ArNet: A Deep Learning Architecture for Pixel-wise Semantic Segmentation of Images

Arik Md. Isthiaque[1]*, Arifa Islam Champa[2]

[1] Student, Department of Computer Science and Engineering, Bangladesh Army International University of Science and Technology, Cumilla, Bangladesh.
e-mail: isthiaque@baiust.edu.bd
tel.: +8801521493029

[2] Lecturer, Department of Computer Science and Engineering, Bangladesh Army International University of Science and Technology, Cumilla, Bangladesh.

Abstract

Detecting objects from images and making decisions from those collected data is a major part of an intelligent machine. Nowadays many real-time machines like robots, autonomous vehicle works on the basis of detecting objects from visual scenarios. So, the technological advancement of computer vision is an utmost importance. The semantic segmentation of images is a part of computer vision to understand a visual scenario. Here we present a deep learning architecture for semantic image segmentation that is motivated by SegNet and Fully Convolutional DenseNet for semantic image segmentation for achieving state-of-the-art accuracy. We have a global accuracy of 91.00% in the current "CamVid" dataset we are working on.

Keywords- Convolutional Neural Network, Encoder-Decoder Network, Semantic Segmentation, SegNet, DenseNet.

1. Introduction

Image segmentation is a process where the objects of an image are classified by their pixel information. In an image, certain pixels share some common characteristics. By identifying them we can label the pixels, so they can be detected from the image. We use image segmentation to find objects in images like text (OCR), cars (automated vehicle), cancer, tumor (medical image). In recent days, deep learning algorithms got success in handwritten OCR, NLP [1] [2] [3]. It has also a huge active interest in pixel-wise semantic segmentation. Semantic segmentation is a machine learning process where a machine can learn from a visual scenario by adding class to every pixel that it contains. It is not like other instance segmentations. Instance segmentation can differentiate between objects of the same classes. In semantic segmentation, it can only detect and it doesn't care about the instances. Our ArNet is based on the SegNet architecture which is a deep convolutional encoder-decoder architecture for robust semantic image segmentation [4]. Our model is also based on encoder-decoder network architecture. One of the major drawbacks of SegNet is that it had too many layers. As we know by increasing the layers of an architecture, we can get higher accuracies but the cost will be more increased [5]. So, we build an architecture called ArNet with lower network layers with more accuracy.

2. Literature Review

2.1 Fully Convolutional Network (FCN8)

Fully Convolutional Network (FCN8) is a deep learning algorithm for semantic segmentation in the image [6][6]. The architecture is built with some blocks of convolution and max pool layers. The purpose of these layers is to decompress the input image and reduce it to 1/32th of its original size. It generates some classes from this decompressed image by using some random predictions. After that, an up-sampling and deconvolution layer work on that decompressed image and the image returns to its original dimension [7][7].

This architecture is not fully connected. It captures the contextual information via down-sampling. But it doesn't limit the input image because the original image size can be regenerated via up-sampling. So, the result we got after the process is the same as the original image size. Here the architecture also uses skip connections of layers to fully regenerate the original image. In skip connection of layer architecture, the process can skip one or some of the layers before completing its process [9] [9].

2.2. SegNet

SegNet is an encoder-decoder architecture built with a deep convolutional neural network for pixel-wise semantic segmentation for images and videos. It is generally built for roads, buildings, cars, pedestrians and for differentiated the context between roads and side-walks. SegNet extracts low-resolution pixel information from the input image and classifies them as per pixel. This extraction must generate some patterns which can be used in boundary localization [4].

SegNet built with an encoder layer and a decoder for a pixel-wise segmentation. The model is in Figure 1. The encoder layer is built with 13 convolutional layers that follow the VGG16 network [3]. The training process can be initialized from weights trained for classification with large datasets [10].

SegNet is built with a stack of encoders with a similar number of decoders. The decoders made the output the same size as the input. This is an important drawback of the architecture as the increasing number of layers increase the parameters of the model [4].

2.3. Bayesian SegNet

This is an extension of the main SegNet architecture. Its main job is to find the uncertainty in the SegNet architecture [4] [11]. It consists of the same sequence of non-linear processing layers called encoders and a corresponding set of decoders followed by a pixel-wise classifier. In this Bayesian architecture, the performance is improved for some baseline datasets. It can probabilistic segmentation by finding the posterior distribution over convolutional weights. Here Monto Carlo dropout is used to approximate the inferences [12].

2.4. Fully Convolutional DenseNet

FCN8 is built with downsampling, upsampling, and some skip layers. In this FCN8 there is a major drawback that is the features explanations in the up-sampling model. The performance of FCN8 could be extended by avoiding features explorations in the upsampling model. This problem is solved in this new densely connected network which is called the Fully Convolutional DenseNet. In the Fully Convolutional DenseNet, the convolutional operations are substituted by a new operation which is referred to as the transition up. It consists of several convolutional modules that upsampled the compressed image. Input images are concatenated by the skip layer with a dense layer [5].

Fully Convolutional DenseNet extended the FCN8 architecture to tackle the problems of image semantic segmentation by making it fully convolutional and by building dense blocks and add them with the extracted feature maps.

4. Methodology

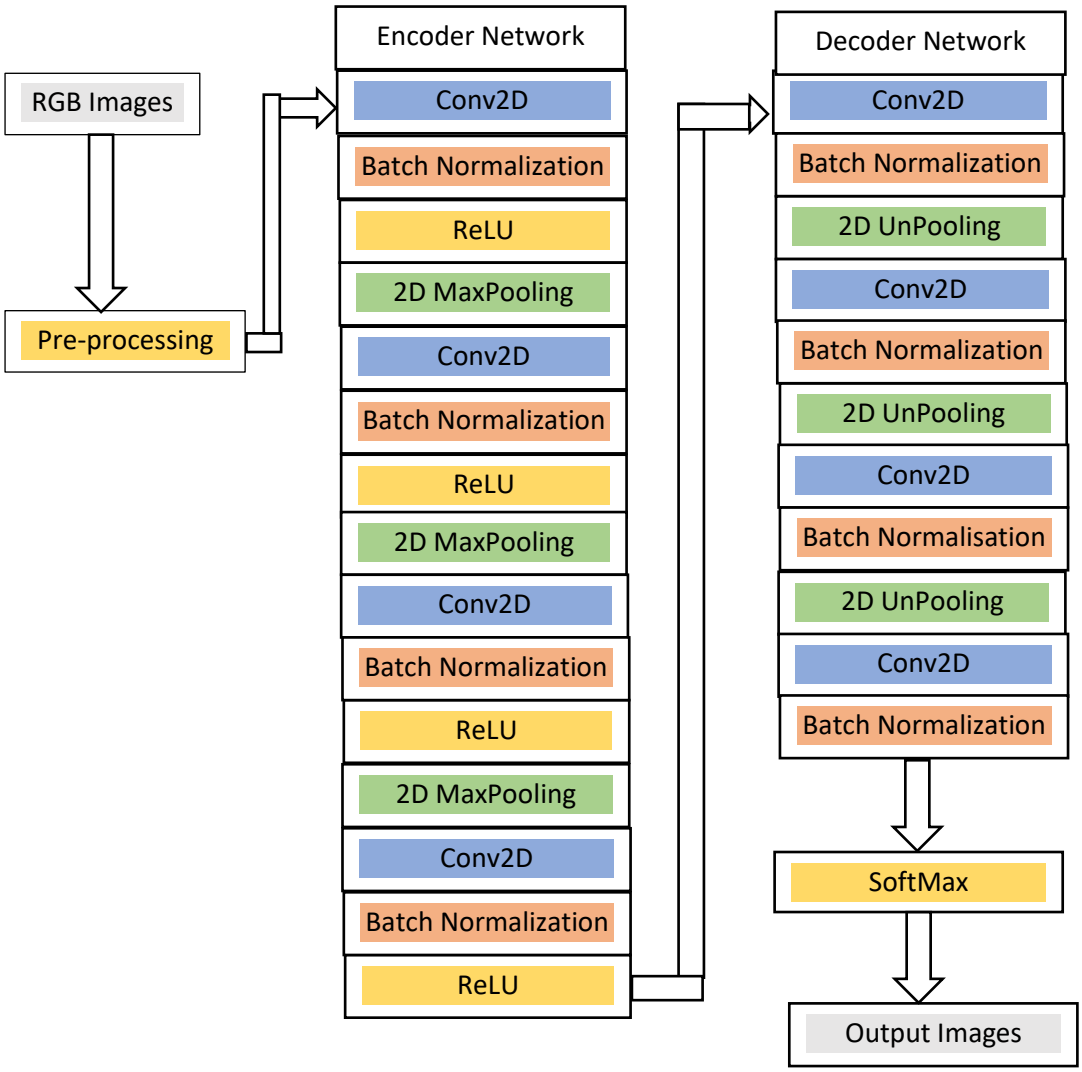

Figure 1: The ArNet Architecture.

The architecture can take RGB image as input. The input image of the dataset at first goes into a pre-processing step where we will process it for the encoder network. In the pre-processing, we will extract the binary level information of the image and the histogram information. By using the extracted information, the image will be reshaped at the size of 360x480. Then the dataset will be fed into the encoder network.

The encoder network is a combination of some sequence of layers. The layers sequence is identical to the VGG16 network. The first layer is the 2D convolution layer with the kernel size 3 and filter size 64. Before the convolution, the input must be padded with zero to match the filter size. After that, a batch normalization will be applied to the convoluted data. It is a normalization process for adjusting the data for scaling and activations [4]. It also speeds up the learning process. So, we used batch size 10. After adjusting the data for the activation process, we apply the Rectifier Linear Unit activation known as ReLU. In the neural network, an activation function is too much imported for transforming

the sum of weights of the input to output and prepare it for feeding into further layers. The Rectifier Linear Unit activation function is a linear function that directly transfers the output if it gets a positive input otherwise it will output zero. Now a 2D MaxPooling will be applied to the data with pool size 2. The 2D MaxPooling is a discretization process. Its main goal is to down-sample the input based on its weight and reduce the input dimension. After this, we will again pad the data with zero and again a 2D convolution will be applied but now the size of the filter is increased to 128 and kernel size 3. After that, all the processes will be applied as described in figure 1. In the 2D convolution, the size of the filter will be increased to 256 and the last layer of convolution will be applied with the filter size of 512. Then the data will be fed into the decoder layer for further processing.

In the decoder layer first, the data will be padded with zero than a 2D convolution will be applied with the filter size of 512 and kernel size 3. After that, a regular batch normalization will be applied with batch size 10 to feed the data into the 2D UnPooling layer. In the UnPooling the down-sampled data will be again up-sampled. After all the processes applied to the data that are described in figure 1, we will get the output. The size of the output will be the same as the size of the original input that we get after the pre-processing of input.

After we get the output from both the encoder-decoder network now, we apply a SoftMax activation function. It is a normalized exponential function. It output a probability distribution variable that will list all possible classes that we can get from the neural network.

5. Results

Table 1: Global accuracy comparison with some of the best architectures of semantic image segmentation with ArNet on the "CamVid" dataset.

Architecture	**Global Accuracy**
FCN8 [6]	88.0
SegNet [4]	62.5
Bayesian SegNet [11]	86.9
DeconvNet [13]	85.9
Dilation8 [14]	79.0
Dilation8 + FSO [15]	88.3
FC-DenseNet56 [5]	88.9
FC-DenseNet67 [5]	90.8
FC-DenseNet103 [5]	91.5
ArNet	91.0

6. Discussion

We trained our model ArNet with the "CamVid" dataset which is the first collection of videos and images with object class semantic labels and contains complete metadata. In ArNet we have a greater number of layers in comparisons to FCN8. In comparison to others, it contains a much lower number of layers. But in the comparisons of accuracy, we have an efficient accuracy with this label of lower layers as we know higher layers can dramatically increase the accuracy. FC-DenseNet103 is a perfect example. It has a combination of 103 layers, which is a very big number and these layers take a huge time to make conclusions but they have great efficiencies. The dataset we used contains 367 images that contain ground truth labels. The images are captured with CCTV-styles cameras from the perspective of driving an automobile. We are currently working on gathering more images for our

dataset so that we can evaluate our architecture for more research and making it generate more accuracies.

7. Conclusions

Semantic segmentation of the image is one of the key processes of machine automation like robotics and autonomous vehicle. So, it is important to develop these basic processes more efficient to make our machines smarter. From FCN8 to Fully Convolutional DenseNet many things are changed for achieving state-of-the-art accuracies and make the process more efficient. In our architecture, we try to make this more efficient by decreasing some of these layers. We are still working on the layers and trying to make it more efficient if possible. In the future, we have a plan of working with the "Canny Edge Detector" algorithm. If we can interface the algorithm the efficiency of object detection and labeling will be easier as we know this algorithm can easily define the edges of any object from an image.

References

[1] P. Luc, N. Neverova, C. Couprie, J. Verbeek, Y. LeCun; The IEEE International Conference on Computer Vision (ICCV), 2017, pp. 648-657C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "*Going deeper with convolutions*," in CVPR, pp. 1–9, 2015.

[2] K. Simonyan and A. Zisserman, "*Very deep convolutional networks for large-scale image recognition*," CoRR, vol. abs/1409.1556, 2014.

[3] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "*Learning hierarchical features for scene labeling*," IEEE PAMI, vol. 35, no. 8, pp. 1915–1929, 2013.

[4] V. Badrinarayanan A. Handa R. Cipolla "*SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling*" arXiv preprint arXiv:1505.07293 2015.

[5] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, Y. Bengio; *The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation* The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2017, pp. 11-19

[6] J. Long, E. Shelhamer, T. Darrell, "*Fully Convolutional Networks for Semantic Segmentation*" The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431-3440

[7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. *DeCAF: A deep convolutional activation feature for generic visual recognition*. In ICML, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun. *Spatial pyramid pooling in deep convolutional networks for visual recognition*. In ECCV, 2014.

[9] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. *Caffe: Convolutional architecture for fast feature embedding*. arXiv preprint arXiv:1408.5093, 2014.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "*ImageNet Large Scale Visual Recognition Challenge*," International Journal of Computer Vision (IJCV), pp. 1–42, April 2015.

[11] V. Badrinarayanan A. Handa R. Cipolla *Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene* Understanding (Submitted on 9 Nov 2015 (v1), last revised 10 Oct 2016 (this version, v2))

[12] V. Badrinarayanan, F. Galasso, R. Cipolla. *Label propagation in video sequences*. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 3265–3272. IEEE, 2010.

[13] H. Noh, S. Hong, and B. Han. *Learning deconvolution network for semantic segmentation*. arXiv preprint arXiv:1505.04366, 2015.

[14] F. Yu and V. Koltun. *Multi-scale context aggregation by dilated convolutions.* In International Conference of Learning Representations (ICLR), 2016.

[15] A. Kundu, V. Vineet, and V. Koltun. *Feature space optimization for semantic video segmentation*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.