COMPLEX INPUT CONVOLUTIONAL NEURAL NETWORKS FOR WIDE ANGLE SAR ATR

Michael Wilmanski, Chris Kreucher & Alfred Hero

University of Michigan
Ann Arbor, MI 48109

Integrity Applications Incorporated
Ann Arbor MI, 48108

wilmansk@umich.edu

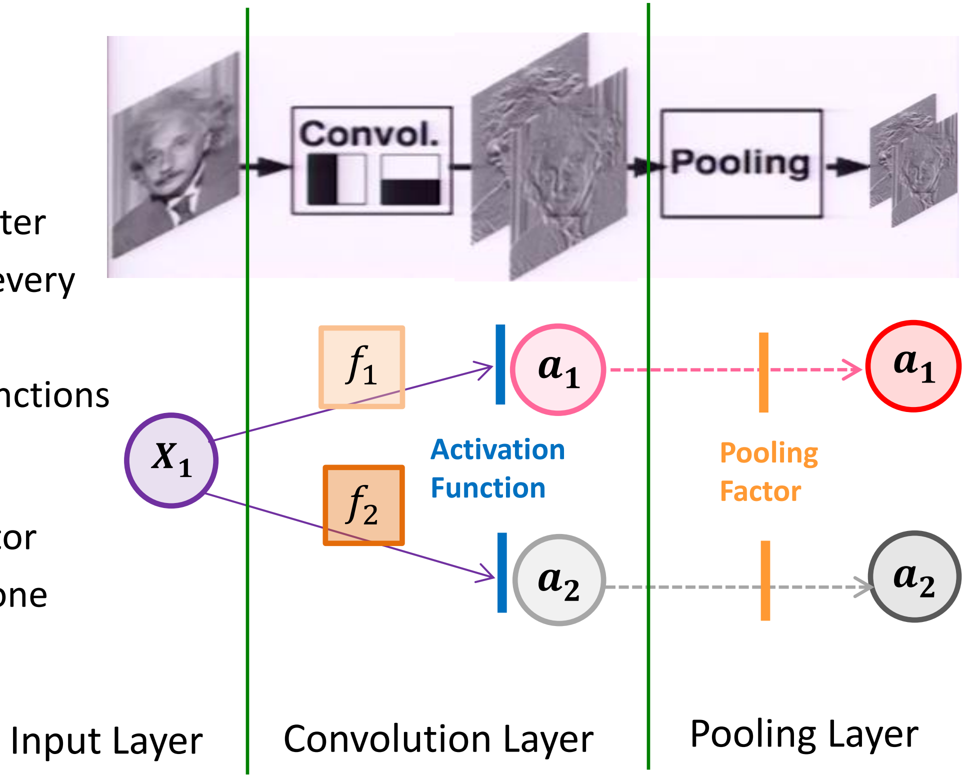
IEEE Global Signal & Information Processing Conference
December 2016

Overview

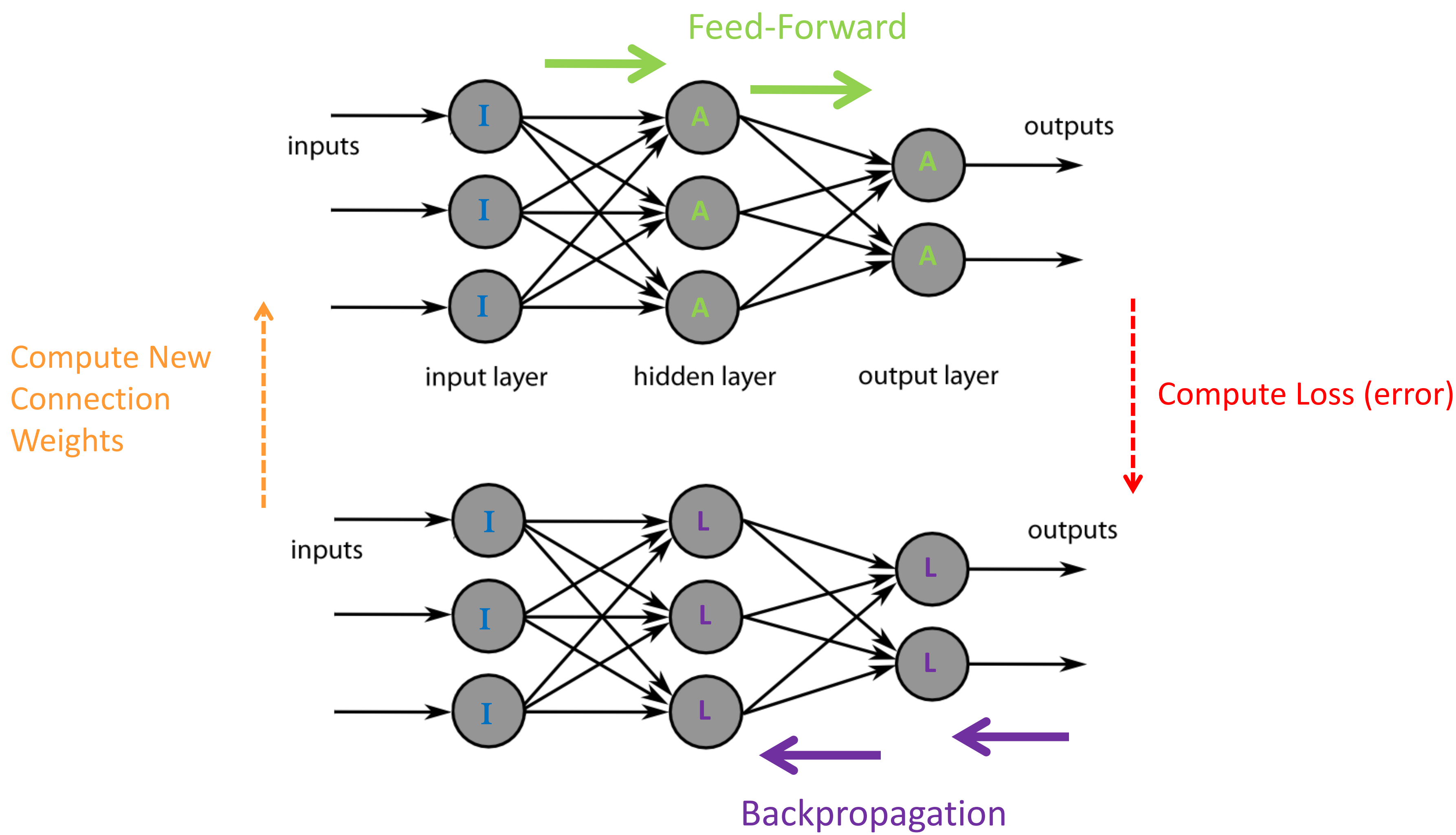
- Overview of standard convolution neural networks
- Explanation of complex-value deep learning problem
- Description of proposed implementation
- Performance comparison on 6-class data set

Convolutional Neural Network

- A **convolutional neural network** (or **CNN**) is a type of artificial neural network where the individual neurons are tiled in such a way that they respond to overlapping regions in the visual field
- Different types of layers
  - Input layer
  - Convolutional layer
    - Each connection contains a weighted filter
    - Each incoming neuron is connected to every outgoing neuron
    - Outgoing neurons contain activation functions
  - Pooling layer
    - Inputs are downscaled by a pooling factor
    - Each incoming neuron is connected to one outgoing neuron
  - Output



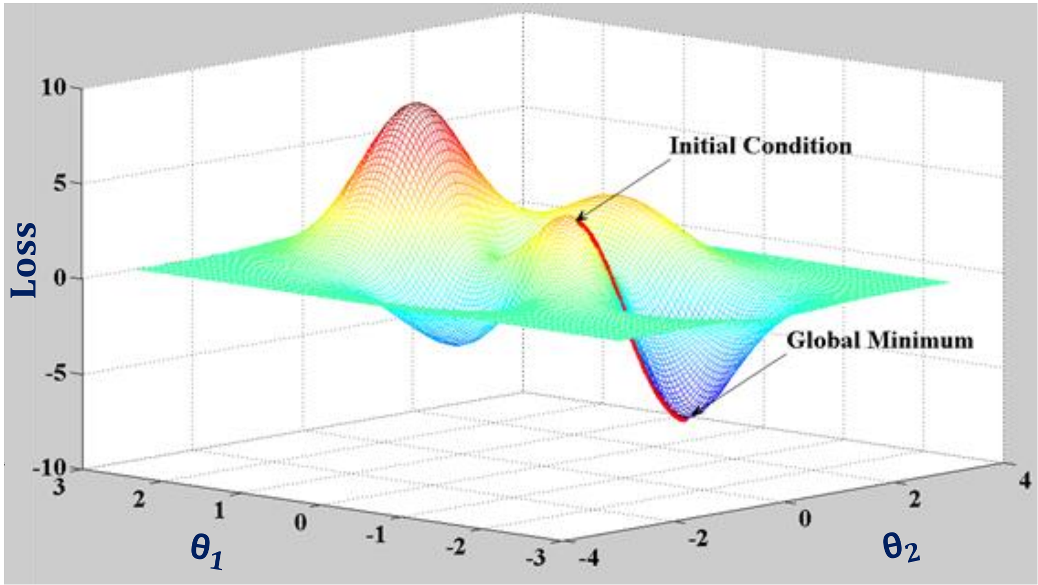
Backpropagation Algorithm Visualization



Gradient Descent

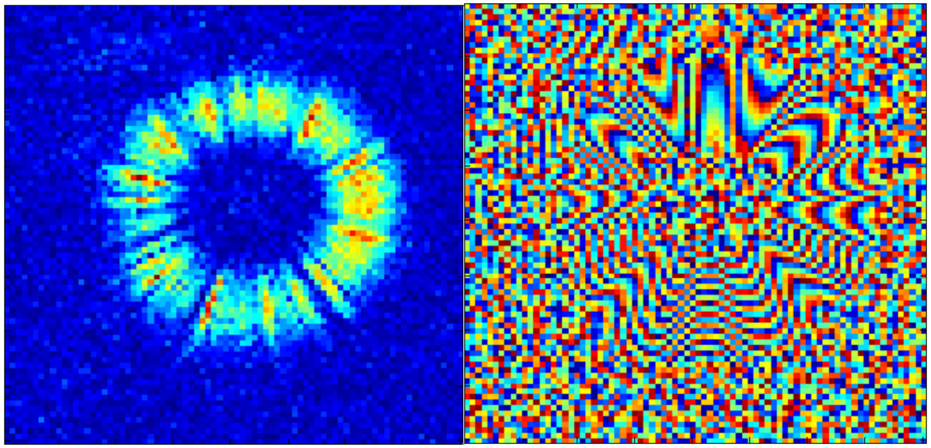
- The **Gradient Descent** algorithm aims to minimize **loss**, where loss is a function of the network's learned weights & current inputs
- The **learning rate** represents the step size taken in the direction of the gradient (downward)

Visualization of Gradient Decent with 2 weights



Complex-Value Deep Learning

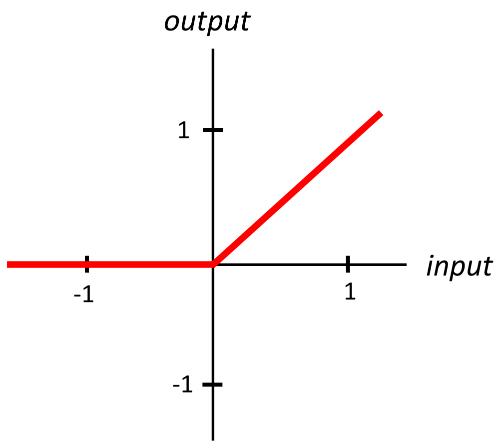
- Motivation: A significant amount of information is discarded when using only magnitude of complex imagery
- Problem: Modern activation functions do not work with complex numbers
  - Activation functions need to be nonlinear
  - Activation functions need to be differentiable



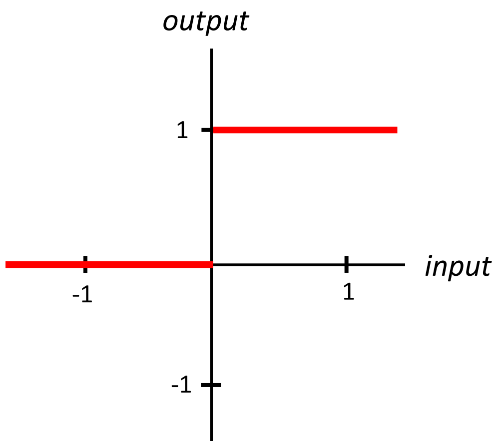
Magnitude (L) & phase (R) of tophat1

Activation Function Problem

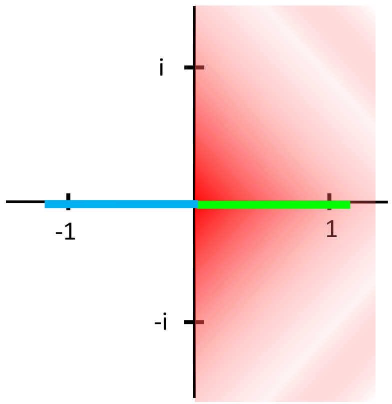
- Proposed solutions have negative consequences
  - Splitting real & imaginary parts & processing each as a real number quickly distorts the phase
  - A phase-only activation function discards a significant amount of useful magnitude information
  - A straight-forward generalization doesn't converge in training



Rectified Linear Unit activation function



Rectified Linear Unit derivative



Complex input

Proposed Implementation

- Proposed implementation is fully-complex at input & first convolution layer
- Let a_i & b_i be filters for real & imaginary parts respectively of input x for ith node
- Let A_i & B_i be the respective outputs for a_i & b_i

$$A_i = a_i * \mathcal{R}(x) \qquad \& \qquad B_i = b_i * \mathcal{I}(x)$$

- f is the activation function, which can be thought of as taking the absolute value of the complex convolution

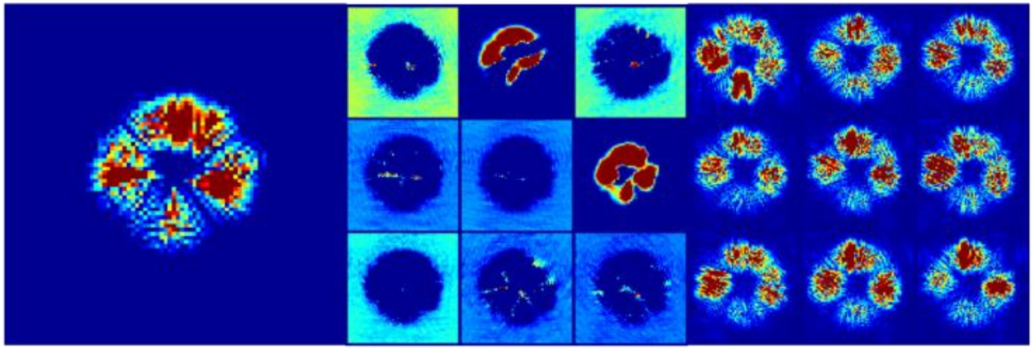
$$f(A_i, B_i) = \sqrt{A_i^2 + B_i^2}$$

Training Derivatives

- Activation functions must be nonlinear & differentiable, making use of complex numbers difficult
- The filters a_i & b_i can be trained with the following partial derivatives from the proposed activation function

$$\frac{\partial f(A_i, B_i)}{\partial A_i} = \frac{A_i}{f(A_i, B_i)} \quad \& \quad \frac{\partial f(A_i, B_i)}{\partial B_i} = \frac{B_i}{f(A_i, B_i)}$$

- After training, activations from complex-input CNN's look distinctly different from that of magnitude-only CNN's



Magnitude-detected input (L), Magnitude-only CNN activations (M), Complex-input CNN activations (R)

Normalization

- Normalization typically involves normalizing input images such that the range between max & min pixels is 1 while mean is 0

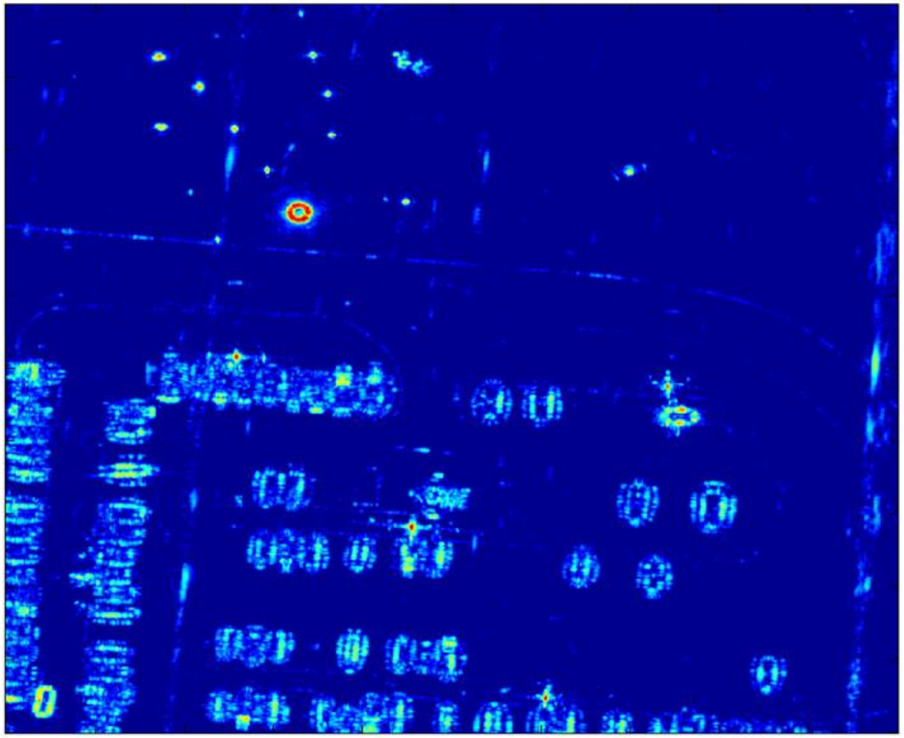
$$p_n(i,j) = \frac{p(i,j) - \frac{1}{N}\sum_{i,j} p(i,j)}{\max\limits_{i,j} p(i,j)}$$

- Subtracting the mean distorts the phase, so we normalize by magnitude only

$$p_n(i,j) = \frac{p(i,j)}{\max\limits_{i,j} |p(i,j)|}$$

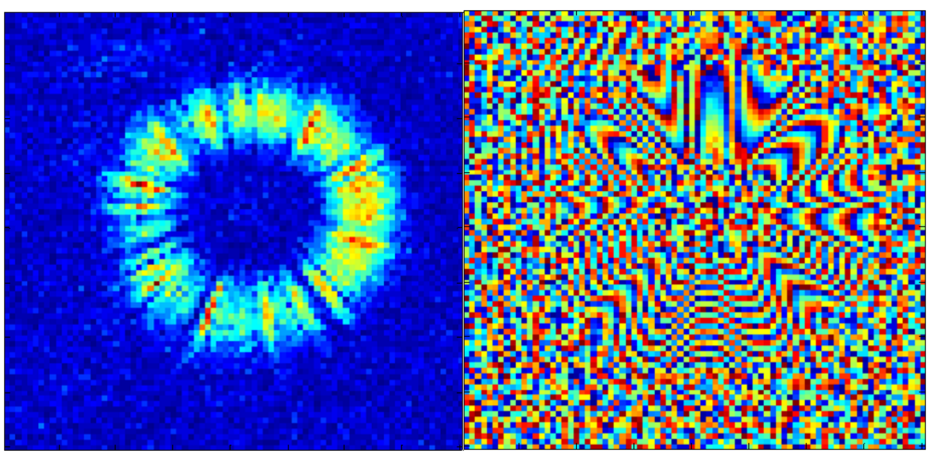
GOTCHA

- The GOTCHA publicly released data set was collected in 2008 by the Air Force Research Laboratory (AFRL)
- The data set consists of SAR phase history data collected at X-band with a 640 MHz bandwidth with full azimuth coverage at 8 different elevation angles with full polarization
- The imaging scene consists of numerous civilian vehicles and calibration targets
- The data set is distributed with a basic backprojection function for synthesizing images

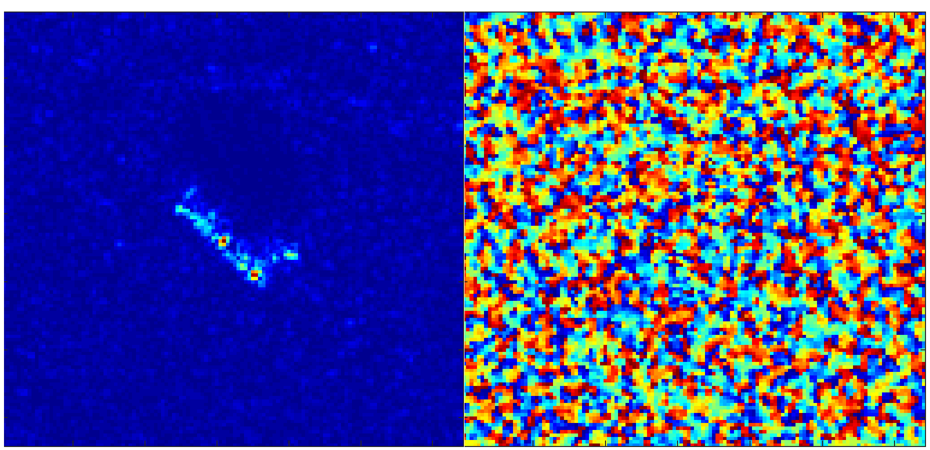


Importance of Phase Structure

- GOTCHA was chosen over competing SAR data sets like MSTAR because there is visible structure to the phase
- Complex-input result on MSTAR was overfit compared to magnitude-only
  - Complex-input accuracy: ~98% testing, 100% training
  - Magnitude-only accuracy: ~99% testing, 100% training

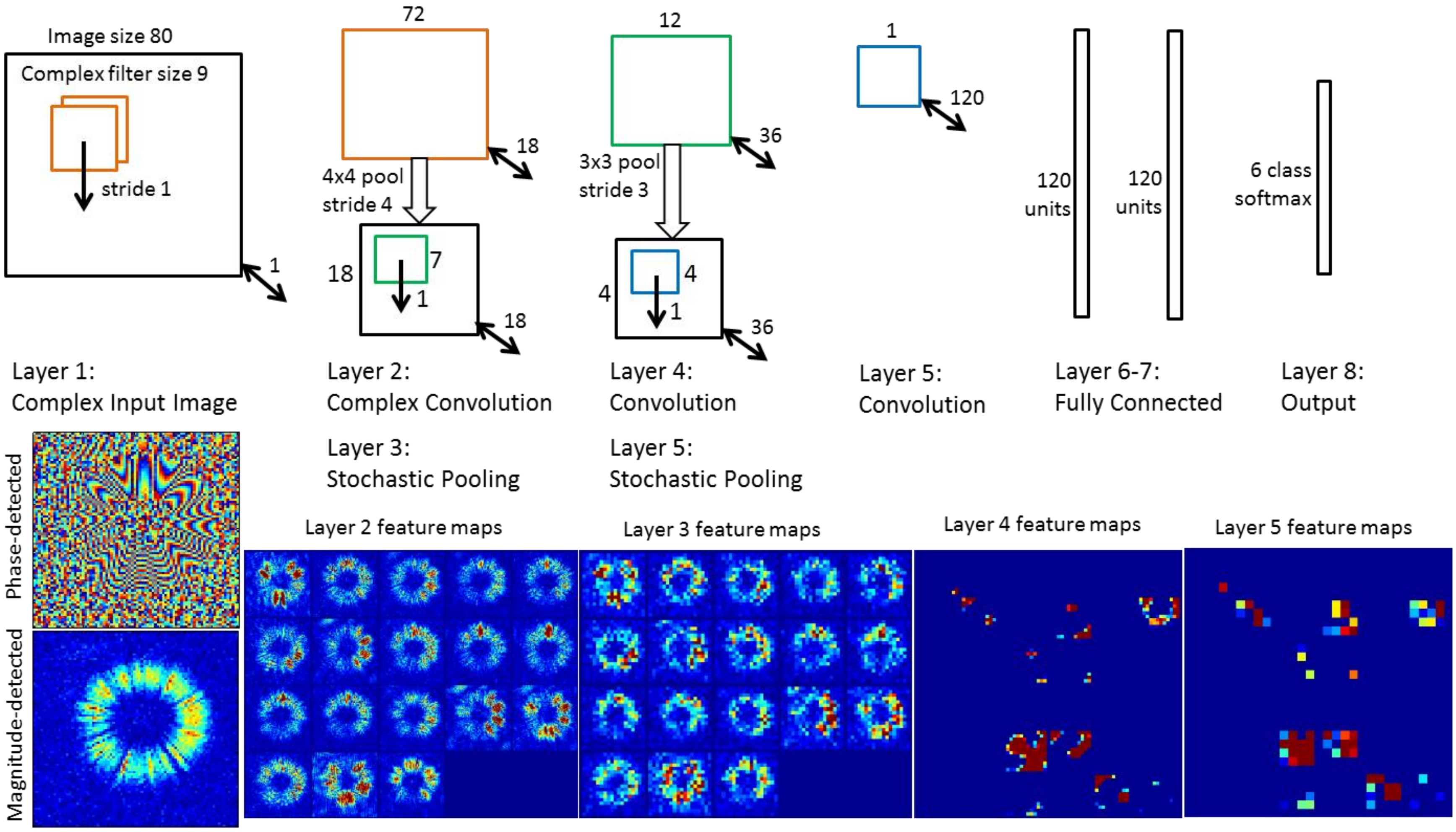


Magnitude (L) & phase (R) of tophat1 from GOTCHA



Magnitude (L) & phase (R) of 2S1 from MSTAR

Complex-input CNN Structure



Results

- Magnitude-only accuracy
  - 100% training set (525/525)
  - 87.30% testing set (110/126)

ACTUAL \ PREDICTED	brud	dihed	scr1	stri	tophat	tri	%CORR
brud	6	0	0	0	0	0	100
dihed	0	6	0	0	0	0	100
scr1	0	0	5	0	1	0	83.3
stri	0	0	0	10	1	1	83.3
tophat	0	0	0	2	10	0	83.3
tri	0	0	0	0	10	74	88.1

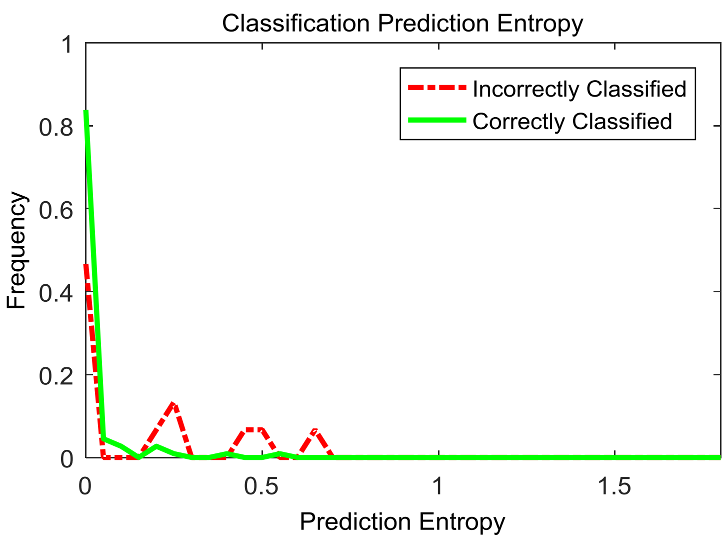
- Complex-input accuracy
  - 100% training set (525/525)
  - 99.21% testing set (125/126)

ACTUAL \ PREDICTED	brud	dihed	scr1	stri	tophat	tri	%CORR
brud	6	0	0	0	0	0	100
dihed	0	6	0	0	0	0	100
scr1	0	0	6	0	0	0	100
stri	0	0	0	12	0	0	100
tophat	0	0	0	1	11	0	91.7
tri	0	0	0	0	0	84	100

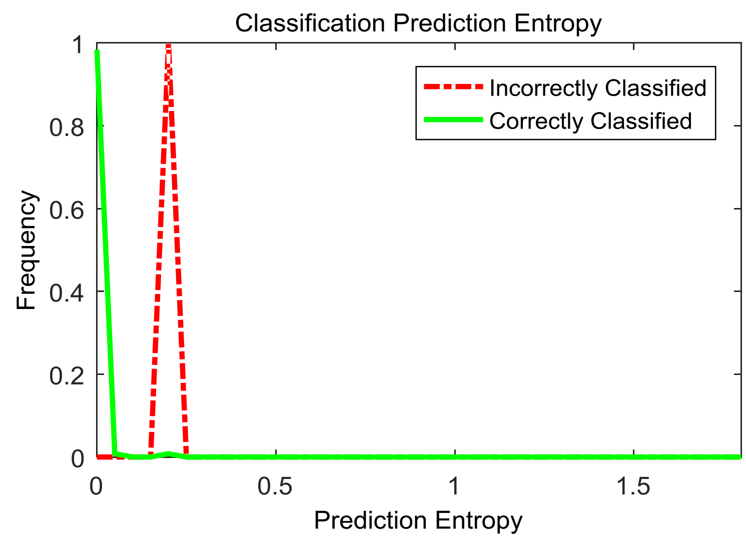
Prediction Entropy

- Prediction entropy is a useful metric for determining how certain a classifier is with its prediction
- High prediction entropy can signify to the user which predictions have a high probability of being incorrect
- For probability distribution $p(x)$, entropy can be quantized as

$$E\big(p(x)\big) = -\sum p(x)\log(p(x))$$



Histogram of prediction entropy for magnitude-only CNN



Histogram of prediction entropy for complex-input CNN