Perceptive Parallel Processes Coordinating Geometry and Texture

Marco A. Gutierrez[1], Rafael E. Banchs[2] and Luis F. D'Haro[2]

Abstract— **Finding and classifying specific objects is a key part in most of the tasks autonomous systems could face. Properly being able to reach objects and find their exact location is very important for successfully achieving higher level robotic behaviors. To perform full object detection and recognition tasks in a wide environment several perception approaches need to be brought together to achieve a good performance. In this paper we present a dual parallel system for object finding in wide environments. Our system implements two main parts. One texture based approach for wide scenes, composed by a Multimodal Deep Learning Neural Network and a syntactic distribution based parser. And another specific geometry based process, using three dimensional data and geometry constrains to look for specific objects and their position within a whole scene. Both systems run in parallel and compliment each other to fulfill an object search and locate task. The major contribution of this paper consists on the success of combining texture and geometry based solutions running in parallel and sharing information in real time to allow a full generic solution to be able to find almost any present object in a wide environment. To validate our system we test it with real environment data injected into a simulated environment. We test 25 tasks in a household environment obtaining a 92% overall success rate finally delivering the correct position of the object.**

I. INTRODUCTION

Significant amount of work has been done in scene understanding from 2D images since the beginnings of computer vision research, achieving significant results. Hand-designed features such as SIFT [1], ORB [2] or HOG [3] underpin many of these successful object recognition approaches. They basically capture low-level textured information with the difficulty on effectively capturing mid-level cues (like edge intersections) or high-level representation (like different object parts). Recent developments in deep learning based solutions have shown how hierarchies of features can be learned in an unsupervised manner directly from data. Learned features based solutions proved significant improvements on object recognition and detection, achieving some of them up to around 90% success rates on different benchmark training/testing sets (i.e. The Pascal VOC Challenge [4]). Recently even full semantical well structured image descriptions are generated by the latest multimodal neural language models [5]. Still when using 2D based scene understanding a lot of valuable information about the shape and geometric layout of objects is not considered. Adding geometric information on these solutions could generally improve their results as well as enrich the information they deliver as an output.

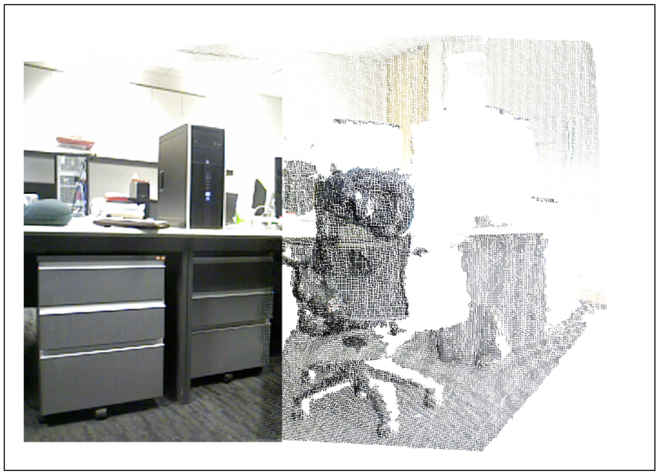


Fig. 1. Our system combines the best of 2D and 3D data information through to coordinated parallel process.

On the other hand, 3D model based approaches make easy to reason about a variety of properties from volumes, 3D distances and local convexities. Solutions focusing on object shapes and geometric characteristics have had also intense computer vision research focus, specially due to the recent new range of inexpensive and fast RGB-D sensors available in the market. 3D features such as FPFH [6] or NARF [7] are some examples of robust features that describe the local geometry around points for 3D point cloud datasets. However 3D solutions have some drawbacks when dealing with heavily clustered scenes or very general views of the environment.

Although good solutions exist on both, image and point cloud based approaches, when it comes to solving tasks in real environments, a more generic approach to achieve a solution for the problem is needed. Systems with a use of both 2D images based solutions and 3D geometry aware processes can provide a more generic purpose robotics architecture with more reliable and rich information. Our approach combines the rich information obtained from new multimodal neural networ object classification techniques on general 2D image scenes with a 3D geometric, distance and shape aware process (figure 1). This allows us to minimize the drawbacks of each of each approach with the strengths of the other.

For the evaluation of our model we used a hybrid simulation-real scenario. A simulator tool was used to man-

*This work was supported by by the A*STAR Research Attachment Programme.

[1] Marco A. Gutiérrez is with the robotics and artificial vision laboratorio (RoboLab), University of Extremadura, Spain marcog@unex.es

[2] Rafel E. Banchs and Luis F. D'Haro are with the HLT dept., I2R, A*STAR, Singapore. rembanchs@i2r.a-star.edu.sg, luisdhe@i2r.a-star.edu.sg

age the robot movements around the environment while sensor data was injected into the system from real scenario captures. This allowed us to test our approach with real environment data, since all perception information used as an input for our application comes from real sensors. As a result we obtain quite promising results on the object finding tasks tested.

The remaining of the paper is organized as follows: in section II we provide an overview of some related works. Following section III gives a detailed general description of the perception system. Section IV and V explain more specific details regarding each of the two main processes, the texture aware process and the geometry aware one respectively. Finally we evaluate the system with an experiment in section VI and give some conclusions and future lines of work in section VII.

II. RELATED WORKS

There is a wide range of research in the area of scene understanding and object recognition from 2D and point cloud data. With RGB-D increased popularity, bringing an easy to access way to RGB and depth data at the same time, several researches have tried combining the two sources of information.

Sensor fusion approaches are the most common ones, they take both sources of data and combine them into one system to improve performance. I.e. in [8] they associate groups of pixels with 3D points into multimodal regions that they call regionlets, then they measure the structure of each regionlet using bottom-up cues from image and range features. This way they are able to determine the scene structure separating it into the meaningful parts discarding the background clutter. Although they do not relay on any rigid assumptions about the scene like we do (we consider objects are placed on tables), the output provides a basic structure discovery over a scene with detection of the main objects while our solution solves a specific object search and locate task on a wider environment.

The machine learning based approaches take features from both depth and color data sources and combine them into one multimodal space to perform later searches for a given input. Koppula et al. [9] perform a labeling task on over-segmented 3D RGBDSLAM sensed scenario. They build a graphical model capturing 2D images information (local color, texture, gradients of interests, etc.) as well as local shape and geometry, and geometrical context (where object most commonly lay to each other). This model then uses approximate inference and is trained using a maximum-margin learning approach. They show the benefits of using image and shape against separated solutions. Also, Lai et al [10] present an RGB-D Object Dataset and evaluate some object recognition and detection techniques. They combine 2D SIFT descriptors with efficient match kernel (EMK) features computed over spin images on randomly subsamples set of 3D points. These features are then used for the evaluation of three classifiers: a linear support vector machine (LinSVM), a gaussian kernel support vector machine (kSVM) and a random forest (RF). The main difference with these works is that they restrict the search to a certain scene while our solution provides a framework to solve a find and locate an object in an entire household environment.

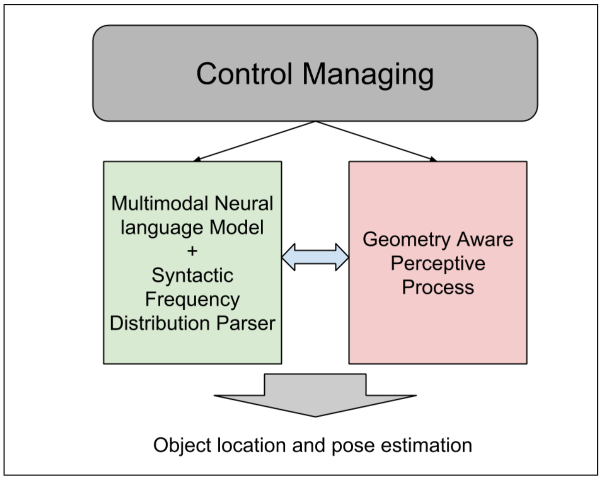

Control Managing
Multimodal Neural language Model + Syntactic Frequency Distribution Parser
Geometry Aware Perceptive Process
Object location and pose estimation

Fig. 2. Overview of the architecture of the perception system.

The work on [11] combine high-resolution 3D laser sans with 2D images to improve object detection. Their solution relies in using a sliding window approach over a combination of visual and depth channels and use those patches to train a classifier. It solves the same problem as the one presented here although they do not perform any optimization in terms of the path to reach the object most probably leading to a slower solution for an object search and location like the one explained here.

Also in [12] they use binary logistic classifiers on 2D and 3D features. The 2D features are small patches selected from images on a training set. They, then, compute 3D features from distance from robot estimation, surface variation and orientation and object dimensions. These features are then learned by the classifier over two-split decision for each object class. The difference with our solution is that they learn multimodal models per object while here the rgb and point cloud data are used by two different process and the outcome combined in a final solution.

III. THE PERCEPTION SYSTEM

As shown in figure 2, the system's architecture has a control manager for decisions and mediation among two perception parallel process. This manager takes care of the information shared between both processes and delivers notifications according to them.

The *texture aware perceptive process* (showed in figure 2 in green) exploits 2D images information data. It runs a multimodal neural language model as described in [13] along with a syntactic frequency distribution based parser to process and evaluate the neural network output. The second one, the *geometry aware perceptive process* is exploiting the geometric features of the environment. This one takes care of two main tasks, looking for tables through the point cloud data and segmenting tabletop setups, recognizing the object

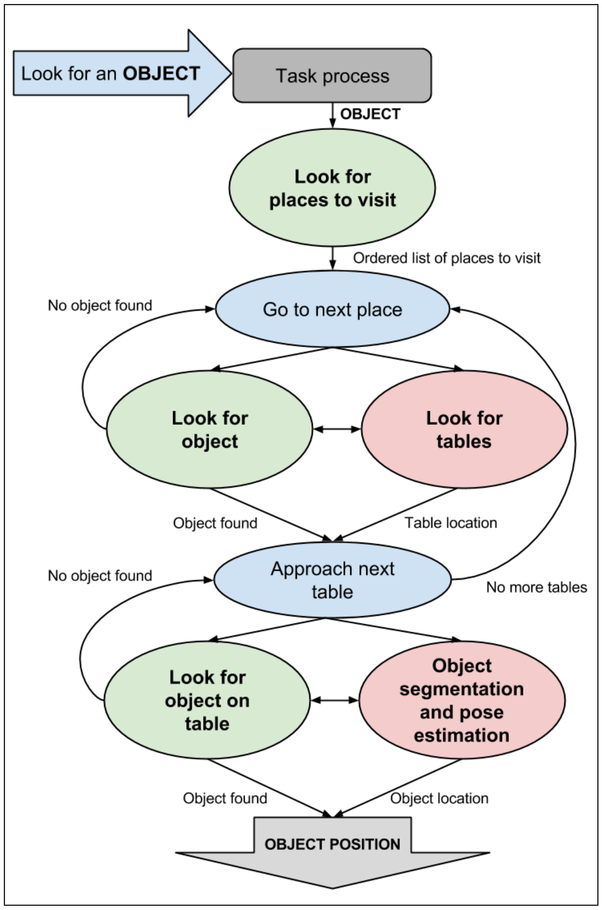


Fig. 3. Flow of the different states and process on the system.

and estimating its position through a shape and position aware histogram based feature matching system.

Figure 3 describes the states and process of the system for a certain object search and locate task. Green tasks are performed by the *texture aware perceptive process* while the ones in red belong to the *geometry aware perceptive process*. A restriction the system assumes is that objects are placed on tables. A simple task is given to the robot in the form of “Look for the OBJECT”, and the object name is extracted and passed to the *texture aware perceptive process* for the “look for places to visit” step. A list of generic images for each available place is stored in our database and evaluated by this perceptive process. A frequency of appearance of possible objects histogram is built for each place. Places to visit are then ordered according to highest appearance of label of the object on this output. Places with no object appearances are left to visit last and ordered randomly.

Once the list of places to visit is ready, the robot visits them in order. When the first place is reached both processes start to work in parallel for the required object. The *texture aware perceptive process* provides a frequency distribution of objects on images taken from the current place while the geometry aware one will start looking for tables on the scene point cloud data. If the object is found in a scene image and a table has been detected the robot will start moving towards the table. Once a table is reached the *texture aware perceptive process* keeps validating the appearance of this object in the scene, then a tabletop segmentation process will be started by the *geometry aware perceptive process* in order to segment, recognize and locate the object.

If no object seems to be present when the tabletop segmentation is performed, the robot continues with the next table or with the next place in list if no more tables are available in the current place. We will only conclude that we cannot find an object once all places have been visited and no object has been found.

IV. TEXTURE AWARE PERCEPTIVE PROCESS

This texture based perceptive process is intended to get quick scene labeling from wide overviews of the environment. It contains a previously trained multimodal neural model that outputs image descriptions. Then, taking into account the top nearest descriptions in the model, a parser extracts the object candidates and builds a frequency distribution histogram on the appearances of these objects class names. This frequency distribution histogram helps obtain a more robust output against false positives as the objects that are present in the scene tend to keep appearing with higher frequency over time in the sentences while the false positives have usually a much lower frequency.

A. Mulitmodal neural model

As previously mentioned the multimodal neural model follows the structure in [13]. This is a neural model pipeline that learns multimodal representations of images and text. The pipeline uses a long short-term memory [14] (LSTM) recurrent neural network for encoding sentences. We use a convolutional network architecture provided by the Toronto Convnet [15] in order to extract 4096 dimensional image features for the neural model. These image features are then projected into the embedding space of the LSTM hidden states. A pairwise ranking loss is minimized in order to learn to rank images and their descriptions. For decoding, the structure-content neural language model (SC-NLM) disentangles the structure of a sentence to its content, conditioned on distributed representations produced by the encoder. Finally, the output is generated by sampling from the SC-NLM the image top descriptions.

B. Syntactic frequency distribution parser

After the system obtains the top scenes generated descriptions, it extracts potential object classes from them, using a syntactic parser. Using the Neural Language Toolkit [16] we syntactically analyze the sentences to extract object candidates that could be present in the image. A frequency distribution histogram is computed over this object candidates. This histogram is then used to evaluate the believe that an object is present in a scene, allowing us to compare different scenes according to the probability of finding an object there and therefore discriminate possible false positives.

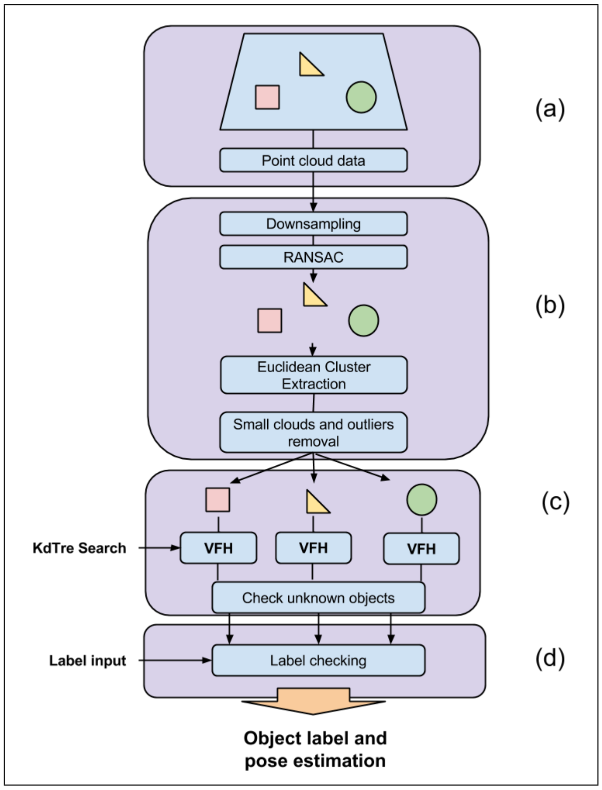


Fig. 4. Tabletop segmentation and object recognition pipeline using point cloud data.

V. GEOMETRY AWARE PERCEPTIVE PROCESS

This process exploits the geometry present in the environment to extract a wide variety of information. For our approach we have restricted the task of finding objects, to objects placed on top of tables. Therefore this process performs two main tasks, one is looking for tables in broad scenes and another one consists on a tabletop segmentation with shape based object recognition and pose estimation.

A. Looking for tables

We describe tables as planes that are parallel to the floor and found at a height between 40 and 110 centimeters. Therefore, we use the RANdom SAmple Consensus (RANSAC) [17] for plane model fitting in the scene point cloud data with a previous downsample of 1cm. Using this algorithms we recursively look for planes matching the previously mentioned constrains and label them as tables.

B. Object recognition and pose estimation

The tabletop segmentation is used when a table is approached and in order to recognize the objects on top of it as well as to estimate their final position.

In the first part, shown in figure 4.b, the RANSAC algorithm provides us with the plane equation and the points that match that equation. Since the RANSAC uses a threshold to deal with sensor noise, points matching the model are not in a perfect plane but within a certain range, so we first project this points to fit the plane equation to obtain a perfect plane point cloud. Then we obtain the convex hull of these plane point cloud and perform a bounding box on top of it up to a certain high. Points within the bounding box are then considered to correspond to objects sitting on top the table. Then it is performed an euclidean clustering extraction to segment the object candidates point clouds.

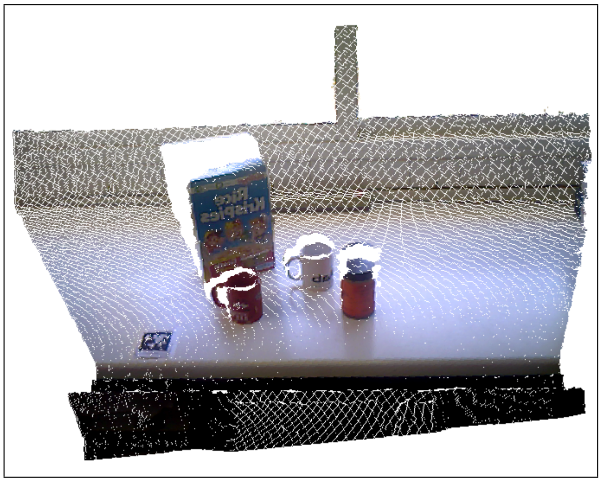


Fig. 5. Example of one of the tabletop setup used in the experiment.

As the next step (figure 4.c) we compute these point clouds Viewpoint Feature Histograms [18] (VFH) and look for the nearest match in our database. For this database we have a previously computed VFH of single views of objects. These VFHs are stored and retrieved through fast approximate K-Nearest Neighbors (KNN) searches using kd-trees [19]. The construction of the tree and the search of the nearest neighbors places an equal weight on each histogram bin in the VFH and spin images features.

Finally the system would check if any of the labels from the objects correspond to the one we are looking for, see figure 4.d, and call it a success or not.

VI. EXPERIMENT

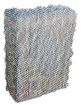
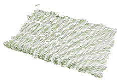
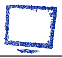
We perform several experiments sending the robot to retrieve different objects in a wide household environment. For the experiment an hybrid simulator-real data environment has been used. We used the simulator for the robot movements between places, while sensor data has been acquired with real RGB and RGB-D cameras (i.e. the tabletop showed in figure 5) and matched to the specific locations on the virtual plane. When the robot needs to move around the simulator takes care of it, once certain positions in the map are reached, the previously obtained real data is injected and used as input for the algorithms. The robot always starts at the entrance of the apartment and from there performs the most optimal way to find the object and delivers its estimated position as a final result.

A. System setup

The LSTM encoder and SC-NLM decoder from the multimodal neural model have been trained using a combination of the Flikr30k [20] dataset and the Microsoft COCO

TABLE I

SUCCESS RATES ON THE DIFFERENT PARTS OF THE ALGORITHM

OBJECT	1. Places to visit Ordering	2. False Negative	3. False Positive	4. Success Rate
Cereal box 	5	0	0	100%
Cup	5	0	1	80%
Bottle	5	0	1	80%
Laptop 	5	0	0	100%
Monitor 	5	0	0	100%
Overall rate	**100%**	**0%**	**8%**	**92%**

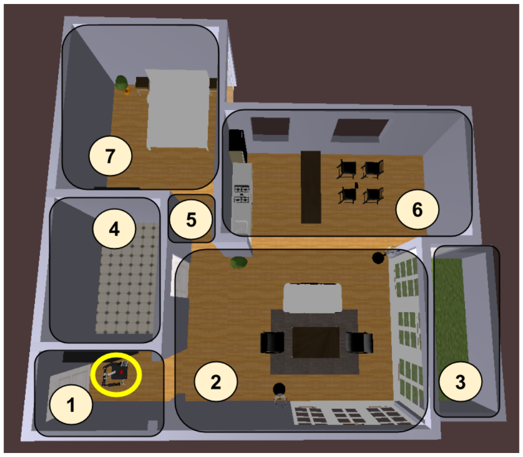


Fig. 6. An overview of the simulation household environment. The rooms are labeled as follows: 1.- Entrance, 2.- Living room, 3.- Patio, 4.- Bathroom 5.- Hallway, 6.- Kitchen, 7.- Bedroom. Circled in yellow is the robot at its starting point.

dataset [21]. The 4096 dimensional image features for the multimodal neural model training are extracted using the Toronto Convnet with their provided models. The frequency histogram is built using the NLTK toolbox on the top 5 generated sentences over at least 5 frames, to achieve a robustness on the objects observed. This NLTK tagging and syntactic analysis is performed using the Treebank Part of Speech Tagger (Maximum entropy) they have available. For the rooms representation, images in the house 5 generic different images of parts of a house are used for each of the places in the house: entrance, room, kitchen, living room, bathroom, patio and bedroom. This images have been selected so they contain the usual set of items presents in those rooms. For the point cloud analysis a kd-tree stores 3729 VFH from different views of 75 different objects.

All the system is developed using the RoboComp robotics framework [22] and the simulation is performed in a virtual scenario using the RoboComp simulator tool. See figure 6 for an overview of the simulation environment.

B. *Results on the experiments*

We run 5 different tasks 5 times and collect the results in the table I. First we measure if the ordering of places to visit after the “Look for places to visit” step in our system was optimal (check figure 3 for details). This turned out to work perfect for all of our test cases, basically because some of the description pictures of the places contained those items and the *texture aware perceptive process* was able to detect them. It is important for this step to select a good range of images representing the different places to visit (see figure 6), specially those images that clearly show an average of the objects you can usually find in those places.

Then we count the false negatives occurrences, this is when we are done with the searching and no object was found. Along our testing this never happened and an object was always found. However we obtained two false positives when the system mistaken a cup for a bottle and when a bottle was mistaken for a bottle of glue. Those mistakes are basically due to the similarity on these objects shape. We could avoid this in the future reinforcing this step with other object features. Specially since the objects to be found where actually present in the table being segmented at the time.

The final success rate on obtaining the proper location of the object and pose estimation is quite high which results promising for further real applications of the system.

VII. CONLUSIONS AND FUTURE WORK

We presented a hybrid perception system that combines 2D data based solutions and approaches using point clouds running in parallel and sharing information in real time in order to achieve a finding object task. The system is able to successfully predict a route through the places with higher probability of finding this objects. We obtained a high rate of success in our experiments as we only obtained two false positives among all our test cases.

An interesting future work would be to perform further testings with a wider range of objects. This could help find some weak points on the system that we might have not found yet and that should be worth to strength with more processes interaction. In the same line and although the sensor data used in the testing where taken from real sensors, integrating the solution with a real robot could bring a more accurate overview of how the system performs in real environments.

False positives obtained during experiments are mainly because of a bad performance of the *geometry aware perceptive process*. Since similarity on the shape of different objects confuses the VFH search, exploiting texture based features on this last step could most probably benefit the whole system final output. Also, since we are using an euclidean clustering extraction method for objects on top of the table, our system cannot deal with heavy cluttered scenes or objects touching each other. Adding alternatives to the segmentation process could help improve this in order to cover a more varied range of scenarios. It would be also desirable to avoid the assumption that objects are always on tables, so we should look into new ways of scene segmentation to improve this step.

Finally adding a learning process in the system would be an interesting enhancement, both parallel process could complement each other, correcting each other mistakes and providing the fixed mistake as a new source of learning, leading to improvements in the following overall system performances.

REFERENCES

[1] D.G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2, 1999.

[2] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2564–2571, Washington, DC, USA, 2011. IEEE Computer Society.

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision & Pattern Recognition*, volume 2, pages 886–893, INRIA Rhône-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334, June 2005.

[4] Mark Everingham, S.M.Ali Eslami, Luc Van Gool, ChristopherK.I. Williams, John Winn, and Andrew Zisserman. The Pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2015.

[5] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. *CoRR*, abs/1411.4555, 2014.

[6] R.B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 3212–3217, May 2009.

[7] Bastian Steder, Radu Bogdan, Rusu Kurt, and Konolige Wolfram Burgard. Narf: 3d range image features for object recognition. In *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics*, Int. Conf. on Intelligent Robots and Systems, IROS '11. IEEE Computer Society, 2010.

[8] Alvaro Collet, Siddhartha S. Srinivasa, and Martial Hebert. Structure discovery in multi-modal data: A region-based approach. In *ICRA*, pages 5695–5702. IEEE, 2011.

[9] Hema S Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *Advances in Neural Information Processing Systems*, pages 244–252, 2011.

[10] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.

[11] M. Quigley, Siddharth Batra, S. Gould, E. Klingbeil, Quoc Le, Ashley Wellman, and A.Y. Ng. High-accuracy 3d sensing for mobile manipulation: Improving object detection and door opening. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2816–2822, May 2009.

[12] Stephen Gould, Paul Baumstarck, Morgan Quigley, Andrew Y. Ng, and Daphne Koller. Integrating Visual and Range Data for Robotic Object Detection. In *ECCV workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (M2SFA2)*, 2008.

[13] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, abs/1411.2539, 2014.

[14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[15] Toronto University. Convolutional Neural Nets. `https://torontodeeplearning.github.io/convnet/`, 2015. [Online; accessed 04-March-2015].

[16] Steven Bird. Nltk: The natural language toolkit. In *Proceedings of the COLING/ACL on Interactive Presentation Sessions*, COLING-ACL '06, pages 69–72, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[17] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[18] R.B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2155–2162, Oct 2010.

[19] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications*, pages 331–340, 2009.

[20] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[22] P. Bustos Marco A. Gutiérrez, A. Romero-Garcés and J. Mart ńez. Progress in robocomp. *Journal of Physical Agents*, 7(1), 2013.