

# A Patch-Based Convolutional Neural Network Architecture for Semantic Segmentation of a Blood Sample

Siddhanth Pillay

May 20, 2019

## Abstract

In this report, I have presented the results of the performance of a patch-based convolutional neural network architecture for the task of semantic segmentation of blood sample images. I have also presented the results of an SVM approach for the same and performed a comparative analysis between the two approaches to demonstrate the superior performance of the CNN Architecture.

## 1 Architecture

In this experiment, LeNet was chosen as the Convolutional Neural Network architecture. LeNet is composed of 7 layers(not counting the input), all of which contain the trainable parameters (weights). The input to the network is the image of a patch of size 81x81. In the following description, convolutional layers are labeled Cx, subsampling layers are labeled Sx, and fully-convolutional layers are labeled Fx, where x is the index of the layer.

Layer C1 is a convolutional layer with 6 feature maps. Each unit in each feature map is connected to a 5x5 neighborhood in the input. The size of the feature maps is 77x77. C1 contains 156 trainable parameters.

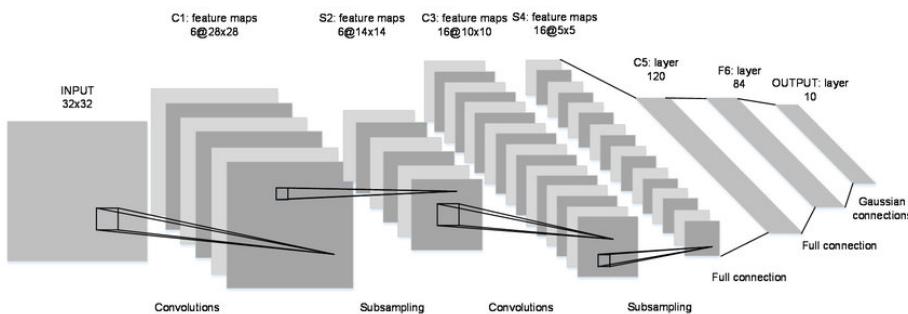


Figure 1: Original LeNet Model

Layer S2 is a subsampling layer with 6 feature maps. Each unit in each feature map is connected to a 2x2 neighborhood in the corresponding feature map in C1. The four inputs to a unit in S2 are passed through a max function to select the maximum among these values. The 2x2 receptive fields are non-overlapping, therefore feature maps in S2 have half the number of rows and columns as feature maps in C1. Layer S2 has no trainable parameters.

Layer C3 is a convolutional layer that has 16 feature maps. Each unit in each feature map is connected to several 5x5 neighborhoods at identical locations in a subset of S2's feature maps. Layer C3 has 1516 trainable parameters.

Layer S4 is a subsampling layer with 16 feature maps. Each unit in each feature map is connected to a 2x2 neighborhood in the corresponding feature map in C3, in a similar way as in C1 and S2. Layer S4 has no trainable parameters.

The feature maps obtained from S4 are then arranged in a vector form. Each unit in the vector is connected to each node in the Layer C5. Thus, this forms a fully connected layer. The total number of nodes in the layer C5 is 120.

Layer F6 has 84 units and is fully connected to C5. It has 10,164 trainable connections.

Finally, the units in F6 layer are fully connected to output layer which contains 4 nodes (one for each class). The values of the output layer are passed through a softmax function to give us the probability values for each of the class.

The ReLU activation function is used after performing computations at layers C1, C3, C5 and F6.

## 2 Experiments and Results

The first task that is to be performed for training the model is the creation of the patch images. For this, we selected 100 pixels at random from every image and created images of patches around them. This step is then repeated for all the images in the dataset. Consequently, 32,800 background class, 17,915 platelet class, 31,500 red blood cell class and 32,300 white blood cell class images were obtained as a training set. As a data preprocessing step, we divide the intensity values at every pixel by 255.

The LeNet model is then trained over this dataset by using Cross Entropy Loss and Adam optimizer with default parameters for 50 epochs. For obtaining the pixel-level classification images, we iterate over all the pixels in the entire image and at each pixel, collect values of the pixels falling around the pixel in the defined patch size of 81x81. This patch is then passed through the LeNet model and the label with the maximum value is chosen as the pixel label.

As a baseline, we train an autoencoder network on the patch images and reduce the number of features to 100. We train an SVM with a Radial Basis Function Kernel using these 100 features for every patch image created as de-

fined above. The results of the SVM model and LeNet model are shown in Fig 3 and Fig 4.

From the images, it is clear that the LeNet architecture performs better than the SVM architecture. LeNet is able to discern between the boundaries better than SVM. It is also able to distinguish between White Blood Cell/Platelet and Red Blood Cell/Background to a greater extent than the other model.

However, the LeNet model results when analyzed independently also have quite a few drawbacks that need to be addressed. For instance, both the architectures are showing patches of platelets where there are instances of WBCs. There is also some amount of ambiguity in the boundary regions between Red Blood Cells and Background. To analyze this observation, we generate the probability map for each of the classes considering the LeNet model. We also calculate the unalikability parameter for every pixel, given by:

$$M1 = 1 - \sum_{i=1}^K p_i^2 \quad (1)$$

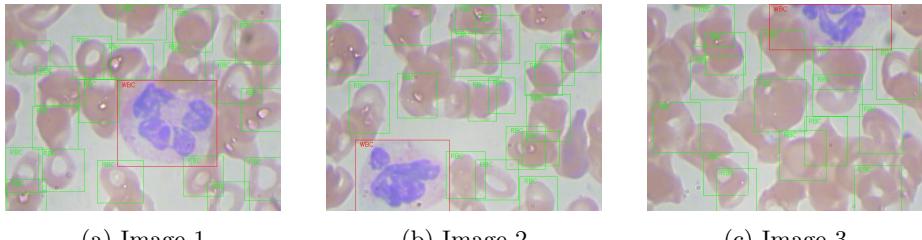
where  $p_i$  is the probability value of each class, obtained from the output of the final layer. We observe the results of these in Figures 5, 6, 7 and 8.

The results from these figures confirm our suspicion. There is some amount of confusion in the LeNet model in categorizing boundary regions between Red Blood Cells and Platelets. This can be attributed to the fact that when we take a patch around the pixels lying in the boundary region, we notice that the patch will contain almost the same amount of visual characteristics of both, background class and Red Blood Cells. Hence, the network is generating higher probabilities for these classes, which can be seen from the probability maps and which leads to higher ‘confusion’, as seen by the unalikability map.

There are some high unalikability areas in the regions present in the White Blood Cells as well. This is because the WBC and Platelet classes are assigned a relatively higher probability score, which can be attributed to the visual similarity in the patches of the two classes. We must also recognize that the dataset provided for this task did not contain pixel-level annotations, which would be ideal, but were provided with bounding boxes around the region of interest. It was also found that not all the regions of interest had been annotated. Due to these limitations in the data, there is a certain amount of noise inherent in the training data.

### 3 Conclusion

In this report, the results of the patch-based LeNet architecture for the task of semantic segmentation of a blood sample is presented. It was compared with a baseline of Autoencoder+SVM model and shown to perform much better than this model. Through analyzing the probability maps and unalikability maps, the drawbacks of the model were also studied.

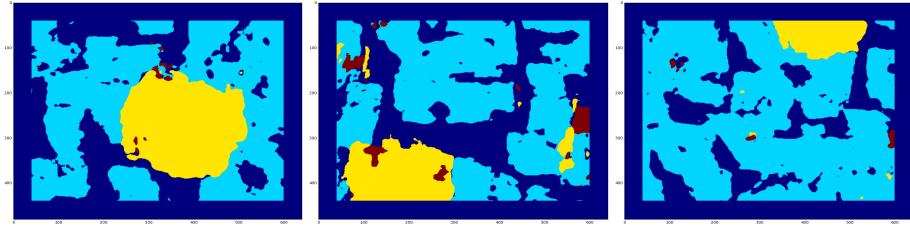


(a) Image 1

(b) Image 2

(c) Image 3

Figure 2: Original Images with Bounding Boxes

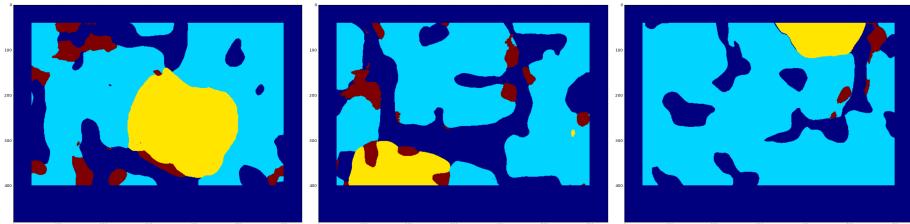


(a) Image 1

(b) Image 2

(c) Image 3

Figure 3: Results of LeNet Architecture

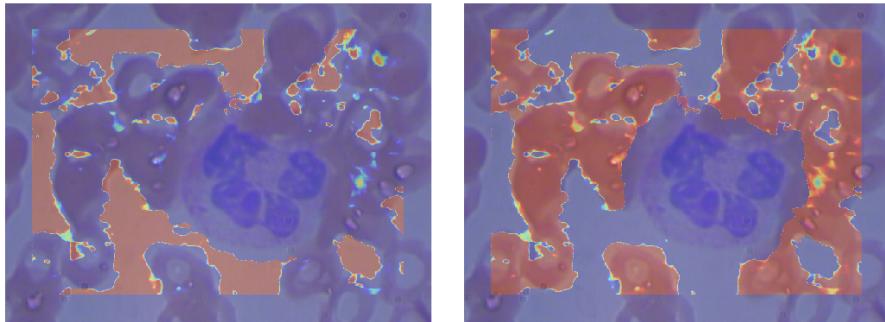


(a) Image 1

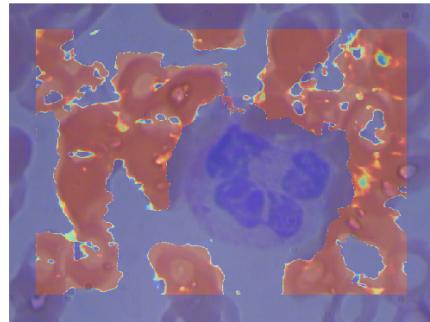
(b) Image 2

(c) Image 3

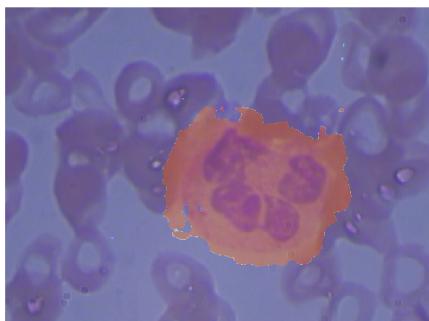
Figure 4: Results of SVM Architecture



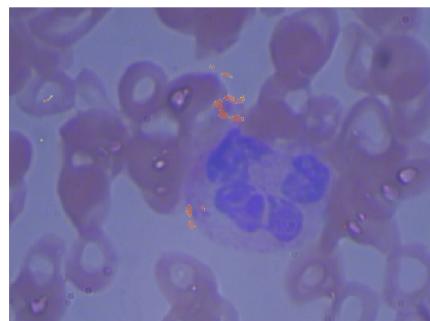
(a) Background



(b) Red Blood Cells

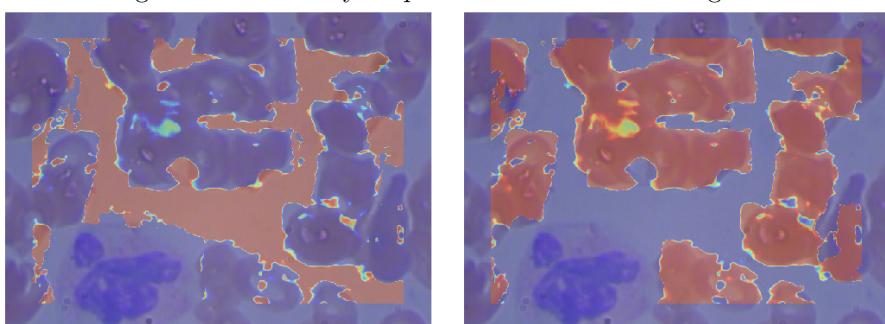


(c) White Blood Cells

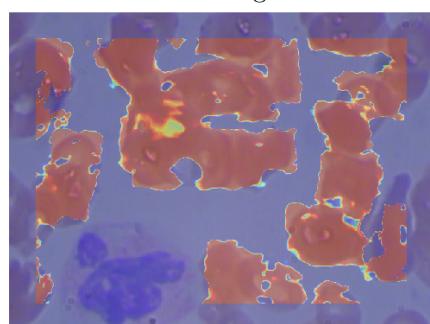


(d) Platelets

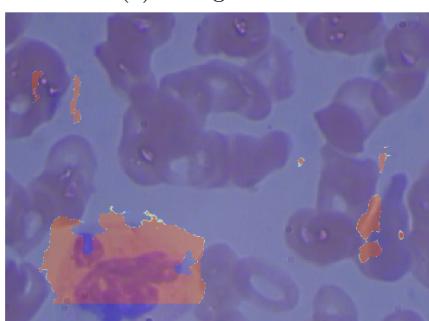
Figure 5: Probability Maps for each classes for Image 1



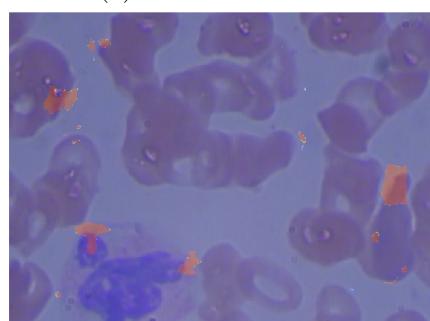
(a) Background



(b) Red Blood Cells



(c) White Blood Cells



(d) Platelets

Figure 6: Probability Maps for each class for Image 2

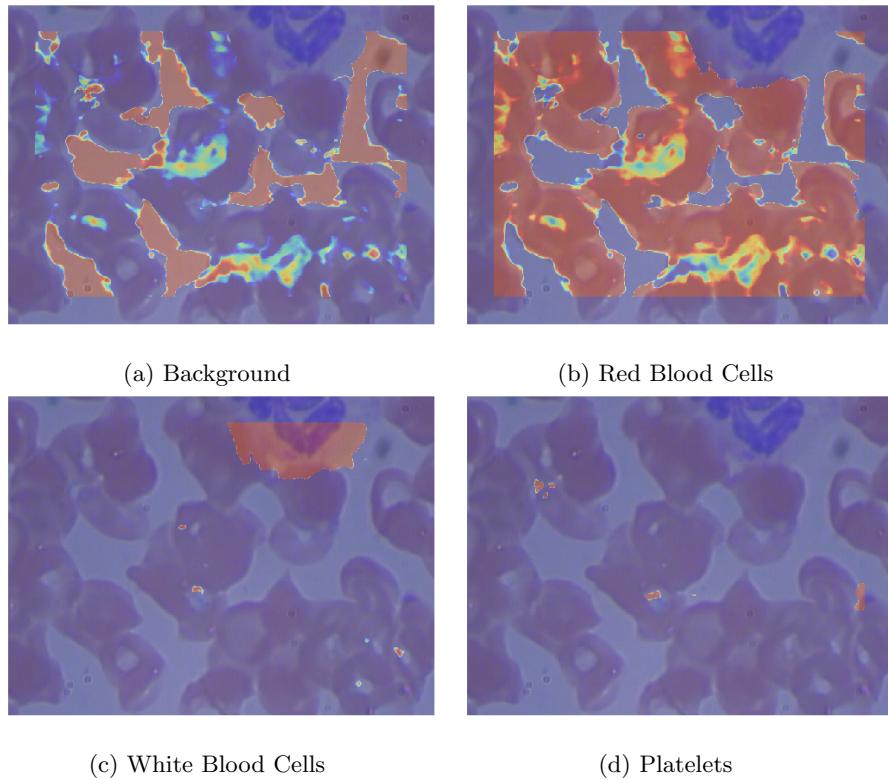


Figure 7: Probability Maps for each class for Image 3

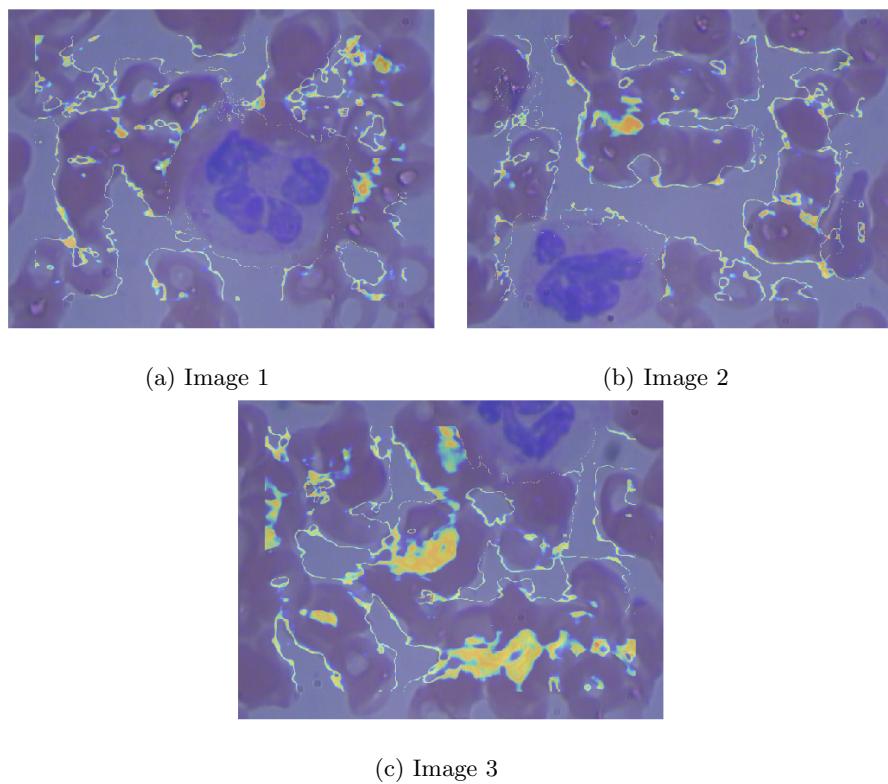


Figure 8: Unalikability Maps for each image