Image Classification Using CNN

Vignesh shanmugam

vshangmug@buffalo.edu

Chennakesava sukesh myla

cmyla@buffalo.edu

code: https://drive.google.com/drive/folders/1xpPNU-eRRGOQUZZWum_5-p15c4CnjmkH?usp=share_link

Abstract

We know that the image classification is a common problem. In this paper we are using the convolution neural network along with data augmentation and tuning of hyper parameters to get better accuracy than the standard convolutional neural network models. We are then using the CNN model trained for the image classification on three different datasets to get a better view on how our model works with all types of datasets.

1. Introduction

Image classification with deep learning involves the use of convolutional neural networks to extract meaning from an image or to get the features useful for categorizing and labeling a given image to assign a label or to identify it as one of the several fixed classes. Image classification is one of the classic problems in computer vision. The given image needs to be identified and recognize which group it belongs from one of the several fixed classes. So to recognize which category the given image belongs we try to extract features from the image like the texture or some unique trait in the feature which extinguishes it from the rest is collected and using this features we can categorize the given image into one of the several fixed classes.

In CNNs, the nodes in the hidden layers don't always share their output with every node in the next layer (known as convolutional layers). Deep learning allows machines to identify and extract features from images.

2. Problem statement

One of the problems we face during the image classification are intra-class variation where some of the images of different classes are identical. The scale variation is a very common problem in image classification where the images of the same class are having multiple sizes. We have the view-point variation where an object can be oriented in different dimensions with respect to how a image is taken. There might be a lot of images where we want to classify them because of the object or subject is being hidden behind another object or subject which makes it difficult to classify and the variation in intensity of the images or the intensity level of pixels of each image can also effect the classification of image. For all the problems mentioned above our model is able to handle and classify.

3. Best accuracy model

CNN or Convolutional Neural Network is a Deep Learning algorithm and a subset of machine learning. A CNN model can be used in different applications and data types. Usually, CNN is used for image classification (or) process involving processing of pixel data

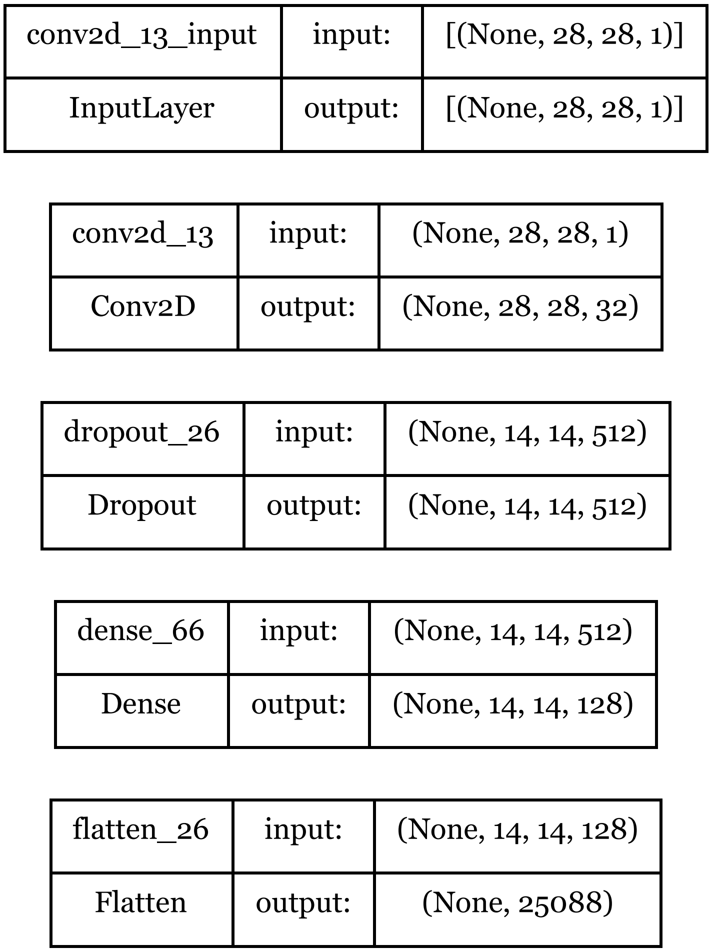
Applications:

Face detection

Object Recognition

Digit Classification

Biometrics



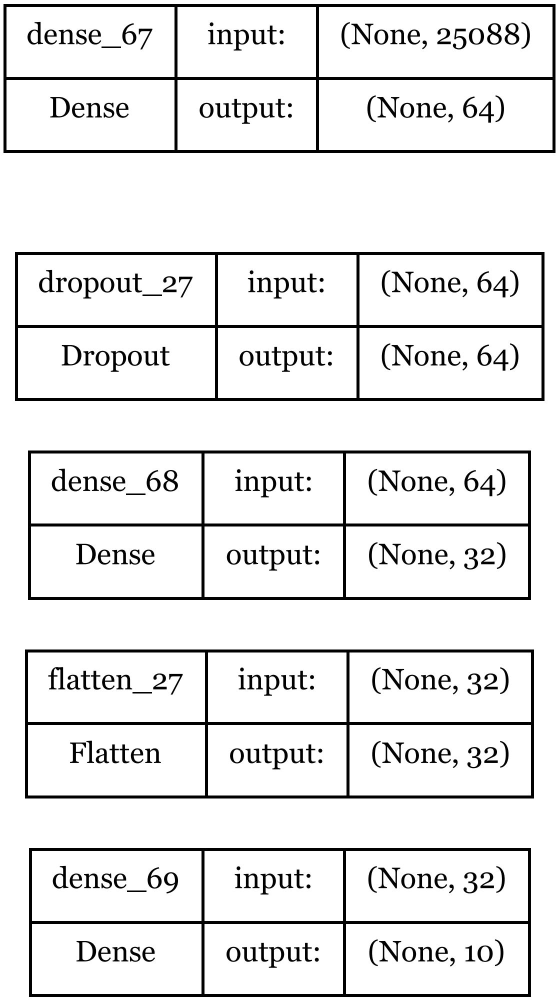
conv2d_13_input	input:	[(None, 28, 28, 1)]
InputLayer	output:	[(None, 28, 28, 1)]

conv2d_13	input:	(None, 28, 28, 1)
Conv2D	output:	(None, 28, 28, 32)

dropout_26	input:	(None, 14, 14, 512)
Dropout	output:	(None, 14, 14, 512)

dense_66	input:	(None, 14, 14, 512)
Dense	output:	(None, 14, 14, 128)

flatten_26	input:	(None, 14, 14, 128)
Flatten	output:	(None, 25088)



dense_67	input:	(None, 25088)
Dense	output:	(None, 64)

dropout_27	input:	(None, 64)
Dropout	output:	(None, 64)

dense_68	input:	(None, 64)
Dense	output:	(None, 32)

flatten_27	input:	(None, 32)
Flatten	output:	(None, 32)

dense_69	input:	(None, 32)
Dense	output:	(None, 10)

4. Data augmentation

Data augmentation is used to artificially increase the size of the training set by making modification to the existing dataset which will helps in preventing overfitting and enhance the performance of the model. It also reduces the operation cost related to the lack of data.

4.1. Rescaling

We use the rescaling to get the pixel values in the targeted range. In this model we are using rescaling = 1./255 which will make the pixel values in the range [0, 1]

4.1.1. Rotation

The Rotation is used to rotate the given image in either the clockwise and anticlockwise by some number of degrees which changes the orientation of the image in the frame. In our model we used the 45 degree to add in extra data for our train data.

4.1.2. Brightness and zoom

The Brightness is a important factor as we are not sure if the images of a dataset are taken with good lighting so to make sure we add a brightness range for the image.

The image can be zoomed in and out using the zoom.

4.1.3. Flipping and height adjustment

The image can be flipped either vertically or horizontally depending on the image if we take a bike or any other vehicle and flip it vertically there is no use.

The height adjustment is used to adjust the height of the given image.

5. Tuning hyper parameters

In order to get the best accuracy model the hyperparameters of the model such as Activation functions, Initializers, Kernel Size, Optimizers are changed and tested on fashion_mnist dataset. Tuning hyper parameter values to compare the best accuracy model

Tuning drop out layer value	
Value	Accuracy
0.2	91.64%
0.4	91.1%
0.6	91.01%
0.5	92.01%

Changing activation function	
Name	Accuracy
SeLu	90.7%
Tanh	90.38%
GeLu	91.32%
ReLu	92.01%

Changing optimizer functions	
Name	Accuracy
SGD	88.35%
RMSprop	87.69%
ADAdelta	71.6%
ADAM	92.01%

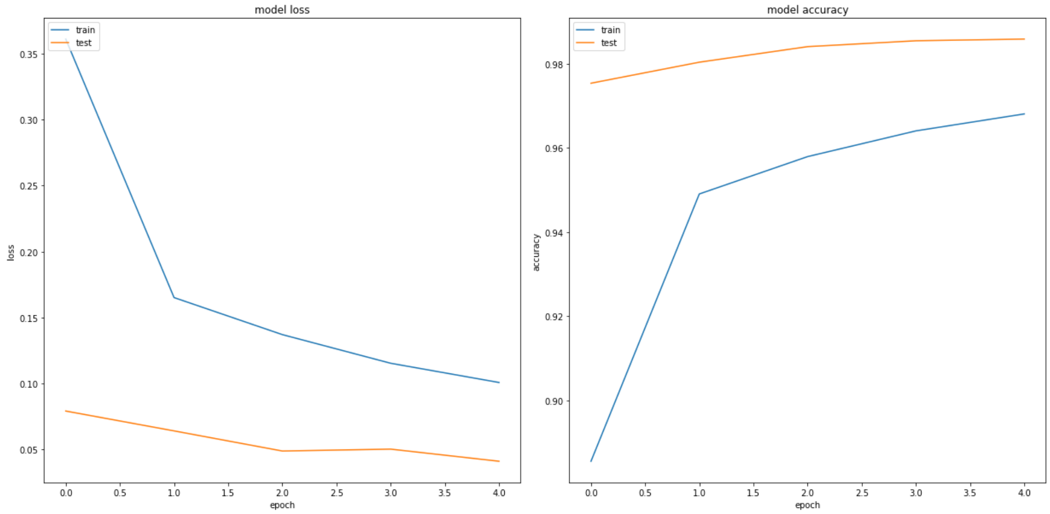
Changing kernel weight initializations	
Name	Accuracy
Glorot Normal	91.22%
Glorot Uniform	91.14%
Random Normal	91.54%
He Normal	92.01%

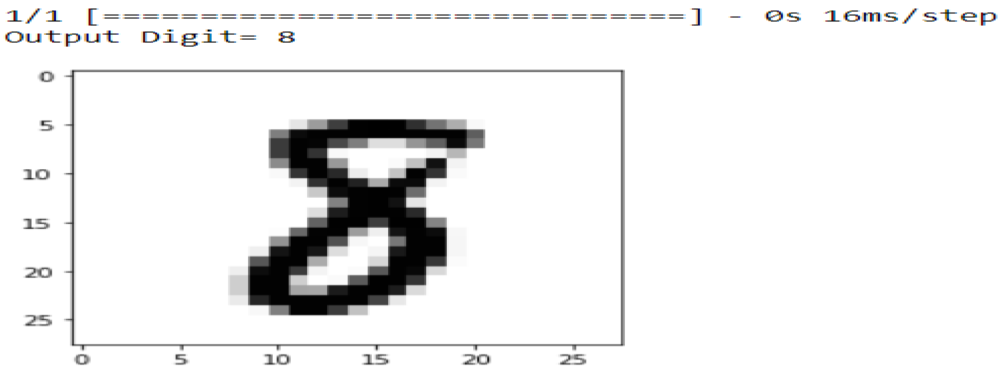
Changing maxpool layer and padding values		
Maxpool	padding	Accuracy
Same	Same	91.39%
Valid	Valid	91.43%
Same	Valid	91.28%

Changing kernel size	
Kernel size	Accuracy
(1,1)	86.16%
(2,2)	91.42%
(4,4)	91.57%
(3,3)	92.01%

6.Results

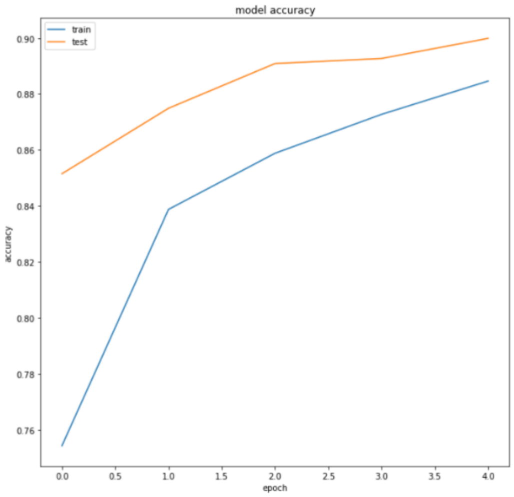
Accuracy Graphs and Results for MNIST

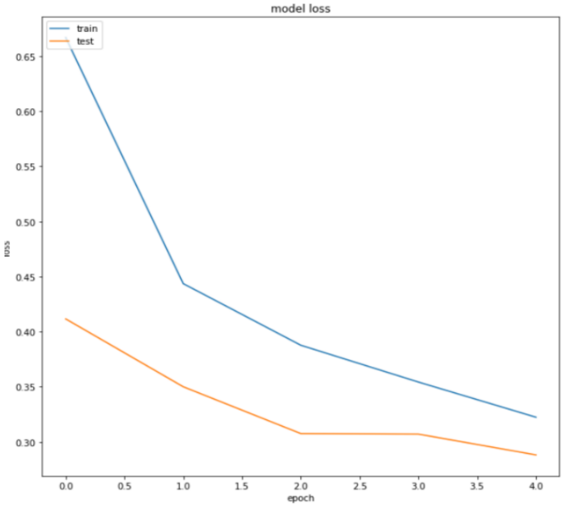


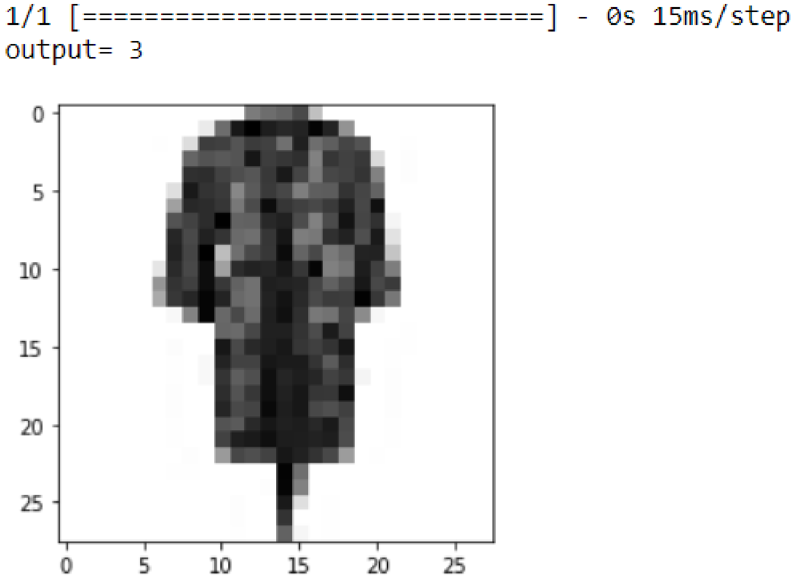


```
1/1 [==============================] - 0s 16ms/step
Output Digit= 8
```

Accuracy Graphs and Results for Fashion MNIST

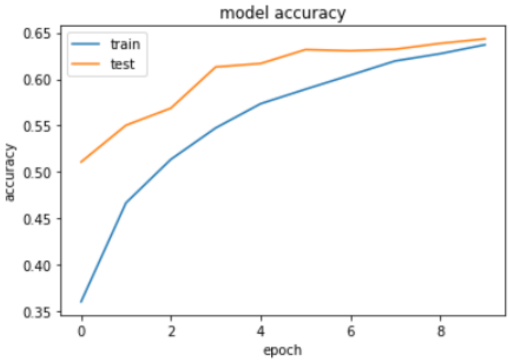


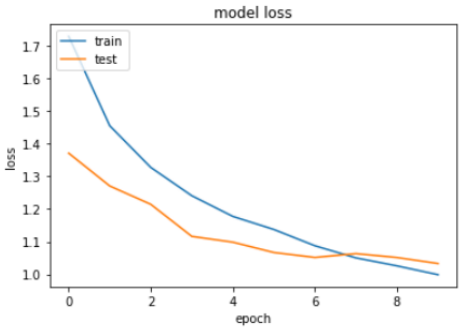




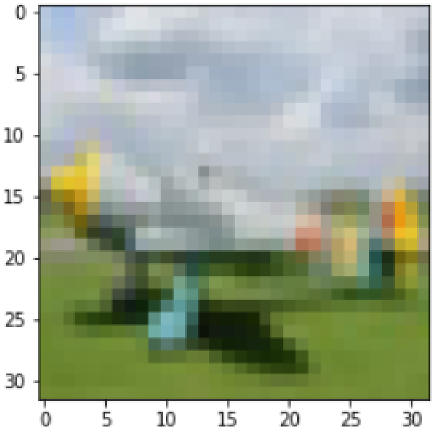
```
1/1 [==============================] - 0s 15ms/step
output= 3
```

Accuracy Graphs and Results for CIFAR10





```
1/1 [==============================] - 0s 16ms/step
Output Digit= 1
```



7.References

References follow the acknowledgments. Use unnumbered first-level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to small (9 point) when listing the references. Note that the Reference section does not count towards the eight pages of content that are allowed.

[1] Keiron O'Shea, Ryan Nash, An Introduction to Convolutional Neural

Networks, 2015.

[2] Xizhou Zhu, Han Hu, Stephen Lin, Jifeng Dai, Deformable ConvNets

v2: More Deformable, Better Results, 2019.

[3] . E. Newman, M. Kilmer, L. Horesh, Image classification using local tensor

singular value decompositions (IEEE, international workshop on computational advances in multi-sensor adaptive processing. IEEE, Willemstad, 2018), pp. 1–5.

[4] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, Yichen Wei, Deformable Convolutional Networks, 2017.

[5] François Chollet, Xception: Deep Learning with Depthwise Separable Convolutions, 2017.

[6] Min Lin, Qiang Chen, Shuicheng Yan, Network In Network, 2013.

[7] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, Eugenio Culurciello, ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation, 2016.

[8] Sachin Mehta, Mohammad Rastegari, Linda Shapiro, Hannaneh Hajishirzi, ESPNetv2: A Light-weight, Power Efficient, and General Purpose Convolutional Neural Network, 2019.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition, 2015.

[10] X. Wang, C. Chen, Y. Cheng, et al, Zero-shot image classification based on deep feature extraction. United Kingdom: IEEE Transactions on Cognitive & Developmental Systems, 10(2), 1–1 (2018).

[11] A.A.M. Al-Saffar, H. Tao, M.A. Talab, Review of deep convolution neural network in image classification (International conference on radar, antenna, microwave, electronics, and telecommunications. IEEE, Jakarta, 2018), pp. 26–31.

[12] A.B. Said, I. Jemel, R. Ejbali, et al., A hybrid approach for image classification based on sparse coding and wavelet decomposition (Ieee/acs, international conference on computer systems and applications. IEEE, Hammamet, 2018)