AI Sensing for Robotics using Deep Learning based Visual and Language Modeling

Yuvaram Singh
HCL Technologies Limited,
Analytics CoE
Noida, India, 201304
yuvaramsingh94@gmail.com

Kameshwar Rao JV
HCL Technologies Limited,
Analytics CoE
Noida, India, 201304
kameshjvkr@gmail.com

Abstract

An artificial intelligence(AI) system should be capable of processing the sensory inputs to extract both task-specific and general information about its environment. However, most of the existing algorithms extract only task specific information. In this work, an innovative approach to address the problem of processing visual sensory data is presented by utilizing convolutional neural network (CNN). It recognizes and represents the physical and semantic nature of the surrounding in both human readable and machine processable format. This work utilizes the image captioning model to capture the semantics of the input image and a modular design to generate a probability distribution for semantic topics. It gives any autonomous system the ability to process visual information in a human-like way and generates more insights which are hardly possible with a conventional algorithm. Here a model and data collection method are proposed.

1 Introduction

In a world gifted with visible light facilitating information sharing, the living creatures have developed organs for sensing the light to understand their surrounding. In an autonomous system, this information is captured in IR, UV, and visible spectrum involving sophisticated sensors and is processed using complex algorithms. At Consumer Electronics Show (CES) 2020, Samsung presented Ballie which is a personalized robot with ideas to make it self-aware of its surroundings and control IoT devices around home to make the environment better. With companies targeting to launch smart home robots with capabilities of following voice commands, there is a need to develop a system that can automatically understand the semantics of the environment and take appropriate decisions on its own.

(a) a person cutting cake while others cheering. (b) Fire fighters are trying to put out fire in the building.

Figure 1: Variety of scenarios that a human can describe comfortably.

The latest work in scene understanding involved construction of knowledge graph for visual semantic understanding(Jiang et al., 2019). The authors used ontology graph in combination with visual captioning to describe the scene. Another approach for functional scene understanding was introduced using semantic segmentation(Wald et al., 2018). All these scene understanding approaches make a system specialized in certain tasks and working environment while failing to generalize across various types of situations and capture the human emotions.

It is efficient to make decision, based on a structured description of the scene instead of working on raw pixel information. Fig.1 shows scenarios where a human can easily interpret the meaning of the scene. It is easy to tell from the Fig.1b that firefighters are trying to put out the fire from building. This is also true for all the Fig.1a, 1b where a human can understand and explain the scene easily through a language representation.

In this work we recommend an AI sensing system that can semantically interpret the environmental conditions, objects, relations and activity carried out from the visual feed. These interpretations are converted into text for human understanding and probability distribution for the control system to process and take decisions. The main intention of this work is to have a neural network based sen-

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 60–63
Seattle, USA, July5 - 10, 2020.
https://doi.org/10.18653/v1/P17

sor processing unit capable of extracting semantic context while deployed on low powered compute hardware.

This paper is divided as follows:

- Section 2 explains various modules of the proposed approach.
- Section 3 discusses about the dataset and considerations to make while implementing this approach.

2 Proposed Approach

In this work, a modular approach is proposed to represent the semantic content of the outside world through vision sensor. A detailed flow diagram of the proposed method is shown in Fig.2. It consist of three sub-modules namely, CNN feature extractor, language module, and environment context probability detector module. It combines visual, language, and context detection modules to assist the control unit to make decisions based on non-task specific environment details.

2.1 CNN Feature Extractor

This module process the visual feed and convert them into feature tensor($\boldsymbol{f}$) which is used to generate semantic understanding of the surrounding. This feature tensor($\boldsymbol{f}$) encodes the information present in the incoming frame. A CNN based feature extractor(Xu et al., 2015) trained for image classification task on Imagenet dataset(Deng et al., 2009) is used. There are variety of CNN based pre-trained architectures are available to be used as feature extractors. Architectures such as Mobilenet(Sandler et al., 2018), ResNet(He et al., 2016), InceptionNet(Szegedy et al., 2015) and DenseNet(Huang et al., 2017) have their own benefits and drawbacks. Based on the deployment hardware, expected response time and environment nature, specific architecture can be chosen.

2.2 Language Module

In this module, the information from the feature tensor($\boldsymbol{f}$) are extracted and represented in a human interpretable language($\boldsymbol{l}$). This is achieved by using Long Short Term Memory unit(LSTM)(Sak et al., 2014) which is a deep neural network(DNN) for generating sequential output(Xu et al., 2015). A combination of soft-attention mechanism(Xu et al., 2015) and LSTM is used to describe the contents extracted from the frame(Vinodababu, 2018). This is a recursive step where the execution comes to a halt when the end token $< end >$ is predicted or maximum sentence length is reached.

$$\boldsymbol{l} = \{w_0, w_1, w_2, ...w_n\}$$

where

$$w_i \in R^k$$

Here R^k is the vector of tokenized words in the vocabulary and ($\boldsymbol{l}$) is the generated word sequence. The byproduct of having language representation is explainability of action.

The process of caption generation happens recursively were to sample a word w from R^k it goes through the following process. Ref Fig.3.

At a time step t,

- The attention mechanism computes the mask m_t for feature tensor$\boldsymbol{f}$ using$\boldsymbol{f}$ and hidden state H_{t-1}.
- $\boldsymbol{f}$ weighted by m_t combined with the previous word detected w_{t-1} is passed onto the LSTM along with hidden state H_{t-1} and cell state C_{t-1} from the previous step.
- The LSTM output a probability distribution for the words in the vocabulary R.

This process is carried out until the end token $< end >$ is predicted or the max length of caption is reached. The effectiveness of this module depends on generation of dense caption for the scene.

2.3 Environment Context Detector

The verbal representation from language module is used to generate probability distribution over various groups of semantic context. The input sequence is tokanized, vectorized and converted into probability distribution by using fully connected network. It is constructed by single or multiple neural net operating parallel, perform prediction over various context. Fig.4 provides the overall view of this module where different fully connected network(FCN) are used for prediction. The caption are tokanized and vectorized to act as input. Here GloVe embedding(Pennington et al., 2014) is used to vectorize the sentence. The activation of the output layer can use either softmax or sigmoid based on the nature of the data. The topics of the context should be decided based on the workspace and

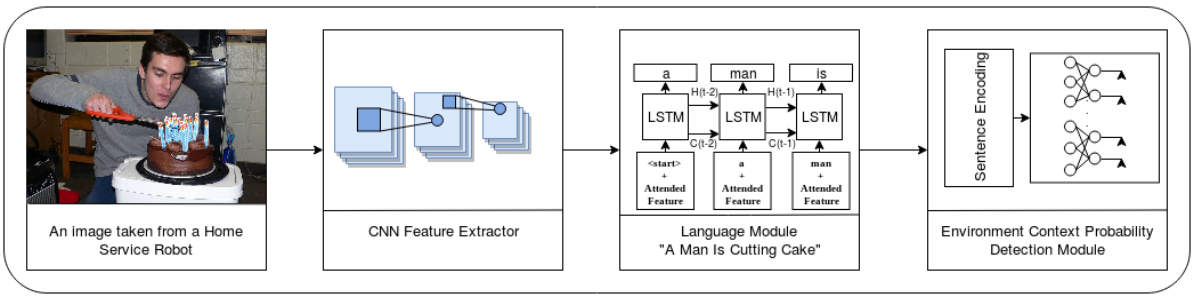

Figure 2: A block diagram of the proposed model.

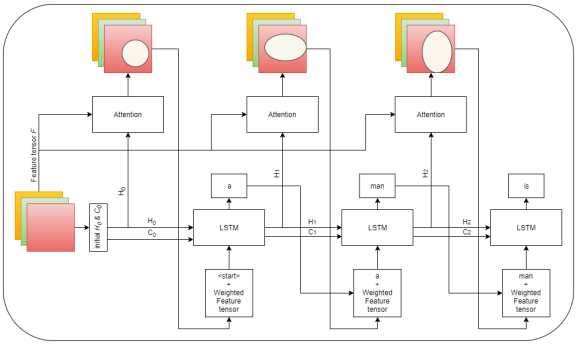

Figure 3: A flow diagram explaining how language module prediction the next word in sequence.(Xu et al., 2015)

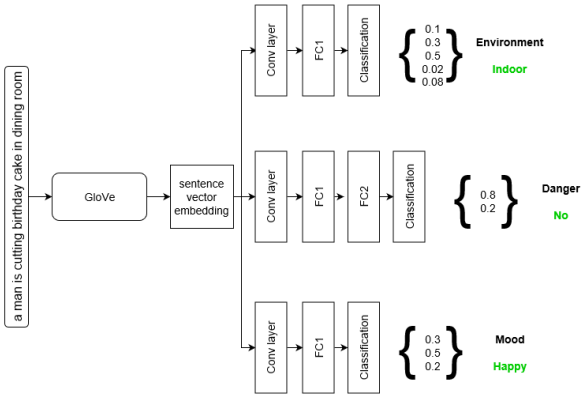

Figure 4: A block diagram of environment context detector.

preference of the robotics designer. Fig.4, shows environment context detector block diagram.

$$\mathbf{E} = [c_0, c_1, ...c_d]$$

where c_i is the prediction vector of i^{th} context net and $\mathbf{E}$ is the collection of c vectors. Here d is the desired number of context net. The generated probability distribution is sent to the control system which takes the final decision whether to react or not. The proposed solution serves as an add-on to the existing control system.

3 Dataset and Considerations

The CNN feature extractor is a pre-trained model trained on Imagenet dataset(Deng et al., 2009) for classifying 1000 objects. The language module is trained using COCO image captioning dataset(Lin et al., 2014) which consist of image and captions in target language. A BLEU-1 score of 70.7 is achieved for the language module.

The dataset for the environment context module is similar to the text sentiment classification dataset. The input will be a sentence and the labels are one-hot vector of target class. A dataset is created from a portion of COCO caption where the semantic context topics are environment, situation, mood, presence of human, and objects in the scene as shown in Fig.4. There are several logical considerations to be take while adopting this method. few of them are,

- On board compute capability to carryout DNN calculation.
- Robot deployment environment and its nature.
- The actual intention and task of the robot.
- How the control system should react to the generated probability distribution.

4 Conclusion

The main objective of the work is to use neural networks to understand and represent the physical environment around the system. This work serve as an add-on to the existing control system by providing additional set of inputs capturing the semantic meaning. An image captioning based approach is used to obtain semantic content of the surrounding and it is represented in a probability distribution.

References

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

Chen Jiang, Steven Lu, and Martin Jagersand. 2019. Constructing dynamic knowledge graph for visual semantic understanding and applications in autonomous robotics. *arXiv preprint arXiv:1909.07459*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Haşim Sak, Andrew Senior, and Françoise Beaufays. 2014. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth annual conference of the international speech communication association*.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Sagar Vinodababu. 2018. a-pytorch-tutorial-to-image-captioning. https://github.com/sgrvinod/a-PyTorch-Tutorial-to-Image-Captioning.

Johanna Wald, Keisuke Tateno, Jürgen Sturm, Nassir Navab, and Federico Tombari. 2018. Real-time fully incremental scene understanding on mobile platforms. *IEEE Robotics and Automation Letters*, 3(4):3402–3409.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.