Volume 1, Issue VII, June 2022, **pp. 85-100**

Submitted 2/4/2022

Final peer reviewed 31/5/2022

Online Publication 11/7/2022

IMPROVING THE PERFORMANCE OF OBSTACLE DETECTION AND AVOIDANCE AUTONOMOUS MOBILE ROBOT USING TRANSFER LEARNING TECHNIQUE

[1]Onah C.B., [2]Asogwa T.C.

[1,2] Enugu State University of Science and Technology
onakchukwuebuka7@yahoo.com

Abstract

This paper is targeted at improving the performance of an obstacle avoidance autonomous mobile robot using transfer learning technique. This work was embarked on to address the problem of obstacle avoidance in mobile robots. To address this problem, transfer learning was adopted with Alex.Net algorithm and used to develop an improved obstacle recognition algorithm using convolutional neural network. The method was guided by Dynamic Systems Development Model (DSDM) methodology, while the system design was done using objected oriented system analysis approach. The model was implemented with Simulink and evaluated. The result after cross validation showed high obstacle detection and recognition accuracy of 98.7%. The accuracy achieved by the algorithm was further justified using comparative algorithm with other state of the art obstacle detection and recognition algorithms. The result showed that the adopted algorithm have a percentage improvement of 1.89% against the best existing state of the art algorithm.

Keywords: Autonomous Robot, Transfer Learning, Alex.Net, Convolutional Neural Network

1. INTRODUCTION

In the last six decades, robotic technology has revolutionize the industrial sector as a redundancy means to help facilitate technical process and maintain standard of production. This is because, robot has the capacity to perform complex and dangerous task repeatedly, execution of multi functions tirelessly and accurately in faster time at a lower cost compared to humans. Due to this reason the technology is one of the fastest expanding fields of scientific research with versatility in applications ranging from surveillance, planetary exploration, patrolling, emergency rescue operations, reconnaissance, petrochemical applications, industrial automation, construction, entertainment, museum guides, personal services, intervention in extreme environments, transportation, medical care, and so on, as well as many other industrial and nonindustrial applications. Most of these are already available on the market (Denica et al., 2018).

According to Liu et al. (2018), robots are classified as either standalone robots, autonomous or mobile robots. The standalone robot was the first generation of robots such as the traditional robot arm or robot manipulator designed with various degrees of freedom. These are employed for the manipulation of heavy payloads from one position to another or tasks which are dangerous, such as the manipulation of toxic or hot objects (Inyama and Agbaraji, 2015). However due to the static base design nature of the robot, they are only specialized to one task within the particular location they are installed. This as a result limits the application and hence calls for a mobile robot which is autonomous in nature.

According to Eneh et al. (2019), the autonomous robot classified as the second generations of robot are designed with the ability to move without assistance (autonomous navigation) from its current location to a target goal position based on the on-board sensory inputs received from the environment. These mobility features broadened the application of the robot compared to the traditional industrial standalone robot. However the challenges encountered in autonomous by these mobile robots are their complete dependency on programming structure and command for the execution of tasks, inability to maneuver constraints during trajectory, not able to detect errors in their own performance or to robustly interact with a complex environment and safety challenges. These have affected the integrity and confidentiality of the robotic system, and as a result, today these robots are deployed mainly for dangerous task and are not allowed to function in an area close to human workers. Hence full autonomy was never achieved in this generation of robot due to this reason. Universally the autonomous mobile robots are the most employed due to their ability to perform effectively, but their inability to function and perform complex operations without human intervention remained a major problem. According to (Eneh et al., 2019), the main reasons robots linger in effective execution of many task is due to their limited ability to recognize certain objects in their environment. This has affected their ability to navigate autonomously, detect and recognize objects and obstacles at all times and as a result has remained a big challenges for researchers over the years.

Overtime, many studies have proposed various approaches to address these problems using advance technologies like machine and deep learning as in (Clark et al., 2014; Jiyong and Woojin, 2019) among others, but despite their success, the dataset used in training the robot are limited to certain few objects which are not good enough for robust simultaneous localization and mapping. To address this problem, this research proposes the use of transfer learning to improve the performance of autonomous mobile robots. Transfer learning are already developed deep learning algorithms to improve object detection and recognition intelligence. This will be adopted in this research and used to improve the performance of the case study robotic system.

2. LITERATURE REVIEW

Seyyed et al. (2019) researched on obstacle avoidance of mobile robots using modified artificial potential field algorithm, according to them, the algorithm has the capacity to control the robot to the targeted optimal environments without any challenge of obstacle delay. The work has simple mathematical model which is easy to understand. However, the major setback of this work is at the local minimum and the inaccessibility of the target when obstacles are in the propagation path. Ashiwini (2019) proposed a simple obstacle avoidance robot; this work presented a robotic design without micro controller. The aim is to implement a robotic system at the lowest economic cost, the work employs sensors and transistors as the main components. The setback with this work is that the circuit is very bulky with lots of transistors and sensors which can be minimized if micro controller has been employed. Also this methodology cannot be recommended for advance robotic design due to the fact that it lacks proper adaptive and intelligent features.

Athira and Archana (2016) presented a mathematical modeling and control of a mobile robot for path tracking; the work performed through differentiable control laws based on kinematic controller; they induced that issues such as uncertainty in sensing and action, planning, learning and reliability should all be integrated into one system in future work. Clark et al (2014); researched on a navigation and obstacle avoidance algorithm for mobile robots operating in unknown maze type environments; the work first characterize a maze type environment using a navigation algorithm and employs a stochastic learning automation approach for obstacle avoidance; however application of deep learning for object detection will give a better result and operability accuracy.

According to Ian et al (2014) their work on Deep Learning technique for Detecting Robotic Grasps where two deep networks were integrated into this system in order to have a robust and faster system. The first network has fewer features and it is faster to run and can prune out unlikely candidates effectively, the second one has higher number of features and is slower but has to run on the top few detected features. This system achieved 84% accuracy in grasping algorithm and 92% in its detection and recognition algorithms. However, building a better algorithm to improve the grasping algorithm of this system is required for reliability; therefore an advanced study is advised for this work.

According to Sulabh and Christopher (2017), their work on Robotic Grasp Detection using Deep Convolutional Neural Networks which presents a robotic grasping approach in order to predict and attain the best grasping pose for a parallel-plate robotic gripper. This model makes use of shallow convolutional neural network to predict the configuration of the grasp for the interior of the object, then uses deep convolutional neural network to extract features from the scene. This work got a performance accuracy of 89.2% on the standard dataset used to run it in real-time speed. Therefore, this work requires an improvement for a better performance in future works. Antonio et al, (2017) researched on Bridging between Computer and Robot Vision through Data Augmentation, this work proposes a technique that zoom an object detected by a layer and modifies the image of the object in other to improve the vision system of a robot. The part that uses the deep convolutional architecture helps to increase the object recognition performance to a good rate. The performance accuracy of this work is reported to be 90.1%, and it requires an improvement for a better system performance and reliability.

3. METHODS

The methods used for the development of the proposed system are data acquisition, convolutional neural network, Alex.Net, deep transfer learning, training, obstacle detection and avoidance.

3.1 Data acquisition: This is the first process in robotic localization and mapping of its environment. This process involves the use of sensing devices which in this case is camera for data collection from the environment. The data are collected and feed to the controller for processing and decision making.

3.2 Convolutional Neural Network (CNN): This is a deep learning based algorithm which is the most effective in solving image based pattern recognition problem. The CNN is a series of inter connected artificial neural network in multi number of layers which are the input layer, the convolutional layer, fully connected layer and output later as shown in figure 1;

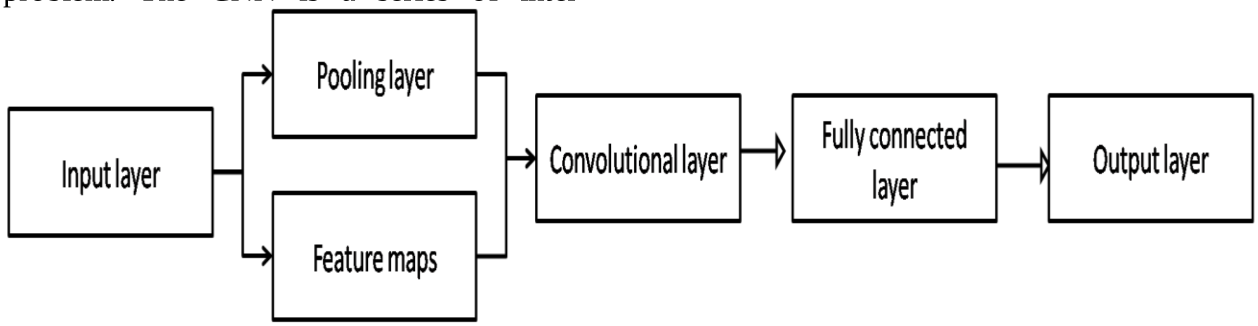

Figure 1: CNN block diagram

In the CNN diagram above the input layer was responsible for the configuration of the input data collected from the sensor into a specific dimension before feed to the convolutional layer. The convolutional layer extracts the images features using filters and pooling layers to feed to the fully connected layer for training and then classification in the output layer. The system design is presented in the next chapter.

3.3 Alex.Net: One major challenge in solving the problem of obstacle detection and avoidance has remained the unavailability of dataset which contains enough data for training. This has remained a major limitation over the year until transfer learning was developed recently. This transfer learning is the ability to transfer algorithm to improve the learning efficiency of others. There are many transfer learning algorithm such as Google cloud vision, ResNet, Alex.Net, among other as identified in Alex (2016). However Alex.Net is the most recognized with over one million different objects used to develop the algorithm (Alex, 2016). This was used to improve the performance of the robot for enhanced object detection and recognition performance.

3.4 Deep Transfer Learning (DTL): Deep transfer learning is a process which married the convolutional neural network and the transfer learning algorithm together as the proposed intelligence algorithm used for the detection and recognition of obstacles with then propagation path of the robots. The system design is presented in the next chapter.

3.5 Training: This is a process of learning the algorithm with new data collected from the environment and then makes classification decisions based on a comparative process with the reference DTL algorithm already developed to detect and recognize obstacle in the environment.

3.6 Obstacle detection and Avoidance: This is the process where the robot used the output of the training process to make decision. When the output of the training is an object, the robot localized the position of the object and then maneuvers it to avoid collision.

4. SYSTEM DESIGN OF THE MOBILE ROBOT

The robot under study is a four wheel mobile holonomic robot. The robot was designed using three main block section which are the controller section, dynamics and kinematics. This aforementioned section presented the various functionalities, variables and attributes which works collectively to present the case study obstacle avoidance robot. The model of the robot was developed using the relationship between the front and rear wheels (wr and w_L) where (r) is right, (y) is left, (l) is left and (x) is right, torque (F), velocity (v) and time (t) as presented in the model in figure 2;

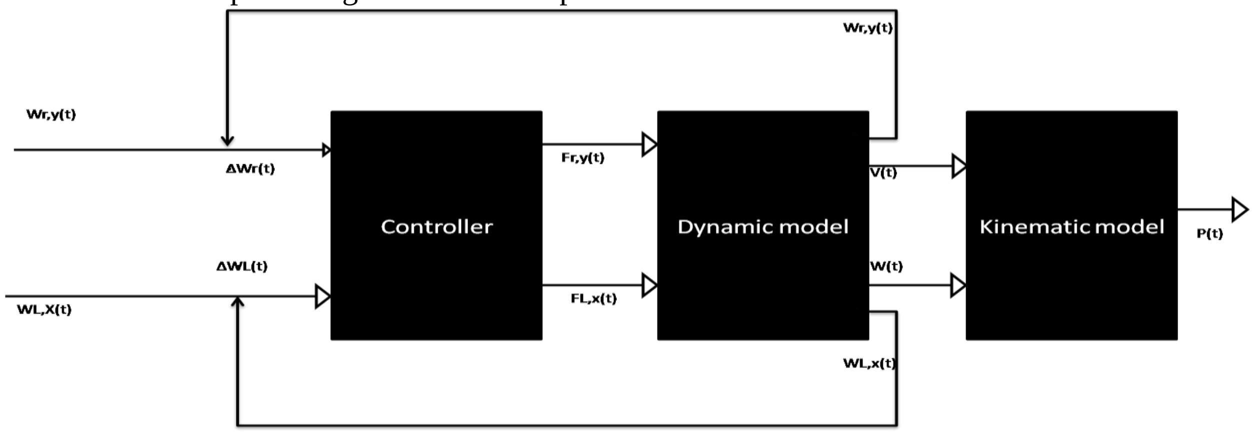

Figure 2: Model Diagram of the mobile robot

The figure 2 presented the modeling diagram of the mobile robot using the relationship between the control parameters, dynamics attributes and kinematic variables to formulate the holonomic structure.

4.1 The Robot Kinematics

The kinematics model presents the geometry which related the multi degree of freedom kinematic chains that form the structure of the mobile robot. This means that the links are modeled as rigid bodies with the joints making provision for pure translation or rotation due to the action of force. The kinematic model of the robot presents the mechanics structure which describes the odometry, showing how forces act on the robotic structure. It presents the analysis of the robot odometry with consideration to velocity and position which formed the robotic mechanism. The wheel mobile robot kinematics are of three types (Eneh al, 2019); Internal kinematics, External kinematic, and forward and inverse kinematics. The internal kinematic defines the relationships between external variables of the systems, such as the rotation of the wheel and robot movement (Pai et al., 2012). External kinematics defines the robot's position and orientation with respect to some coordinate frame of reference. Forward and reverse kinematics or direct kinematics solves the robot's state depending on its inputs. The inputs can be wheel speeds, movement of joints, steering of wheels, etc. A four-wheel mobile robot's steering is based on controlling the robot's four left and right differential wheels' relative velocities.

4.2 The Robot Dynamics

The dynamic model describes the state of the robotic system variables, as the operation evolves with time. It explains the relationship between the forces acting on a robotic structure with respect to the acceleration produced by the structure. However, it has been a challenge in understanding the kinematics and the dynamics of the four-wheel mobile robot platform due to the complex wheel ground interaction and the kinematic constraints imposed on the platform by the four wheels. However, since a mobile robot is dynamic system in nature, as many attributes such as lateral and longitudinal slip are attributed to it, an appropriate torque needs to be applied to the wheels of the robot to obtain the needed motion. Basically, the dynamic equation of motion concepts and techniques can be approached using the Newton-Eular and Lagrange methodologies as presented in (Moreno et al., 2016).

4.3 The control system

This is the most important part o the robot which coordinates the action and reaction to object. The control system coordinated the kinematic model with the dynamic model so as to achieved controlled motion, with ignorance to nonlinear constraints as already aforementioned. From the literature reviewed it has been established that this control system suffer some limitations, apart from the précised control of the kinematics against dynamic constraints. The research gap submitted that the ability o the robot to detect and recognize unknown obstacles within the propagation part has remained a major problem in time memorial, mainly due to the unavailability of robust training dataset for pattern recognition problems. To address this problem, the study developed a deep transfer learning algorithm and used to improve the recognition intelligent of the robot for better efficiency.

4.4 Modeling of the deep transfer learning algorithm

Deep learning is a branch of machine learning as already discussed in the literature review for processing huge amount of data, so as to achieve a regression or classification result. There are various types of deep learning technique like the unsupervised pre-trained neural network, convolutional neural network, recurrent neural network and recursive neural network, each compatible to solve various problems, but the one adopted for this research is the convolutional neural network (CNN) (Paitnaik et al., 2014). CNN was adopted over the other deep learning predecessors due to its automatic feature extraction for learning ability and achieving very high precision rate, especially for the large image-sets. This implies that CNN have the intelligence to identify the key features from an image data and then learn it to make predictions with a correct accuracy level competing with human intelligence more that other deep learning technique for image data types. This was achieved easily here by fine tuning a transfer learning algorithm called the Alex.Net. The Alex.Net is an already developed deep learning model trained with over a million images classified into 1000 subsets (Anakwenze, 2021). The transfer learning algorithm contains five convolutional layers, three fully connected layers alongside the max pooling and drop out layers and was presented using the modeling architectural diagram in figure 3;.

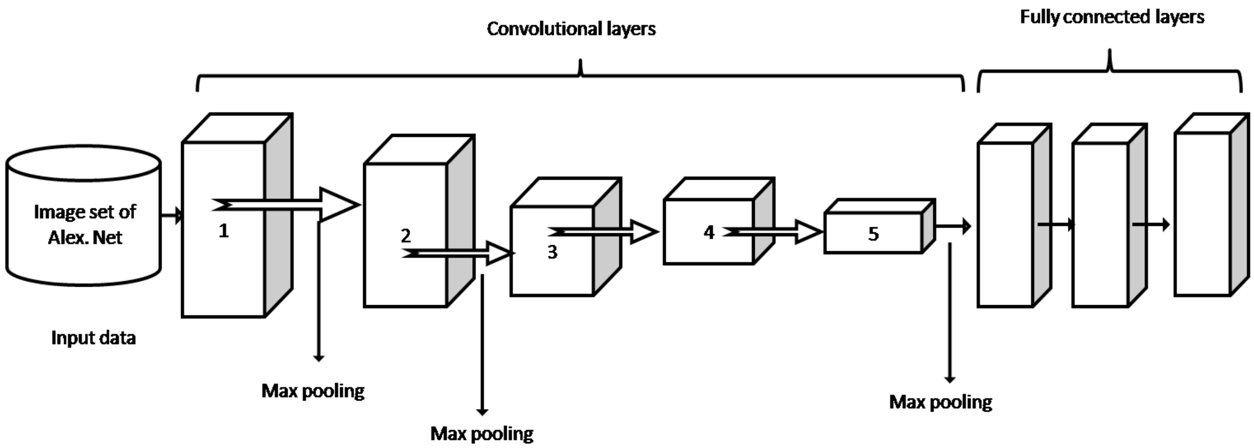

Figure 3: Architecture model of the Alex.Net

The architecture presented in figure 3 shows the configuration of the transfer learning model with the input from the image data set specified as 227 by 227, while the channel size is 3. The learning rate factor is for weight is 15, and the bias learning rate factor is 15. The convolutional layers slowly learned all the over a million images in the dataset incrementally for convolution and then classified them in the fully connected layer. The pooling function is used to transfer pre-learned data from one layer to the next until it gets to the fully connected layer.

4.5 Fine Tuning the Alex.Net

This was done developing CNN architecture and using the convolutional layers to replace the last three convolutional layers of the Alex.Net architecture already presented in figure 4.2. The development of the CNN architecture to be used for the fine tuning was designed using the data model of the objects data, input layers, three convolutional layers, pooling layers, fully connected layers and output layers presented figure 4;

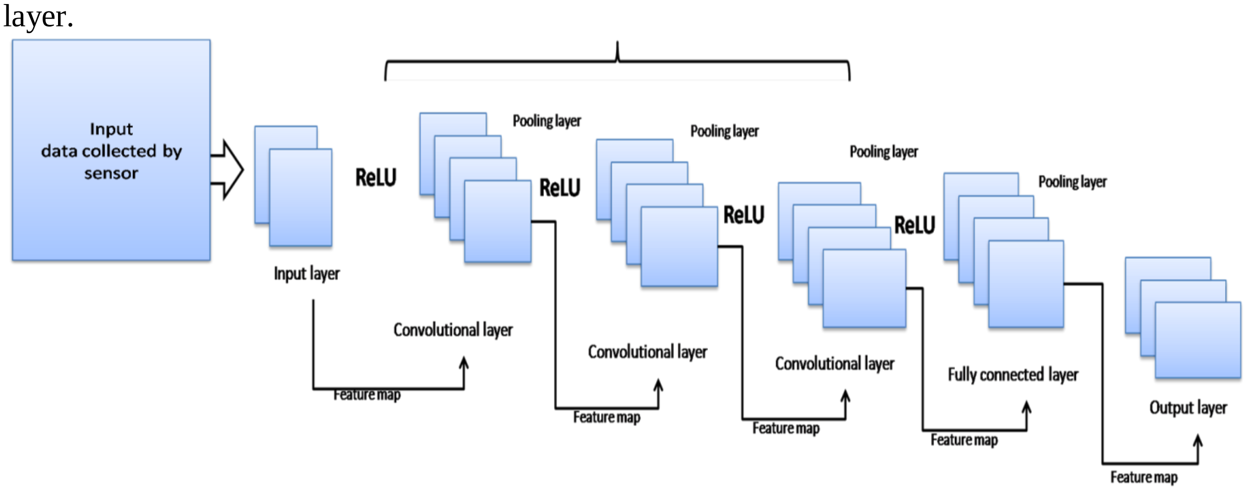

Figure 4: CNN architectural Model

The figure 4 presented the CNN architecture developed with input layer which sizes the input data from the dataset into desired dimension of resolutions and color channel as (227 x 227 x 3) where the value for the color channel represents the RGB color. The dimensioned data identified with the input matrix (h x w x d) and filter dimensioned as f_w x f_h x d are feed to the convolutional layer as (h-f_w + 1) x (w – f_w + 1) x number of convolutional layers. Where h is height of the image pixels, w is weight of the image pixels, d is the dimensions, f_w is the filter weight, f_h is the filter height. The filter specification for this case is 5 x 5 and image size is (227 x 227 x 3). These filters specified in (5 x 5 x 3) are used to scan the interesting part of the image for each convolution and label the feature map learned per convolution as 75 based on the relationship between the filter dimension and color channel. This scanning process continuous until the whole image is scanned and arranged in an array of feature map vectors of 223. During the scanning process, at times when the filter don't size the image arrays very well, valid padding was used to ensure that only interesting part of the image was scanned and also rectified linear unit (ReLU) was used to introduce nonlinearity to ensure that only real values are extracted in a total feature map of 223 x 223. Pooling layer was used to pool the feature maps in a compressed spatial size and then feed to the second convolutional layer via the neurons with weights of 154587 via back propagation process. The process continuous again in next two convolutional layers until the data model is learned. Now that the data model is learned by the convolutional layers of the CNN, the three layers was sued to replace the last three layers of the Alex.Net architecture to form a deep transfer learning model as shown in the architecture below;

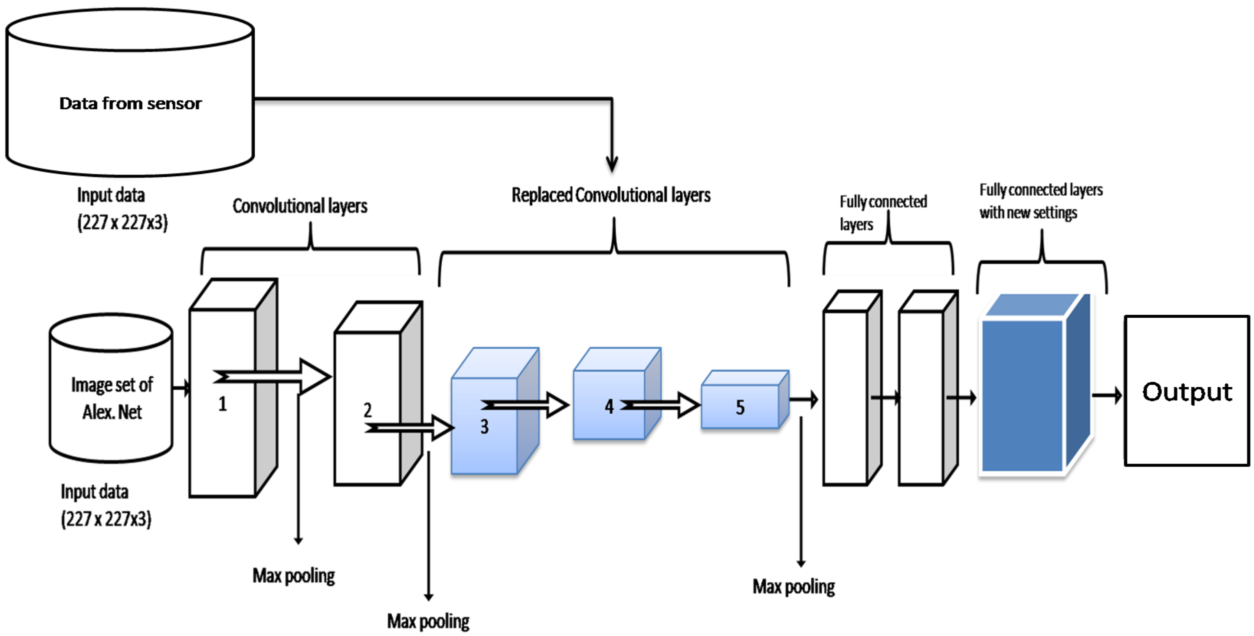

Figure 5: Architectural model of the deep transfer learning algorithm

The architectural model in figure 5 presented the new training algorithm which has the intelligence to recognize and correctly classify over a million objects and hence improve the performance of the robotic system. The pseudo code for the deep transfer learning algorithm is presented below;

1. **Start**
2. ***Activate camera sensors***
3. ***Collect data from environment***
4. ***Load data from sensor***

5. *Load pre trained Network (Alex.net)*
6. *Load CNN Network*
7. *Replace the last three layers of the Alex.net with the CNN three layers*
8. **If**
9. *Weight and bias learning factor = < 10*

    Increase by 20

10. **Else**
11. *Train the network*
12. *Classify images*
13. **End if**
14. **Return**
15. **End**

4.6 Activity Diagram for the proposed System

The activity diagram describes the process flow among multiple objects of a class during the activity processing. They are used in association with the UML modeling methods with a major objective of molding templates for workflow behind the system being developed. It is employed in describing a use case by giving details of a complex algorithm as a require action that is needed to take place and when they are to occur. The activity diagram presenting the logical data flow of the mobile robot is presented in the figure 6;

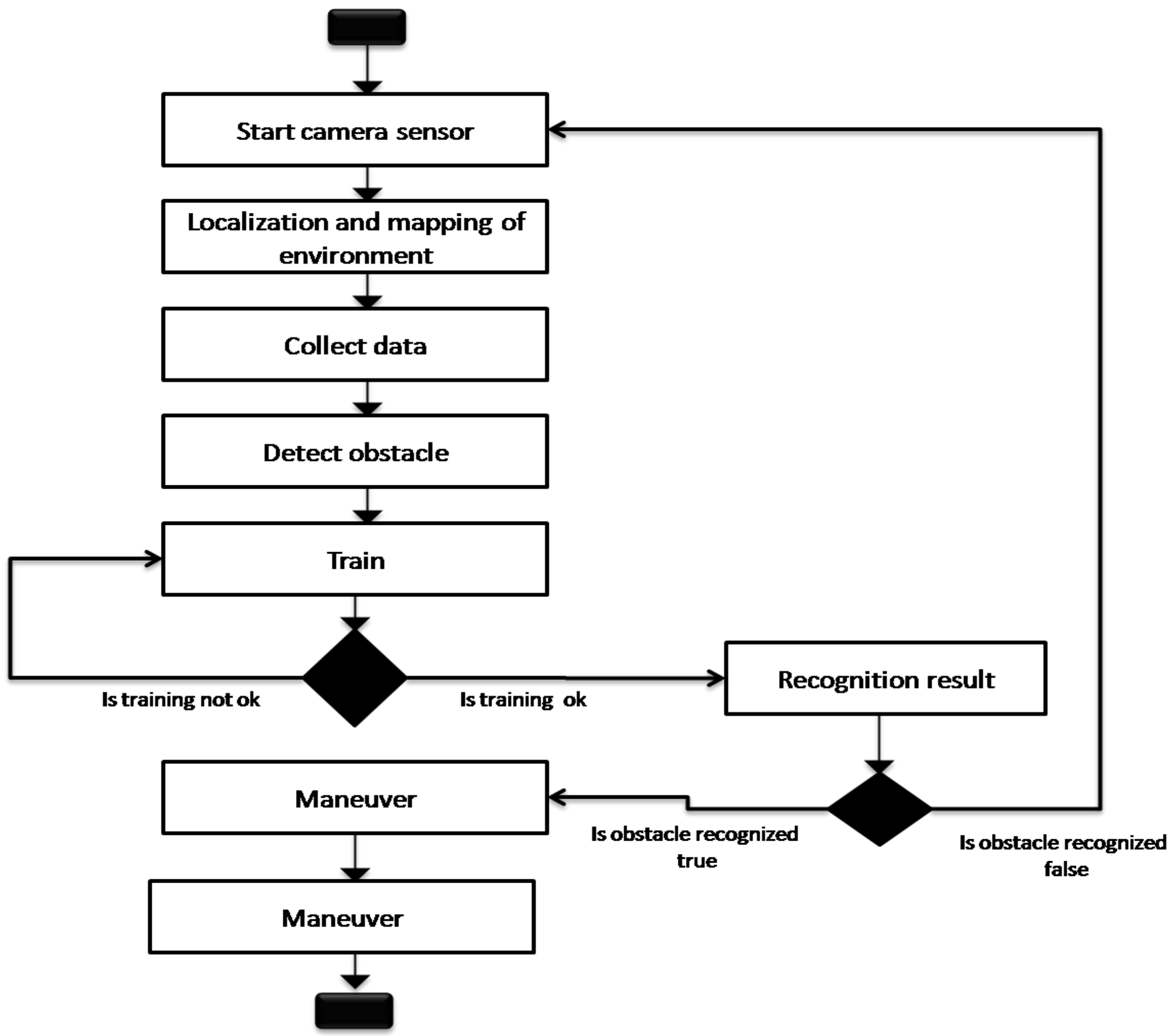

Figure 6: Activity model of the system

The activity diagram in figure 6 presented the performance model of the robot when objects or obstacles are detected. When the robot is active, data are captured via the sensors and feed to the training algorithm for recognition, when the training is not completed, the training is restarted, while when it is completed, the obstacle is labeled and then with the result, the robot maneuvers and continuous navigation.

5. System Implementation

The system was implemented using the models developed, control system toolbox, deep learning toolbox and simulink. The system was implemented using the model of the robot in figure 2 and presented below in the simulink form;

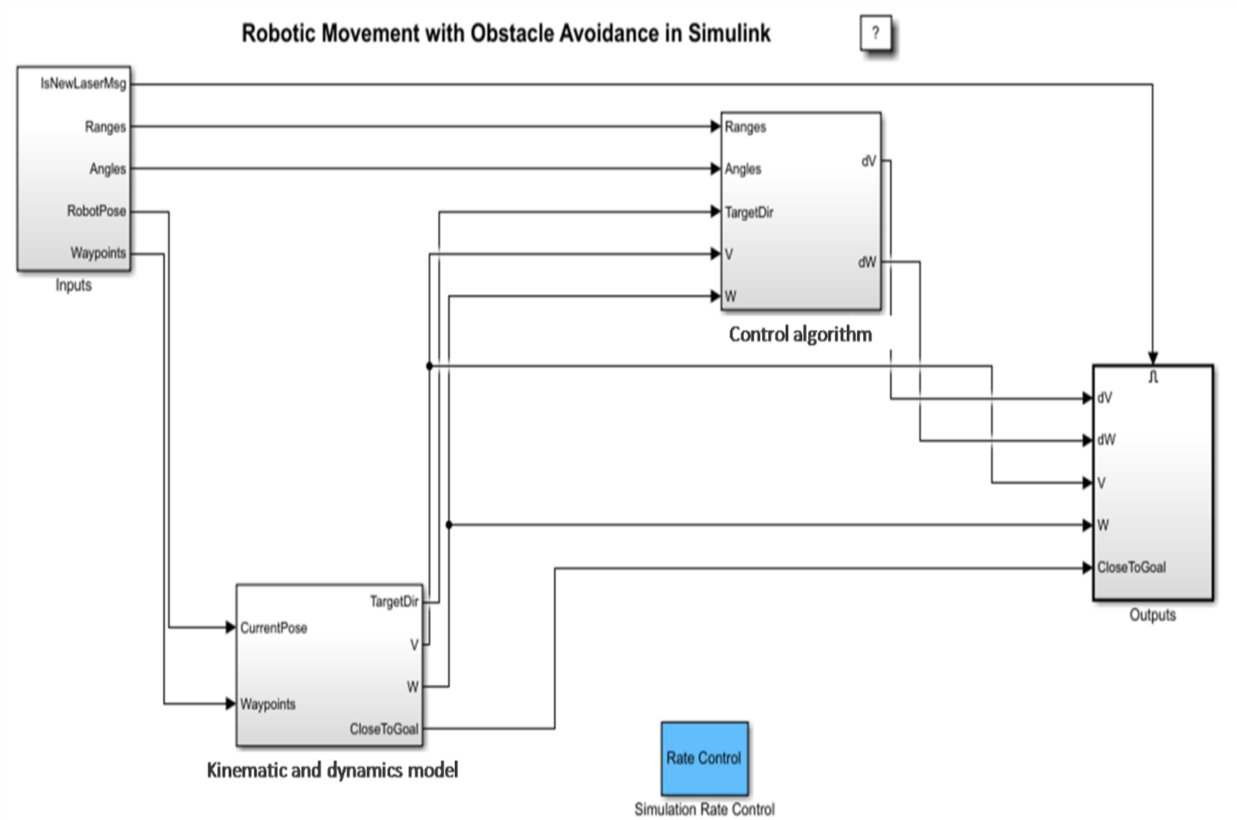

Figure 7: The simulink model of the robot

The simulink in figure 7 presented the relationship between the kinematics, dynamics and control system. The control system was developed using the proposed deep transfer learning algorithm developed. The algorithm was achieved using the transfer learning model in figure 3 and the CNN model in figure 4 to achieve the improved obstacle recognition intelligence. During the training process the deep learning tool automatically splits the feature vectors into test, training and validation sets respectively in a ration of 70:15:15 before training. During this learning process, the filters scan the images for learning feature vectors in each layer of convolution and feed to the other layer to continue until the whole image is complete learned. The training process was performed using the training parameters in table 1 below; the table will be used as a reference point for specifications of values as the result continues.

Table 1: deep leaning training parameters

Parameters	Values
Training epochs	100
Size of hidden layers	10
Iteration per epoch	31
Training segments	30
Neuron weights	681
Filter size	5 x 5 x 3
Image dimension	227 x 227 x 3
Image pixel	154587
Total feature maps	176
Initial feature map	75
Learning rate	0.001
Minimum reference value	-0.7
Maximum reference value	0.7

The training process was performed loading the dataset designed into the deep learning training toolbox, then train and monitor the performance as shown below;

6. RESULTS AND DISCUSSIONS

The overall accuracy of a classifier is estimated by dividing the total correctly classified positives and negatives by the total number of samples in the drowsy dataset. The accuracy is computed by the deep learning tool using the model below;

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad \text{Equation 1}$$

Where TP is true positive rate, TN is true negative rate, FP is false positive rate, FN is false negative rate.

6.1 The Deep Learning Training Result

This section presented the result of the deep learning training performance which was done in the previous chapter using the training parameters in table 1; the result presented the accuracy and loss function of the training process. The aim is to know how well the algorithm developed was able to train and recognize obstacles in the robot propagation path. The training result was measured using the model in equation 1 and presented in figure 8;

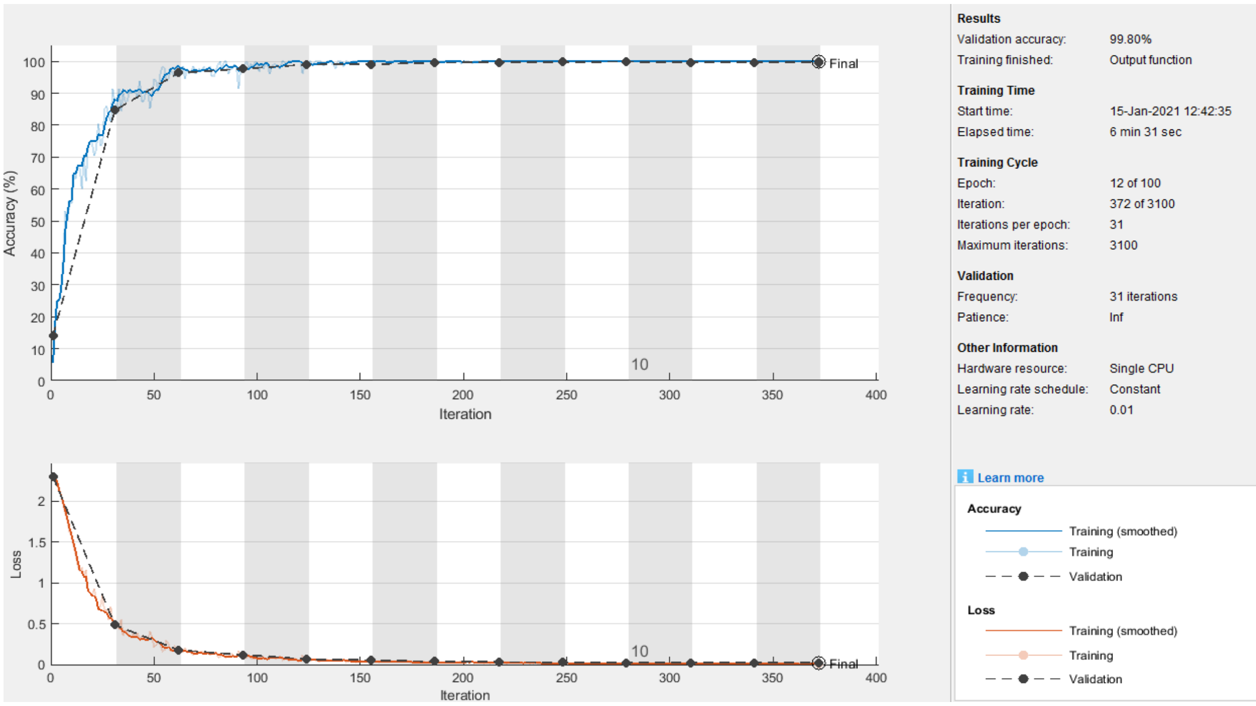

Figure 8: The training result with deep learning tool

The result in figure 8, presents the training performance of the deep learning toolbox. The result was obtained when the Alex.net was feed forward to the CNN architecture as shown in figure 5. During this training process, the test vectors are used to check the check the system performance using a multi class entropy function presented earlier in equation 1 and the result recorded is 99.8%. The implication of this result showed that the deep transfer algorithm was able to train and recognizes obstacles at a very high accuracy of 99.8% which shows reliability and precision in efficiency. The next section presented the validation of the result.

6.2 System Validation

To validate the system developed using the result achieved, tenfold cross validation technique was used which iteratively performance the training, testing and validation process ten times and compute the mean as the standard result based on the model in equation 2 was used;

$$CVA = \frac{1}{10}\sum_{1}^{10} Ai \qquad \text{Equation 2}$$

Where CVA stands for Cross Validation Accuracy, A is the accuracy measure for each fold. The results are generated performing other series of training based on the validation model above and then reported as shown below;

Table 2: Ten-fold validation training results

Iteration number	Accuracy (%)	Loss (%)
1	98.9	1.1
2	98.1	1.9
3	98.1	1.9
4	99.1	0.8

5	98.2	1.8
6	99.0	1.0
7	98.1	1.9
8	99.7	0.3
9	97.9	2.1
10	98.9	1.1
Average	98.7	1.3

From the result presented in the table 2 showing the performance of the algorithm in tenfold, the mean was computed using the model in equation 2 and the result is 98.7%. This accuracy recorded is very good and will be further justified using comparative analysis which considered some of the state of the art algorithm for object detection and recognition as presented in the table 3;

Table 3: comparative results

Author and year	**Technique**	**Accuracy (%)**
Pietro (2020)	Transfer learning	94.70
Xin et al. (2018)	Reinforced guided policy learning	90.00
Ajay et al. (2018)	Clustering algorithm	86.00
Ian et al. (2016)	Deep learning	84.00
Andreas et al. (2015)	Convolutional Neural Network	91.30
Ziyan et al. (2016)	Multi model deep learning	93.80
Ricardo and Herman (2019)	Deep learning based escape route approach	96.81
	New algorithm with Alex.Net	98.70

The table 3 presented a comparative result of the some of the state of the art algorithms alongside the new algorithm developed. The graph in figure 9 presented the comparative result.

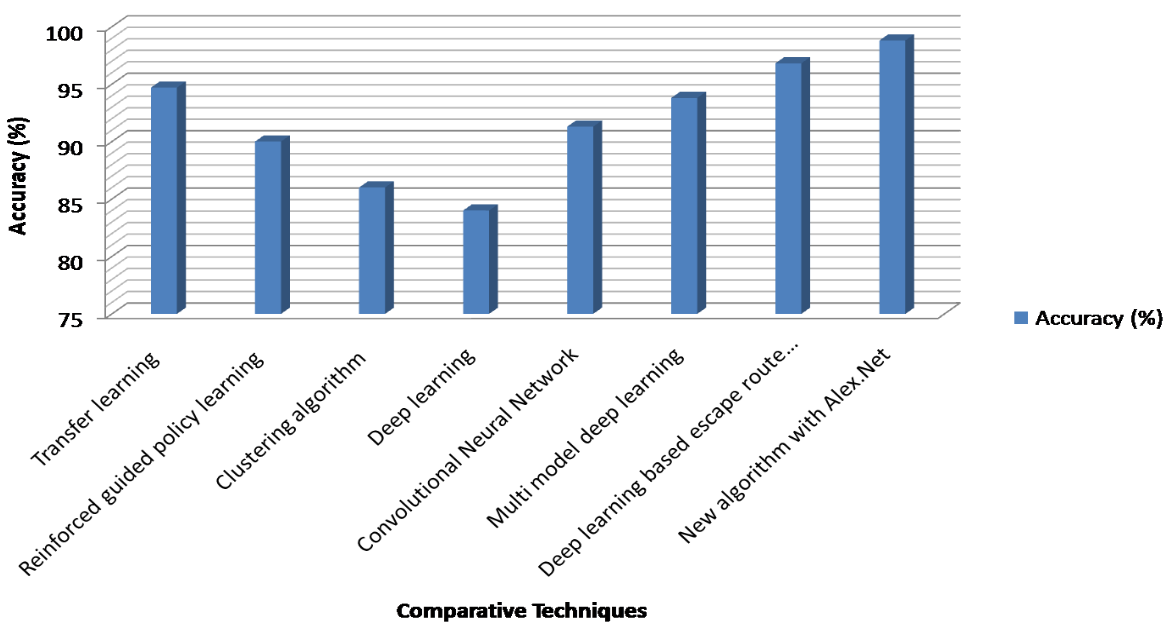

Figure 9: Comparative analysis of techniques

The result in figure 9 presented the comparative analysis of the various selected object detection and recognition algorithm and from the result it was observe that the new algorithm is 1.89% better then the best existing state of the art algorithm in (Ricardo and Herman, 2019).

7. CONTRIBUTION TO KNOWLEDGE

i. A deep transfer learning algorithm was developed to improve SLAM efficiency in mobile robotic systems

ii. Percentage improvement of 1.89% was achieved against the best existing state of the art algorithm

iii. A reliable autonomous holonomic mobile robot was presented

8. CONCLUSION

This research work has successfully presented a holonomic robot with the full potential of artificial intelligence. The literatures reviewed have shown that the major challenge with the existing robotic designs is mainly in their ability to recognize certain objects which always presented itself as obstacle during SLAM and navigation. We believe that since the human brain can recognize everything once seen, there is a possibility for a robot to replicate this effect using transfer learning algorithm. Alex.ne was adopted as a common transfer learning algorithm, trained with over a million objects and then used to develop an improved recognition algorithm with convolutional neural network. The result was tested and then validated with other state of the art algorithms. The result showed that the new system has a percentage improvement of 1.89% ahead of the best achieved so far according to the literatures reviewed.

9. REFERENCE

Ajay Kumar Tanwani, Nitesh Mor, John Kubiatowicz, Joseph E. Gonzalez, Ken Goldberg (2019). Application to Object Recognition and Grasp Planning in Surface Decluttering. arXiv:1903.09589v1 [cs.RO] 22 Mar 2019.

Alex, K., Sergey, L., Peter, P., and Deirdre, Q. (2016). Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection. In ISER, 2016.

Anakwenze U.C (2021) " Development of a video based early fire hazard detection system using computer vision with deep learning" PHD Dissertation; Unizik, Nnamdi Azikiwe University, Awka.

Anderson D. and McNeill G. (1992). Artificial Neural Network Technology. New York, Utica.

Borenstein J. & Koren Y. (1989). Real-time Obstacle Avoidance for Fast Mobile Robots. EEE Transactions on Systems, Man, and Cybernetics, pp. 1179-1187, Vol. 19, No. 5.

Antonio D'Innocente, Fabio Maria Carlucci, MircoColosi, Barbara Caputo (2017). Bridging between Computer and Robot Vision through Data Augmentation: a Case Study on Object Recognition. arXiv:1705.02139v1 [cs.CV] 5 May 2017.

Athira Ramesh1, Archana (2016) Mathematical Modelling and Control of a Mobile Robot for Path Tracking; *International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064: 6.391 Volume 5 Issue 9, September 2016*

Clark R, A. El-Osery, K. Wedeward, and S. Bruder (2014); A Navigation and Obstacle Avoidance Algorithm for Mobile Robots Operating in Unknown, Maze-Type Environments; *Proc. International Test and Evaluation Association Workshop on Modeling and Simulation, Las Cruces, NM, December 2014; Intelligent Systems and Robotics Group (ISRG)*

Clark R, A. El-Osery, K. Wedeward, and S. Bruder (2014); A Navigation and Obstacle Avoidance Algorithm for Mobile Robots Operating in Unknown, Maze-Type Environments; *Proc. International Test and Evaluation Association Workshop on Modeling and Simulation, Las Cruces, NM, December 2014; Intelligent Systems and Robotics Group (ISRG)*

Danica Kragic, Joakim Gustafson, Hakan Karaoguz, Patric Jensfelt and Robert Krug, "Interactive, Collaborative Robots: Challenges and Opportunities" Robotics, Perception and Learning lab, KTH Royal Institute of Technology, Stockholm, Sweden; Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

Eneh P.C, Eneh I.I, Egoigwe S.V and Ebere C.U (2019), "Deep artificial Neural Network Based Obstacle Detection and Avoidance for a Non Holonomic Mobil Robot"; *International Research Journal of Applied Sciences, Engineering and Technology, Vol.5, No.1; ISSN (1573-1405).*

Ian Lenz, Honglak Lee, and Ashutosh Saxena (2014). Deep Learning for Detecting Robotic Grasps. arXiv:1301.3592v6 [cs.LG] 21 Aug 2014.

Ian Lenz, Honglak Lee, and Ashutosh Saxena (2014). Deep Learning for Detecting Robotic Grasps. arXiv:1301.3592v6 [cs.LG] 21

Inyama H.C and Agbaraji C.E (2015) "Survey of controller design methods for a robot manipulator in harsh environment"; European Journal of

Engineering and Technology; Vol 3 No. 3, ISSN 2056 -5860

Jiyong Jin and Woojin Chung (2019); Obstacle Avoidance of Two-Wheel Differential Robots Considering the Uncertainty of Robot Motion on the Basis of Encoder Odometry Information; *School of Mechanical Engineering, Korea University, Seoul 02841, Korea*

Liu Y., D. Chen, S. Zhang,. Obstacle Avoidance Method Based on the Movement Trend of Dynamic Obstacles (2018), *3rd International Conference on Control and Robotics Engineering* pp. 45–50.

Moreno J., Clotet E., Lupianez R., Tresanchez M., Martinez D., Palleja T., Casanovas J. & Palacin J. (2016). Design, Implementation and Validation of the Three-Wheel Holonomic Motion System of the Assistant Personal Robot (APR). Sensors, doi:10.3390/s16101658

Pai N. S., Hui H., & Lai Y. C. (2012). Implementation of Obstacle-Avoidance Control for an Autonomous Omni-Directional Mobile Robot Based on Extension Theory. sensors, pp. 13947-13963, Vol 12.

Patnaik A., Khetarpal K., and Behera L. (2014). Mobile robot navigation using evolving neural controller in unstructured environments. Third International Conference on Advances in Control and Optimization of Dynamical Systems (pp. 758-766, Vol. 13, No. 15). Kanpur, India: Advances in Control and Optimization of Dynamical Systems.

Pietro Falco, Shuang Lu, Ciro Natale, Salvatore Pirozzi, and Dongheui Lee (2020). A Transfer Learning Approach to Cross-modal Object Recognition: from Visual Observation to Robotic Haptic Exploration. arXiv:2001.06673v1 [cs.RO] 18 Jan 2020

Ricardo Buettner and Hermann Baumgartl (2019). A highly effective deep learning-based escape route recognition module for autonomous robots in crisis and emergency situations. Proceedings of the 52nd Hawaii International Conference on System Sciences | 2019. URI: https://hdl.handle.net/10125/59506 ISBN: 978-0-9981331-2-6 (CC BY-NC-ND 4.0).

Seyyed Mohammad Hosseini Rostami, Arun Kumar Sangaiah, Jin Wang & Xiaozhu Liu (2019); Obstacle avoidance of mobile robots using modified artificial potential field algorithm; *EURASIP Journal on Wireless Communications and Networking* volume 2019,

SulabhKumra and Christopher Kanan (2017). Robotic Grasp Detection using Deep Convolutional Neural Networks. arXiv:1611.08036v4 [cs.RO] 21 Jul 2017.

Xin Ye, Zhe Lin, Haoxiang Li, Shibin Zheng, Yezhou Yang (2018). Active Object Perceiver: Recognition-guided Policy Learning for Object Searching on Mobile Robots. arXiv:1807.11174v1 [cs.AI] 30 Jul 2018.

Ziyan Wang, Jiwen Lu, Ruogu Lin, Jianjiang Feng, Jie Zhou (2016). Correlated and Individual Multi-Modal Deep Learning for RGB-D Object Recognition. arXiv:1604.01655v3 [cs.CV] 9 Dec 2016.