Trained model for the supplied data set

Augmenting our data by inverting the original images. Turn the white pixels to black and vice-versa. Then we will train our network using the original and inverted images and I saw that classification is improved.

I trained a model using the LeNet network for 15 epochs. The reason for chosen this network that datasets contain grayscale images and this the only choice and performed well which give me on validation data 99.01% accuracy.

When classifying 8 new images of digit-written that weren't in the training set this model showed 90.19% accuracy on test set.
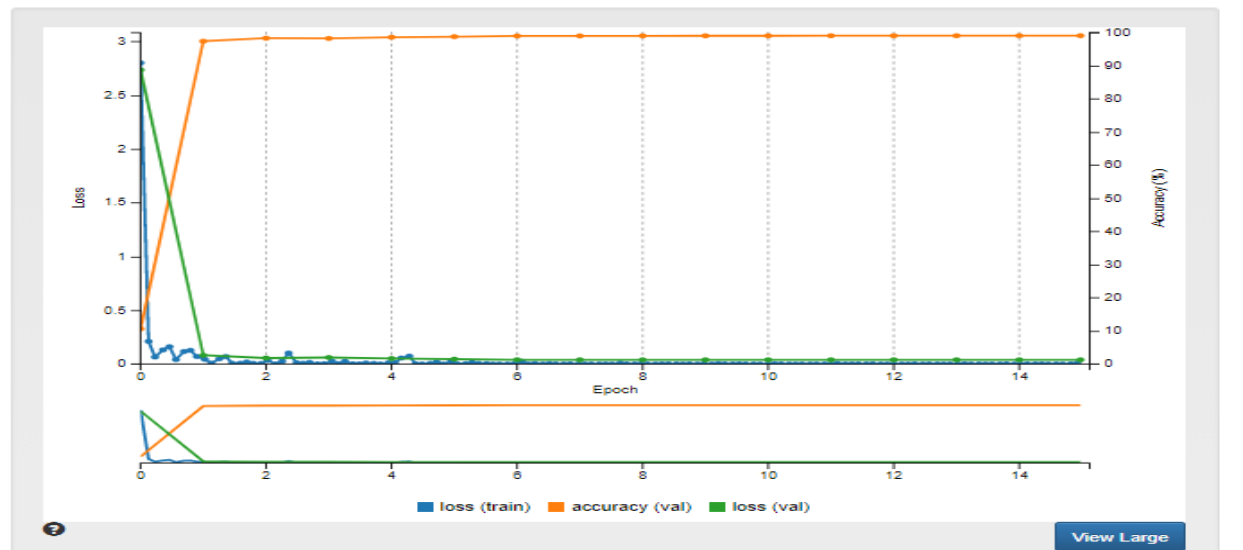

Fig 1.0

All classifications

	Path	Top predictions								
1	/data/notebooks/test_images/image-1-1.jpg	1	89.71%	7	10.08%	9	0.09%	4	0.07%	2	0.02%
2	/data/notebooks/test_images/image-2-1.jpg	2	99.67%	7	0.29%	0	0.02%	1	0.02%	3	0.0%
3	/data/notebooks/test_images/image-3-1.jpg	3	99.97%	2	0.02%	1	0.01%	5	0.0%	7	0.0%
4	/data/notebooks/test_images/image-4-1.jpg	4	99.97%	7	0.02%	1	0.01%	9	0.0%	5	0.0%
5	/data/notebooks/test_images/image-7-1.jpg	7	93.95%	1	5.67%	2	0.36%	0	0.01%	8	0.0%
6	/data/notebooks/test_images/image-8-1.jpg	8	100.0%	3	0.0%	2	0.0%	5	0.0%	0	0.0%
7	/data/notebooks/test_images/image-8-2.jpg	2	99.66%	8	0.12%	7	0.12%	1	0.05%	3	0.04%

When I write evaluate this me this message it give me 0.0 accuracy .

```
Calculating average inference time over 10 samples...
deploy: /opt/DIGITS/digits/jobs/20201124-070914-4b1c/deploy.prototxt
model: /opt/DIGITS/digits/jobs/20201124-070914-4b1c/snapshot_iter_21105.caffemodel
output: softmax
iterations: 5
avgRuns: 10
Input "data": 1x28x28
Output "softmax": 10x1x1
name=data, bindingIndex=0, buffers.size()=2
name=softmax, bindingIndex=1, buffers.size()=2
Average over 10 runs is 0.221888 ms.
Average over 10 runs is 0.19745 ms.
Average over 10 runs is 0.204845 ms.
Average over 10 runs is 0.236598 ms.
Average over 10 runs is 0.250016 ms.

Calculating model accuacy...

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 19064  100 16748  100  2316   2577    356  0:00:06  0:00:06 --:--:--  3210

Your model accuacy is 0.0 %
```

Inference Robotics

Abstract:

In this project I am going to talk about sorting cups, my problem that I used to drink and left them cups where I was sitting and my mother was always complaining about that bad habit so I decide to make robot to help her to bring cups to kitchen.

Introduction:

my problem that I used to drink and left them cups where I was sitting and my mother was always complaining about that bad habit so I decide to make robot to help her to bring cups to kitchen.

I used to leave the cups under chairs everywhere in the sitting room which is not easy to see them.

So, I decided to make robot that can perceive the environment and classify the cups in the home to bring them back to kitchen.

Background / Formulation:

To classify images of cups two types of standard neural network models, AlexNet and GoogleNet. The reason for chosen these network that datasets contain colored images of cups, and these networks are commonly used in image classification problems and has shown great results in accuracy and time. In both models base learning rate is set to 0.01, solver type is SGD (Stochastic Gradient Decent), and models are trained 30 with epochs.

Numerous other learning rates and solving types were tested, but these showed the best result. I tried to used AlexNet but it given me a 93.5% accuracy (val), then I tried GoogleNet performed best with 30 epochs,

achieving 99.43 % accuracy (val).

Data Acquisition:

Dataset is consisted of three classes cup of coffee, cup of ice cream and mug.

Two of classes have 2 different type the last one with one type.

200 images for each type with 400 for each class with total 1200 images.25% of images is used for validation, and the data was a color image with size 256 x 256 pixel.

I used my phone to collect the data then I used online site to resize and format them in *.PNG.

Results:

When training with AlexNet network model, on 30th epoch validation accuracy was 93.83% accuracy. When classifying new images of cups that weren't in the training set this model showed 99.17% accuracy for mug and cup of coffee with 98.18% for one type and other misclassified as you can see in the figures below. As well it takes 4 inference time.

Fig classifying cup of coffee using AlexNet

Fig classifying cup of coffee with 8.49 prediction using AlexNet

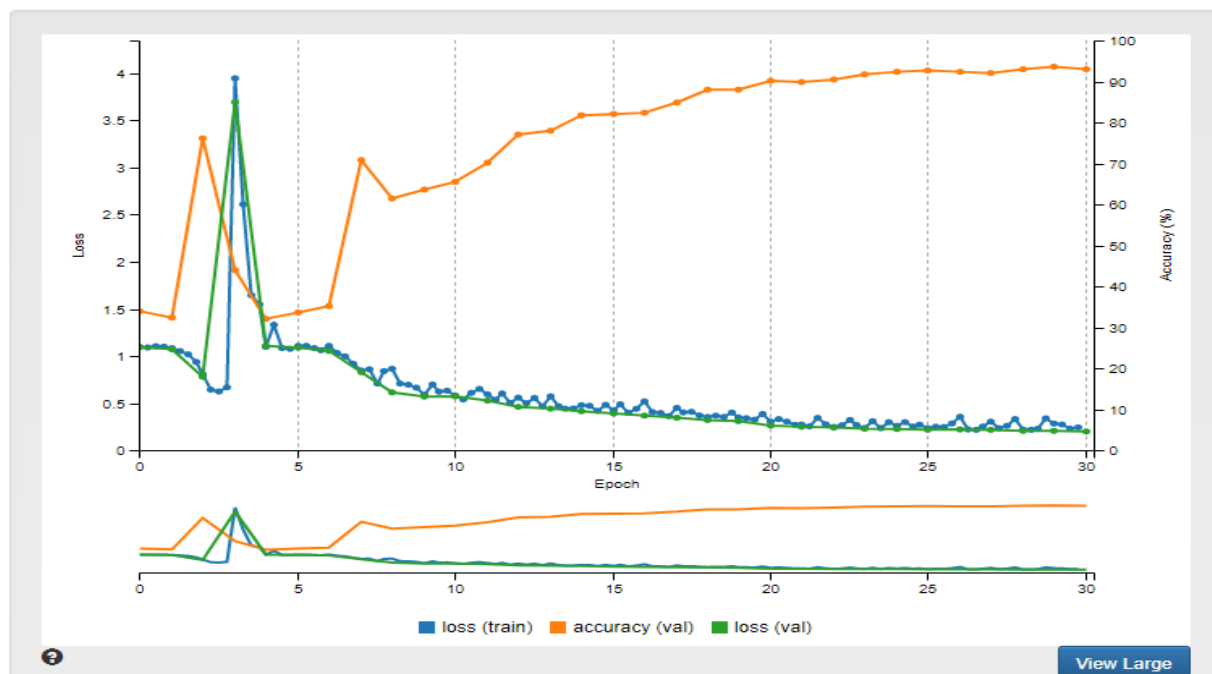Fig classifying mug using AlexNet

my_model Image Classification Model

- Initialized at 05:56:47 AM (1 second)
- Running at 05:56:48 AM (3 seconds)
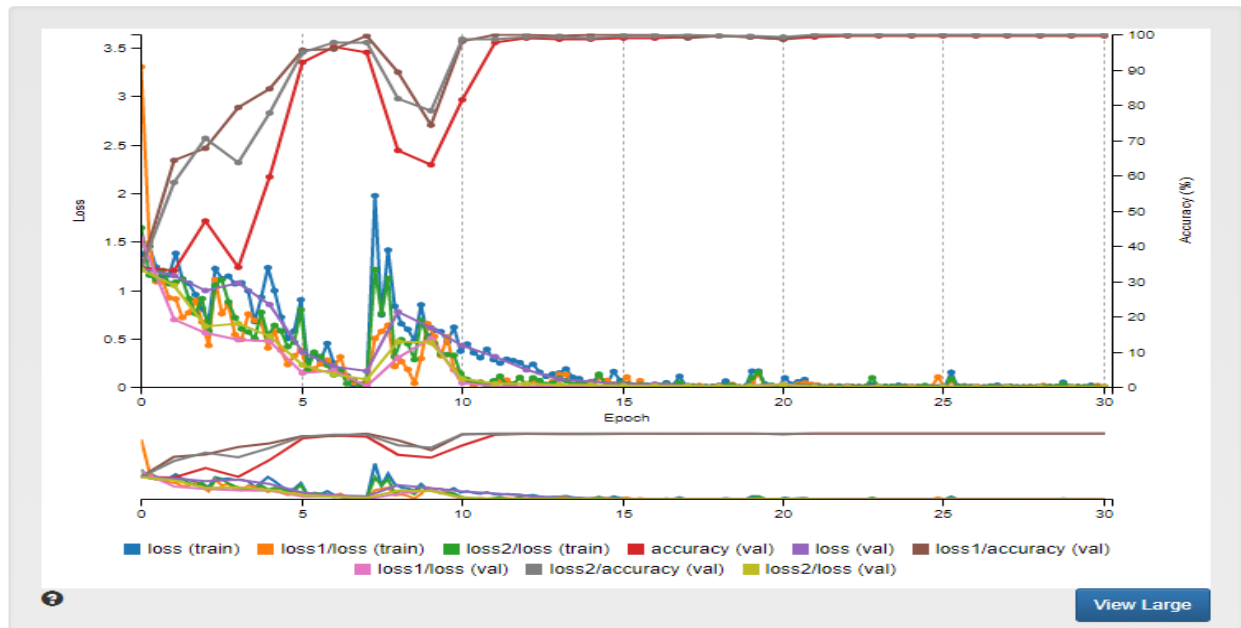- Done at 05:56:51 AM
  (Total - 4 seconds)

Infer Model Done ▾

Predictions

ice cup	97.33%
cup coffee	2.67%
mug	0.01%

Notes

None

Fig classifying ice of cup using AlexNet

AlexNet on Vaildation and Training Set

GoogleNet on Vaildation and Training Set

Fig classifying cup of coffee using GoogleNet

Fig classifying ice cup using GoogleNet

my_model_google Image Classification Model

Predictions

mug	99.25%
ice cup	0.65%
cup coffee	0.1%

Job Status Done

- Initialized at 03:59:58 PM (1 second)
- Running at 03:59:59 PM (3 seconds)
- Done at 04:00:02 PM
  (Total - 4 seconds)

Infer Model Done ▾

Notes

None

Fig classifying mug using GoogleNet

Discussion:

As can be seen from the results even though GoogLeNet model showed better result when classifying data set and validation set, AlexNet model resulted higher accuracy on test images. And AlexNet model showed more than 90% accuracy in 2 classes, as well GoogLeNet but with high accuracy on training and validation set. Additionally looking at inference time, two model performed similarly when classifying one image, GoogLeNet and AlexNet model only took 4 seconds.

Classifying images of cups for making robot help older people at their home. Given the results, for this particular problem the AlexNet model might be more suitable.

One may believe the accuracy on the dataset is low because some images are too small to show the necessary detail. That's why cup of coffee is misclassified.

In regards to my project, Accuracy and inference time are in a hyperbolic relationship which with high accuracy given me 4 fps on

Conclusion / Future Work:

Both network models showed similar results classifying validation and testing sets, but the model using AlexNet network resulted higher accuracy when classifying images of cups that were not in the training set. Both models showed more than 90% accuracy classifying images of cup of coffee.

It might be due to the fact that training set for cup of coffee class had images with mostly look like ice cup in color tone.

Collecting images of cups with different colors should improve the accuracy. To be commercially viable product both number of cups types and the variety of cups in one class with different color schemes and patterns need to be increased.

when collecting data, along with angle and ratio, the way that cups lays on the floor should be considered too. These types of classification can also be applied in areas like e-commerce.

Also I notice that after more training it increases accuracy(val) of the model I think this network going to memorize the dataset.