Modelling of a Bidirectional Network

Maham Haroon

00000117578

MS-RIME'15

Supervisor

Dr. Yasar Ayaz

Co-Supervisor

Dr. Muhamamd Naveed

DEPARTMENT OF ROBOTICS AND ARTIFICIAL INTELLIGENCE

SCHOOL OF MECHANICAL AND MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

July, 2017

Modelling of a bidirectional Neural Network

Maham Haroon

00000117578

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Robotics and Intelligent Machine Engineering Degree

Thesis Supervisor:

Dr. Yasar Ayaz

Thesis Supervisor's Signature:_____

DEPARTMENT OF ROBOTICS AND ARTIFICIAL INTELLIGENCE

SCHOOL OF MECHANICAL AND MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

JULY, 2017

Declaration

I certify that this research work titled "*Modelling of a Bi-directional Neural Network"* is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

Maham Haroon

MS RIME'15

Language Correctness Certificate

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.

Signature of Student

Maham Haroon

00000117578

MS RIME'15

Signature of Supervisor

Copyright Statement

Acknowledgements

I'm thankful to Allah Almighty for giving me the strength and courage to see this project through. I also had immense support from a lot of people during the process. I'd like to specially thank my supervisor, Dr. Yasar Ayaz for his support, guidance and insight throughout my thesis, which enabled me to complete this project in a timely manner. I'd also like to thank my co-supervisor, Dr. Muhammad Naveed for being extremely helpful during this entire process. Apart from my supervisors, I'm very thankful to Dr. Hasan Sajid, whose insightful comments and vast knowledge of the field aided me a lot in my project. I'm grateful to my lab mates for their technical and moral support. Last but not the least I'd like to thank my friends and family for always being there for me.

Dedicated to the most frustrating problems that help us explore and expand the limits of our minds.

Abstract

Visual place and object categorization has been an important aspect of research for a number of years now. One reason for its popularity is the wide number of applications in Mobile Robotics and HRI. Some of these include, behavior based navigation, mapping, task based planning, semantic SLAM and active object search. Recently, there has been a trend in place categorization based on the objects associated with that place. The reason that object based place classification proves useful is that a standard camera image represents partially observable environment at any one point in time and where simple place based classification will fail in such an environment, object based place classification is more successful.

Inspired from the successful results of these algorithms, this research formulates a new approach where the correlation between two input types aids the better classification or categorization of both the inputs.

The purpose of this research is to contribute to simultaneous place and object classification. This research augments the ongoing research in three areas; i) A novel method is introduced for bidirectional classification ii) A randomized object detector and localizer is introduced iii) The approach shows superior performance to separate predictions of object and place classification.

Key Words: *Place Categorization, Object Categorization, Machine Learning, Neural Networks*

Table of Contents

List of Figures

List of Tables

Chapter 1 **INTRODUCTION**

Visual place and object categorization has been an important aspect of research for a number of years now. A reason for its popularity is wide number of its applications in Mobile Robotics and HRI, such as; behavior based navigation, mapping, task based planning, semantic SLAM, active object search and rescue. Recently, there has been a trend in place categorization based on the objects associated with that place [1], [2], [3], [4]. One reason that object based place classification proves useful is that a standard camera image represents partially observable environment at any one point in time and where simple place based classification will fail in such an environment, object based place classification proves more useful.

Learning based approaches enable robots to acquire conceptual information of the environment and help them to represent the information in human-centric terms. What about a problem where there are two independently learnt models which are codependent and even a correlation is found but that cannot be easily learned. For instance in a task of simultaneous object and place classification either place classification on the basis of objects or the other way round is done [6], [7] , [8], [9]. What if we want to use this correlation to achieve better classification? A mechanism at the end of classifiers that improves the overall accuracy seems independent and generalized.

In Robotics, task based robots have to perform in a certain environment and therefore it is more crucial for them to have the quality knowledge of the place rather than a poor quality knowledge of a lot of quantity. For this reason, the model uses existing models and applies them to new environments to show superior performance.

1.1 Background

Bidirectional Associative Memory (BAM) is a two way network that was first presented by Bart Kosko in 1980 [10] which served the purpose of regenerating lost information. BAM is a form of recurrent Neural Networks [11]. It's similar to a Hopfield Network [12] but where Hopfield Networks are a form of Auto-associative memory [13] i.e. the patterns on the output and input must be the same size, the BAM on the other hand is hetero associative [14] and can learn an association between binary patterns of different sizes as long as the conditions are met. Many

modifications of the network exist to date. This model uses a variation of BAM to learn an association between certain places and objects.

1.2 Motivation

A typical neural network takes input and some labels and learns to relate that input to the said labels. If there are multiple types of information a straight forward approach is to use multiple such networks together to separately classify this information. When humans take in the same information, they are also using the correlation between this information for successful comprehension of this information. Two separate neural networks will not do that unless they have been trained to do so.

Motivation for this model is two-fold. Firstly, as shown in Figure 1, the performance of a Neural Network depends on how much data is available and how large the network is. For a network to be ideal we therefore will either require infinite data to train it on, the network will have to be very large in size or possibly both. The progress in the past decade has provided us with both; a large amount of data to train on and the presence of GPU's enable making of large CNNs. A large number of CNN models for various tasks have been therefore, generated in the past few decades [19], [20], [26]. Instead of modelling a new CNN, this approach uses a combination of various existing CNNs to respectively improve their performance based on the results already generated.

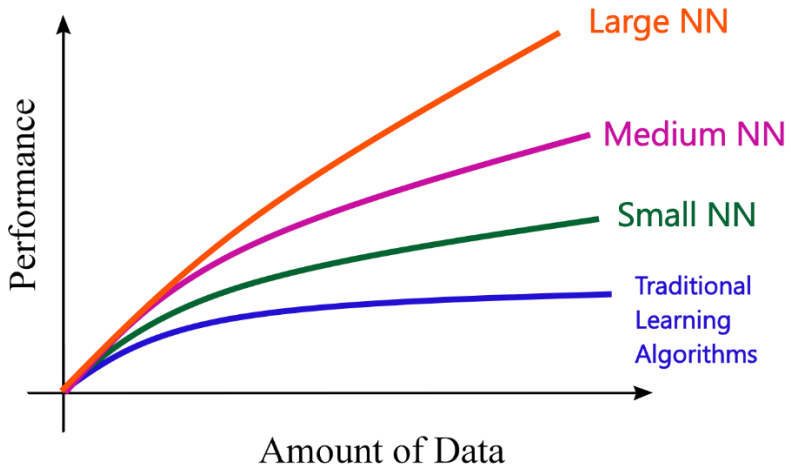

Figure 1 Correlation between amount of Training Data and Performance of Neural Networks

The Second motivation for this approach was human brain itself. Human brain uses data from five senses i.e. touch, taste, smell, see and hear, then it uses correlation of this data to produce multiple outputs. For instance hearing the word house causes the brain to be reminded of a certain structure and similarly looking at a house causes the brain to remember the word associated with the picture. The neural network in the brain are multidirectional, which inspired the bidirectional neural network model presented in this thesis.

Section II enlists a short summary of the related research done in this domain so far. Section III describes the proposed approach and implementation steps in detail. Section IV presents results and conclusions.

Chapter 2 **LITERATURE REVIEW**

The section is divided into further categories such as; relational learning, sensors, learning approaches, dataset and finally observed deficiencies.

2.1 Relational Learning

In machine learning, relational machine learning (RML) is very common subfield these days. The major contributions started after 1990s. There are forms like Collective Learning i.e. various objects are simultaneously learned based on the probability of their presence with respect to each other [27]. These approaches always assume that there is a certain relational connection in their target domain [28]. Where these algorithms have advantages like better predictive capacity, better understanding of domains and are definitely a path of growth for machine learning, there are also certain problems associated with these networks i.e. learning is much harder, inference becomes a crucial issue and greater complexity for user. Our algorithm does not learn the association while training the network instead in simply uses a large number of pretrained models already extensively trained and uses the relational data between them to better their performance. It does not consume excessive resources or requires extensive data to be trained like most RML algorithms.

2.2 Sensors

Most commonly the robots are equipped with four kinds of sensors; laser range finders, standard cameras, omnidirectional cameras and RGB-D sensors, that assist in map building, localization, semantic place classification and other related problems. Out of these sensors laser range and standard camera images are most commonly used. Pronobis et al. [2] and Hirokazu et al. [8] use camera images, whereas Shi et al. [3] [4] [5] [6] and Sousa et al. [7] use laser range finder data for the problem of place classification. Our approach uses camera images for this application which has two major reasons;

(1) Due to the aid of object classifier, partially observable scene is acceptable for classification.

(2) It is inexpensive and captures all the information required while camera images are far too commonly available compared to other sensor data types.

Although, RGB-D Data is richer in information, but if an object classifier is found that works well with standard camera information then RGB-D capacity is merely extra information and being 4D, RGB-D becomes computationally very expensive in problems such as deep learning specially, in the scenario, where we do not use GPUs.

2.3 Learning Approaches

The idea of enabling robots to form human-like understanding of places and concepts is the root of learning algorithms. Moreover, learning approaches enable robots to adapt to changing environment whereas classical classification techniques are rigid and work for only certain environment.

Pronobis et al. [2] very effectively emphasizes the fact. In problems such as place classification learning applications are surfacing, M. Mozos et al. [23] uses a supervised learning approach, identifying environment to four classes with a maximum accuracy of 92On the other hand, S. Jaeyong et al. [9] use a deep learning approach for robot operating in realworld environment, performing real-world tasks, M. Hirokazu et al. [8] use a combination of SIFT and GIST features as Input to Deep Neural Net for scene classification claiming an accuracy of 96Computer vision has tried dealing with dealing the problem by taking aid from CNN methods and GPUs and large collection of online available data. with quite of success specially in case of Places-CNN [17] i.e. Sun-CNN, Places- CNN, ImageNet-CNN [16], [17], [18] using CNNs. but in case of robots we do not have a lot of computational power and we lack large amount of data in real time. We use learning algorithms based deep learning architecture due to two major reasons: (1) images present highly non-linear structures in learning problems which is better handled by deep neural nets. (2) Convolutional Neural Network extensively trained have far better accuracies than any other kind of Neural Network available.

2.4 Dataset

A large number of robot sensor datasets are available online with certain image databases too. Choosing an appropriate dataset to train the network on is a very important consideration. S. Jaeyong et al. [9] use images from Robobarista Dataset [10] for learning about object manipulation tasks. M.

Hirokazu et al. [8] use images from KTH IDOL dataset [22] for scene classification. L. Shi et al. [3] [4] [5] [6] use laser range data from COLD Database [1] for place classification.

This approach tested a lot of datasets and settled on obtaining images from Google source because of the diversity and freedom available for the data available.

2.5 CNNs

There is a wide variety of CNNs available for place classification and object categorization. We required a network that could:

- Give a likelihood score for all the available scene categories that it distinguishes for each image.
- An object classifier that could provide probability and presence of all possible objects in a scene at the same time instance.
- A platform that could be used to merge the two separate type of Convolutional Neural Networks for best performance.

Given these requirements, Caffe was chosen as the standard platform as most CNN models have a Caffe interface available, other option was Matconvnet [29] but the number of CNNs available with Matlab interface are rather limited.

Places205 was chosen for scene categorization although Places365 [30] is a better version but given our limited number of objects that were categorized, Places 205 offered diverse enough scene categories with state of the art accuracies.

For Object classification a Single Shot Multibox Detector was used, which was ideal because it provided prediction and presence of all the possible objects in a scene at the same instance. Although one limitation was the small number of objects that SSD classified and their extremely diverse nature which made it hard to relate them with scenes very strongly.

Chapter 3 **ARCHITECTURE**

3.1 Bidirectional Associative Memory (BAM)

Bidirectional Associative Memory is a form of recurrent Neural Network [11] that is similar to Hopfield Network [12]. When two Hopfield Networks are connected head to tail, a BAM network is formed. As shown in the figure, a BAM contains two layers of neurons which are fully connected to each other and once the weights have been established, input in layer one generates the pattern in layer to and vice versa.

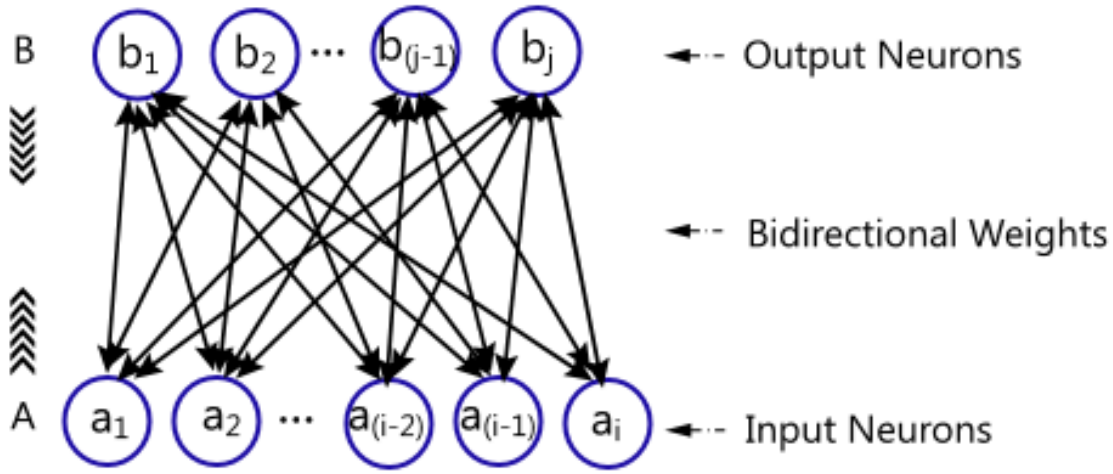

Figure 2 Bidirectional Associative Memory

We can store ‘n’ no. of associations between matrix A and B, where ‘n’ corresponds to number of rows in A or B and number of columns in A and B can vary.

Initially A and B are converted into bipolar form []. To get the association matrix ‘M’:

$$M = A'B$$

To retrieve A and B:

$$A = BM'$$

$$B = AM$$

A BAM network can reliably recall up to min(x,y) independent vector pairs where ‘x’ corresponds to number of column in A and ‘y’ corresponds to number of columns in B.

3.2 Bidirectional Neural Network

An ideal bi-directional Neural Network when trained well should be able to reproduce the data on its input only with the data available at the output and the opposite must also be true. Unlike typical neural networks that learn a set of specific parameter values that when combined with the input, provide an output with the least error; squared or other. When a certain minima is reached, the said parameters are stored and the network is said to be trained with respect to that data for minimum error on the data of that particular kind. The network yet does not have the capability to reproduce its input. A typical neural network is shown in figure.

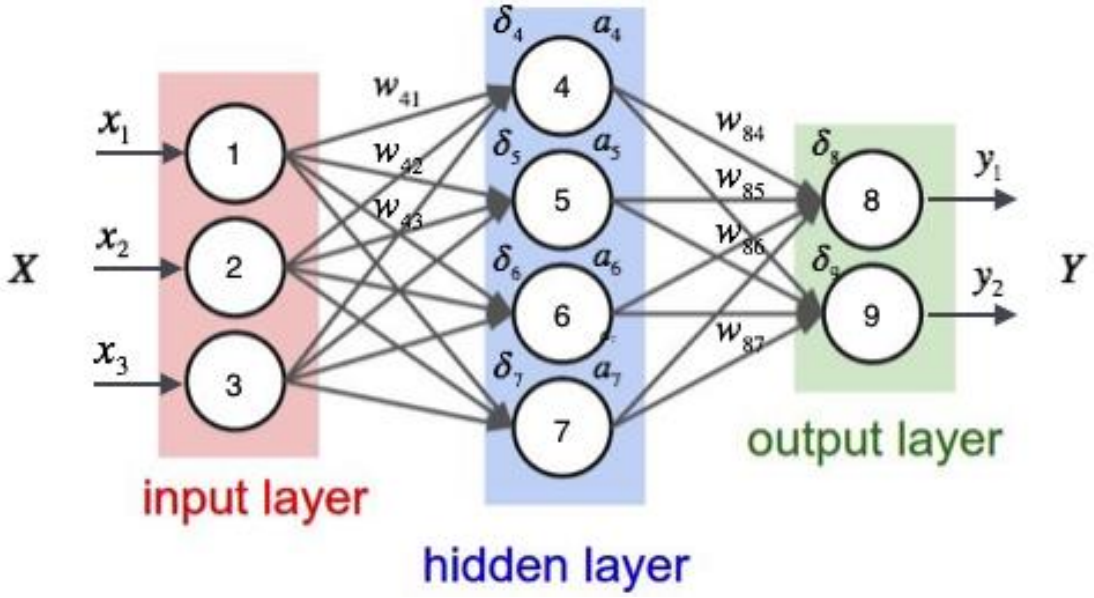

Figure 3 Architecture of a Neural Network with one hidden layer

On the other hand a bi-directional network has the capability to reproduce its input as described in the previous section.

3.3 Proposed Model

Let us assume we have two separate neural networks through which we can distinguish between ‘n’ scene categories and ‘m’ object categories with some certainty.

A is a diagonal matrix which is of size ‘Pn’ x ‘Pn’ where ‘Pn’ is the number of scenes or places that a certain pre-trained network can classify with some success.

$$A = \begin{bmatrix} P & P1 & P2 & P3 & \cdots & Pn \\ P1 & 1 & 0 & 0 & \cdots & 0 \\ P2 & 0 & 1 & 0 & \cdots & 0 \\ P3 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ Pn & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Whereas B is a matrix of size ‘Pn’ x ‘Om’ where ‘Om’ is the number of objects that a certain network can classify to some extent.

$$B = \begin{bmatrix} O & O1 & O2 & O3 & \cdots & Om \\ P1 & 0.9 & 0.6 & 0 & \cdots & 0 \\ P2 & 0 & 1 & 0.3 & \cdots & 0.7 \\ P3 & 0.1 & 0.8 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ Pn & 0 & 0.2 & 0.9 & \cdots & 1 \end{bmatrix}$$

Where A is a diagonal Matrix. B can have more than one values in a row i.e. A place can have more than one objects but the opposite is unlikely. B can have scores between 0 and 1 which are determined so that they correspond to probability of presence of a certain object in a certain scene. If an object is likely to be common in other scenes as well, its probability drops again with correspondence to that scene depending on how common it is in other scenes.

Another requirement for the network is to meet the maximum recalling capacity of the network. Also the network can reliably retrieve only min(x,y) vector pairs. Where ‘x’ corresponds to number of columns in A and ‘y’ correspond to number of Columns in B. For this reason the vectors are augmented with zeroes i.e. for 6 scenes and 10 objects min number of columns in both A and B must be 6.

Our requirement is the relational Matrix ‘M’ that can be obtained by multiplying matrix A and B.

3.4 Proposed Network Architecture

The network architecture proposed is a variation of Bidirectional Associative Memory. The algorithm for the architecture is shown in the figure.

Algorithm

Algorithm 1

Inputs: Labels for objects and scenes: L1, L2 and number of categories for objects and scenes: S, O

Outputs: Association Matrix: M, Place and Object Matrices: M1, M2 and Relational Matrix M3

Function W BRAM
$NS \leftarrow rows(L1)$ {*where NS is No. of Image Samples*}
$NO \leftarrow columns(L2)$ {*where No is No. of Objects in each Samples*}
for $i \in \{1 : NS\}$ **do**
  $M1 \leftarrow M1plusLabel(i)$
  for $j \in \{1 : NO\}$ **do**
    $M2 \leftarrow M2plusLabel(i, k)$
  end for
end for
$A \leftarrow$ *No. of instances for each scene*
$B \leftarrow$ *No. of instances for each objects*
for $ii \in \{1 : S\}$ **do**
  $M1(ii) \leftarrow M1(ii)/A$
  $M2(ii) \leftarrow M2(ii)/A$
  for $jj \in \{1 : O\}$ **do**
    $M2(ii, jj) \leftarrow M2(ii, jj)/B$
  end for
end for
$M3 \leftarrow$ *No. of non-zero elements in each row of M2*
$M \leftarrow M1/M2$
end Function

Algorithm 2 Scene Classification

Inputs: Softmax probability distribution for scene and objects: PredS, PredO, Association Matrix: M and Place Matrix: M1

Outputs: Prediction for Scene: Pred

Function Prediction
$PredO \leftarrow normalized(PredO)$
$PredS \leftarrow normalized(PredP)$
$Top \leftarrow max3(PrepP)$ {*where max3 gets top three values and indices for PredP*}
for $i \in \{1 : size(Top)\}$ **do**
  $Top \leftarrow M1(i)xA(i) + M1(i)xMxB(i)$ {*i represents the label associated with the prediction*}
end for
$Pred \leftarrow max(Top)$
end Function

Figure 4 Algorithm of Bidirectional Neural Network

In the algorithm provided above, L1 corresponds to image labels determining scene category of a certain image and L2 lists the labels of each object in a certain scene depending on the identification power of the object classifier.

S is the number of Scenes that can be distinguished and O corresponds to the number of distinguishable objects.

What we require from the network is to obtain an Association Matrix ‘M’.

M1 and M2 are basically A and B described in the previous section. M3 simply determines how many objects, the network associates with each scene.

The second algorithm uses this M matrix, combines it with the results of the pre-existing qualifier to achieve higher prediction capabilities.

A Bidirectional Network can be therefore called a dictionary between Scenes and Objects which increases the chances of better classification given partial information.

Chapter 4 **EXPERIMENTATION**

4.1 Proof of Concept

In order to verify working of the model, a test is devised. Two sets of images each containing nine 9 distinct images are generated. In the first set 9 geometric shapes are generated and a certain amount of noise i.e. salt and pepper and Gaussian noise is added to the images containing these shapes as show in the figure. Each figure contains a different shape i.e. hourglass, vertical slabs, triangle, trapezoid, pentagon, horizontal slabs, square, circle and a star.

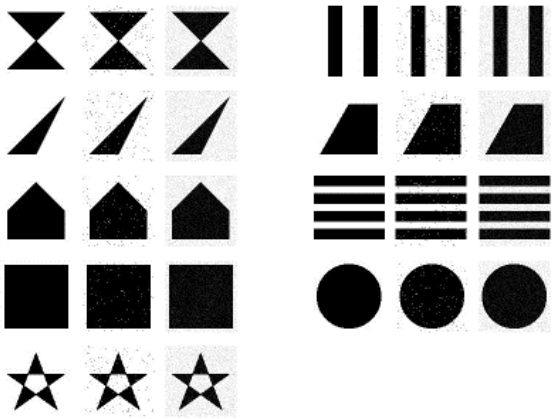

Figure 5 Set of nine shapes with added noise for training an auto-encoder

For the second set, each of these 9 different geometric shapes are places in 9 different positions on a larger image i.e. box 1 through box 9 if we consider the large image to be a square with 9 boxes. Similar salt & pepper and Gaussian noise was added to these images as well.

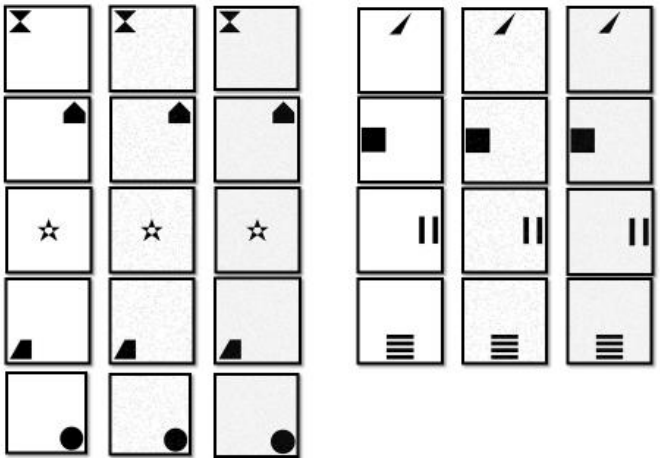

Figure 6 Set of corresponding patterns with added noise for training an auto-encoder

The test images for pattern and shapes were generated by adding excessive noise to the original images as shown in the figure below:

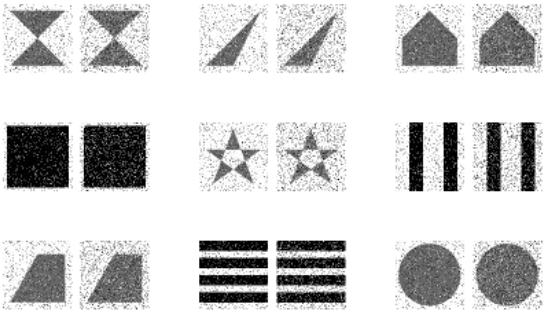

Figure 7 Shapes used for testing

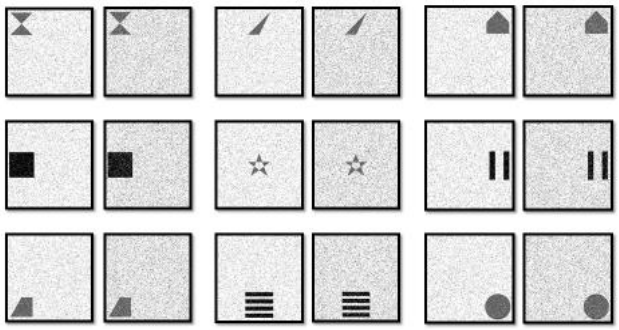

Figure 8 Corresponding patterns used for testing

An auto-encoder based neural network is used for the proof of concept. Initially two auto-encoders were separately trained on the training set. The specifications of the auto-encoder are shown in the following figure:

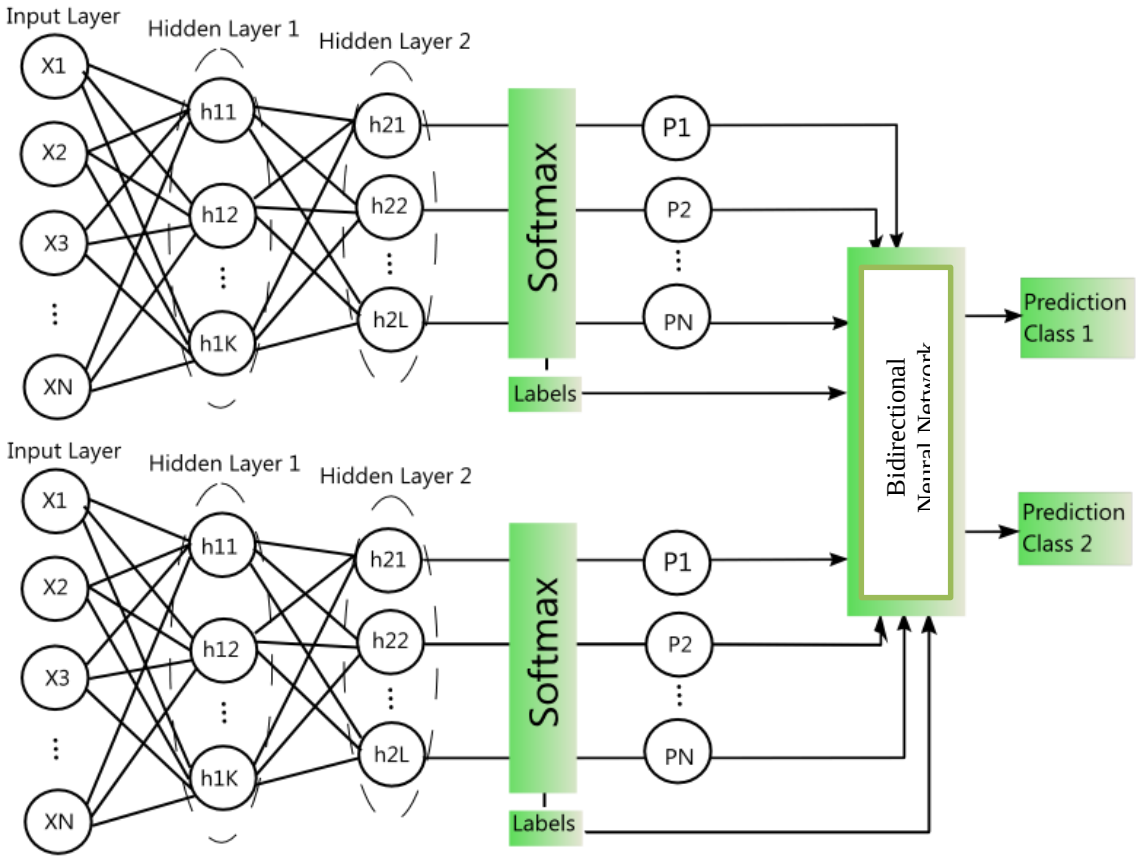

Figure 9 Architecture for the proof of concept

After training both auto-encoders separately. Bidirectional Network was used at the end to see if the results were improved.

a Test 1

In the first test the training auto-encoders both has 300 units in both hidden layers with the input size being 2500 unit.

The pattern 1 corresponds to the patterns containing 9 shapes in 9 positions and Pattern 2 corresponds to the 9 shapes.

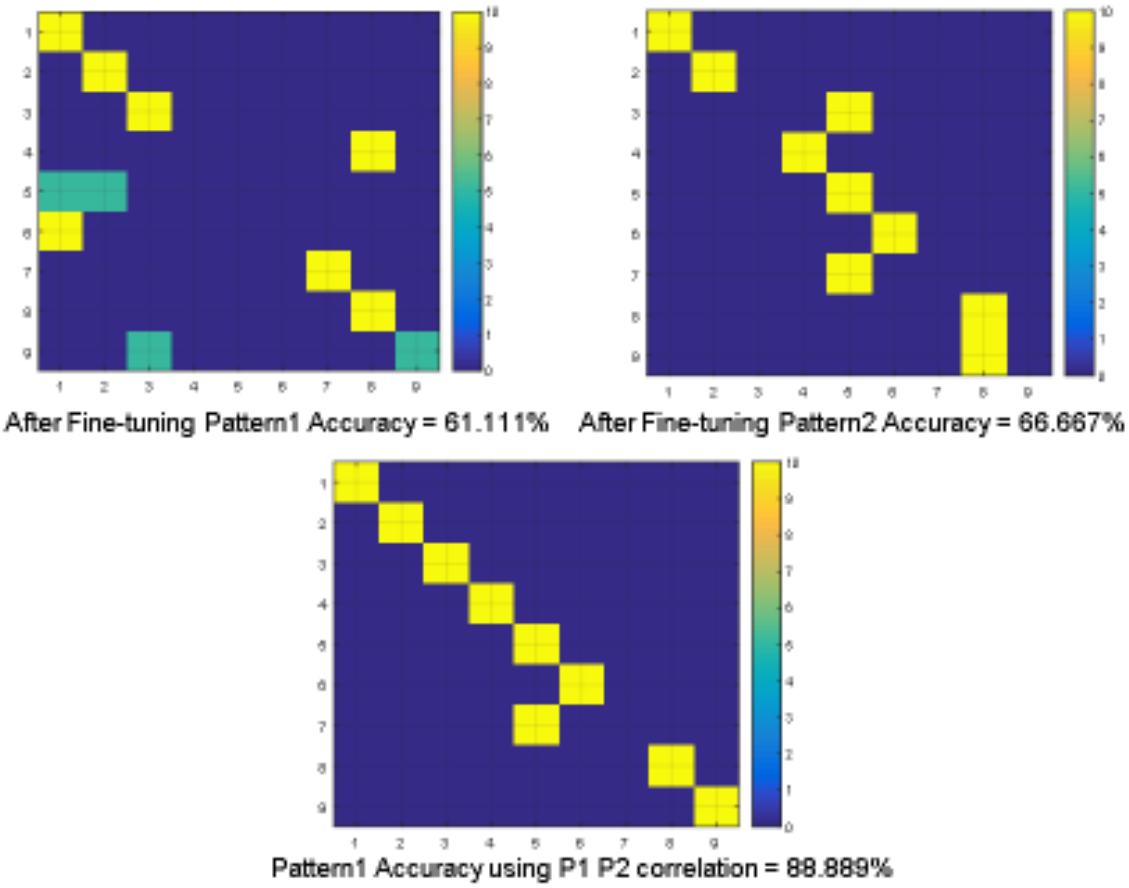

Figure 10 Confusion matrix set 1 for proof of concept

The results show superior performance to separate testing results and even in the areas where both networks give false results the collective network gives a correct result i.e. in case of 3rd shape and pattern.

b Test 2

In the first test the training autoencoders both has 400 units, instead of 300, in both hidden layers with the input size being 2500 unit.

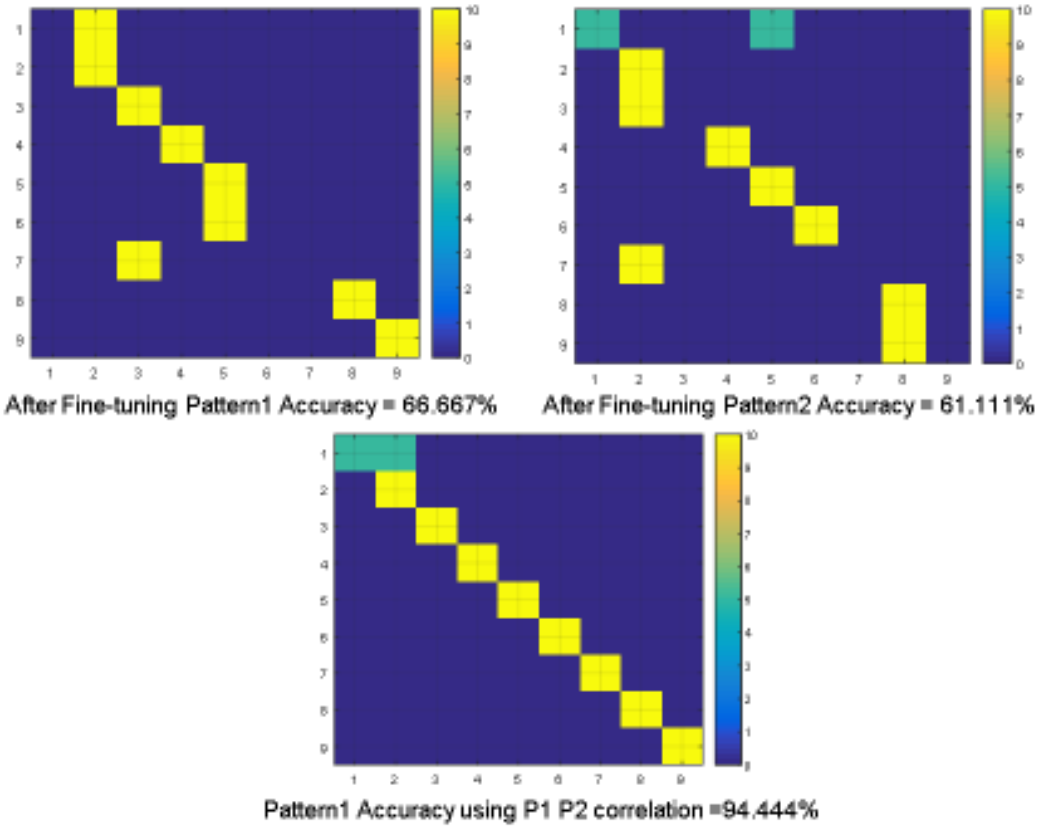

Figure 11 Confusion matrix set 2 for proof of concept

We can see in test 2 as well that the performance of the final network was superior to both individual networks.

After the algorithm was tried and tested to be true, it was deemed necessary to test it on some already existing classifiers and merge them together to see the performance. Since an object detector was required and not merely a classifier that could have useful objects with respect to the place classifier, in order to achieve best results. After having less success in finding an appropriate object detector, a randomized object detector was formulated.

4.2 Randomized Object Detector

8 classes that are commonly found in an office environment are chosen for training the object detector. For each class 250-300 images are taken from Google for testing and training a two hidden layered based auto-encoder. Classes include sink, stairs, printer, fridge, sofa, chair, table and computer monitor.

Some of the resized and grey scaled image samples of each class are shown in the figures below.

Figure 12 Training Images for Object Classifier

Initially an object classifier is trained to distinguish between the 8 classes. This classifier is auto-encoder based with a softmax layer at the end. The architecture used for training the classifier is shown in the figure.

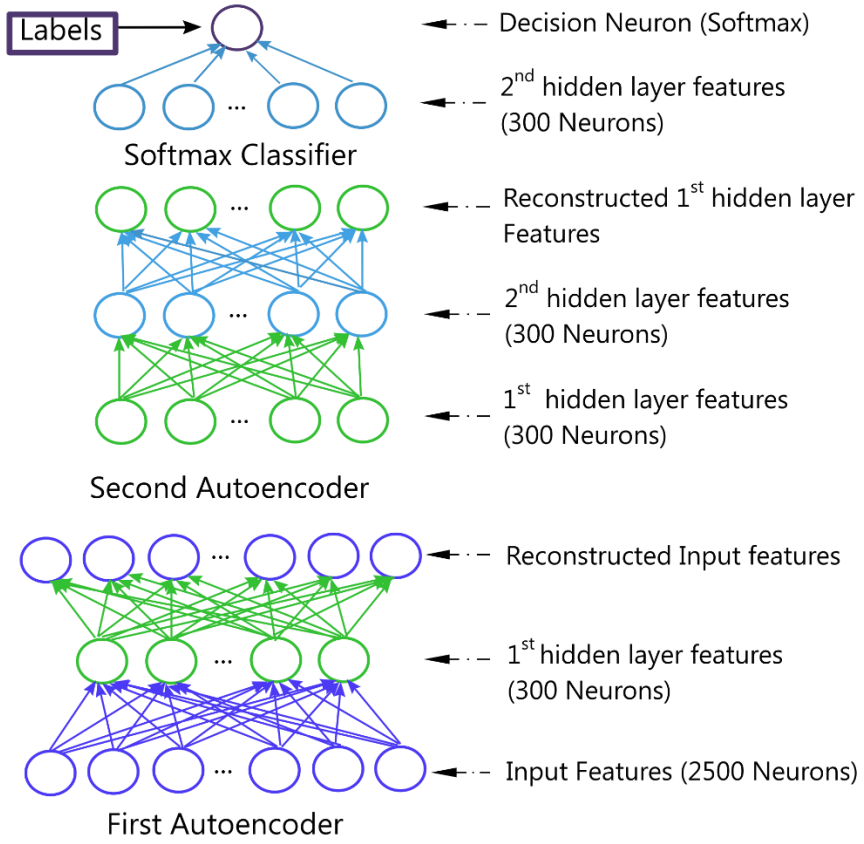

Figure 13 Auto-encoder specifications for proof of concept

The auto-encoder uses greedy layer wise approach. The input has 2500 units while each hidden layer has 300 neurons. The labels are added in the final layer, i.e. softmax layer to distinguish between the 10 classes.

- Total number of test images for all 10 classes = 200
- Total number of training images for all 10 classes = 2018
- Overall testing accuracy of the network before the fine-tuning layer = 71.5%
- Overall testing accuracy of the network after the fine-tuning layer = 77 %
- Value of the sparsity parameter = 0.25, lambda = 0.003 & beta=1

Performance of individual classes is shown in the table below.

Table 1 Performance of Object Classifier

Class	Label	Images	Accuracy before Fine Tuning	Accuracy after Fine Tuning	Specificity	Sensitivity	TP	TN	FP	FN	Training Images	Testing Images
Chair	1	274	84%	76%	98.86%	76%	19	173	2	6	249	25
Sofa	2	279	96%	88%	98.29%	88%	22	172	3	3	254	25
Fridge	3	276	84%	76%	98.29%	76%	19	172	3	6	251	25
Monitor	4	275	92%	92%	98.29%	92%	23	172	3	2	250	25
Printer	5	279	44%	64%	98.29%	64%	16	172	3	9	254	25
Sink	6	278	52%	72%	94.86%	72%	18	166	9	7	253	25
Staircase	7	278	48%	64%	93.14%	64%	16	163	12	9	253	25
Table	8	279	72%	84%	93.7%	84%	21	164	11	4	254	25

- True Positive (TP) corresponds to correctly identified positive instances
- True Negative (TN) corresponds to correctly identified negative instances
- False Positive (FP) corresponds to negative instances classified as positive instances
- False Negative (FN) corresponds to positive instances classified as negative instances

Sensitivity, also known as true positive rate is the percentage of identified positive instances to total number of positive instances i.e. likelihood of an instance being positive when classified positive by the classifier.

Specificity also known as true negative rate is the percentage of identified negative instances to total number of negative instances i.e. likelihood of an instance being negative when identified so by the classifier.[1]

[1] Note: Formula's for sensitivity and Specificity are given in Appendix A.

After training the classifier, random sized blocks of image containing a certain object are used and passed through the classifier to detect the object. This technique takes as small as 1 second to as long as 13 second to find a certain object. An illustration of the localizer is given in the figure where the localizer identifies the location of a sofa in an image. The corresponding time taken to find the object is also shown in the figure.

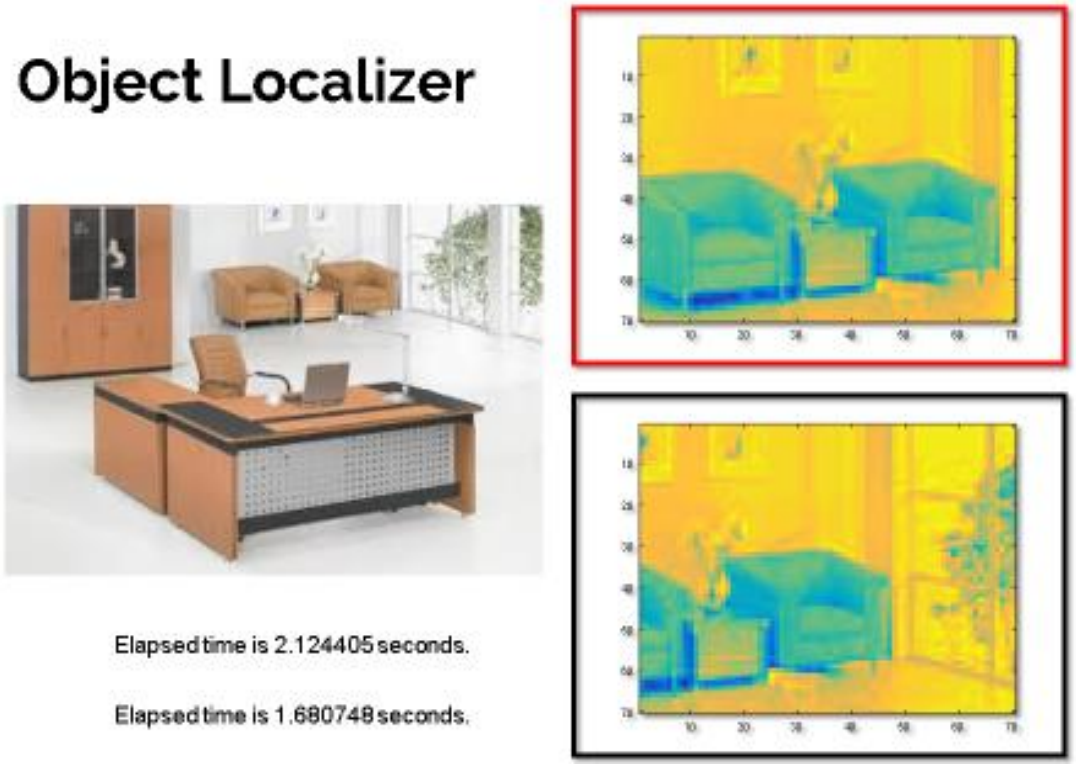

Figure 14 Localization of an object in a scene

a Issues in utilizing the Randomized Object Localizer Approach

In order to effectively prove the usefulness of the bidirectional network, it was important that both place classifier and object identifier can be merged. The Object Localizer in this scenario took unpredictable time and therefore made it hard to effectively sync with a pre-trained place classifier.

Secondly, the described object localizer only identified one object at a time and for the bidirectional network a classifier that could detect multiple objects simultaneously was more preferable.

Finally, due to limited size of the training network, i.e. a two layered auto-encoder architecture, the training accuracies were not very high.

Given the said reasons, a state of the art object localizer was instead used.

4.3 Single Shot Detector

In order to take full advantage of the Convolutional Neural Networks available with stellar performance, place and object classifiers were both chosen to be state of the art. This was hard because in order for a successful performance of the network, a strong correlation between objects and places was very important. The best object localizer found for this job was a Single Shot Multi-box Detector (SSD) [19]. The network was successfully able to localize 20 object categories which included:

1. Aero plane
2. Bicycle
3. Bird
4. Boat
5. Bottle
6. Bus
7. Car
8. Cat
9. Chair
10. Cow
11. Dining Table
12. Dog
13. Horse
14. Motorbike
15. Person
16. Potted Plant
17. Sheep
18. Sofa
19. Train
20. TV Monitor

Given the diversity and limited number of objects detected by the network, the place classifier was required to be large enough to incorporate places that strongly correlated with the said objects.

4.4 Places205

After testing a number of CNNs, Places CNN [20] was chosen to be best suited for place classification in the Bidirectional Network. The version of Places CNN used can classify 205 scene categories. It is a deep neural network for scene categorization. It has been trained on 2.5 million images with the help of multi-GPU architecture by the CSAIL Lab at MIT.

Chapter 5 **CONCLUSION AND RESULTS**

Initially, a Jupyter/IPython Notebook interface was generated for Places205 CNN and SSD using their corresponding Caffe Zoo models [21] in Linux. After the successful interface, a subset of scenes was chosen based on the objects that SSD was able to identify. For each of these categories, a certain number of images were obtained. This was divided into a train set and a test set. Initially all the images were labeled w.r.t. the object and scenes that were present in those scenes. These labels were used to generate an association matrix between places and objects as previously shown in the Bidirectional Network Algorithm. The value of the correspondence Matrix (M) was normalized to remain between 0 and 1 and is shown in the figure below.

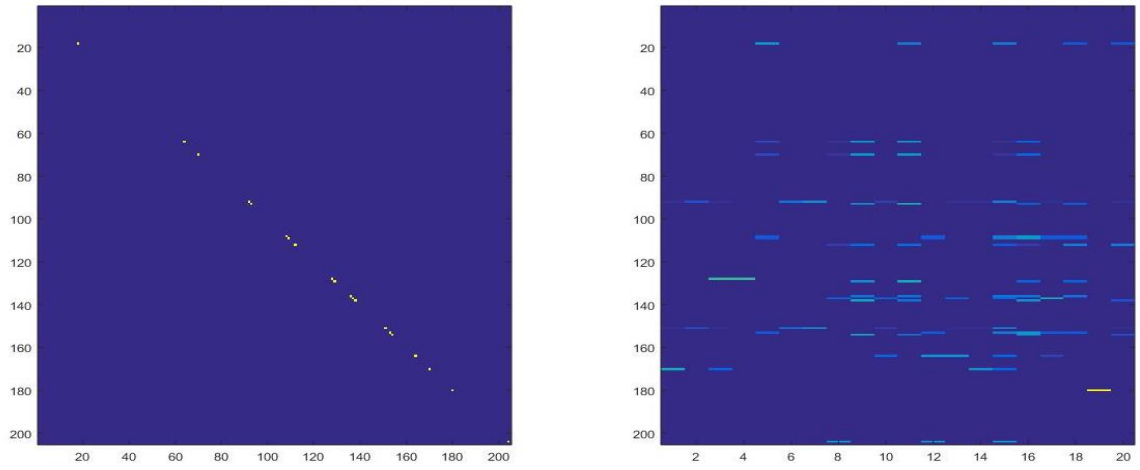

Figure 15 Place matrix 205 x 205 and Object matrix 205 x 20

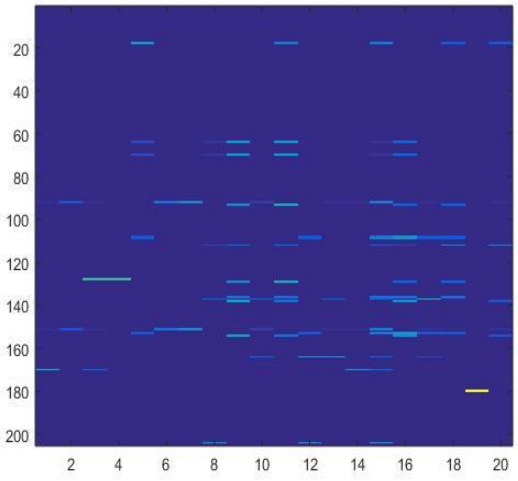

Figure 16 Correspondence matrix M (205 x 20)

5.1 Results without the use of Bidirectional Network

After the successful training the test images were used to individually analyze performance of each network for the selected classes.

The figures below show the True Positives, False positive graphs as well as Sensitivity and Specificity results for SSD.

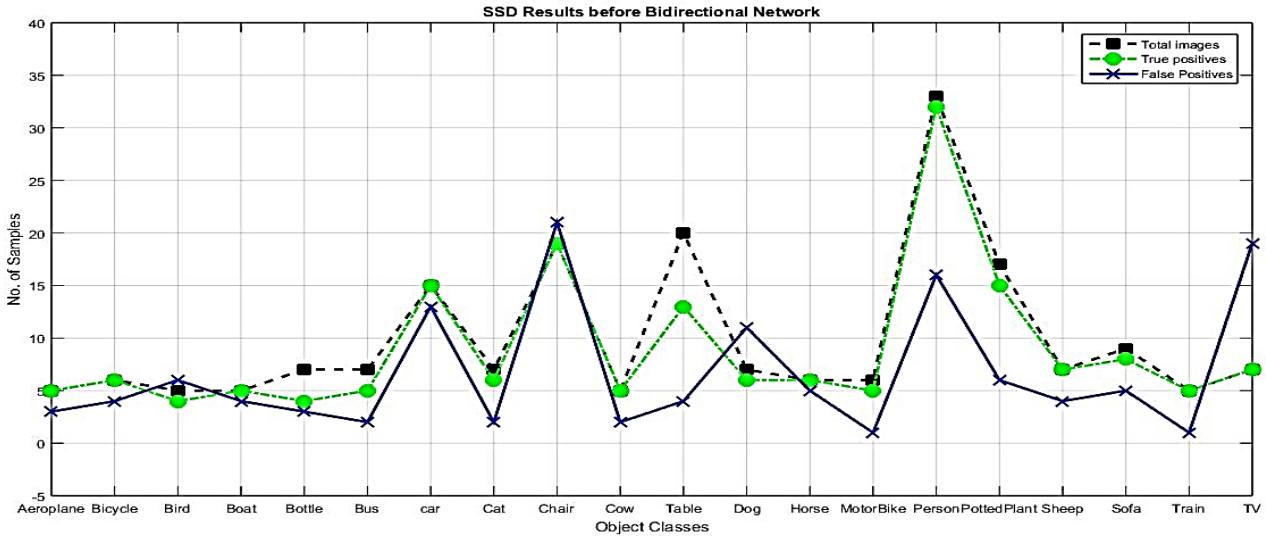

Figure 17 True Positive and False Positives for SSD

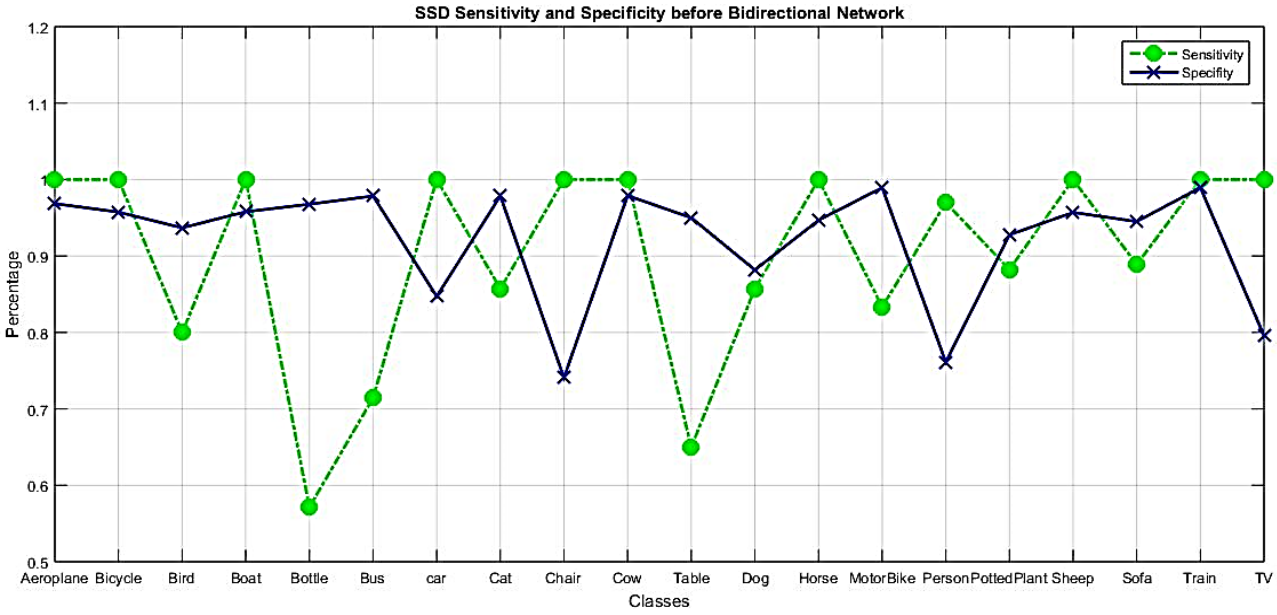

Figure 18 Sensitivity and Specificity for SSD

The figures below show the True Positives, False positive graphs as well as Sensitivity and Specificity results for Places205 before adding the bidirectional Network.

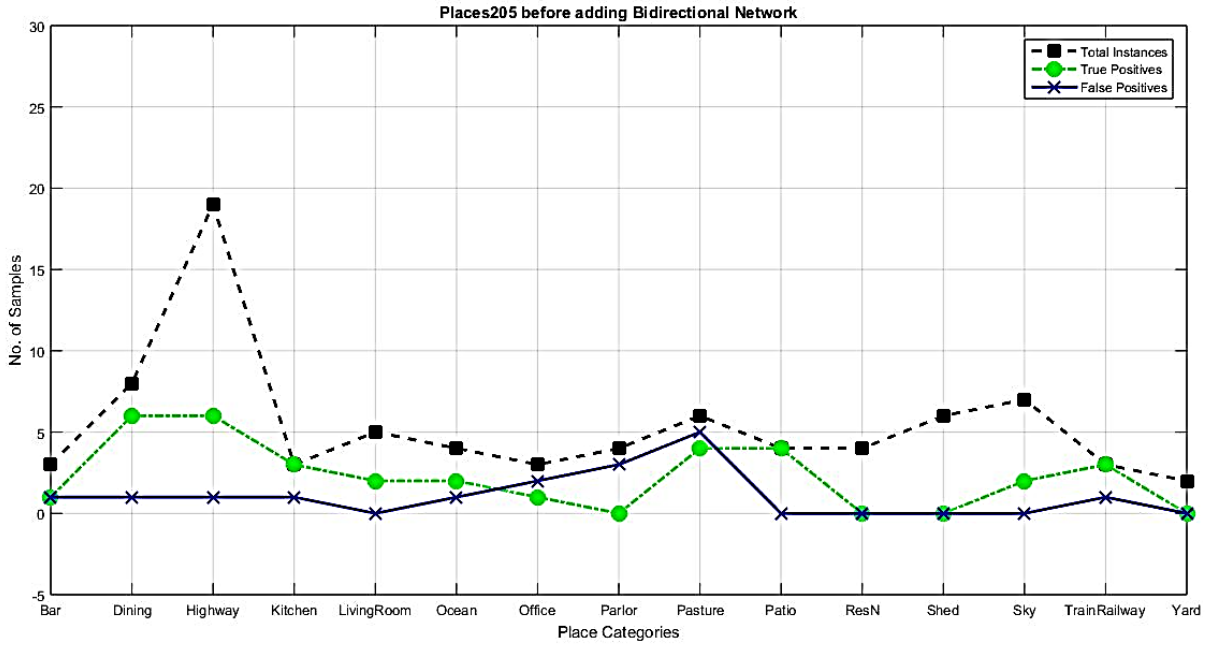

Figure 19 True Positive and False Positives for Places205

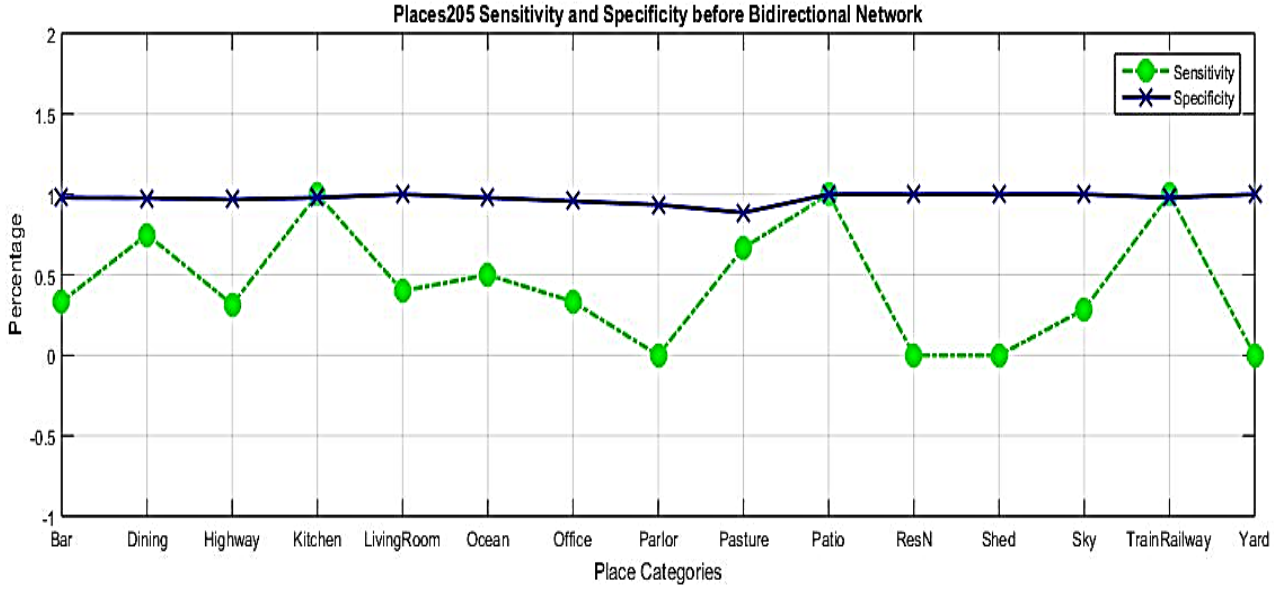

Figure 20 Sensitivity and Specificity for Places205

5.2 Results after using Bidirectional Network

a Case 1

In case 1, only the correspondence matrix M, generated via training images is used with SSD, without using predictions available with Places205. Results are shown in the figures below.

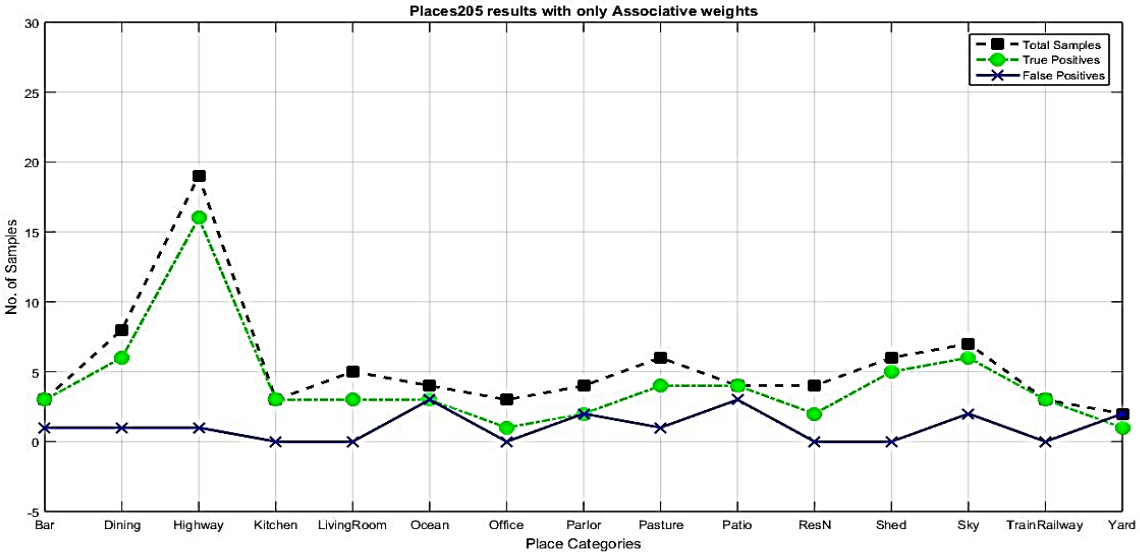

Figure 21 True Positive and False Positives for M matrix with SSD

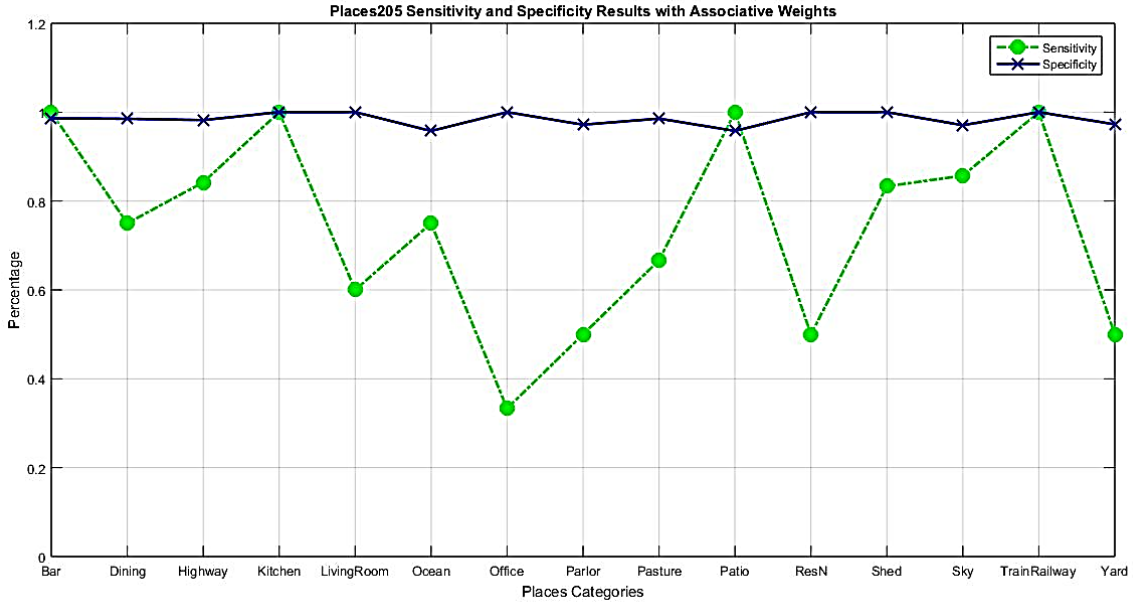

Figure 22 Sensitivity and Specificity for M matrix with SSD

b Case 2

In case 2, the predictions obtained from Places205 are also utilized with SSD and M matrix to obtain better scene classification results. Again the results are shown in the figures below.

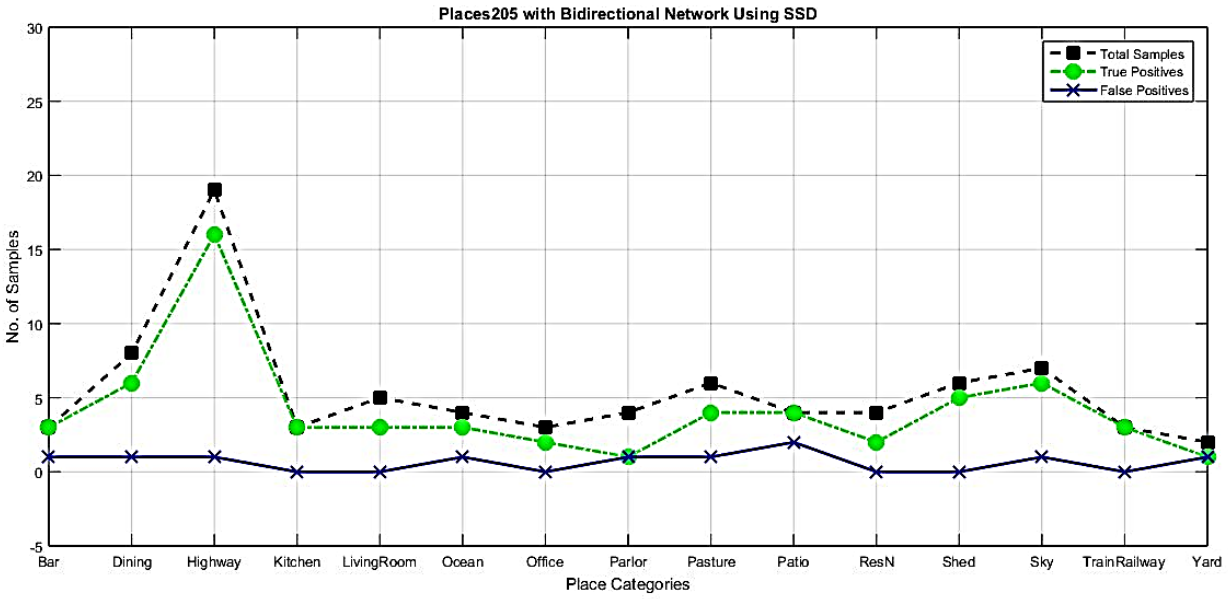

Figure 23 True Positive and False Positives for Places205 with Bidirectional Network

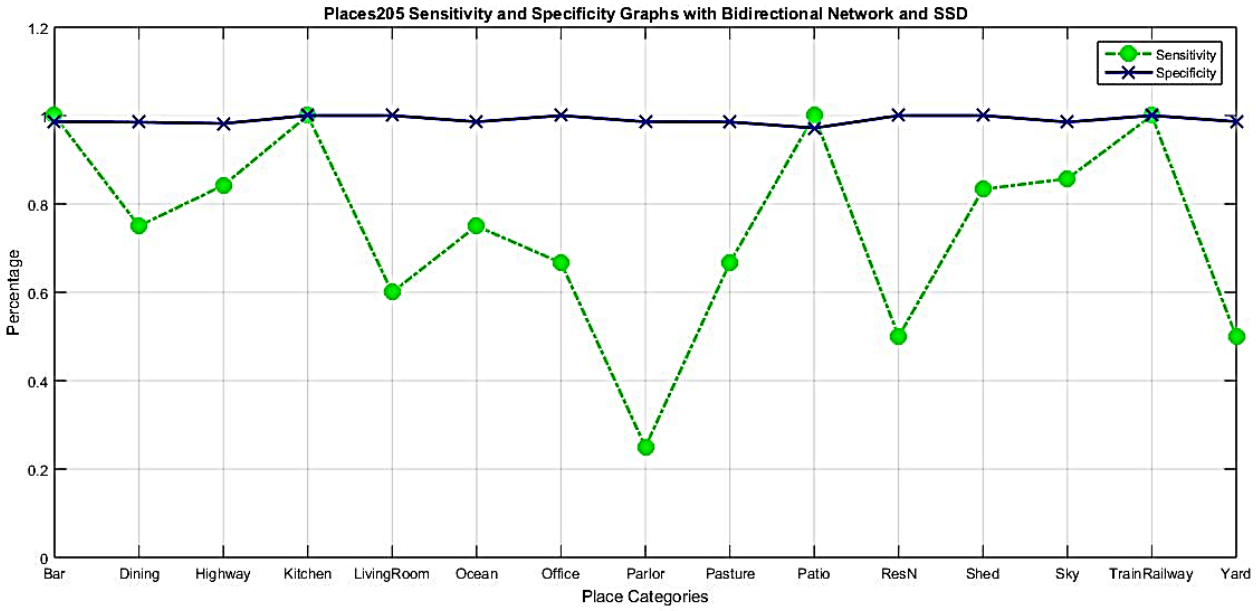

Figure 24 Sensitivity and Specificity for Places205 with Bidirectional Network

Some Images that show the improved performance of Scene Categorization before and after are also shown in the figures below:

Figure 25 Individual results of Places205 and SSD on an image

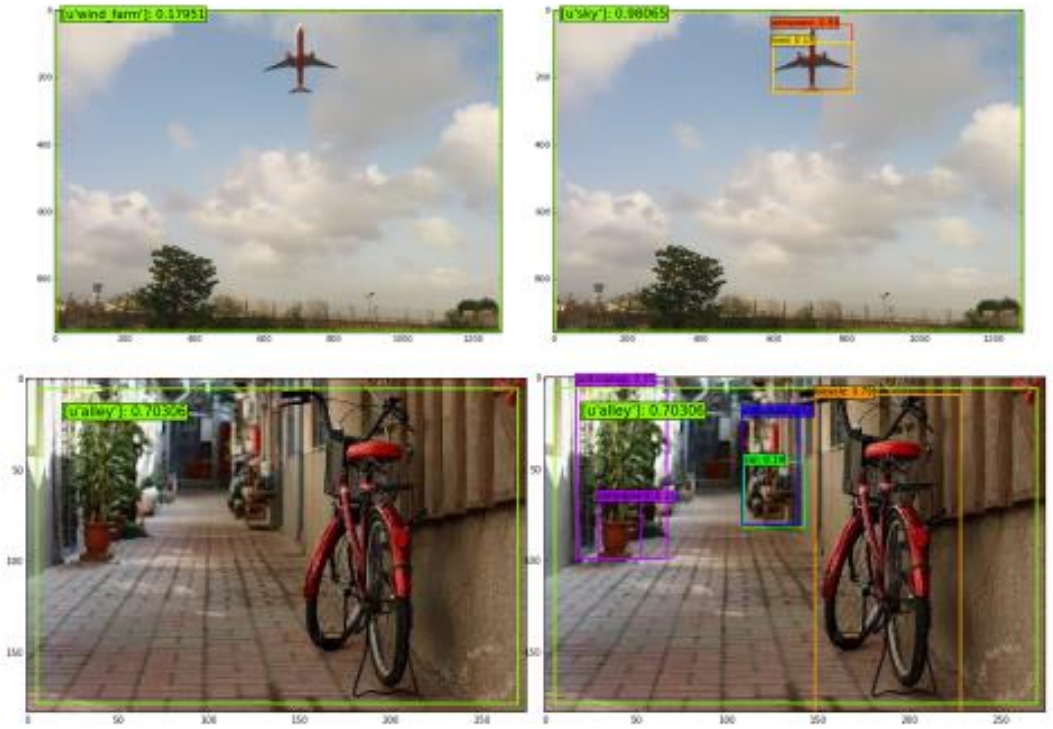

Figure 26 Images which show improved performance with classes that took aid from Bidirectional Network and unaffected classes that were not used with Bidirectional Network but still performed correctly after Bidirectional Network

Finally the Table shows the comparison of Places205 with Case 1 and Case 2 mentioned earlier. We can see the improved performance by the addition of Bidirectional Neural Network.

Formulas for Recall, Precision, F1.score and Accuracy are given in Appendix A.

Table 2 Comparison of Techniques

Method	Recall	Precision	F1.score	Accuracy
Places205	0.5286	0.9433	0.6775	94.8%
SSD with Correlation Matrix	0.6714	0.8236	0.7398	95.3%
SSD with Correlation Matrix and Places205	0.8561	0.9232	0.8890	97.86%

5.3 Conclusion and Future Recommendations

The approach shows better performance and a way of improving performance for already existing Networks which is definitely a path to work in to progress in Machien Learning. In future, the capabilities of the Network can be shown to do the same for object categorization moreover they can be used for connecting different types of Data i.e. Voice and text, Images and audio, words and alphabets and so on.

Another recommendation would be to generate a multi-directional Neural Network that can utilize capacity of all other data types and sync them effectively to generate better predictions but this approach will be heavier on computation because even testing a CNN requires storing heavy variables and too many of them will definitely slow down the speed of the Network.

Appendix A

1. $Specificity = \frac{TN}{TN+FP}$
2. $Sensitivity = \frac{TP}{TP+FN}$
3. $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
4. $Precision = \frac{TP}{TP+FP}$
5. $Recall = \frac{TP}{TP+TN}$
6. $F1.Score = 2.\frac{Precision\ .Recall}{Precision+Recall}$

REFERENCES

[1] Pronobis, Andrzej, and Barbara Caputo. "COLD: The CoSy localization database." The International Journal of Robotics Research 28.5 (2009): 588-594.

[2] A. Pronobis, L. Jie, and B. Caputo, "The more you learn, the less you store: Memory-controlled incremental SVM for visual place recognition," Image and Vision Computing, vol. 28, pp. 1080-1097, 2010.

[3] Shi, L., Sarath Kodagoda, and Gamini Dissanayake. "Application of semi-supervised learning with Voronoi graph for place classification." Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. IEEE, 2012.

[4] L. Shi and S. Kodagoda, "Towards Simultaneous Place Classification and Object Detection based on Conditional Random Field with Multiple Cues" International Conference on Intelligent Robots and Systems (IROS), November 3-7, 2013.

[5] L. Shi and S. Kodagoda, "Towards generalization of semi-supervised place classification over generalized Voronoi graph," Robotics and Autonomous Systems, vol. 61, pp. 785-796, 2013.

[6] Liao, Yiyi, et al. "Place classification with a graph regularized deep neural network model." arXiv preprint arXiv:1506.03899 (2015). In Press.

[7] P. Sousa, R. Araiijo, U. Nunes, "Real-Time Labeling of Places using Support Vector Machines," in Proc. IEEE Int. Symp. on Industrial Electronics (ISIE), Vigo, 2007, pp. 2022-2027.

[8] Madokoro, Hirokazu, Yuya Utsumi, and Kazuhito Sato. "Scene classification using unsupervised neural networks for mobile robot vision." SICE Annual Conference (SICE), 2012 Proceedings of. IEEE, 2012.

[9] Sung, Jaeyong, Ian Lenz, and Ashutosh Saxena. "Deep Multimodal Embedding: Manipulating Novel Objects with Point-clouds, Language and Trajectories." arXiv preprint arXiv:1509.07831 (2015). Unpublished

[10] Sung, Jaeyong, Seok Hyun Jin, and Ashutosh Saxena. "Robobarista: Object Part based Transfer of Manipulation Trajectories from Crowd-sourcing in 3D Pointclouds." arXiv preprint arXiv:1504.03071 (2015). Unpublished

[11] Medsker, L. R., and L. C. Jain. "Recurrent neural networks." Design and Applications 5 (2001).

[12] Guan, Zhi-Hong, and Guanrong Chen. "On delayed impulsive Hopfield neural networks." Neural Networks 12.2 (1999): 273-280.

[13] Samad, Tariq. "Neural network auto-associative memory with two rules for varying the weights." U.S. Patent No. 5,050,095. 17 Sep. 1991.

[14] Hassoun, Mohamad H. "Dynamic heteroassociative neural memories." Neural Networks 2.4 (1989): 275-287.

[15] Costante, Gabriele, et al. "A transfer learning approach for multi-cue semantic place recognition." Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013.

[16] Xiao, Jianxiong, et al. "Sun database: Large-scale scene recognition from abbey to zoo." Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. IEEE, 2010.

[17] Razavian, Ali, et al. "CNN features off-the-shelf: an astounding baseline for recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2014.

[18] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database."Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.

[19] Liu, Wei, et al. "Ssd: Single shot multibox detector." European conference on computer vision. Springer, Cham, 2016.

[20] Wang, Limin, et al. "Places205-vggnet models for scene recognition." arXiv preprint arXiv:1508.01667 (2015).

[21] Jia, Yangqing. "Caffe model zoo." (2015).

[22] KTHIDOL2 database," Technical Report CVAP304, Kungliga Tekniska Hoegskolan ,CVAP/CAS ,Oct. 2006.

[23] O. Mart´ınez Mozos, C. Stachniss, and W. Burgard, "Supervised learning of places from range data using AdaBoost," in Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA), Barcelona, Spain, April 2005, pp. 1742–1747.

[24] LeCun, J., Corinna Cortes, and Christopher JC Burges. "The mnist dataset of handwritten digits." URL http://yann. lecun. com/exdb/mnist (1999).

[25] Ng, A. "Ufldl Tutorial on Neural Networks." Ufldl Tutorial on Neural Networks(2011).

[26] Ballester, Pedro, and Ricardo Matsumura de Araújo. "On the Performance of GoogLeNet and AlexNet Applied to Sketches." AAAI. 2016.

[27] Richardson, Matthew, and Pedro Domingos. "Markov logic networks." Machine learning 62.1 (2006): 107-136.

[28] Khosravi, Hassan, and Bahareh Bina. "A Survey on Statistical Relational Learning." Canadian Conference on AI. 2010.

[29] Vedaldi, Andrea, and Karel Lenc. "Matconvnet: Convolutional neural networks for matlab." Proceedings of the 23rd ACM international conference on Multimedia. ACM, 2015.

[30] Zhou, Bolei, et al. "Places: A 10 million image database for scene recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence (2017).