Transfer Learning for Object Detection

A Deep Neural Network Approach

High Performance Computing Lab, IIT Gandhinagar

SRIP 2017

ABSTRACT

Modern day Computer Vision systems require a large amount of data to perform object detection and classification tasks. Since neural networks constitute a key element of these learning systems, it has become increasingly necessary to increase the dataset capacity to utilize these networks to their full potential. With the development of more complex and efficient computing resources, this data intensive task has been successfully parallelized to achieve very low training times. In such scenarios, the availability of a big enough dataset with both high diversity as well as generalization capability has become a bottleneck to the training process. Annotation and retrieval of such a dataset is a very time consuming and tedious task. Hence generation of synthetic scenes rendered from 3D shape models offer a promising approach to transfer knowledge from synthetic to real domain. The performance of several state of the art CNNs (Convolutional Neural Nets) has been evaluated for their application to the Transfer Learning problem in Object Detection.

INTRODUCTION

The task of Object Detection can be performed using several methods. Early methods used feature engineering, which included the use of handcrafted feature vectors custom to the object under consideration [1] . However, such methods failed to generalize over a vast number of classes as well were limited to a select variety of objects. Hence modern systems rely on neural networks, mostly CNNs to perform deep vision tasks.

Convolutional Neural Networks are a feed forward neural network architecture that have proven very effective in object detection and classification tasks. Their key features [2] include:

- Convolution Layers for feature mapping.
- Weight Sharing for Spatial Coherence.
- Pooling and Sub-Sampling layers for dimensional reduction.
- Fully connected final layer for object classification.

Fig. 1 below shows the general layer wise architecture of a CNN for deep learning.

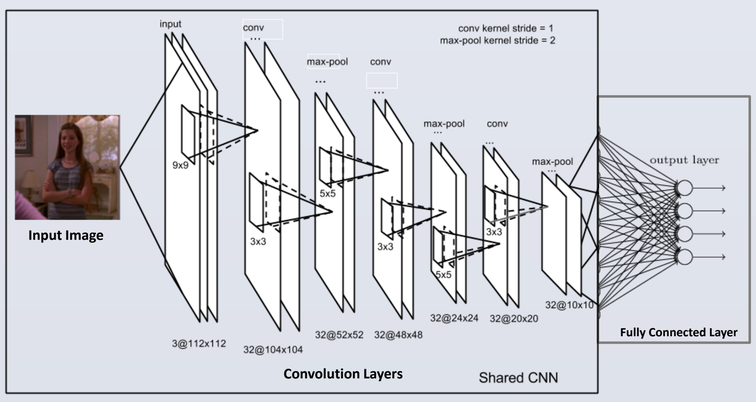

Fig. 1: Basic CNN Architecture

SYNTHETIC DATA: TRANSFER LEARNING

Synthetic images form the key component for training during any transfer learning process. We apply transfer learning strategy to detect packaged food products clustered in refrigerator scenes. We use Blender-Python APIs to load 3D models and automate the process of scene generation. The overall process flow for generation of Synthetic images is given in Fig. 2 below. Fig 3 shows the wide variety and diversity in scene generation through blender API.

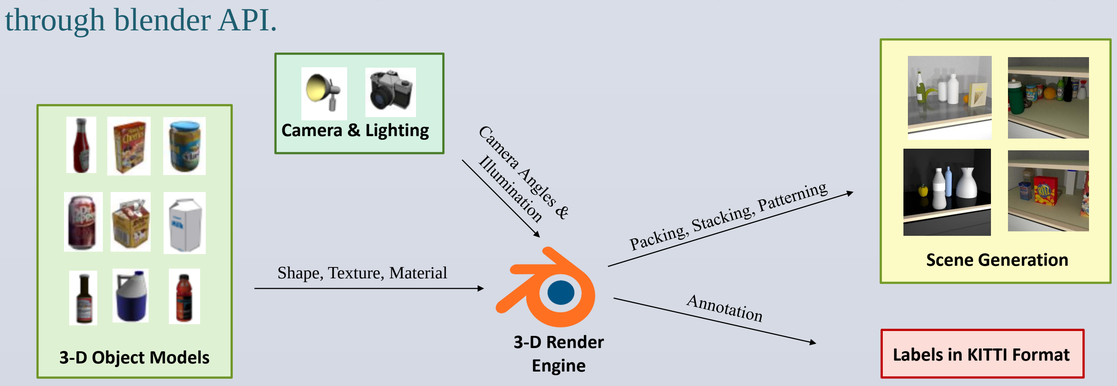

Fig. 2: Synthetic Dataset Generation through Blender, involving Scene Generation through 3D Models and their annotation

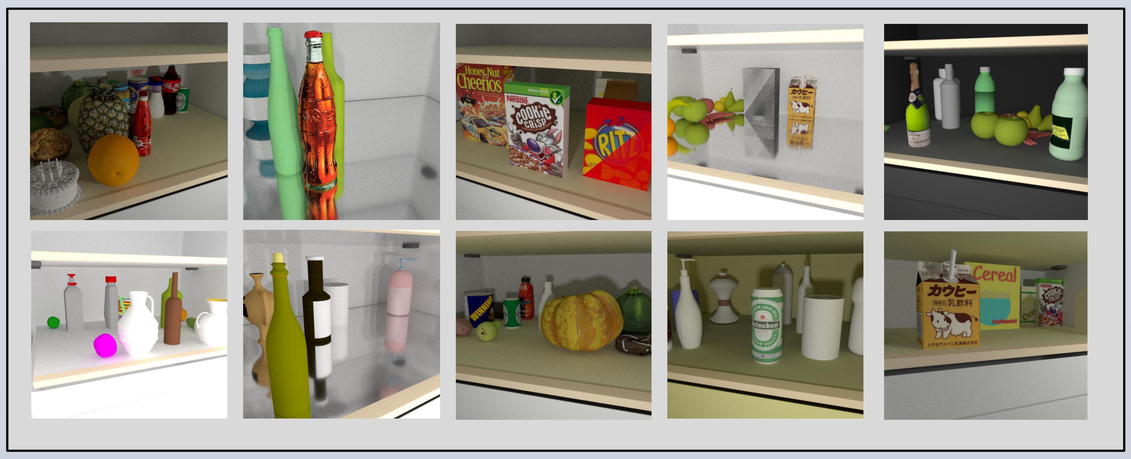

Fig. 3: Synthetic Scenes with different objects, packing, lighting and camera angles, rendered using Blender API for training the transfer learning model.

SYSTEM ARCHITECTURE

Standard process for transfer learning involves generating a synthetic dataset followed by training it on the desired CNN and ends with validation on a real test set. The overall Object Detection pipeline for transfer learning has been shown in Fig. 4. The following state of the art deep neural network architectures have been evaluated for their performance on Transfer Learning:

1. GoogLeNet : NVidia DIGITS object detection framework based on GoogLeNet [3].
2. YOLO: You Only Learn Once, Fast Unified Real Time object detection [4].
3. SSD: Single Shot MultiBox Detector [5].

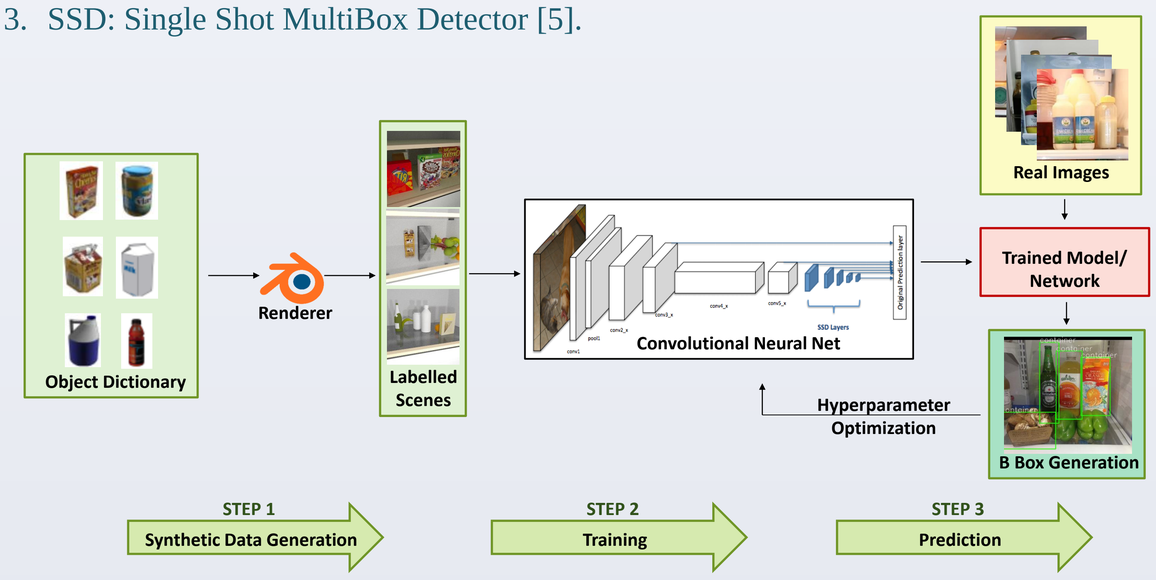

Fig. 4: The standard Object Detection Pipeline for Transfer Learning. Step 1 involves generation of Synthetic Dataset for transfer Learning. Step 2 involves the use of a state of the art CNN for training on the synthetically generated dataset. Step 4 involves testing on a real dataset to evaluate network performance and accuracy.

RESULTS

The performance evaluation of any object detection framework is done using a metric known as mean average precision (mAP). The performance analysis of all the 4 state of the art networks, i.e. Nvidia (DIGITS), YOLO and SSD is shown below. Experiments were carried out on workstation with Intel i7-5960X processor accelerated by NVIDIA GEFORCE GTX-1070. Fig. 5 shows the relative performance of the CNN frameworks.

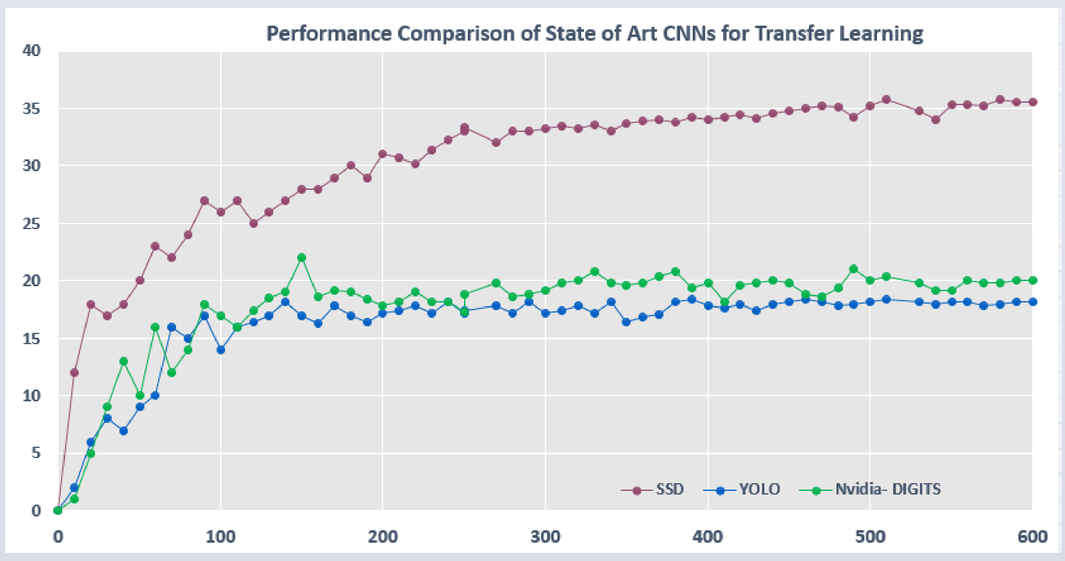

Fig. 5: Performance comparison between SSD, YOLO and Nvidia DIGITS showing the mAP vs the number of epochs trained.

The following **key findings** were observed from the experiments carried out:

1. SSD outperforms YOLO and Nvidia DIGITS in terms of both accuracy and speed.
2. Synthetic dataset needs to be diverse in terms of scene definition for transfer learning.
3. Performance of CNNs is affected by the generalization capability of the synthetic dataset but independent of the number of training images. Increasing dataset size may cause overfitting.
4. mAP finds a direct correlation between no. of epochs and size and diversity of training data.

Fig. 6 shows the test results obtained after training the mentioned networks. The networks successfully detect objects within the real scenes with a high degree of precision.

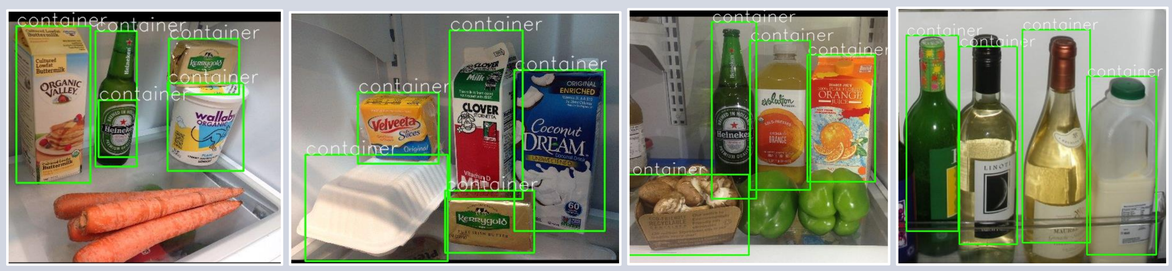

Fig. 6: Test output of the Convolutional Neural Networks trained on Synthetic Images and tested on Real Scenes.

REFERENCES

[1]. S. Khalid et al, “A survey of feature selection and feature extraction techniques in machine learning.” *In Proc. IEEE SAI, 2014*

[2]. Y. LeCun et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.

[3]. C.Szegedy et al. "Going deeper with convolutions." *Proceedings of IEEE* CVPR (2015).

[4]. J. Redmonet al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE* CVPR (2016).

[5]. W. Liu et al. "Ssd: Single shot multibox detector." *arXiv preprint arXiv:1512.02325* (2015).

Prepared By: Jonti Talukdar, ITNU, under the guidance of Dr. Ravi Hegde at HPC Lab, IITGn.