Engineering

Taste of Research Summer Scholarship Program

Machine Learning: RoboCup Soccer Ball Detection

Vintony Padmadiredja

Prof. Claude Sammut

Intelligent & Autonomous Systems

Background

Each year, the rules for RoboCup Standard Platform League (SPL) [1] changes to further push research in robotics and artificial intelligence to enable robotic soccer to become more similar to human soccer.
In 2016, the solid orange ball (Fig.1) was replaced with a standard black and white soccer ball (Fig. 2) so determining a ball can no longer be done by looking for orange pixels on an image.
Currently, UNSW RoboCup Soccer team's ball detection is done by finding regions with significant amounts of white which then goes through a series of heuristic checks to consider whether the region contains a ball. However, this algorithm isn't very reliable.

The NAO v5 robots contains an Intel Atom Z530 @ 1.6GHz with Integrated Graphics, 1GB RAM and 2GB of Memory.
It would be a challenge to run a Neural Network with these specification in real time with acceptable performance.

Figure 1: Orange Soccer Ball

Figure 2: Standard Soccer Ball

Objectives

- Train a Convolutional Neural Network (CNN) to detect soccer balls
- Optimise the model generated to perform its detection within milliseconds
- Use network within NAO robots for ball detection in RoboCup SPL competitions

Method

1. Obtain 2000~ images of soccer ball on the field for both top and bottom camera under various conditions (lighting, objects, field lines, angles).
2. Automatically label images with bounding boxes using existing CNN and manually hand label (Fig. 3) those which were not automated

Figure 3: Hand labelling using LabelImg[10]

3. Train a network using labelled images and experiment with a number of different network configurations until desired speed and accuracy is achieved
4. Implement network to NAO robots

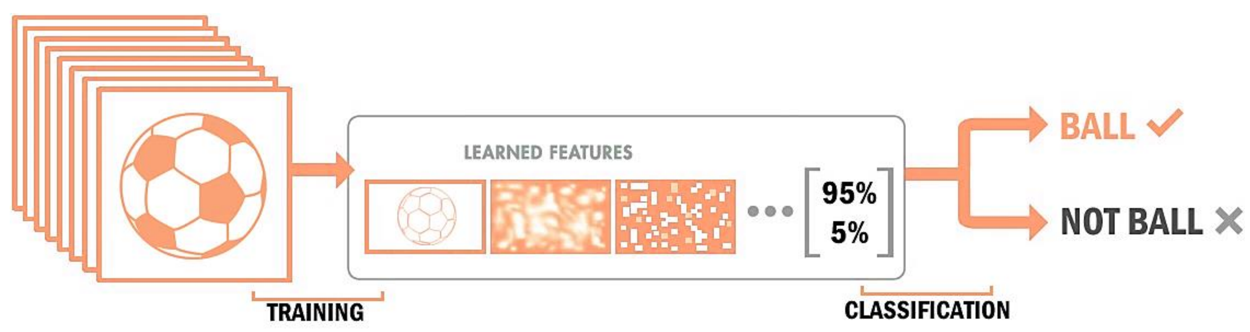

Figure 4: Simple CNN diagram

Results

Figure 5: Speed comparisons of different implementations

Network	Framework	Speed (ms) ± 5 ms	Accuracy (%)	References
MobileNet	Tensorflow	110	95.13	[2]
MobileNet	Caffe	100	99.67	[4], [5]
Tiny-Darknet	Darknet	45	71.50	[3]
Modified-Darknet	Darknet	30	67.42	[3]
Faster R-CNN	Caffe	2000	?	[6]
SqueezeNet	Caffe	15	99.96	[7], [5]
Modified-SqueezeNet	Caffe	8	86.68	[7], [5]
Modified-MiniSqueezeNet	Caffe	6	57.96	[9], [5]

Discussion

- Modified-SqueezeNet performs the fastest with not a lot of accuracy lost
- Darknet is relatively fast however significant amount of accuracy lost
- MobileNet is accurate but the run time is would not be acceptable
- Performance was calculated on a Intel i7-5820 @ 3.30 GHz CPU thus speed is expected to be slower on the Intel Atom within the NAO Robots
- Many other processes would be running concurrently in real time on the NAO Robots which is another factor which would contribute to speed

Figure 6: Ball detection on trained Tiny-Darknet

- Data augmentation is used during training such as brightness, hue, saturation, cropping to allow the network to also learn from synthetic data
- Accuracy is evaluated by testing a random sample from the dataset and how well the network performs on it
- XNOR-Net[8] is a network which uses binarized models and computations which is expected to speed up performance up to 58x. However due to limited documentation and resources, implementation is difficult

Conclusion

- Modified-SqueezeNet is currently the best solution to implement Neural Networks within the NAO as it doesn't sacrifice much accuracy and runs relatively fast
- For further improvement, research and experimentation on XNOR-Net could be pursued

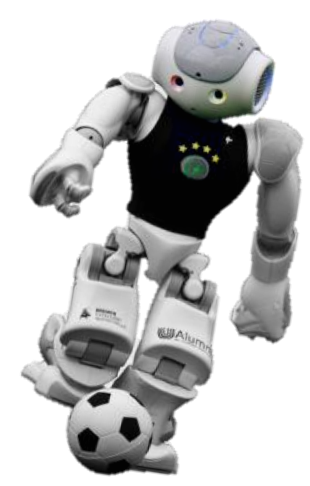

Figure 7: NAO kicking ball[1]

References

1. Kitano, H, Veloso, M, Fujita, M, Asada, M, Duhau, D: RoboCup Standard Platform League; (1998-2018)
2. Abadi, M., Agarwal, A., et al.: TensorFlow: Large-scale machine learning on heterogeneous systems. http://tensorflow.org/ (2015)
3. Redmon, J: Darknet: Open Source Neural Networks in C. http://pjreddie.com/darknet/ (2013-2016)
4. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding; arXiv:1408.5093 (2014)
5. Liu, Wei, Anguelov, Dragomir, Erhan, Dumitru, Szegedy, Christian, Reed, Scott, Fu, Cheng-Yang, Berg, Alexander C.: SSD: Single Shot MultiBox Detector; ECCV; (2016)
6. Ren, S., He, K., Girshick, R., Sun, S.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks; Advances in Neural Information Processing Systems; (2015)
7. Iandola, F., Han, S., Moskewicz, M., Ashraf, K., Dally, W., Keutzer, K.: SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5MB model size; arXiv:1602.07360; (2016)
8. Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A.:XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks; ECCV; (2016)
9. Cruz, N., Lobos-Tsunekawa, K., Ruiz-del-Solar, J.: Using Convolutional Neural Networks in Robots with Limited Computational Resources: Detecting NAO Robots while Playing Soccer; arXiv:1706.06702; (2017)
10. Tzutalin: LabelImg; Git code; https://github.com/tzutalin/labelImg (2015);

Acknowledgements

I thank Professor Claude Sammut, UNSW RoboCup Soccer rUNSWift Team and RoboCup Home Team for their assistance and support throughout this research and UNSW Faculty of Engineering for scholarship