CNN TUTORIAL

Maria Ingold
12693772
Unit 10
Machine Learning
University of Essex Online
13 January 2025

CONTENTS

CNN Tutorial

Wang et al. (N.D.) explains Convolutional Neural Networks (CNN) and provides an interactive tutorial (Figure 1) broken into:

- **input layer**: here, RGB channels
- **convolutional layer**: learned kernels (weights) for classification
- **Rectified Linear Activation function (ReLU) layer**: non-linearity
- **pooling layer (Max-Pooling)**: reduce parameters and network computation
- **flatten layer:** convert 3D layer to 1D vector used by softmax
- **softmax operation**: Ensure CNN outputs sum to 1; good for probabilities

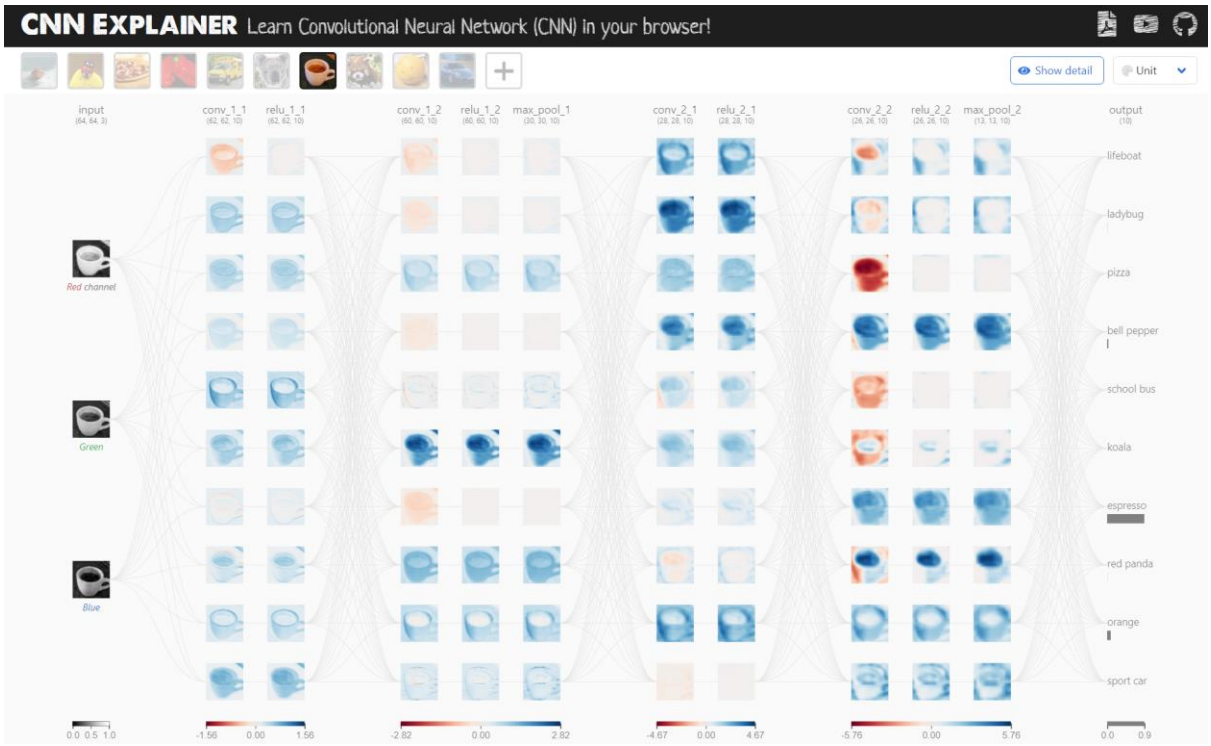


FIGURE 1 | CNN Explainer

Various terms are defined as:

- **tensor**: n-dimensional matrix
- **neuron**: multi-input function with single output
- **activation maps**: neuron outputs
- **layer**: neuron collection with same operation and hyperparameters
- **kernel weights and biases**: unique to each neuron, tuned in training
- **differentiable score function**: class score

Hyperparameters (Figure 2) are defined as:

- **padding:** zero-padding adds zeros around input edges
- **kernel size:** dimensions of input sliding window; smaller is more accurate
- **stride:** smaller stride means more features learned

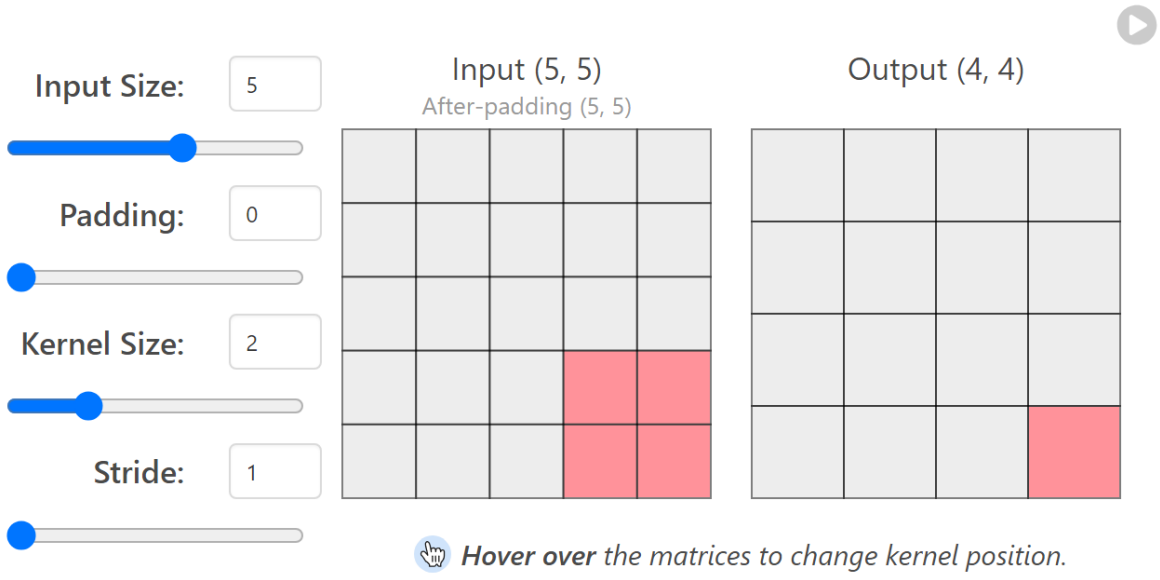


FIGURE 2 | Hyperparameters

Input layer

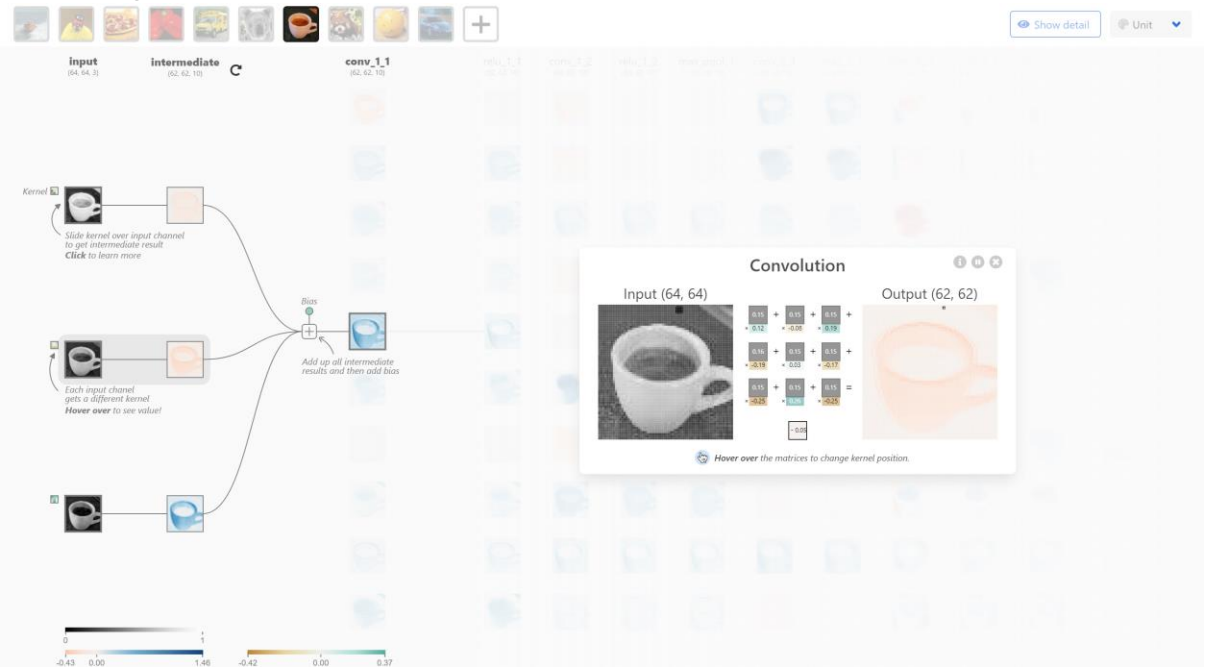


FIGURE 3 | Input Layer to Convolutional Layer

Figure 2 shows the input layer has three neurons corresponding to colour channels: red, green, and blue.

Convolutional Layer

The input layer connects to the first convolutional layer which has ten neurons. The convolutional layer contains the weights that do the feature extraction for classifying images. The link that connects it is a unique kernel; there are 3 x 10 = 30.

ReLU Activation

Figure 4 shows the ReLU activation. Its non-linearity increases accuracy and training speed of CNNs. It is performed after every convolutional layer in this network.

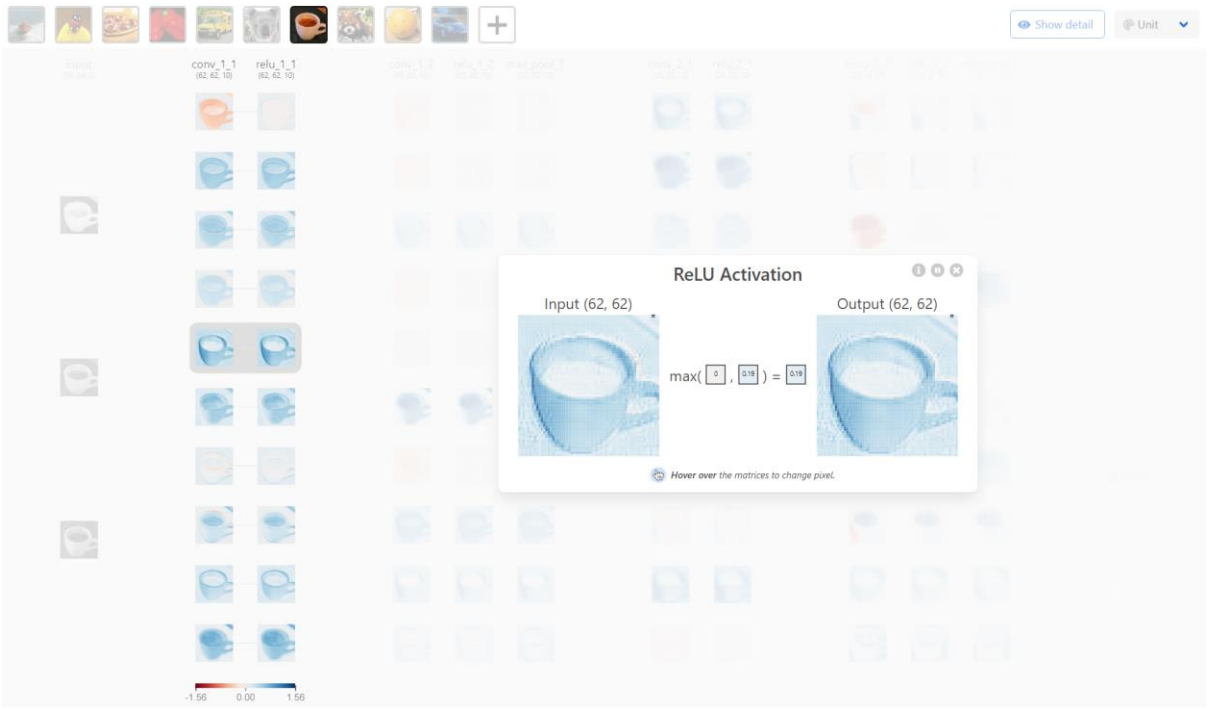


FIGURE 4 | ReLU Activation

ReLU feeds into an intermediate layer then into next convolutional layer (Figure 5).

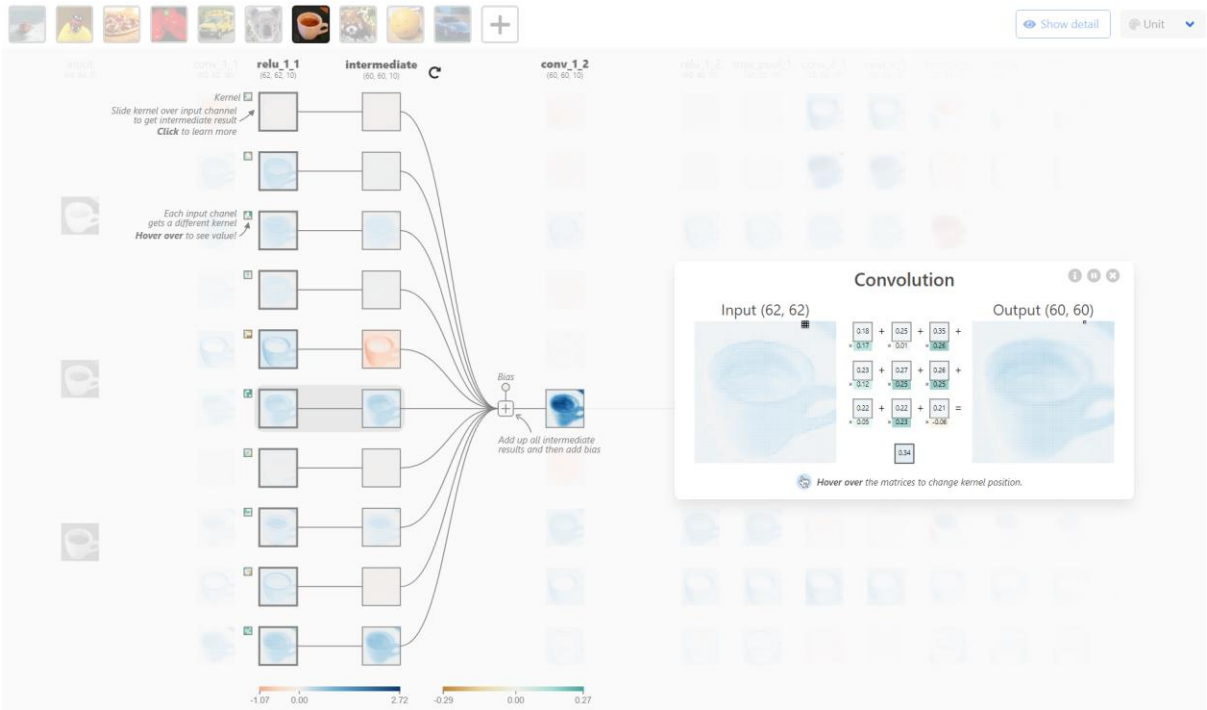


FIGURE 5 | ReLU into Convolution

Pooling Layer

Max-Pooling (Figure 6, 7) decreases network spatial extent using kernel size (e.g. 2x2) and stride length (e.g. 2) to discard (75% of) activations, thereby increasing computational efficiency and avoiding overfitting.

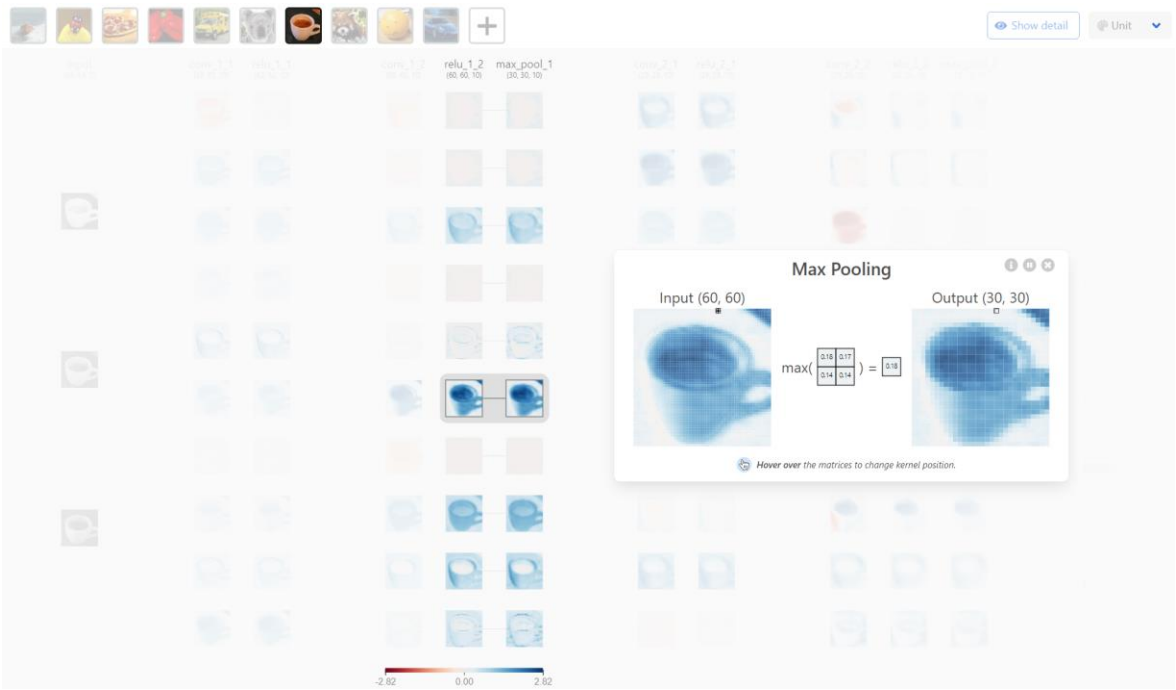


FIGURE 6 | Max-Pooling

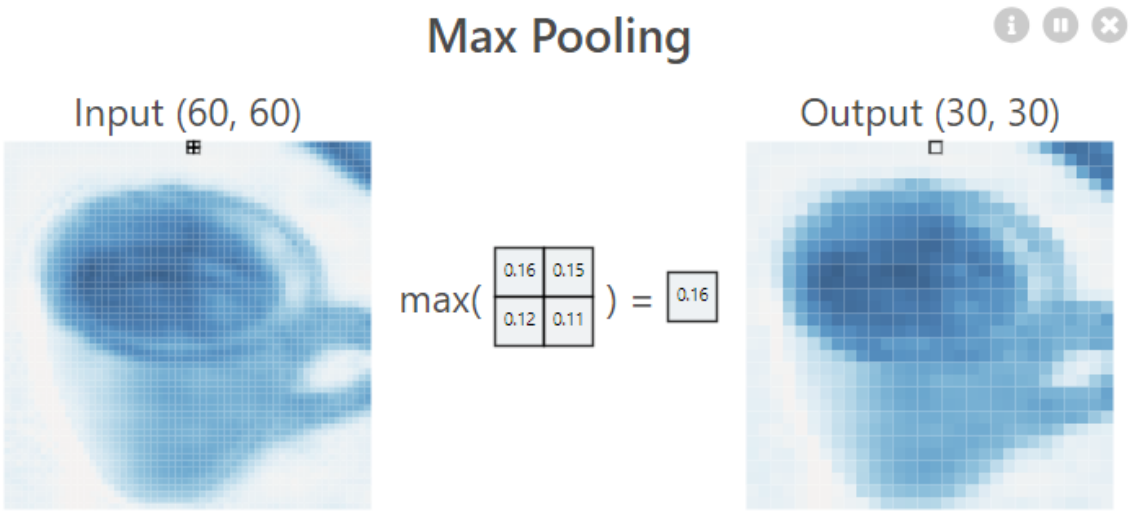


FIGURE 7 | Max-Pooling close-up

Flatten Layer

Flattens three dimensions into one-dimensional vector required for softmax classification.

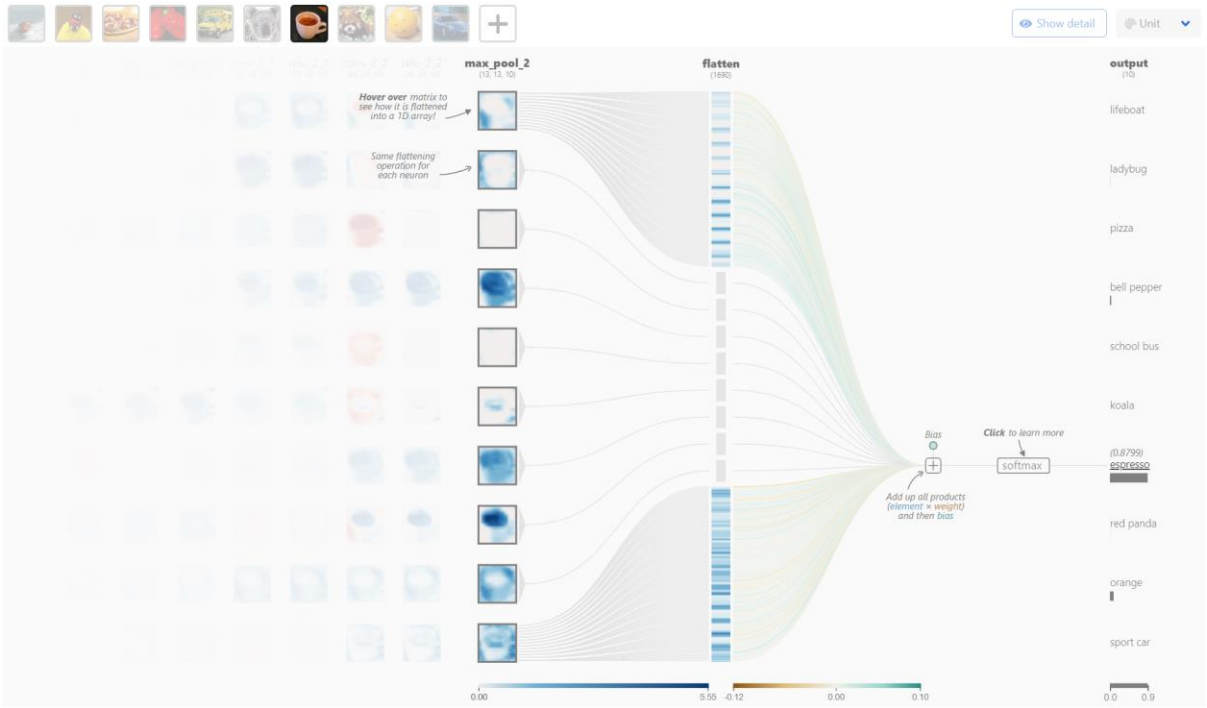


FIGURE 8 | Flatten layer

Softmax

Figure 9 shows the final softmax classification step. Each class corresponds to probability.

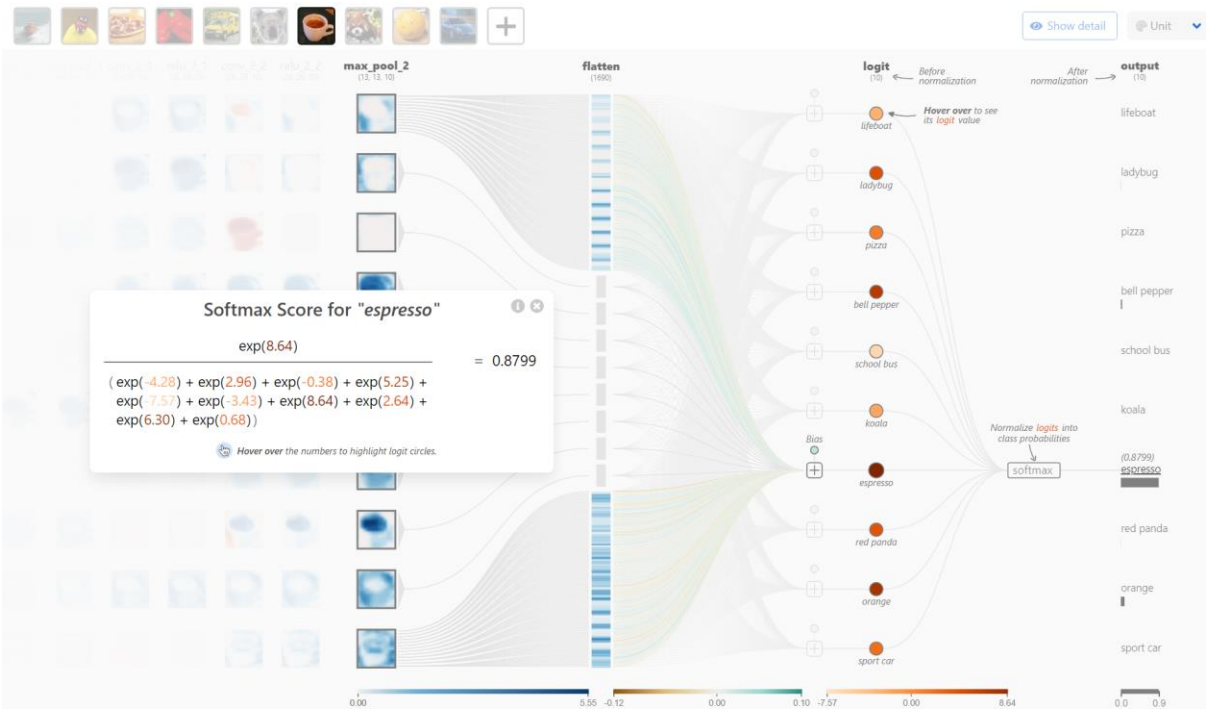


FIGURE 9 | Softmax operation

Testing

Miata (failed as ladybug)

1993 Mazda Miata in black and tan (Figure 10) was defined as a ladybug (Figure 11) instead of a sportscar. Perhaps overfitting as this image is rectangular, not square. Also, sportscar is oriented in opposite direction and is blue.

FIGURE 10 | Source image: Miata

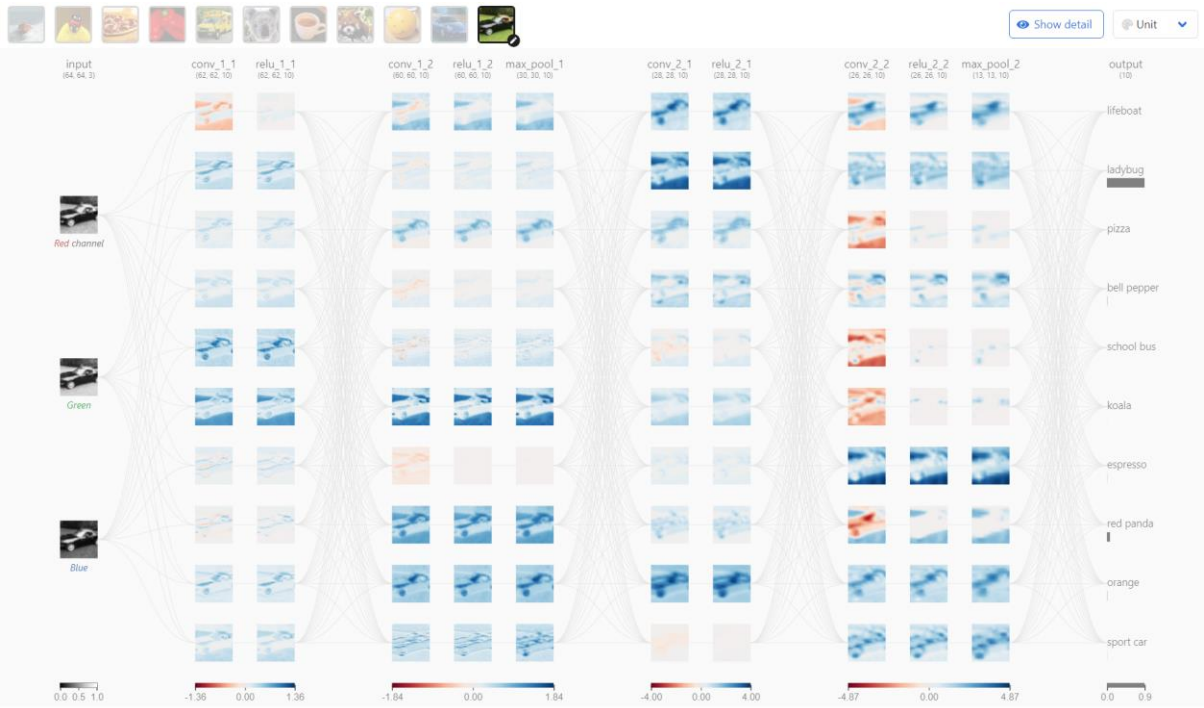
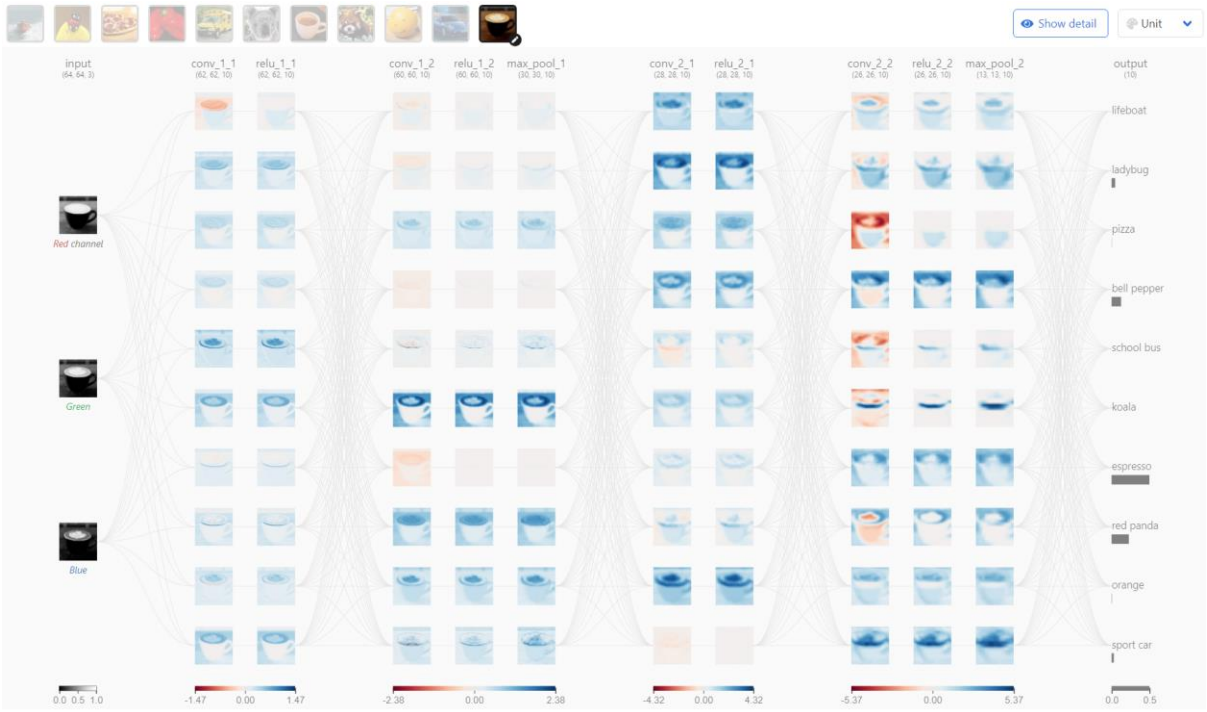


FIGURE 11 | Miata defined as ladybug not sportscar

Flat White (correct as espresso)

FIGURE 12 | Source image: Flat white

This flat white is correctly interpreted as an espresso, but also a bit of red panda. The cup is oriented similarly but is a different colour, black instead of white. However, the rest of the structure is similar.


Show detail
Unit
input
(64, 64, 3)
conv_1_1
(62, 62, 10)
relu_1_1
(62, 62, 10)
conv_1_2
(60, 60, 10)
relu_1_2
(60, 60, 10)
max_pool_1
(30, 30, 10)
conv_2_1
(28, 28, 10)
relu_2_1
(28, 28, 10)
conv_2_2
(26, 26, 10)
relu_2_2
(26, 26, 10)
max_pool_2
(13, 13, 10)
output
(10)
Red channel
Green
Blue
lifeboat
ladybug
pizza
bell pepper
school bus
koala
espresso
red panda
orange
sport car
0.0 0.5 1.0
-1.47 0.00 1.47
-2.38 0.00 2.38
-4.32 0.00 4.32
-5.37 0.00 5.37
0.0 0.5

References

Wang, J. et al. (N.D.) *CNN Explainer*. Available from: https://poloclub.github.io/cnn-explainer/ [Accessed 29 December 2024].