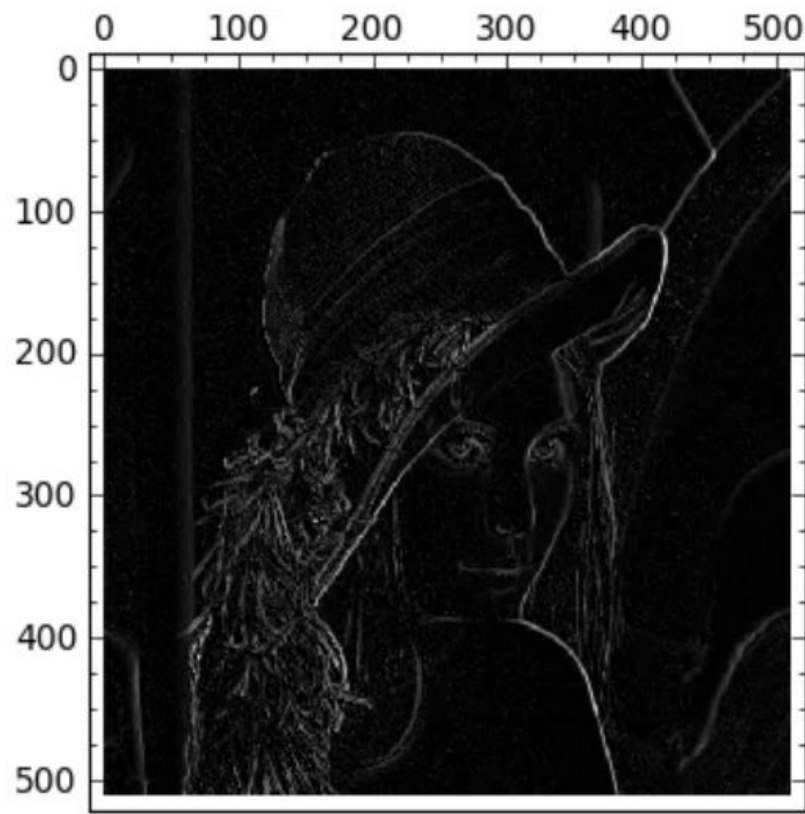
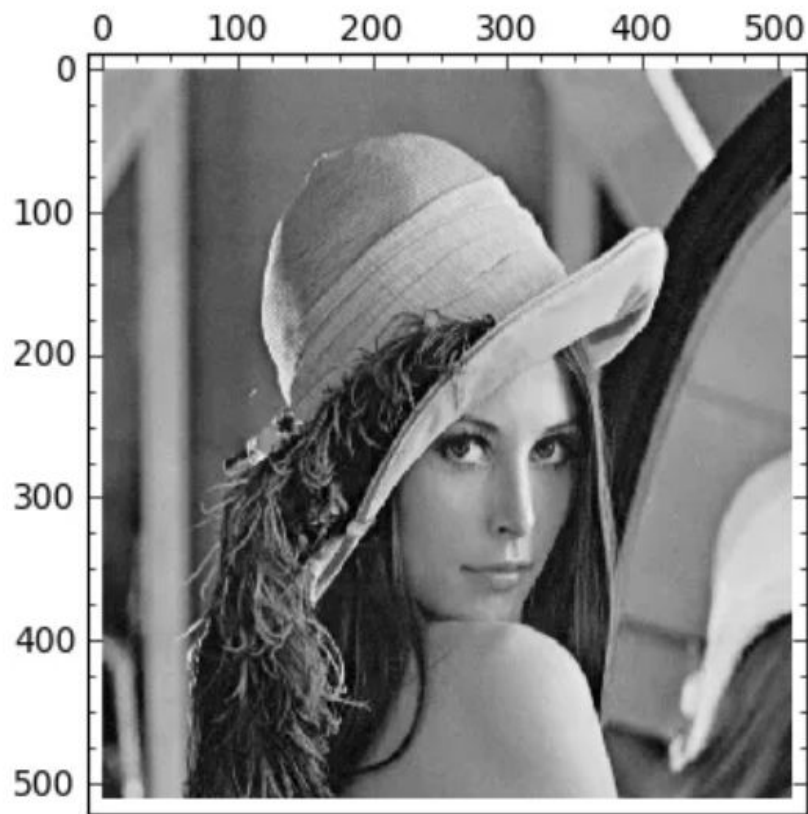


# *Phase Stretch Transform Neural Networks*

Aditya Gorla, Griffin Romanek, Mai Saleh



Source: <https://blancosilva.wordpress.com/teaching/mathematical-imaging/edge-detection-the-convolution-approach/>



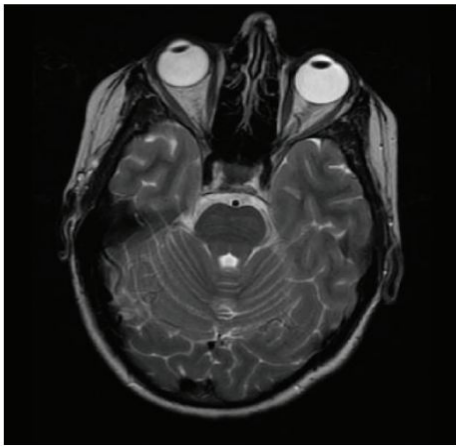
(a) Proposed method, phase strength ( $S$ ) = 0.48, and warp ( $W$ ) = 12.14



(b) Sobel: Thresh = 7.9



(c) Canny: sigma = 1.28; Thresh (min, max) = [0.024, 0.117]



(a) Original image of brain MRI: top view



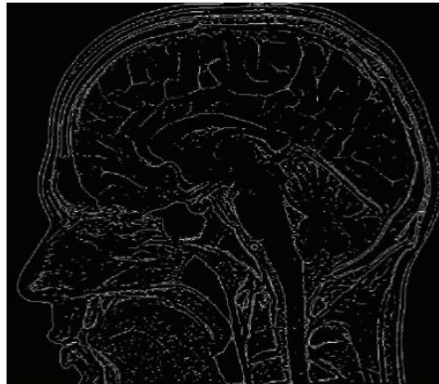
(b) Detected edges  $S = 0.3$ ,  $W = 10$ ,  $T = 0.0019$



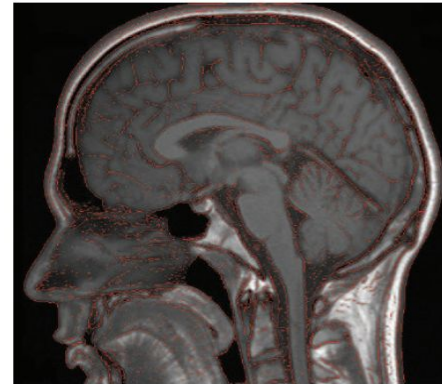
(c) Overlay



(d) Original image of brain MRI: side view



(e) Detected edges  $S = 0.5$ ,  $W = 12$ ,  $T = 0.003$



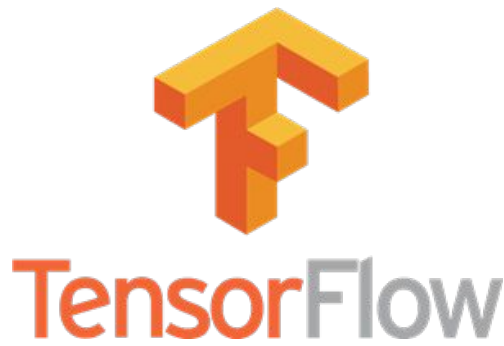
(f) Overlay



## Parameters

-----  
LPF : Isotropic Gaussian localization filter Full Width at Half Maximum (FWHM) (min : 0, max : 1)  
Phase\_strength : PST Kernel Phase Strength (min : 0, max : 1)  
Warp\_strength : PST Kernel Warp Strength (min : 0, max : 1)  
Threshold\_min : minimum threshold (min : -1, max : 0)  
Threshold\_max : maximum threshold (min : 0, max : 1)  
Morph\_flag allows user to compute the analog edge (if Morph\_flag=0) or the digital edge (analog edge followed by thresholding and morphological operations, if Morph\_flag=1).

Parameters of the kernel ( $S$  and  $W$ ) control the edge detection process. In the proposed method, there is a trade-off between spatial resolution and noises of edge detection. A larger phase (larger  $S$ ) results in better noise performance in edge detection but at the expense of lower spatial resolution. Also a larger warp (larger  $W$ ) in the phase derivative results in a sharper edge but it also increases the edge noise. These parameters can be adjusted manually or optimized by iterative algorithms. They can be either globally fixed or locally optimized.



### PST Layer

Takes in Image as input.  
Applies a single PST filter to entire image, training LPF, Warp Strength, and Phase Strength Parameters.

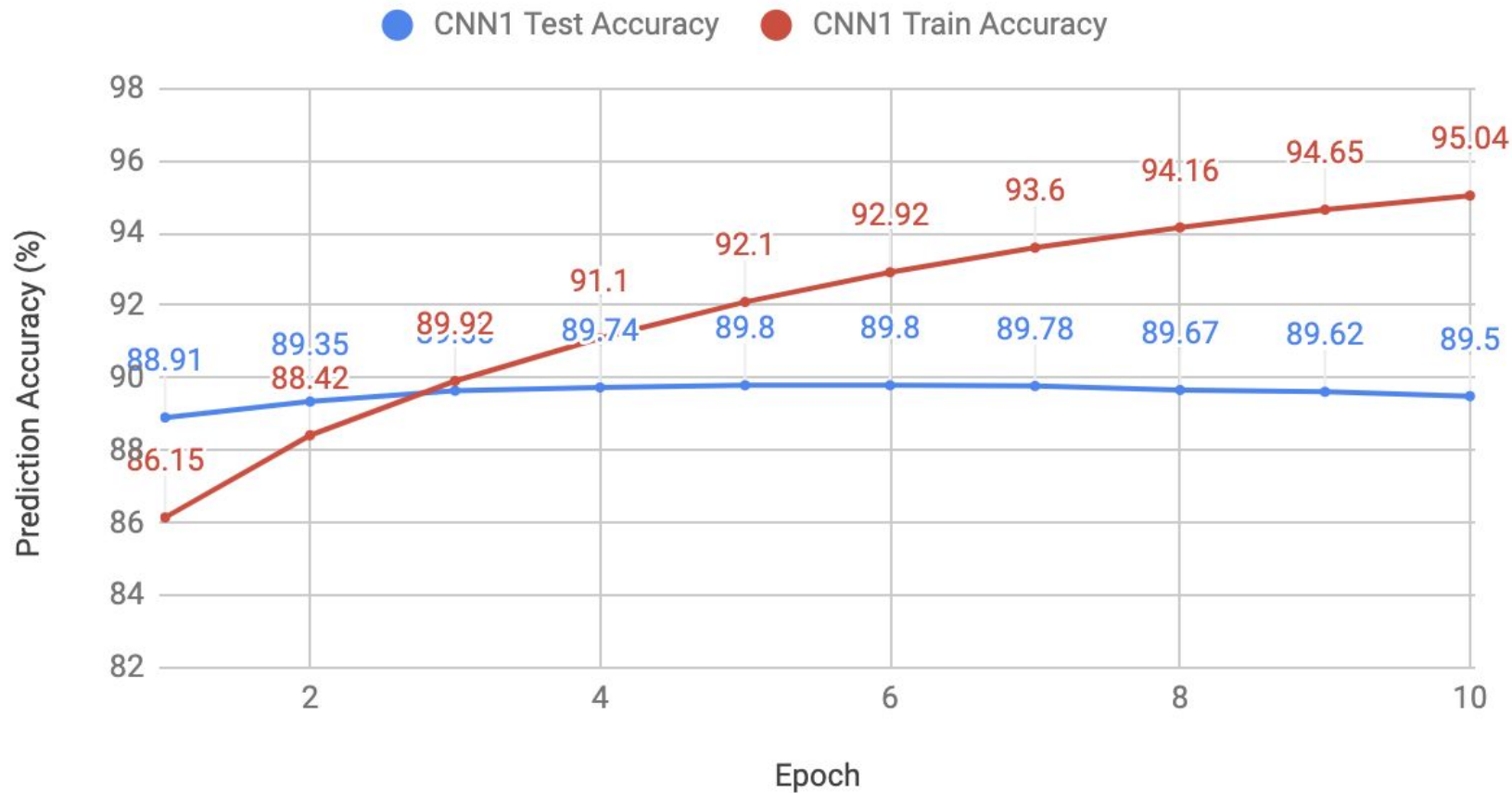
### Dense Layer

128 node densely connected layer, as is typical at the end of convolutional neural networks.

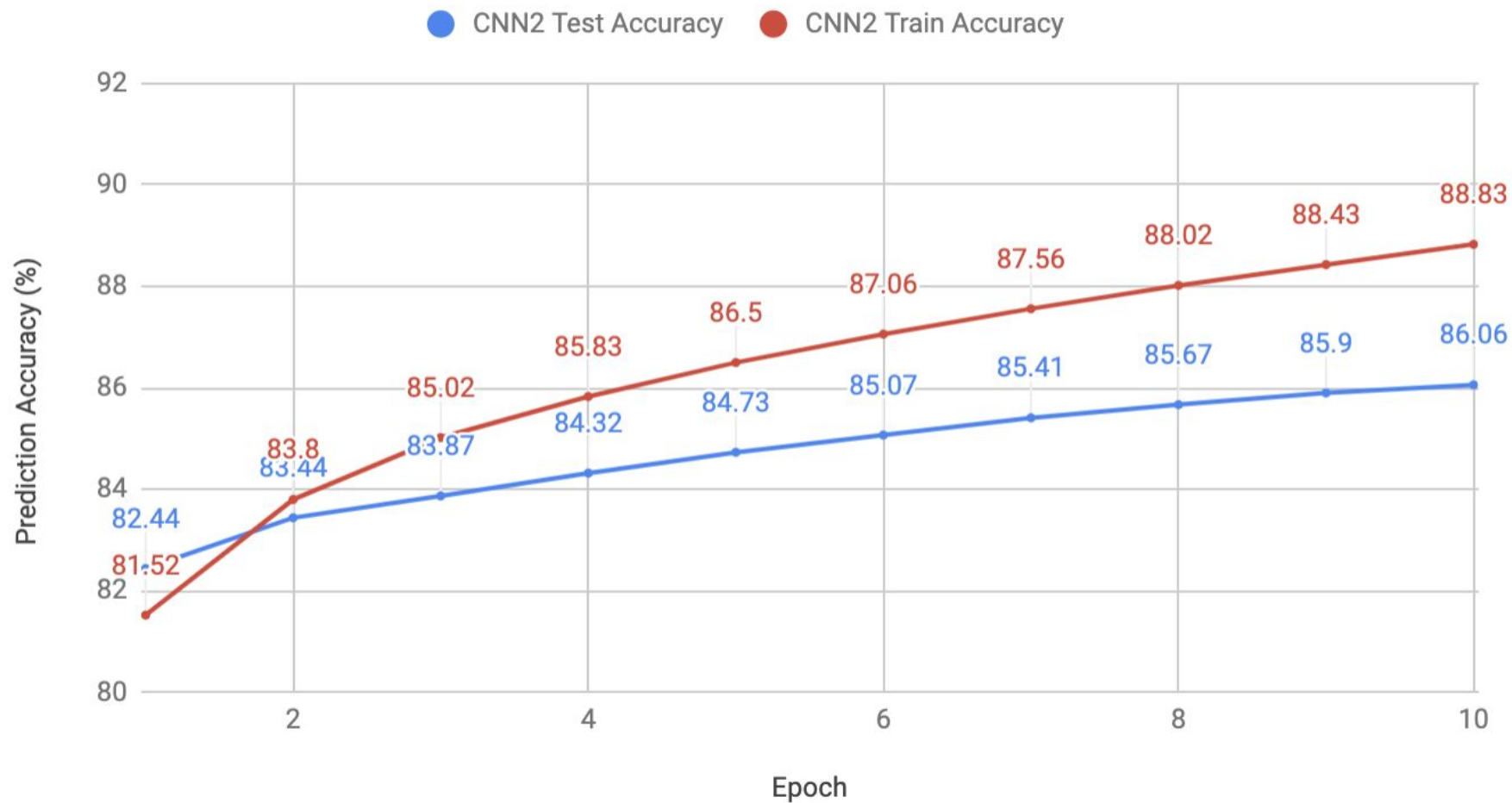
### Dense Output Layer

Another densely connected layer with 10 output nodes, one for each potential class in the Fashion MNIST classification.

# CNN1 - 32 filters, 3x3 Kernel

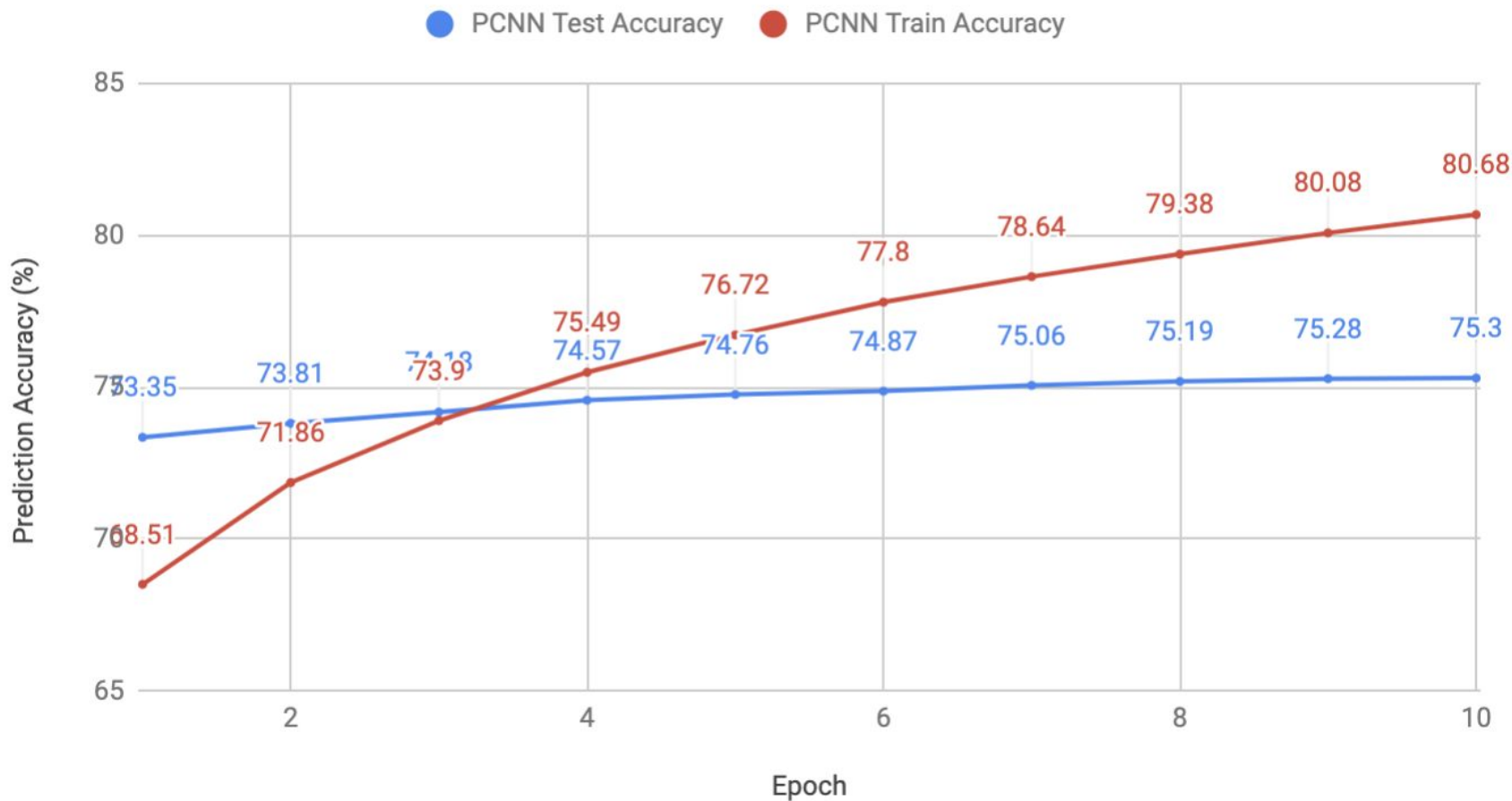


# CNN2 - 1 filter, 3x3 Kernel





# PCNN Accuracy



	CNN1	CNN2	PCNN
Training Loss	0.14	0.31	0.55
Training Accuracy	95.04%	88.83%	80.68%
Testing Loss	0.37	0.38	0.76
Testing Accuracy	89.50%	86.06%	75.30%
Training Time (per epoch)	48.72	9.07	129.81
Testing Time (per epoch)	1.42	0.37	11.92