are also limited.

3 Project Plan

To answer the research problem explained in the previous section, general overview will be presented that will be used for research approach. From the perspective of computer science, the expected workflow will be as follows:

1. **Data collection** we will identify as much as possible about the literature work done in spatial relations among objects and collect the data on which those research and literature works are based.
2. **Analysis of reasoning techniques** Identifying reasoning techniques that can be used in obtaining the proposed goal, we will also evaluate the effectiveness and efficiency of techniques.
3. **Algorithm Design** In order to obtain a solution to the research problem an algorithm will be developed. This step will likely require devising additional heuristics to obtain an efficient method.
4. **Implementation and testing** Using a different dataset of images that have RGB images the algorithm will be tested and implemented. The algorithm will test in a way that will prove whether the research problem is fulfilled or not.

4 Methodology

After the literature review and obtaining the desired datasets the next step in the project plan is to analyse the reasoning techniques that are available to identify the pairwise spatial relations between two or more objects. Analysis of technique will cover the parameters of effectiveness and efficiency of the techniques along with the flexibility as the main goal is to obtain as many spatial relations as possible and that too in the high number of the objects present in the scene. After analysis of the reasoning technique, an algorithm to determine the spatial relations among objects will be created and tested as the project moves forward.

4.1 Faster R-CNN

Object detection is one of the topics in computer vision on which many researchers worked and achieved new success in making it better. Faster R-CNN is one of the techniques used to detect objects in an image. Faster R-CNN is a deep convolution network. This network is single end-to-end and unified. Faster R-CNN detects objects accurately and quickly. The faster R-CNN network detects objects by predicting the accurate locations of the objects in the images which predictions can later be seen in form of bounding boxes. Faster R-CNN is an evolved form of R-CNN and Fast R-CNN. [34][32]

4.2 Non-Maximum suppression (NMS)

Non-maximum suppression is a computer vision technique. The selection of one most common entity from many different entities which are overlapping each other is the main goal of this technique. In most cases, NMS is used as an aid to object detectors, where object detectors provide bounding boxes, and NMS works as a filter and provides the most common bounding box for the object. The bounding box provided by NMS is the best-suited bounding box for the object. Intersection over union (IOU) is the main prediction that helps NMS in filtering out the best bonding box. Working and use of IOU and NMS in this project will be described further in the description section. [3]

4.3 Description

To recognize the spatial relation in an image first we need to distinguish the objects present in the image. After obtaining separate identities for every present object in the image we can move forward with finding the spatial relations among those present and detected objects. Figure 4:4:1 will represents what steps are involved in this research project:

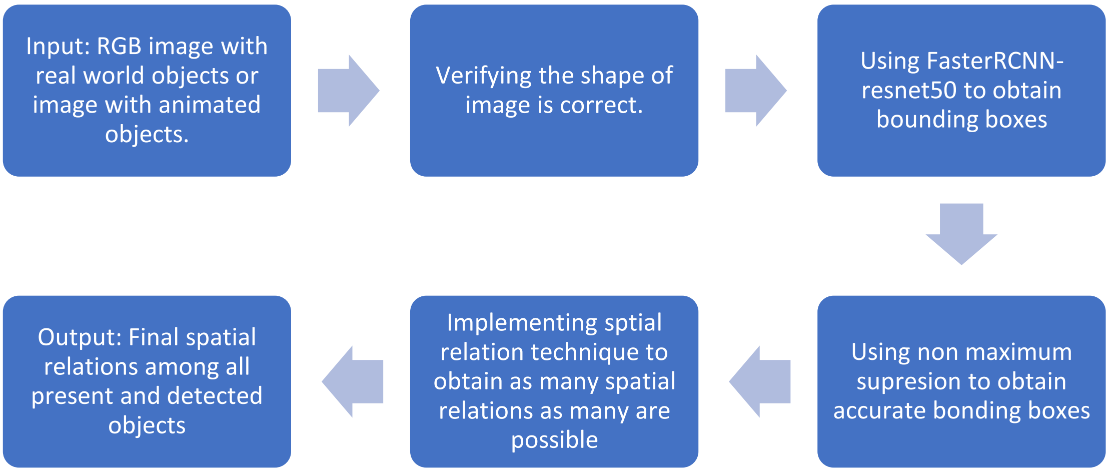

Input: RGB image with real world objects or image with animated objects.
Verifying the shape of image is correct.
Using FasterRCNN-resnet50 to obtain bounding boxes
Using non maximum supresion to obtain accurate bonding boxes
Implementing sptial relation technique to obtain as many spatial relations as many are possible
Output: Final spatial relations among all present and detected objects

Figure 4:4:1: Steps involved in this research project.

Images given as input have no limitations in terms of shape and objects they can be of shape (3,200,200) or (3,2300,2800) and can contain real-world objects or animated objects as the algorithm will detect and work on images of different shapes and object types.

Once it is figured out that the image is of the correct type and shape faster R-CNN taken from deep learning is used to get bounding boxes and used as an object detector. The bounding boxes provided by faster R-CNN contain information about the boxes as the position of the box, labels, and scores. At the initial test, it turned out that provided boxes by faster R-CNN are not so accurate and we obtained more than one box for one object. To solve this problem non-maximum suppression [3] taken from computer vision is used to get the accurate and final bounding boxes which will have one box for one object.

In non-maximum suppression, IOU (intersection area over union area) threshold is used to determine the number of boxes or number of objects to be detected in the scene. With low threshold of the IOU, boxes with small overlaps will be omitted, which would result on having fewer bounding boxes at the end.

IOU is used as a parameter to establish if two boxes are very similar. They describe the similarity between two: pictures, objects, or as in this case boxes. There is already one IOU that uses the area of two boxes and compares if both areas are similar and how much similar, that is called IOUa. IOUa, as shown in equation 4.1, uses the areas of under-study boxes and checks how much similar they are based on the intersection over union areas of the boxes. If IOUa is around 1, which means the intersection area is very much the same as the union area, then both boxes are very similar. And when

0 the non-similarity occurs. The problem with this method was that even though both boxes could be in different places of the picture if they have the same area, they were considered the same, and boxes would be ignored. Hence, a new IOU is created and it is called IOUc as shown in equation 4.2 which is based on the centres of both boxes, for example, the global centre point of both boxes are used(centres are local in the picture but are treated as global while comparing centre of one box with other) the previous method is used to determine the similarity; if both centres are very close or same the IOUc will be 1 and otherwise, the centre is not the same. The combination of both IOUa and IOUc is used because if just centre is used, the two concentric boxes with different dimensions could be interpreted the same and ignored.

$$IOUa = \frac{Intersection\ area\ of\ both\ boxes}{Union\ area\ of\ both\ boxes} \qquad (4.1)$$

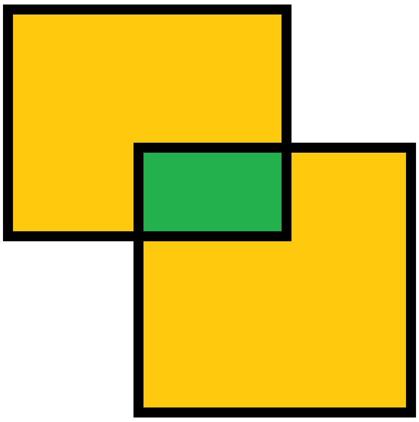


Figure 4:4:2: The image for IOUa green shaded area is the intersection area and the yellow area including the green area is the union area.

$$IOUc = \max\left(0,1 - \sqrt{(c_1x - c_2x)^2 + (c_1y + c_2y)^2}\right. \qquad (4.2)$$

$$IoU = \beta.IOUa + (1-\beta)IOUc \qquad (4.3)$$

$$\text{IOUM} = \text{Mean(IoU, axis} = 0) \qquad (4.4)$$

$$\text{IOU}_{\text{(kept)}} = \text{IOUM} < \alpha \qquad (4.5)$$

In equation 4.2 c_1x and c_1y are the center points of the first box and c_2x and c_2y are the center points of box two. IOU for every picture is determined separately by plotting the curve between the number of boxes needed and the IOU. The equation 4.3 have both IOUa and IOUc, and their ability to manipulate the IoU is depending on β. β decides in equation 4.3 which IOU to follow or how much effect of each IOUa and IOUc should be present in determining the IoU. In equation 4.4 mean of all IoU values calculated in equation 4.3 is taken. The mean calculated from equation 4.4 is used in equation 4.5 to decide on which bonding boxes should be kept. In other words, equation 4.5 suggests what should be the value of IOU by comparing IOUM with pre-set value of α. The curve is plotted which shows how to select IOU value for a specific number of boxes, and a curve is explained in the results chapter using Figure 5:1.

After determining the correct IOU, spatial relations are determined among the final detected object. To determine the final spatial relation the center points of boxes are used to determine the location of objects in the picture. After obtaining the location of objects in the picture they are compared with each other to determine the relation of both objects. To determine the pairwise relation between two objects axis system is used as shown in Figure 4:4:3 and Figure 4:4:4 and equation 4.4 is used to determine the center point of the objects.

$$\tan\theta = \frac{c_j y - c_i y}{c_j x - c_i x} \tag{4.4}$$

In equation 4.4 ciy and cjy are y points of both center points of boxes and cix and cjx are x points of both center points of boxes. After getting values in terms of 0, π/2, - π/2 and 1 relation among the objects is determined. Determined relations are: "in-front left, back left, in-front right, back right, left, right, front, back and on".

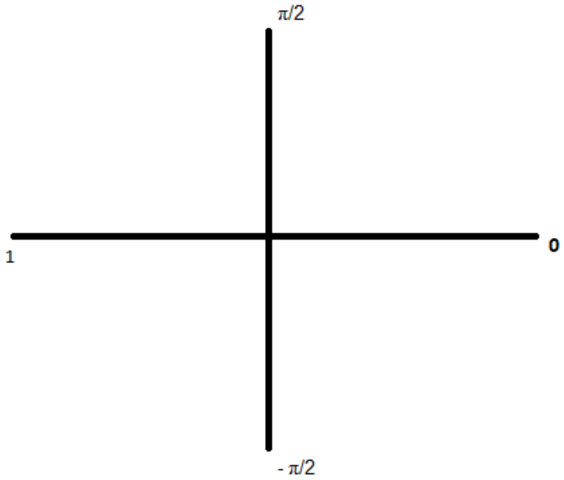


Figure 4:4:3: Axis system used to determine the relations.

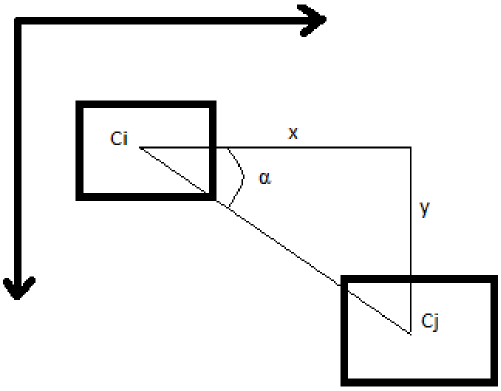


Figure 4:4:4: Demonstration of α calculated by the use of center points of the boxes.

To determine the relation of “on” a different approach is used to obtain the desired result. In this approach, the corner points of both boxes are compared and checked if they are positioned in certain limits if they fulfil that limit the spatial relation of “on” stands correctly.

```
Algorithm 1: Algorithm to calculate box center (center)


Input: box ∈ R4
Cx = $\frac{\text{box(0)+box(2)}}{2}$
Cy = $\frac{\text{box(1)+box(3)}}{2}$
```

```
Algorithm 2: Algorithm to calculate IOU for center (IOUc)
Input: boxs ∈ $R^4xR^n$
Init: i, j = 1, Cx ∈ $R^n$, Cy ∈ $R^n$, IouC ∈ $R^{nxn}$, m ∈ [0, +∞)
  while i ≤ n do
     box = boxes[i]
     Cx[i], Cy[i] = center(box)
  end while
  Cxn = MinMaxNorm(Cx, min = 0, max = m) → Scale the values

  to(min,max)
  Cyn = MinMaxNorm(Cy, min = 0, max = m)
  while i ≤ n do
     while j ≤ n do
        erx = $(Cxn[i] - Cxn[j])^2$
        ery = $(Cyn[i] - Cyn[j])^2$
        er = $\frac{m-\sqrt{erx+ery}}{m}$
        IouC[i, j] = max(0, er)
     end while
  end while
      Output: IouC
```

Algorithm 3: Algorithm to determine which boxes to kept based on Intersection Over Union

```
Input:
  $boxs \in R^4 x R^n$
  $\alpha, \beta \in (0, 1)$
Init: $i, j = 1$, $IouC, IouA, Iou \in R^{nxn}$, $IouM \in R^n$, $kept$: array of
  generic length
   while $i \leq n$ do
      while $j \leq n$ do
      $IouA[i, j] = J(boxs[i],boxs[j])$ → J(A,B): Jaccard similarity index between A and B
      end while
   end while
   IouC = IOU$_C$($boxs$)
   IoU = $\beta * IouA + (1 - \beta) * IouC$
   $IouM = Mean(IoU, axis = 0)$ → Average over axis 0 of the matrix
   while $i \leq n$ do
      if $IouM[i] < \alpha$ then kept.append(i)
```

Algorithm 4: Algorithm to determine 2D relation between two object/boxes given there centre

```
Input:
  $ci, cj \in R^2$
Init: $\varphi \in R$, relation = string
   $tan(\varphi) = \frac{cjy-ciy}{cjx-cix}$
   if $\varphi > 0$ and $\varphi < \pi/2$ then relation = ’front left’
   else if $\varphi < 0$ and $\varphi > -\pi/2$ then relation = ’back left’
   else if $\varphi > 0$ and $\varphi > \pi/2$ then relation = ’front right’
   else if $\varphi < 0$ and $\varphi < \pi/2$ then relation = ’front right’
   else if $\varphi = 0$ then relation = ’left’
   else if $\varphi = \pi/2$ then relation = ’front’
   else if $\varphi > \pi/2$ then relation = ’back’
   else if $\varphi > -\pi$ and $\varphi < \pi$ then relation = ’right’
   end if
        Output: relation
```

```
Algorithm 5: Algorithm to determine 3D relation between two object/boxes
Input:
   $bi, bj \in R^4$
   box1x, box1y = calculate$_c$($b_i$)
   box2x, box2y = calculate$_c$($b_j$)
   centerpoint = ($\sqrt{box1x - box2x}$ + (box1y - box2y)$_2$)
Init: relation = string
  if bi[0] < bj[0] and bi[1]< bj[1] and bi[2] > bj[2] and bi[3] > bj[3] then relation = ’on top’
  else if bi[0] < bj[0] and bi[1] < bj[1] and bi[2] > bj[2] and bi[3] < bj[3] and centerpoint < 1700 then relation = ’on top’
  else if bi[0] < bj[0] and bi[1] < bj[1] and bi[2] < bj[2] and bi[3] > bj[3] and centerpoint < 1700 then relation = ’on top’
  else if bi[0] < bj[0] and bi[1] > bj[1] and bi[2] < bj[2] and bi[3] < bj[3] and centerpoint < 1700 then relation = ’on top’
  else if bi[0] > bj[0] and bi[1] < bj[1] and bi[2] < bj[2] and bi[3] < bj[3] and centerpoint < 1700 then relation = ’on top’
  end if
       Output: relation
```

In the above-provided Pseudo-code of algorithms, algorithm 1 defines how the center point of boxes is calculated. Algorithm 2 defines how IOU for the center is calculated, algorithm 3 is providing information on how final boxes are kept and providing information on how NMS is working. Algorithm 4 is a Pseudo-code for 2-D relations and algorithm 5 is a Pseudo-code for 3-D relation of “on-top”. The bounding boxes kept from using algorithm 3 are used to determine relations from algorithm 4 and algorithm 5.

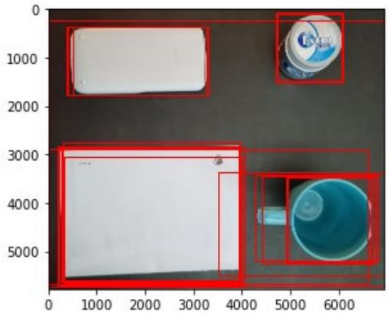
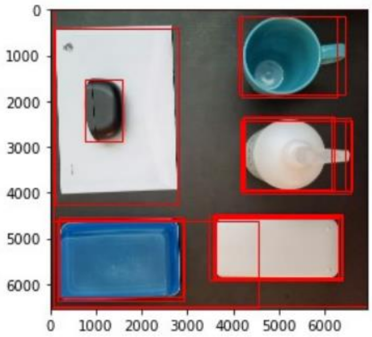
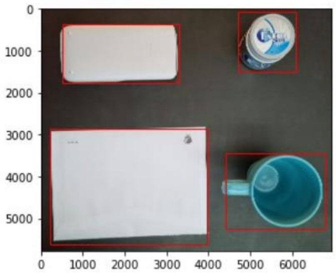
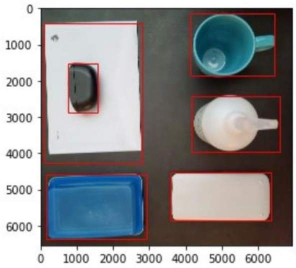
5 Results and Discussion

This section will represent the steps and results to represent the research questions to accomplish. The final result of every image is treated as one case and every case is verified at four different test points to reach the final result. After getting the final result of every case the final result of the algorithm is determined and calculated. Four test points for every case are as follows:

1. Bounding boxes from faster R-CNN.
2. IOU value determination from IOU predictor.
3. Final bounding boxes from NMS by using the correct IOU.
4. Final relations among the detected objects.

A few test cases representing all four test points are presented in the following table 1:

Table 1: Examples representing the test steps.

Picture Number	01	08
Faster RCNN		
IOU Predictor	The IOU prediction curve shown in Figure 5:1 is used to determine IOU for this picture number 01 IOU predicted is 0.17.	The IOU prediction curve shown in Figure 5:1 is used to determine IOU for this picture number 01 IOU predicted is 0.18.
NMS final		

Relations correctly obtained	All 2-D relations among the objects in the image are correctly obtained.	All 2-D and 3-D relations among the objects in the image are correctly obtained.

Picture 08 used in table-1 also has a 3-D relation between the letter envelope and black case. In this case, the black case is on the white envelope, this relation is successfully determined by the algorithm along with all other relations and achieved 100 percent accuracy in determining spatial relationships among all the objects present in the image. Following table 2 I will now determine how the algorithm worked and how many successful results are obtained in every case.

Table 2: Presenting objects present, detected, and accuracy of 2-D and 3-D relations.

Picture number	Objects in picture	IOU	NMS detected objects	2-D relations present	3-D relations present	2-D detected relations	3-D detected relations	2-D relation accuracy	3-D relation accuracy
01	4	0.17	4	12	0	12	0	92,28 %	75%
02	5	0.55	5	20	1	20	1		
03	5	0.55	5	20	1	20	1		
04	6	0.55	5	30	1	20	0		
05	4	0.18	4	12	0	12	0		
06	5	0.35	5	20	1	20	1		
07	5	0.18	5	20	0	20	0		
08	6	0.18	6	30	1	30	1		
09	5	0.18	5	20	0	20	0		
10	6	0.18	5	30	1	20	0		
11	8	0.18	7	56	0	42	0		
12	4	0.205	4	12	0	12	0		
13_1	6	0.1	6	30	2	30	2		
14	6	0.18	6	30	1	30	1		
15	6	0.1	5	30	1	20	0		
16	5	0.17	5	20	0	20	0		
17	5	0.17	5	20	0	20	0		
18	6	0.17	6	30	0	30	0		

19	6	0.17	6	30	0	30	0		
20	7	0.45	7	42	1	42	1		
21	8	0.55	8	56	1	56	1		

The accuracy of determining relations dropped due to object detection errors in NMS. Mainly caused in those cases where 3-D relations were present. It is noted that the IOU predictor didn't perform as well in cases that contain 3-D relations. One reason for this behaviour can be that both the IOU predictor and NMS were based on bounding boxes and the location of multiple boxes in the same area. In such cases where 3-D relations were present or where one object was on top of another object, thus, both IOU and NMS faced some troubles.

If we take the example of case number 4 the NMS should have detected 6 objects. It is observed that it detected 5 objects because it neglected the bounding boxes of the object which was on another object. At the same time, IOU predicted that for 6 objects IOU threshold should be around 0.19, but NMS didn't provide the correct numbers of bounding boxes. So, the IOU threshold increased to 0.55 where NMS was still not providing the bounding box for the 6th object. When the IOU value was set at 0.55, NMS started to provide bounding boxes which were representing more than one object or NMS was providing bounding boxes for those objects which were already detected. Thus, the step of NMS is stopped at an IOU of 0.55 and final relations were determined. The same sort of scenario is faced in other cases (case number: 4, 10, 15) (dataset and result provided in appendix) where NMS failed to detect the correct number of objects which leads to failure in getting all spatial relations. In case 11 faster R-CNN failed to recognize one object as that picture contains animated objects because of which was not completely in the picture and resembles the background. Thus, it got neglected by faster R-CNN and got missed by the algorithm.

Initially, the image is manually analysed, and the number of objects is determined. After that IOU curve is plotted using alpha and beta as shown in equation 4.3. Figure 5:1 shows the curves which are obtained by changing the values of alpha and beta. On the x-axis, the IOU value is plotted and on the y-axis number of boxes is plotted. As the α curve increases and approaches 1, all boxes get included, and as the α curve decreases and approaches 0 all boxes get deducted. Thus, if we take case 12 as an example, we can note that four objects are present so to keep four boxes we can match the values of the y and x-axis in the plot and get an estimated value of IOU. In this case, IOU for four objects' bounding boxes will be around 0.205.

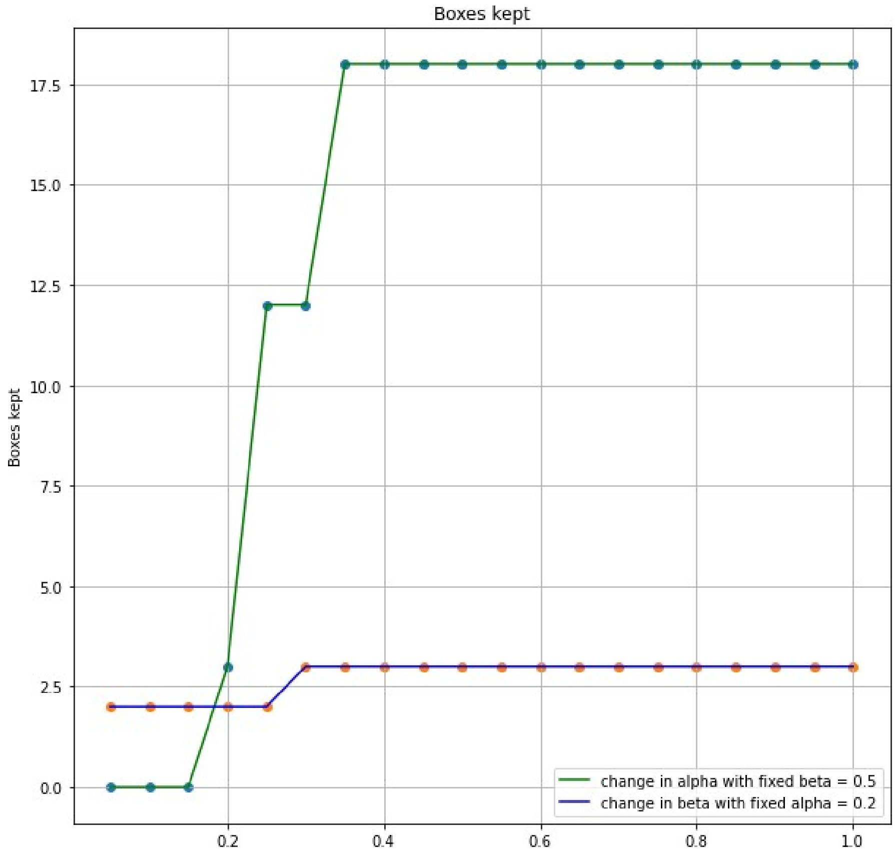


Figure 5:1: IOU prediction curve for picture/case 12, IOU can be from 0.205 to 0.22.

The Figure 5:1 green line representing the change in alpha with fixed beta is determining the correct value of IOU and IOU can vary from 0.205 to 0.22 to keep 4 bounding boxes from NMS.

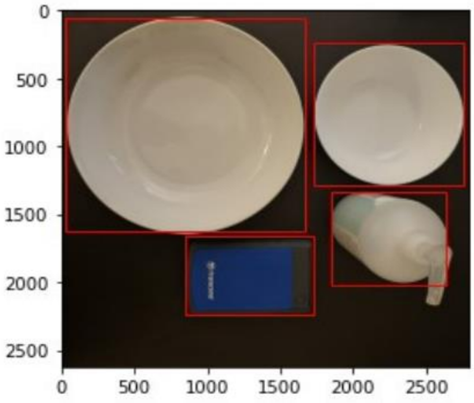
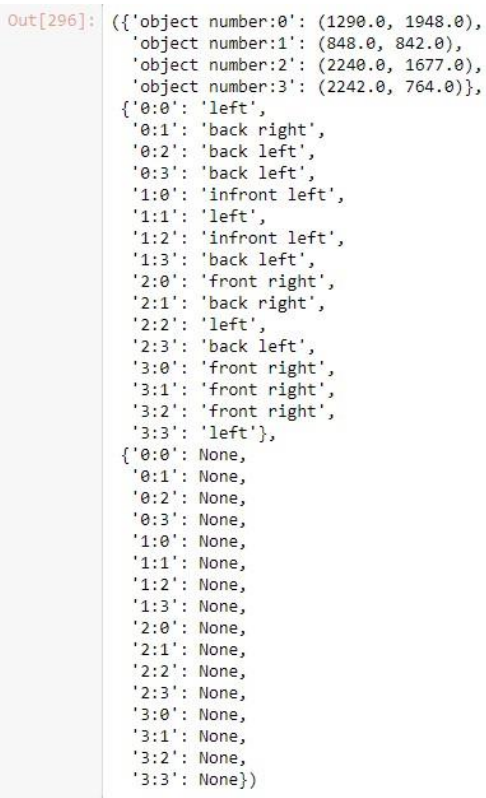


Figure 5:2: Final bounding boxes from NMS at picture/case 12.

The final spatial relations provided by the algorithm are shown in Figure 5:3, which are spatial relations for the same picture/case 12. To read the relation one must note that relations are saved in a dictionary and the second number in the dictionary key the 1 in '0:1' is representing the object from which the other object is viewed. One object is kept as a reference point and relations of other objects to that reference objects are determined. If the key of the dictionary is ‘0:1’ and the value is ‘back right’ it means object 1 is at the back right position of object 0. In beginning, the center points of the objects are printed to easily get information about each object.



```
Out[296]: ({'object number:0': (1290.0, 1948.0),
            'object number:1': (848.0, 842.0),
            'object number:2': (2240.0, 1677.0),
            'object number:3': (2242.0, 764.0)},
           {'0:0': 'left',
            '0:1': 'back right',
            '0:2': 'back left',
            '0:3': 'back left',
            '1:0': 'infront left',
            '1:1': 'left',
            '1:2': 'infront left',
            '1:3': 'back left',
            '2:0': 'front right',
            '2:1': 'back right',
            '2:2': 'left',
            '2:3': 'back left',
            '3:0': 'front right',
            '3:1': 'front right',
            '3:2': 'front right',
            '3:3': 'left'},
           {'0:0': None,
            '0:1': None,
            '0:2': None,
            '0:3': None,
            '1:0': None,
            '1:1': None,
            '1:2': None,
            '1:3': None,
            '2:0': None,
            '2:1': None,
            '2:2': None,
            '2:3': None,
            '3:0': None,
            '3:1': None,
            '3:2': None,
            '3:3': None})
```

Figure 5:3: Final relation of picture/case 12.

In Figure 5:3 at the bottom, ‘none’ entries show that no two objects have 3-D relations among themselves.

The accuracy of the method is lower in 3-D relations and the model was not able to achieve 100 percent in 2-D relations because of the boxes provided by NMS. Faster R-CNN worked perfectly in every case. In NMS it is noticed that in a few cases such bonding boxes also appeared which were representing more than one object. In such cases, the output from NMS was controlled by assuming a bit lower number of IOU. So, we can at least have only those bonding boxes which are for just one true object. In table 2 it can be noted that the number of detected objects by NMS dropped which became the main reason for not achieving true number or higher accuracy in both types of relationships. The model is still used because it can be used on any type of RDB image, is easy to understand and implement, and will provide an accuracy of at least 65% in final results. In this thesis, it is also noticed that this method can easily handle a higher number of objects i.e. 8, and provide 100 percent results with 9 different types of relations, while other methods such as CNN’s accuracy dropped if the number of relations increases [17].

5.1 Experiment:

In the experiment, the dataset of another state-of-the-art model presented by [17] is used and a few test cases from that dataset randomly selected are tested on the above-proposed model.

5.1.1 Differences in Datasets:

The dataset obtained from the research work done [17] has a lot of differences from the dataset I created to test my algorithm. The differences are mentioned below.

1. The images included in the dataset of [17] are taken from an eye-level angle or the camera is at 20 or 30 degrees higher than the eye-level angle. The dataset I created contains images taken from a high angle.
2. The images from the data set [17] include objects which have printed objects on designs that can get predicted as separate objects by the algorithm.
3. Objects in images [17] have almost similar colours or similar colour as the background of the image.
4. Many objects in the imported dataset from [17] have auxiliary objects attached like headphones or umbrellas which can get detected as more than one object.
5. The dataset used in the research work [17] contains above 5000 cases and the average number of objects in the cases is 4 or lower.

5.1.2 Baselines:

The baselines used in the model presented by [17] are ResNet 101 and VGG16. The baseline used in my model is Faster R-CNN. After the baseline [17] used neural network techniques of CNN and graph-based techniques to achieve the results. In case of this thesis on the other hand used the image analysis technique of non-maximum suppression to achieve the results. VGG16 is 16 network layer image classifier. Faster R-CNN is a region proposal algorithm and includes bounding boxes for the detection of desired elements in the provided information for example bounding boxes for objects in images.

5.1.3 Assumptions:

As the images in the dataset of [17] are not taken from a high angle, because of which my algorithm faced a few problems. After detecting the objects and getting the bounding boxes from the faster R-CNN the bounding box for the complete picture was not removable. Upon removing the bounding box of the complete picture few other boxes also got removed which disturbed the lengths for the score of boxes and the total number of boxes. Thus, to overcome this the bounding box is left among other bounding boxes, and the relations of that bounding box with other bounding boxes are removed by manually visualizing and analysing the final relations from the algorithm.

5.1.4 Results:

In the following Table, 3 and 4 results achieved by my algorithm and state-of-the-art model presented by [17] are provided. Images from the dataset of [17] are used to achieve these results. Table 4 also shows the accuracy of my model which I achieve using the dataset I created.

In table 3 mAP is the mean Average Precision. The model GVMRN is a graph-based visual manipulation relationship reasoning network, and GVMRN-RF is GVMRN with relation filtering and these two methods were proposed in [17]. Final accuracies are compared as provided in the following tables 3 and 4:

Table 3: Comparison of the presented model with state-of-the-art models (by mean average precision) dataset used is provided by [17].

Baseline	Models	mAP
ResNet101	Multi-task CNN [30]	-
	VMRN [18]	95.4
	GVMRN [17]	94.5
	GVMRN-RF [17]	94.6
VGG16	VMRN [18]	94.2
	GVMRN [17]	95.4
	GVMRN-RF	95.4
Faster-RCNN	NMS (ours)	82,9

Table 4: Comparison of the presented model with state-of-the-art models (by object numbers) dataset used is provided by [17].

Baseline	Models	Image accuracy				
		Total (%)	Object Number (Image)			
			2	3	4	5
ResNet101	Multi-task CNN	67.1	87.7	64.1	56.6	72.9
	VMRN	65.8	-	-	-	-

	GVMRN	68.0	90.0	68.8	60.3	56.2
	GVMRN-RF	68.8	91.4	69.2	61.2	57.5
VGG 16	VMRN	68.4	-	-	-	-
	GVMRN	69.7	91.4	69.9	62.9	58.9
	GVMRN-RF	70.2	92.9	70.3	63.8	60.3
Faster-RCNN	NMS(ours)	78,3	95,2	92,5	63,9	79,1

5.1.5 Reasoning:

The reason for the lower accuracy in the presented model is that the dataset used in the study consists of images that are not taken from the same place and angle. Thus, the presented model faced difficulties at the stage of NMS where extra bonding boxes were eliminated. In some cases, it worked perfectly but, in a few cases due to the high level of sensitivity of faster-RCNN many such objects were also detected which were printed on other objects. NMS considered them as bonding boxes of true objects and was not able to eliminate those bonding boxes while keeping the true object's bounding boxes. A sample of results is provided in appendix table 5. The dataset used mainly consists of a maximum of 4 objects. As pictures are not taken from one specific angle as right from the top of the table surface it creates another issue with faster-RCNN, while getting bonding boxes faster-RCNN provides one extra bonding box which is for the complete image. In the dataset created for the presented model, the bounding box for the complete image can get deleted very easily. In the dataset of [17] deleting the bounding box of the complete image also removes some other details because which box's number and scores' number don't match. NMS can't perform if the box's number and scores' number are not the same. The bounding box of the complete image was not deleted due to NMS, and the relations of the bonding boxes were ignored while visualizing the final relations.

5.2 Research question 1:

In the provided dataset with depth information, it is impossible to obtain desired point clouds, which were the main base to detect 3-D relations. This method is used on two different datasets but in both datasets, depth information was not sufficient to get desired point clouds and detection of 3-D relations was not possible. In future work, researchers can try to get a dataset with sufficient depth information. After getting a dataset with correct depth information points clouds can be achieved and this will help in getting more complex relations.

5.3 Research question 2:

In 2-D relations, four complex relations such as ‘in-front left’, ‘in-front right’, ‘back left, and ‘back right’ are detected using RGB images and bounding boxes. In the case of 3-D relation, one complex relation of ‘On’ is detected using RGB images and bounding boxes. In 2-D relations, the accuracy of complex relations remains 92.28% and in 3-D relations accuracy remains 75% as represented in table 2. In total 5 complex relations out of a total of 9 relations are detected in this master’s thesis.

5.4 Research question 3:

The depth information available in the current RGB-D dataset was not sufficient to use, due to which depth information available was not used to detect the relations. In future work getting hands-on with the datasets which provide complete depth information can help in achieving spatial relation goals with more accuracy, and a higher number of complex relations can be extracted.

6 Conclusions

In conclusion, this master's thesis has examined different approaches to recognize spatial relations among objects in RGB images. The images used in this process have different characteristics in terms of objects and no two images have the same objects with same orientation. At such a variable dataset it is noted that faster-RCNN performed very well and detected all the objects in almost every image. NMS and IOU predictor missed a few objects in cases where objects had 3-D relations and bounding boxes of both top and bottom objects were of almost the same area or areas of bounding boxes were close to each other. Eventually obtaining bounding boxes from faster-RCNN and reduced by using NMS provided good information which was helpful to obtain eight different types of 2-D relations and one type of 3-D relation with very good accuracy. The accuracy achieved in the case of 2-D relations is 92.28% and the accuracy in the case of 3-D relations is 75%.

This master's thesis also examined if it is possible to use depth information obtained by RGB-D images to determine the 3-D relations. In that scenario work on obtaining the point cloud is being done but due to a lack of information about depth information desired point clouds were not obtained and 3-D relations such as 'in' or 'under' couldn't be obtained.

Datasets were one of the main problems in obtaining the desired results in this master's thesis as work needed to be done on images containing real-life objects but at the moment no open-source dataset is available which has real-life objects in images and also has depth information. To overcome the problem of having real-life objects in images a small dataset is created and above explained algorithm is tested.

6.1 Future work:

In further work on similar projects, researchers can work on making the IOU predictor better for cases of 3-D relations, and work on getting better results from NMS so that algorithm can detect 3-D relations in all presented cases. The presented algorithm is generic and robust enough to be used on any type of image with various dimensions and shapes but adding robustness in NMS and working on depth information to obtain point clouds will make it more efficient and a higher number of 2-D and 3-D relations can be extracted from this algorithm. The accuracy of 3-D relations can be improved from 75% by working on point clouds or getting better results from NMS, by making it possible that NMS doesn't neglect those bounding boxes which are defining the relation of the object being present on another object.

7 Bibliography

[1] O. M. a. W. Burgard, “Composing Pick-and-Place Tasks By Grounding Language,” *Proceedings of the International Symposium on Experimental Robotics (ISER),* pp. 491-501, 2021.

[2] O. M. a. A. E. a. J. V. a. W. Burgard, “Learning Object Placements For Relational Instructions by Hallucinating Scene Representations,” *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA),* 2020.

[3] J. Prakash, “Non Maximum Suppression: Theory and Implementation in PyTorch,” 02 06 2021. [Online]. Available: https://learnopencv.com/non-maximum-suppression-theory-and-implementation-in-pytorch/.

[4] Labelbox, “Semantic segmentation made simple and fast,” 2022. [Online]. Available: https://learn.labelbox.com/semantic-segmentation/?utm_medium=paid-search&utm_source=google&utm_campaign=EMEA-Core-Annotate-C&utm_term=semantic%20segmentation&utm_device=c&_bt=517728825577&_bm=p&_bn=g&gclid=Cj0KCQjwnbmaBhD-ARIsAGTPcfUAzO7sVKQnLeSf0DMFL_c7_.

[5] Michael, “Instance vs. Semantic Segmentation: What Are the Key Differences?,” 8 May 2021. [Online]. Available: https://keymakr.com/blog/instance-vs-semantic-segmentation/.

[6] J. P. C. T. J. C. M. D. Haifeng Li, “What do We Learn by Semantic Scene Understanding for Remote Sensing imagery in CNN framework?,” *arXiv,* 19 May 2017.

[7] P. M. E. P. Leila de Floriani, “Chapter 7 - Applications of Computational Geometry to Geographic Information Systems,” in *Handbook of Computational Geometry*, 2000, pp. 333-388.

[8] Studiobinder, “High Angle Shot — Camera Angle Explained & Iconic Examples,” 12 September 2021. [Online]. Available: https://www.studiobinder.com/blog/high-angle-shot-camera-movement-angle/#:~:text=A%20high%20angle%20shot%20is,to%20directly%20above%20the%20subject..

[9] R. a. T. B. P. a. D. C. a. G. M. J. a. L. S. L. a. W. D. a. V. H. A. Bayareh Mancilla, “Anatomical 3D Modeling Using IR Sensors and Radiometric Processing Based on Structure from Motion: Towards a Tool for the Diabetic Foot Diagnosis,” *Sensors,* 2021.

[10] W. Y. a. C. P. a. K. D. a. D. Fox, “SORN}et: Spatial Object-Centric Representations for Sequential Manipulation,” *5th Annual Conference on Robot Learning,* 2021.

[11] R. a. L. D. a. A. T. Kartmann, "Semantic Scene Manipulation Based on 3D Spatial Object Relations and Language Instructions," *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids),* pp. 306-313, 2021.

[12] C. Y. a. Y. Z. a. L. U. a. N. B. a. Y. D. a. H. D. a. D. S. a. A. M. a. E. Keogh, "Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets," *IEEE Computer Society,* pp. 1317-1322, 2016.

[13] M. G. M. N. J. S. E. S. C. A. A. H. M. J. G. R. &. A. A. S. R. Khan, " Global Incidence and Mortality Patterns of Pedestrian Road Traffic Injuries by Sociodemographic Index, with Forecasting: Findings from the Global Burden of Diseases, Injuries, and Risk Factors 2017 Study," *International journal of environmental research and public health,* 2020.

[14] S. K. P. a. A. K. a. A. K. a. D. R. a. A. Loutfi, "A Novel Method for Estimating Distances from a Robot to Humans Using Egocentric RGB Camera," *Sensors,* pp. 1-13, 2019.

[15] P. G. F. Y. R. C. I. B. Mateus Riva, "Approximation of dilation-based spatial relations to add structural constraints in neural networks," 2021.

[16] Mathworks, "Image Types," [Online]. Available: https://www.mathworks.com/help/matlab/creating_plots/image-types.html.

[17] G. a. T. J. a. L. H. a. C. W. a. L. J. Zuo, "Graph-based Visual Manipulation Relationship Reasoning in Object-Stacking Scenes," *2021 International Joint Conference on Neural Networks (IJCNN),* pp. 1-8, 2021.

[18] H. Z. a. X. L. a. X. Z. a. Z. T. a. Y. Z. a. N. Zheng, "Visual manipulation relationship recognition in object-stacking scenes," *Pattern Recognition Letters,* pp. 34-42, 2020.

[19] L. a. Z. M. a. R. H. a. X. L. Zhao, "Channel Exchanging for RGB-T Tracking," *Sensors,* 2021.

[20] F. a. W. D. a. H. H. Yan, "Robotic Understanding of Spatial Relationships Using Neural-Logic Learning," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* pp. 8358-8365, 2020.

[21] H. a. G. E. a. C. L. a. D. J. Wurdemann, "Slam using 3D reconstruction VIA a visual RGB & RGB-D sensory input," *Proceedings of the ASME Design Engineering Technical Conference,* pp. 615-622, 2011.

[22] Wikipedia, "Spatial relation," [Online]. Available: https://en.wikipedia.org/wiki/Spatial_relation#:~:text=A%20spatial%20relation%20specifies%20how,often%20represented%20by%20a%20point..

[23] Wikipedia, "Object detection," [Online]. Available: https://en.wikipedia.org/wiki/Object_detection.

[24] C. P. T. H. D. F. Weiyu Liu, “StructFormer: Learning Spatial Structure for Language-Guided Semantic Rearrangement of Novel Objects,” 2021.

[25] I. a. A. M. a. A. D. a. O. B. Rocco, “RGB-D and Thermal Sensor Fusion,” 2016.

[26] J. Y. a. J. L. a. S. L. a. D. B. a. D. Parikh, “Graph R-CNN for Scene Graph Generation,” 2018.

[27] O. K. Mohit Sharma, “Relational Learning for Skill Preconditions,” 2020.

[28] T. S. C. a. L. S. S. a. M. P. C. a. N. C. McDonald, “Automated Vehicles and Pedestrian Safety: Exploring the Promise and Limits of Pedestrian Detection,” *American Journal of Preventive Medicine,* pp. 1-7, 2019.

[29] H. a. L. X. a. Z. X. a. T. Z. a. Z. Y. a. Z. N. Zhang, “Visual manipulation relationship recognition in object-stacking scenes,” *Pattern Recognition Letters,* pp. 32-42, 12 2020.

[30] H. a. L. X. a. B. S. a. W. L. a. Y. C. a. Z. N. Zhang, “A Multi-task Convolutional Neural Network for Autonomous Robotic Grasping in Object Stacking Scenes,” pp. 6435-6442, 11 2019.

[31] P. systems, “Semantic Scene Understanding,” 2021. [Online]. Available: https://ps.is.mpg.de/research_fields/semantic-scene-understanding#:~:text=Photo%3A%20Wolfram%20Scheible.,and%20semantic%20relationships%20between%20objects..

[32] K. H. R. G. J. S. Shaoqing Ren, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” in *Advances in Neural Information Processing Systems NIPS*, 2015.

[33] W. Liu, “Data set,” 2022. [Online]. Available: https://github.com/wliu88/StructFormer/blob/main/README.md#quick-start-with-pretrained-models.

[34] A. F. Gad, “Faster R-CNN Explained for Object Detection Tasks,” 2020. [Online]. Available: https://blog.paperspace.com/faster-r-cnn-explained-object-detection/. [Accessed 2022].

[35] Mathworks, “What is object detection.,” [Online]. Available: https://www.mathworks.com/discovery/object-detection.html#:~:text=Object%20detection%20is%20a%20computer,learning%20to%20produce%20meaningful%20results.. [Accessed 2022].

[36] Intel, “Docs RS Online,” June 2020. [Online]. Available: https://docs.rs-online.com/2f4f/A700000006942957.pdf. [Accessed 2022].

[37] W. Afzal, “Data set,” 2022. [Online]. Available: https://drive.google.com/file/d/1iw_I_WI6u9YWI53RPtScp0rva5EMEU80/view?usp=sharing. [Accessed 20 December 2022].

[38] C. P. W. Y. B. S. Y.-W. C. M. C. a. D. F. Andreea Bobu, “Learning Perceptual Concepts by Bootstrapping from Human Queries,” *IEEE Robotics and Automation Letters,* 2021.

[39] A. Bobu, “Data set,” 2022. [Online]. Available: https://drive.google.com/file/d/1h-jcEI-SArBFR4FO4uL09jUYle6zRPjQ/edit. [Accessed 20 December 2022].

[40] P. J. a. F. C. a. B. R. a. A. D. Navarro, “A Machine Learning Approach to Pedestrian Detection for Autonomous Vehicles Using High-Definition 3D Range Data,” *Sensors,* vol. 17, no. 1, 2017.

Appendix

The main cases from the dataset which were not working perfectly are provided below:

Case 4:

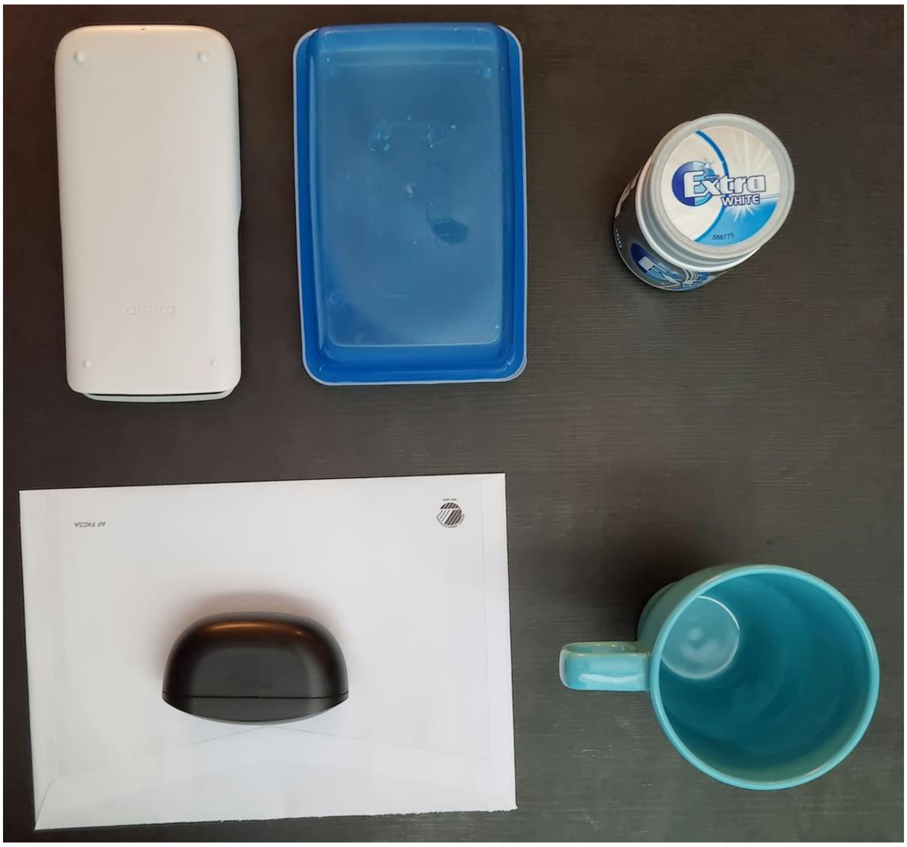


Figure 7:1: Image number 04 used in case number 04.

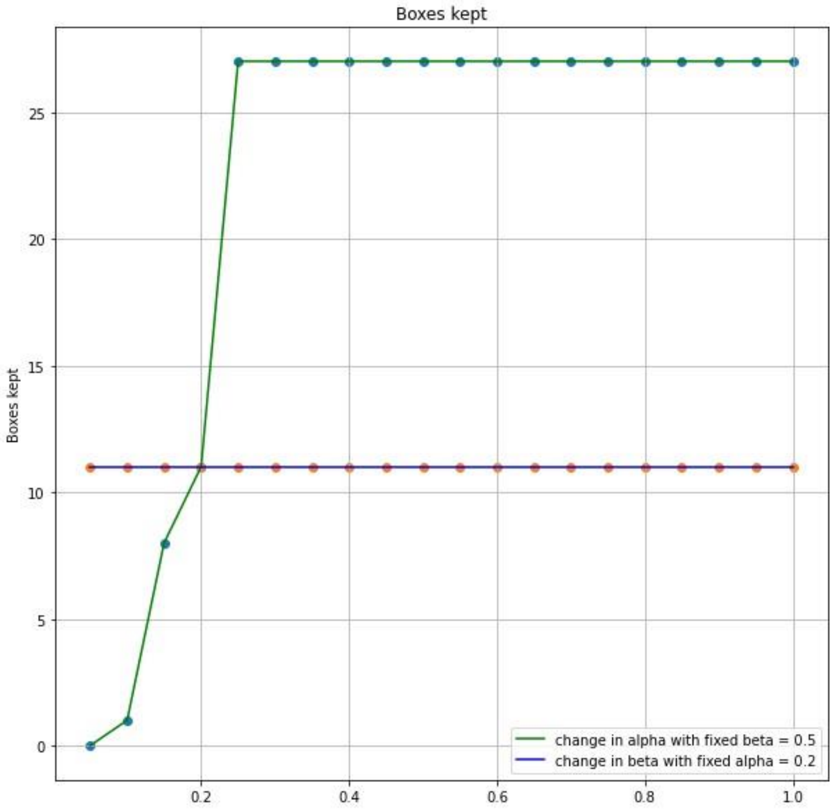


Figure 7:2: IOU predictor, predicting IOU to be of around 0.17 for 5 objects but IOU needed was 0.55.

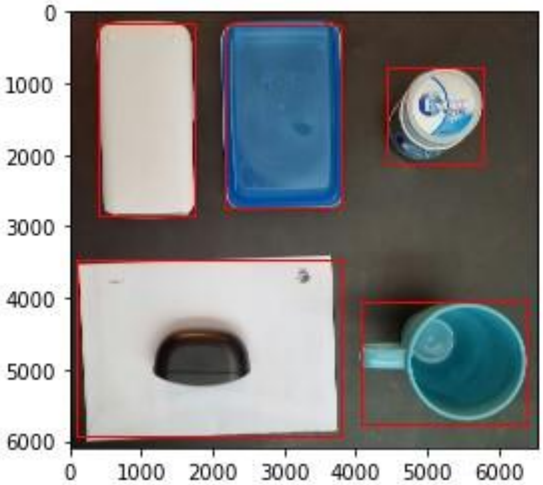
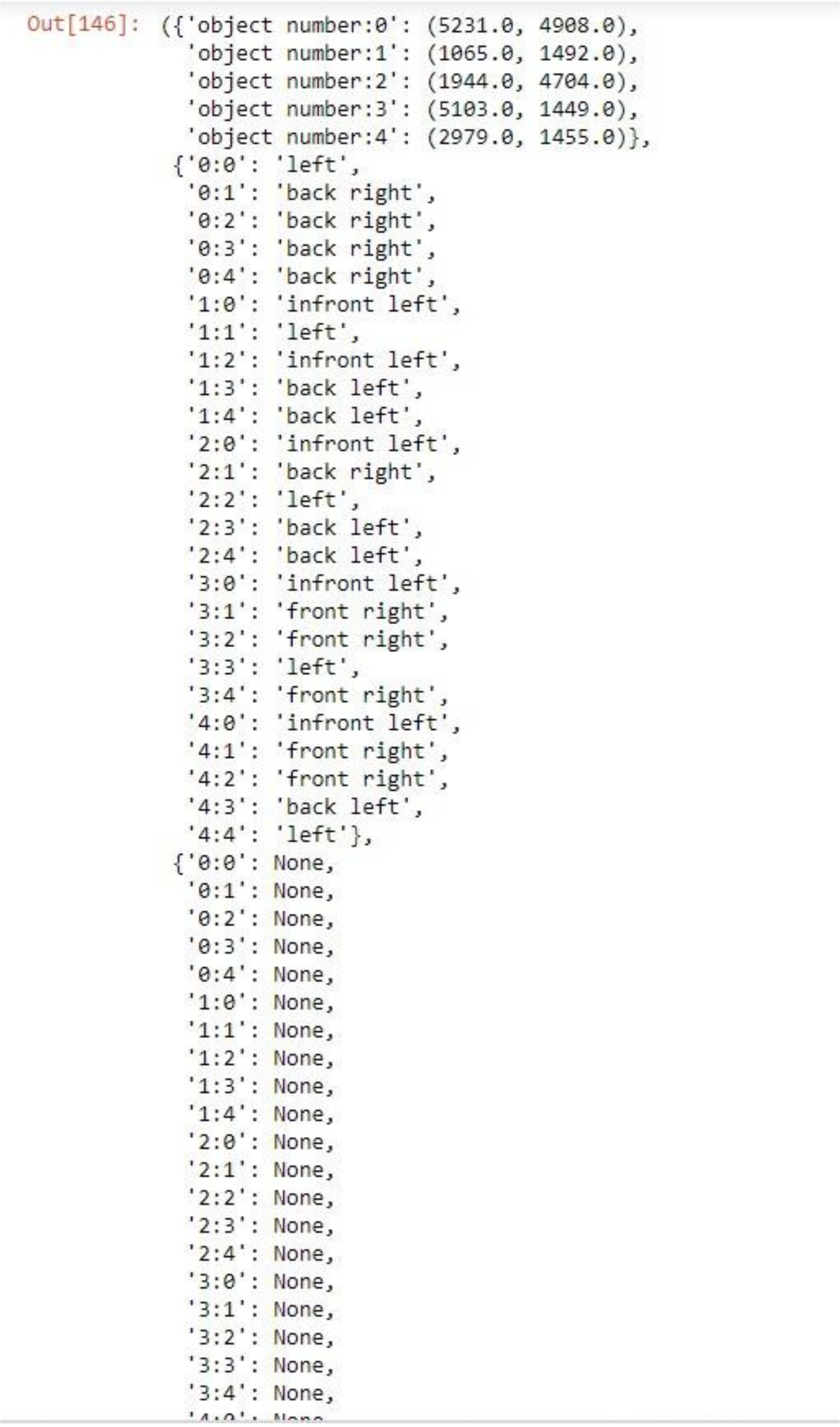


Figure 7:3: Final bounding boxes from NMS, missed the one for black box.



```
Out[146]: ({'object number:0': (5231.0, 4908.0),
            'object number:1': (1065.0, 1492.0),
            'object number:2': (1944.0, 4704.0),
            'object number:3': (5103.0, 1449.0),
            'object number:4': (2979.0, 1455.0)},
           {'0:0': 'left',
            '0:1': 'back right',
            '0:2': 'back right',
            '0:3': 'back right',
            '0:4': 'back right',
            '1:0': 'infront left',
            '1:1': 'left',
            '1:2': 'infront left',
            '1:3': 'back left',
            '1:4': 'back left',
            '2:0': 'infront left',
            '2:1': 'back right',
            '2:2': 'left',
            '2:3': 'back left',
            '2:4': 'back left',
            '3:0': 'infront left',
            '3:1': 'front right',
            '3:2': 'front right',
            '3:3': 'left',
            '3:4': 'front right',
            '4:0': 'infront left',
            '4:1': 'front right',
            '4:2': 'front right',
            '4:3': 'back left',
            '4:4': 'left'},
           {'0:0': None,
            '0:1': None,
            '0:2': None,
            '0:3': None,
            '0:4': None,
            '1:0': None,
            '1:1': None,
            '1:2': None,
            '1:3': None,
            '1:4': None,
            '2:0': None,
            '2:1': None,
            '2:2': None,
            '2:3': None,
            '2:4': None,
            '3:0': None,
            '3:1': None,
            '3:2': None,
            '3:3': None,
            '3:4': None,
            '4:0': None
```

Figure 7:4: Relations obtained from case number 04.

Case 10:

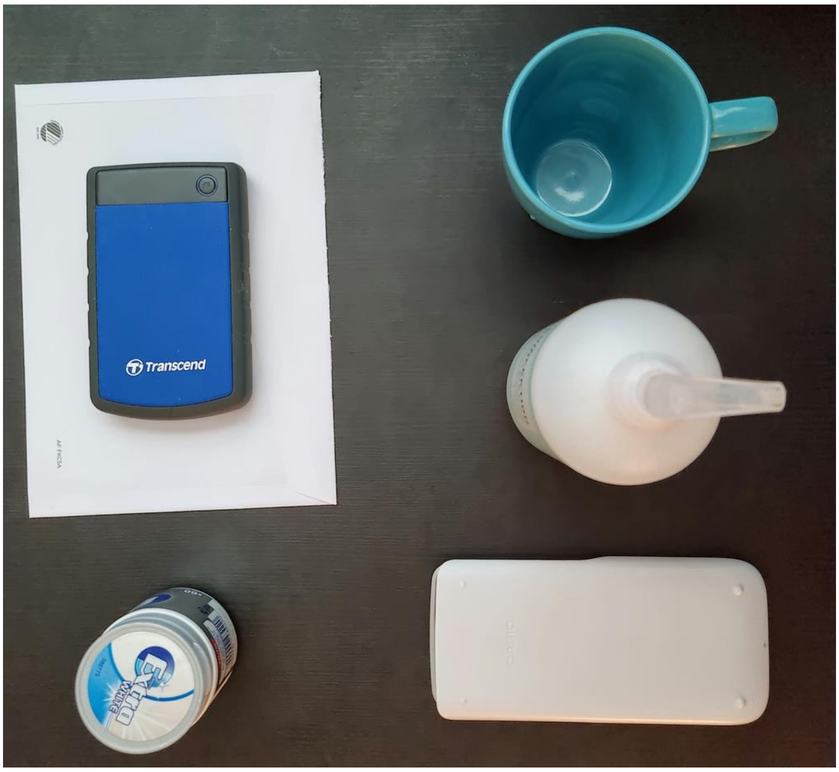


Figure 7:5: Image number 10 used in case number 10.

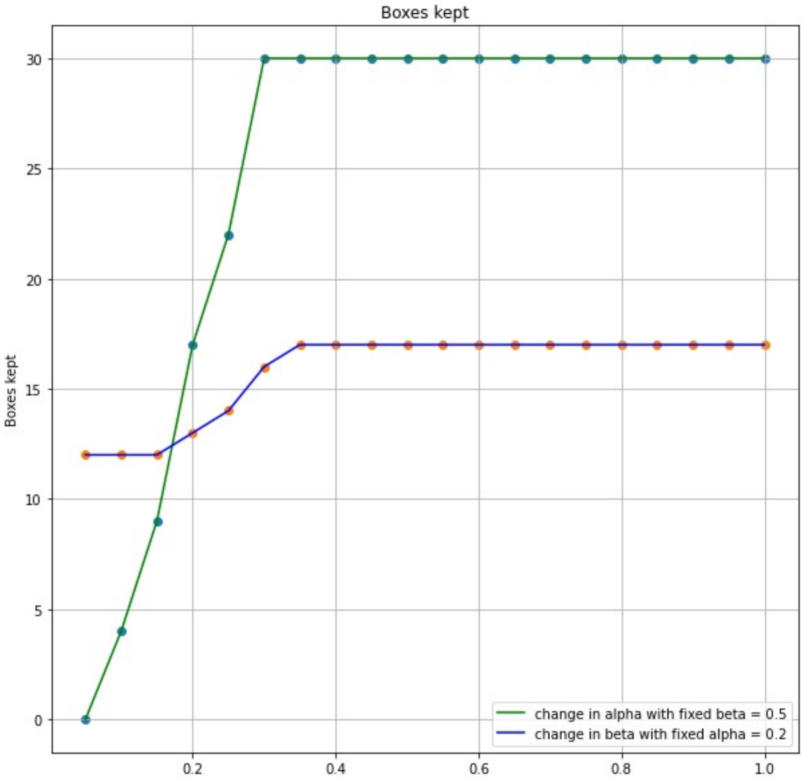


Figure 7:6: IOU predictor, predicting IOU to be at 0.18. Used IOU was also 0.18.

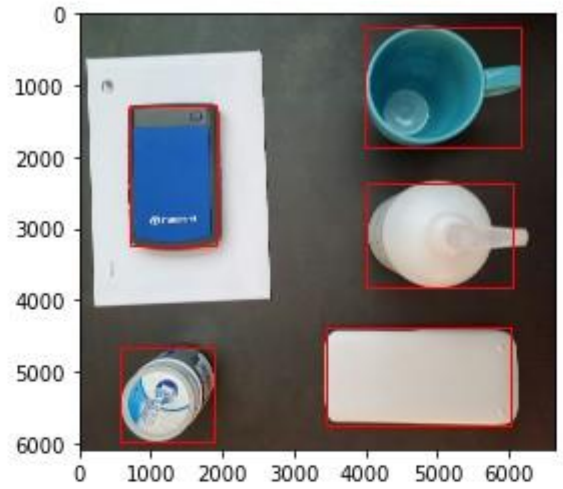
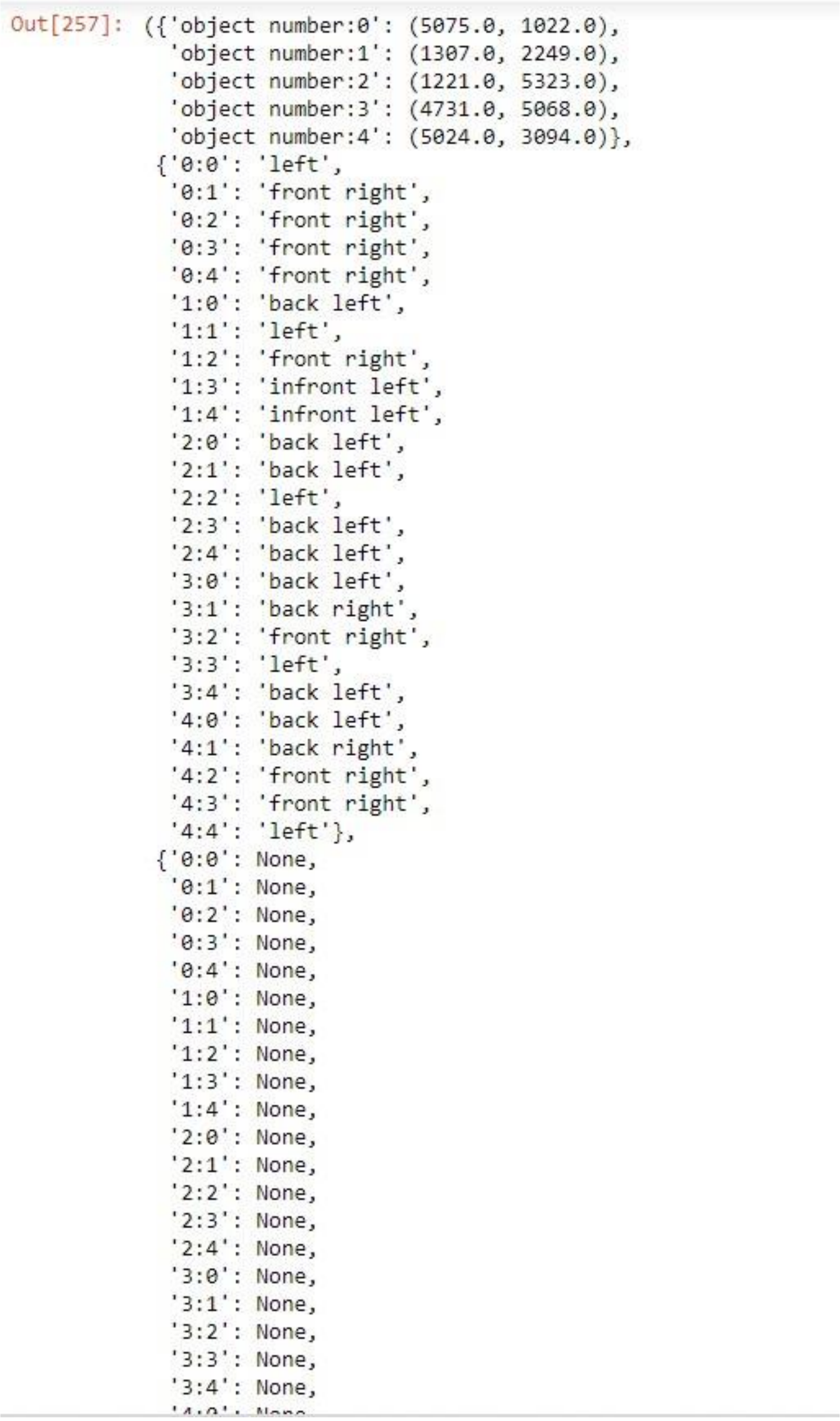


Figure 7:7: Final NMS, missed the bounding box for white envelope at IOU of 0.18.



```
Out[257]: ({'object number:0': (5075.0, 1022.0),
             'object number:1': (1307.0, 2249.0),
             'object number:2': (1221.0, 5323.0),
             'object number:3': (4731.0, 5068.0),
             'object number:4': (5024.0, 3094.0)},
            {'0:0': 'left',
             '0:1': 'front right',
             '0:2': 'front right',
             '0:3': 'front right',
             '0:4': 'front right',
             '1:0': 'back left',
             '1:1': 'left',
             '1:2': 'front right',
             '1:3': 'infront left',
             '1:4': 'infront left',
             '2:0': 'back left',
             '2:1': 'back left',
             '2:2': 'left',
             '2:3': 'back left',
             '2:4': 'back left',
             '3:0': 'back left',
             '3:1': 'back right',
             '3:2': 'front right',
             '3:3': 'left',
             '3:4': 'back left',
             '4:0': 'back left',
             '4:1': 'back right',
             '4:2': 'front right',
             '4:3': 'front right',
             '4:4': 'left'},
            {'0:0': None,
             '0:1': None,
             '0:2': None,
             '0:3': None,
             '0:4': None,
             '1:0': None,
             '1:1': None,
             '1:2': None,
             '1:3': None,
             '1:4': None,
             '2:0': None,
             '2:1': None,
             '2:2': None,
             '2:3': None,
             '2:4': None,
             '3:0': None,
             '3:1': None,
             '3:2': None,
             '3:3': None,
             '3:4': None,
             '4:0': None
```

Figure 7:8: Relations obtained from case number 10.

Case 15:

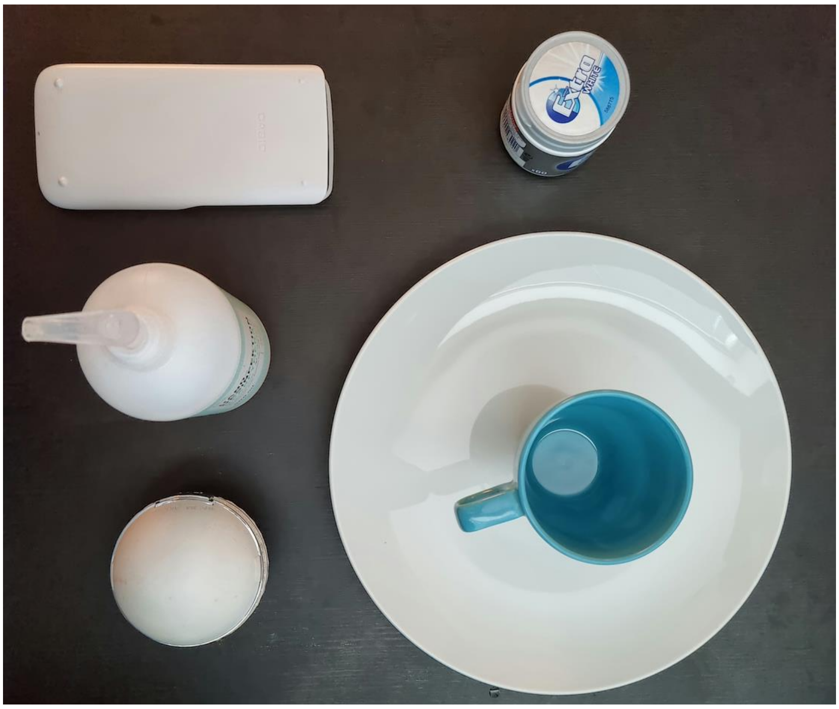


Figure 7:9: Image number 15 used in case number 15.

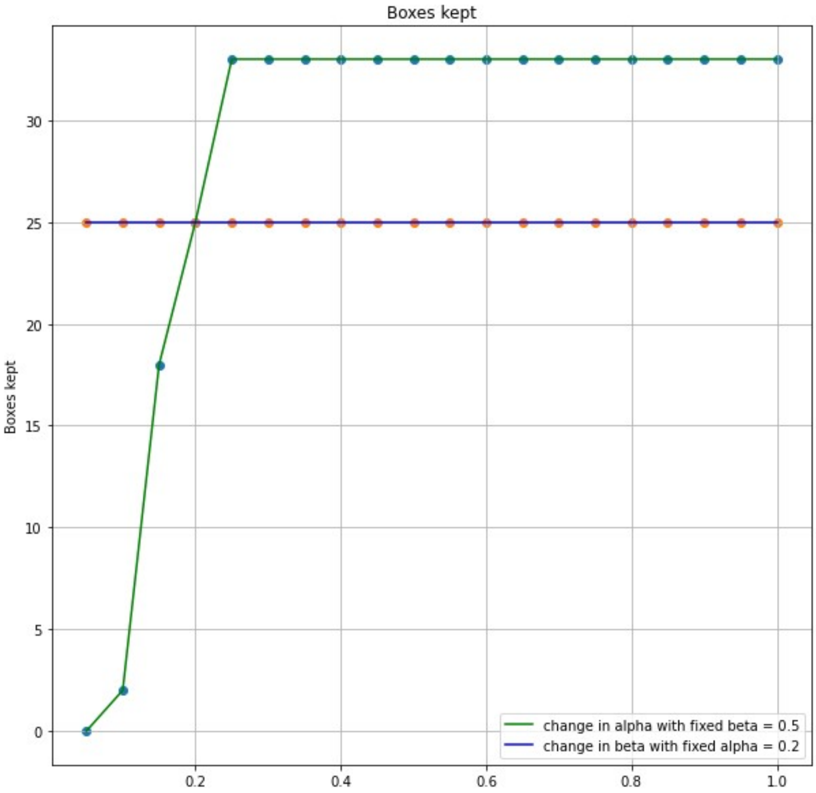


Figure 7:10: IOU predictor, predicting IOU to be of around 0.15 for 6 objects but IOU needed was 0.1.

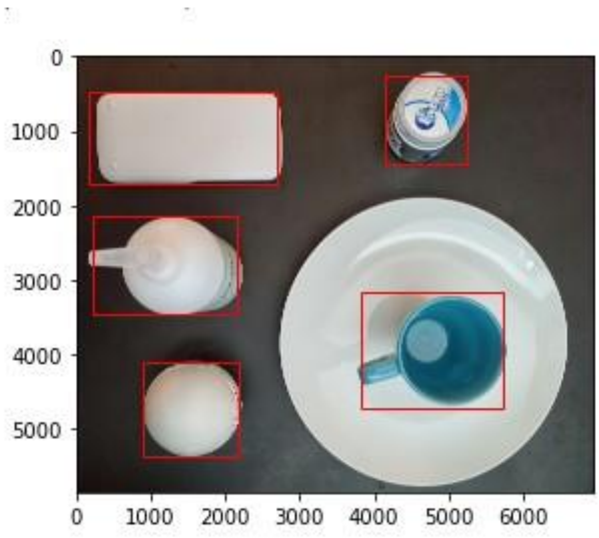
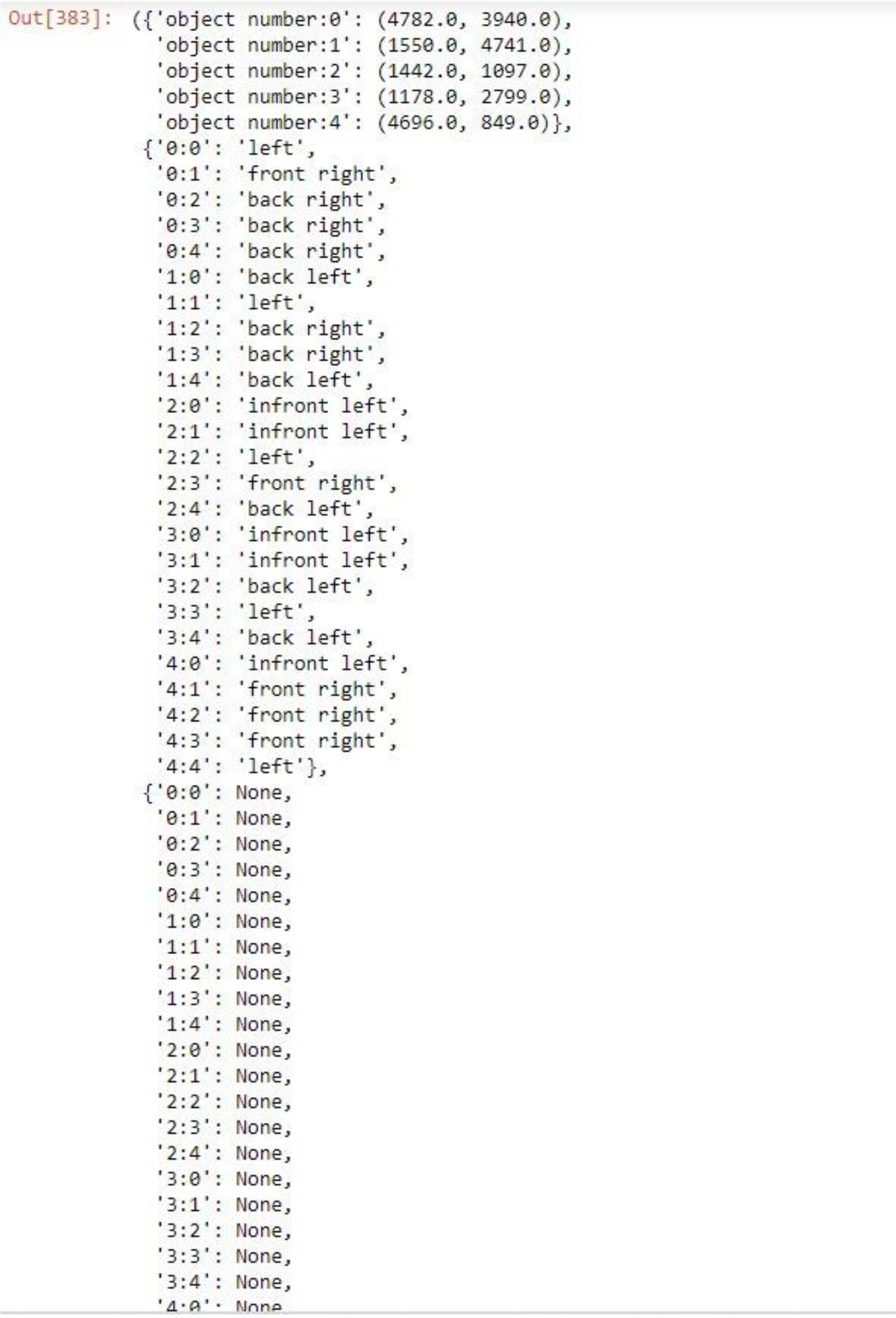


Figure 7:11: Final NMS, missed the bounding box for white plate at IOU of 0.1.



```
Out[383]: ({'object number:0': (4782.0, 3940.0),
            'object number:1': (1550.0, 4741.0),
            'object number:2': (1442.0, 1097.0),
            'object number:3': (1178.0, 2799.0),
            'object number:4': (4696.0, 849.0)},
           {'0:0': 'left',
            '0:1': 'front right',
            '0:2': 'back right',
            '0:3': 'back right',
            '0:4': 'back right',
            '1:0': 'back left',
            '1:1': 'left',
            '1:2': 'back right',
            '1:3': 'back right',
            '1:4': 'back left',
            '2:0': 'infront left',
            '2:1': 'infront left',
            '2:2': 'left',
            '2:3': 'front right',
            '2:4': 'back left',
            '3:0': 'infront left',
            '3:1': 'infront left',
            '3:2': 'back left',
            '3:3': 'left',
            '3:4': 'back left',
            '4:0': 'infront left',
            '4:1': 'front right',
            '4:2': 'front right',
            '4:3': 'front right',
            '4:4': 'left'},
           {'0:0': None,
            '0:1': None,
            '0:2': None,
            '0:3': None,
            '0:4': None,
            '1:0': None,
            '1:1': None,
            '1:2': None,
            '1:3': None,
            '1:4': None,
            '2:0': None,
            '2:1': None,
            '2:2': None,
            '2:3': None,
            '2:4': None,
            '3:0': None,
            '3:1': None,
            '3:2': None,
            '3:3': None,
            '3:4': None,
            '4:0': None
```

Figure 7:12: Relations obtained from case number 15.

In case 10 and 15 it is observed that either IOU value predicted by IOU predictor is used or IOU value less then IOU predictor value is used, that is because in case of increasing the value from the one which is used result in additional bounding boxes which were either for already detected objects or covered more then one objects.

Experiment:

Table 5: Result of cases from dataset of (Zuo, G. a. 2021).

Picture number	Objects in picture	NMS detected objects	2-D relations present	3-D relations present	2-D detected relations	3-D detected relations	2-D relation accuracy	3-D relation accuracy	AP	mAP
00224	2	3	2	1	2	2	82,8%	64,28%	0,75	82,9 %
00264	4	4	12	3	12	3			1	
00945	2	2	2	1	2	1			1	
00946	2	2	2	1	2	1			1	
00949	4	3	12	5	6	3			0,53	
01003	2	2	2	1	2	0			0,67	
01029	5	5	20	4	20	1			0,875	
01222	5	4	20	4	12	1			0,541	
01226	5	5	20	4	20	3			0,95	
02600	3	4	6	2	6	0			0,75	
02618	2	2	2	1	2	1			1	
02620	2	2	2	1	2	1			1	
02622	3	3	6	3	6	2			0,66	
02633	3	3	6	2	6	2			1	
02634	4	3	12	3	6	2			0,53	
02663	2	2	2	1	2	1			1	
02680	4	4	12	2	6	1			0,5	
02683	3	3	6	2	6	2			1	
02684	3	3	6	1	6	1			1	

Table 6: mAP performance comparison of presented model on both datasets.

Model	Dataset	2-D relations	3-D relations	mAP

Faster-RCNN with NMS	On dataset used by (Zuo, G. a. 2021)	82.89%	64.28%	82.9%
	My proposed dataset	92.28%	75%	93.67%

HALMSTAD
UNIVERSITY

PO Box 823, SE-301 18 Halmstad
Phone: +35 46 16 71 00
E-mail: registrator@hh.se
www.hh.se