# Vortex Boundary Identification using Convolutional Neural Network – Supplemental Document

Category: Research

Paper Type: Empirical Study

## ABSTRACT

In this supplemental document, we provide additional information of the neural network architectures that we trained for vortex boundary extraction. We also provide the error plots of the training of the three architectures on the synthetic flow data set generated based on the fitted parameter ranges and distributions for the simulation of the 2D flow behind a cylinder. Finally, the full resolution results of the 2D flow behind cylinder simulation are provided.

## 1 ARCHITECTURES

In this section, we provide the details of the three network architectures that we evaluate.

**CNN** We build our first network as a form of typical CNN, as illustrated in Figure 1. The first part of our CNN consists of 2D convolutional layers. In this part, the dimensionality is reduced and feature maps are extracted with a convolution kernel size of $3 \times 3$ and with strides of $2 \times 2$, followed by a batch normalization and a rectified linear unit (ReLU) activation. The number of features is doubled starting from 64 to 128. The second part of the network exploits fully connected layers instead of convolutional filters for the final inference from the identified high level flow features. Batch normalization and a ReLU activation follow in the same fashion as in the convolutional layers, and a dropout with the probability of 0.5% is used to avoid overfitting. The number of neurons of the last layer is set to match the target resolution, here, the initial $64 \times 64$ target patch.
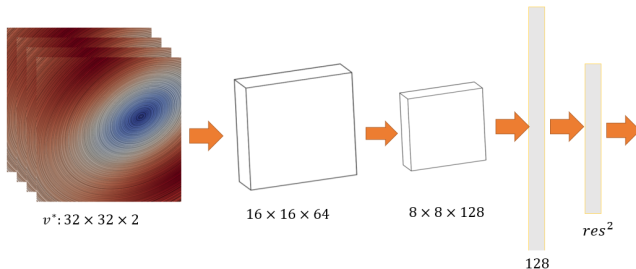


Figure 1: Our CNN architecture. The numbers below each box denote the dimensionality of the feature maps.

**Resnet** Our Resnet model is based on the Resnet20 architecture by He et al. [2]. The core idea of ResNet is to introduce an identity shortcut connection or skip connection that skips one or more layers, as shown Figure 2. The motivation behind that is that deeper model should not produce a training error higher than its shallower counterparts. The authors in [2] hypothesize that letting the stacked layers fit a residual mapping is easier than letting them directly fit the desired underlying mapping. And the residual block above explicitly allows it to do precisely that. In our implementation, we use a learning rate 0.001 and place 6 residual blocks. Each block includes three convolutional layers.

**Unet** Our Unet model is based on the original model proposed by Ronneberger et al. [4]. In addition to skip connection, it also includes
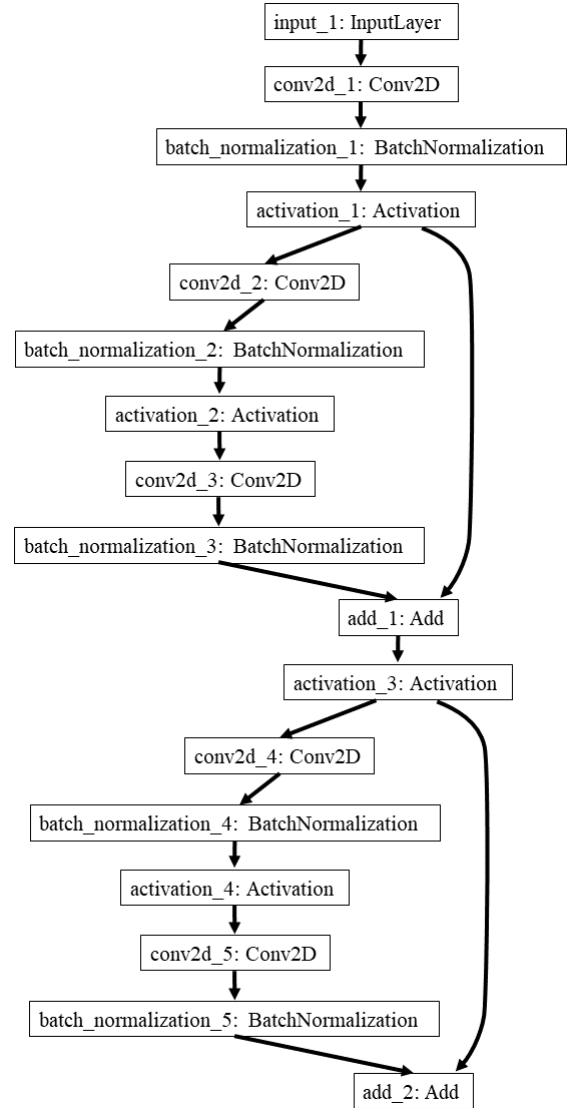


Figure 2: One residual block in the Resnet20 architecture.

a concatenation with the correspondingly cropped feature map from the contracting path. In the end, a $1 \times 1$ convolutional layer is used to make the number of feature maps equal to the number of segments which are desired in the output. Unet uses a loss function for each pixel of the image. This helps in easy identification of individual cells within the segmentation map and it could generate the results with higher accuracy in comparison to ResNet. We use a depth of three and a dropout of 1%. The activation function is set to ReLu and we use max pooling between the layers. Figure 3 shows the structure of our model.
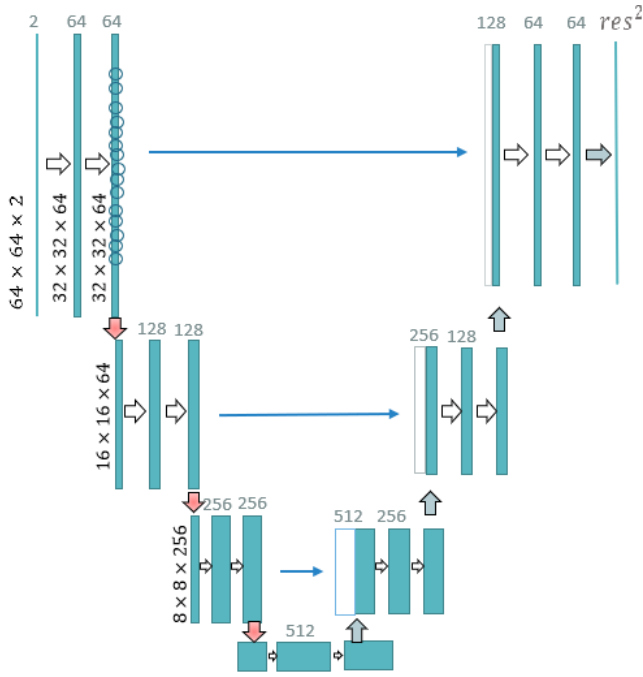
Figure 3: Our Unet architecture. We used three layers of depth. The numbers above each box represent the number of feature maps.

## 2 PARAMETER SPACE FITTING AND TRAINING FOR CYLINDER FLOW

We performed simulated annealing to optimize the parameter ranges for our parametric synthesis framework. The optimized ranges for the individual parameters for the CYLINDER flow were shown in Figure 4. Next, based on these optimized parameter ranges, we generate synthetic flow patches using the proposed parametric synthesis with the combinations of the parameter values sampled with the probability based on the distribution shown in Figure 4. For this flow, we generate $100k$ synthetic flow patches to train the three neural networks as described earlier, including the velocity field and the binary segmentation for each patch. We use a binary cross entropy as loss function, which is plotted in Figure 5 for the three networks. The loss of the training and testing set reduce in well-behaved manner. For Unet, individual spikes on the testing loss can be seen, which are due to the mini batch gradient descent in the ADAM optimizer. We terminated the training after 100 epochs, since further improvements were small enough.
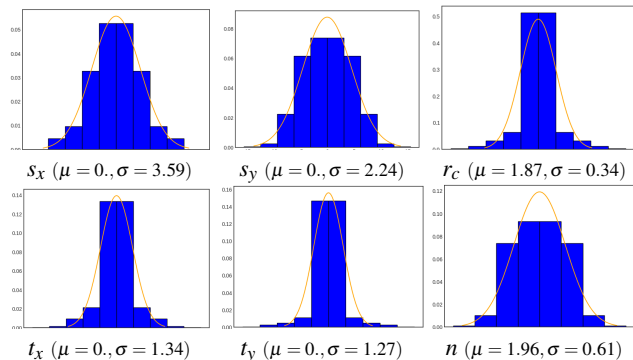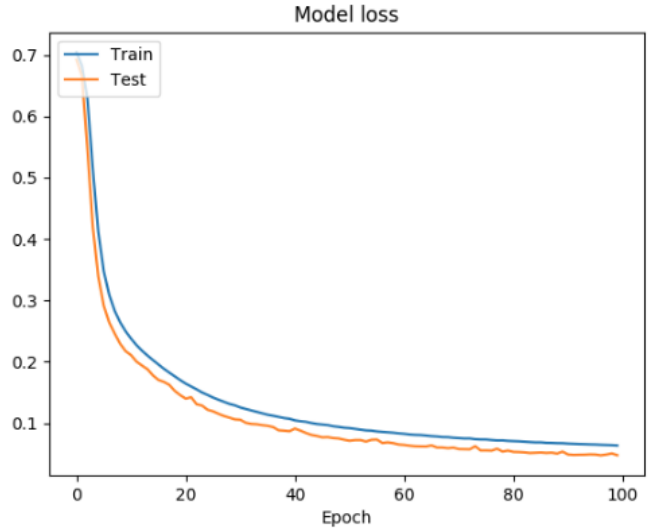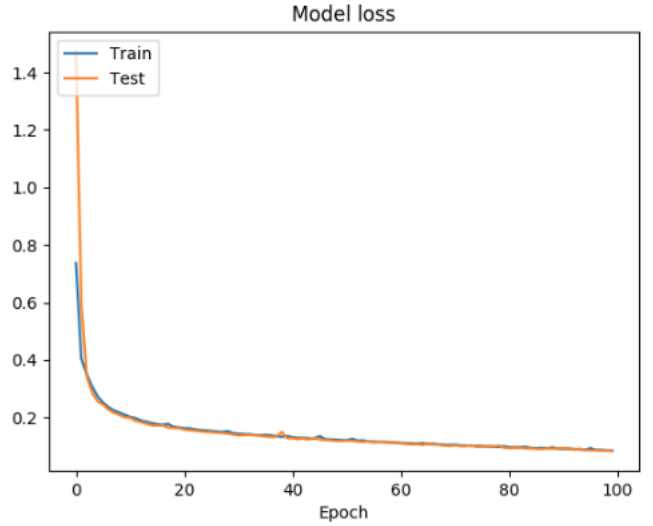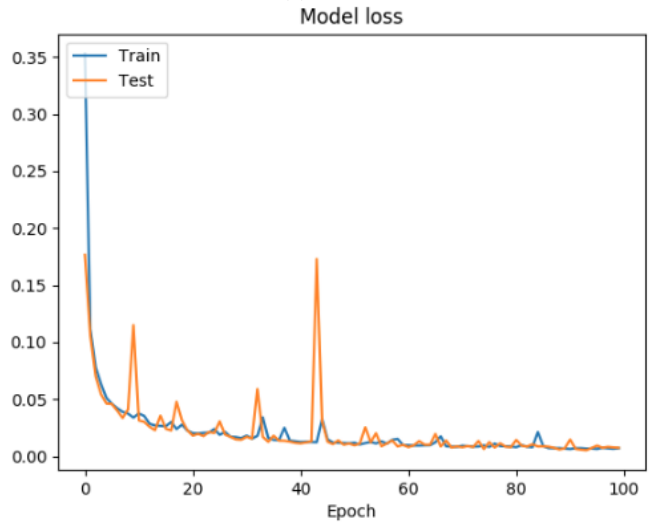


Figure 4: Histograms of the individual model parameters for the CYLINDER flow, showing the near-Gaussian distribution of the individual parameter values.



(a) CNN



(b) Resnet



(c) Unet

Figure 5: Loss plots of the training and testing set for 100 epochs for all three network architectures. Initially, the loss decreases quickly. From 20 to 30 epochs onwards, further training effort is rewarded by only slow improvements.

(a) CNN



(b) Resnet
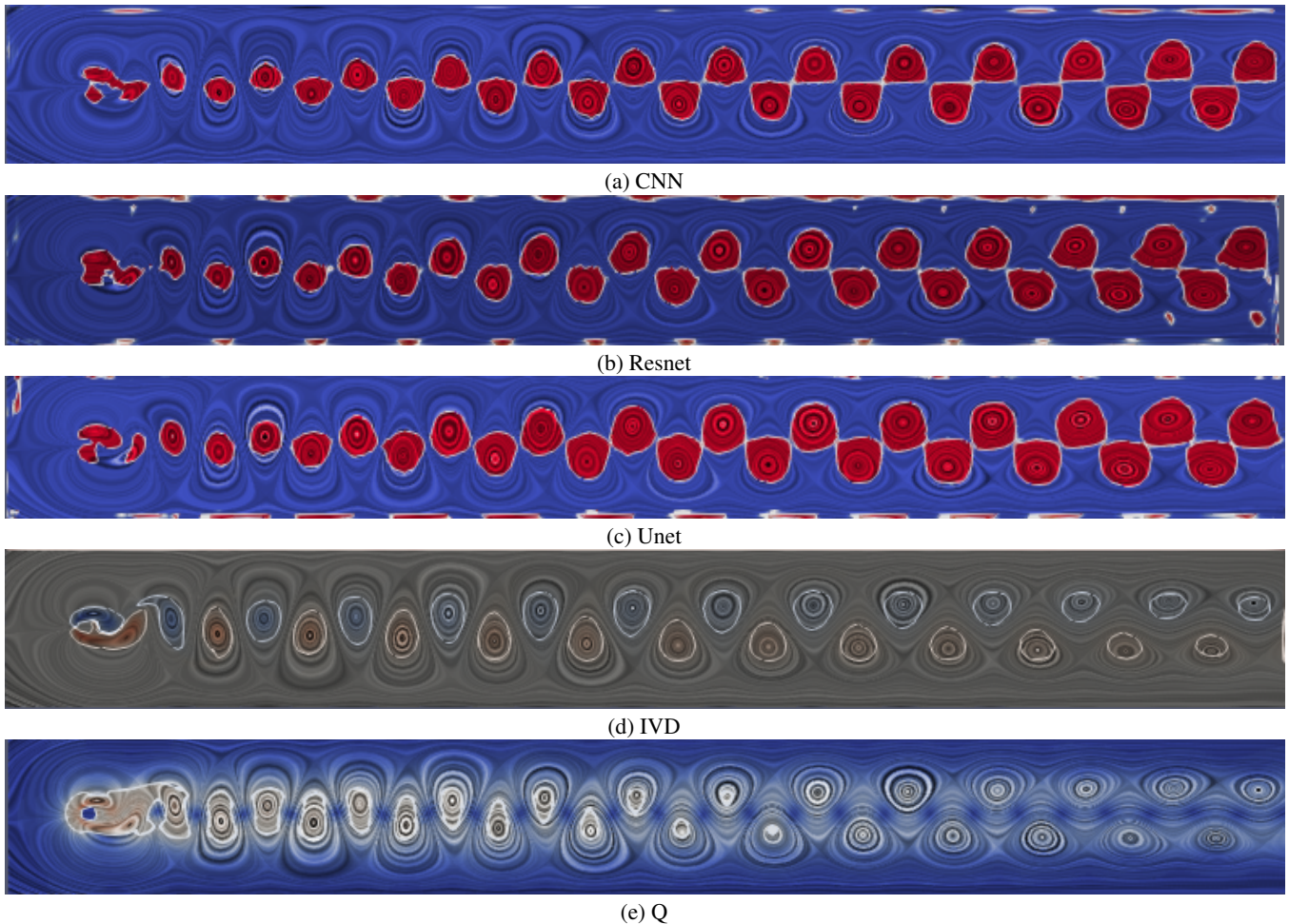


(c) Unet



(d) IVD



(e) Q

Figure 6: Comparison of boundary extraction with cnn (a), Resnet (b), and Unet(c) using the CYLINDER flow. The input of networks are velocity patches. Our method shows Unet outperforms the other networks. (d) shows the IVD result with threshold value of ±0.03. (e) shows the iso-contours of the Q attribute field of the flow with threshold value of 0.019.

## 3 RESULTS FOR CYLINDER FLOW

Figure 6 provides the extracted vortex boundaries of the CYLINDER flow using the three architectures described above (a–c), the IVD field) [1] (d), and the 2D Q attribute field [3] (e), respectively.

## REFERENCES

[1] G. Haller, A. Hadjighasem, M. Farazmand, and F. Huhn. Defining coherent vortices objectively from the vorticity. *Journal of Fluid Mechanics*, 795:136–173, 2016.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[3] J. Hunt. Vorticity and vortex dynamics in complex turbulent flows. *Transactions of the Canadian Society for Mechanical Engineering*, 11(1):21–35, 1987.

[4] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.