HW3

Project Description:

The Images of the mouse are stored in the folder. By using the annotation tool we marked the location of the 7 joints: “nose”, “left ear”, “right ear”, “left hip”, “right hip”, “tail base”, “tail end”. And we also located the position of 2 corners “top left” and “right bottom” to keep the bonding box for the mouse. The input of the model is 14 points which are X and Y locations of the 7 joints mentioned above. And the ground truth is the centroid of the bounding box.

Data Analysis:

The dimension of the image dataset is width width=640, height=360

The total data set of images are 126, there is minimum one mouse in each image, and there are two mouse in some images. After doing the annotation there are total of 205 mouses, where 126 are mouse1 and remaining 79 are mouse 2. In whole data, which we have 39 mouses with all data points(7 joints).

Structure of your machine learning model:

We did annotation with DeepLabCut-master tool

Data analysis and cleaning, and making data in structured format.

We used sequential model from karas to build a model.

Had build different models, and compared the evaluation

Training Procedure:

The model below I had took train test split is 70% and 30%

The model architecture:
This is the model architecture for all the 4 models
Input dimension = 14
First dense layer number of units =13
Second dense layer number of units = 9
Output layer number of units= 2

L2 Loss = MSE
L2 metric = MSE
Optimizer = Adam
Learning Rate= 0.01
Epochs = 500

Standardization:

Model 1 and Model 3:

For model 1 and model 3 I did standardization. After split train test data. The values of the coordinates of mouse some are very large and some are small, Then to get all this data into particular range, I did standardization which will get all the values into particular range from '0' to '1' . Here I used MinMaxScalar where it will make all the values into certain range

Model 2 and Model 4:

Here also i used the same dataset as used for above above model1.
I here I had not standardization the data, like I did in the above model. I take the data without any standardization.

Preprocessing:

Model 1 and Model2:

Here in preprocessing, While doing annotations in DeepLabCut some parts of mouse are not visible so I skipped the marking of points that are not visible and make them as 0. I used this data to train the model. Here I had 206 rows of mouse data, in this data there are 0 where I had skipped in the annotation.

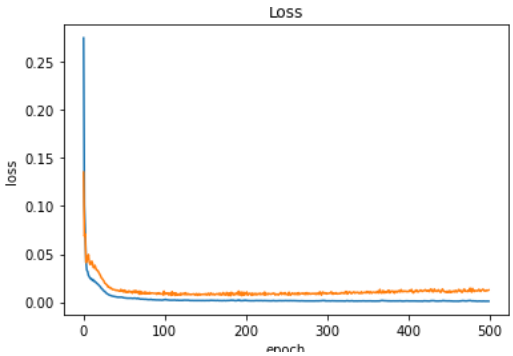
Model 3 and Model 4:

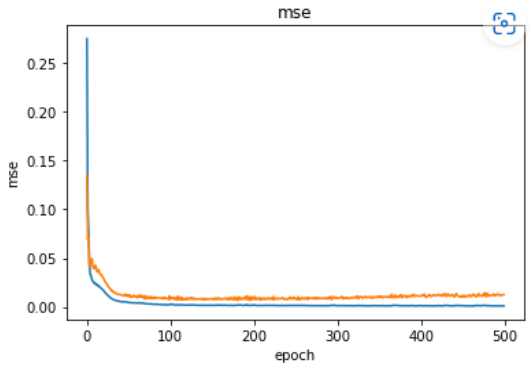
In preprocessing stage, I had removed the mouse which are having missing datapoints. While doing annotations in DeepLabCut some parts of mouse are not visible so I skipped the marking of points that are not visible and make them as 0. Then when reading the csv, I had removed the data of mouses having at least one '0' in its data.So then I was left we 39 mouse data having all data points .

The experiment results:

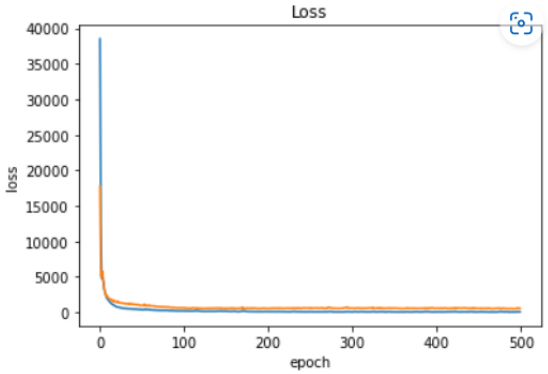
Model no	Train Loss	Train Mse	**Val_Loss**	**Val_mse**
Model 1	5.8083e-04	5.8083e-04	0.0186	0.0186
Model 2	82.3511	82.3511	567.8085	567.8085
Model 3	7.8308e-04	7.8308e-04	0.0071	0.007
Model 4	94.3933	94.3933	204.0330	204.0330

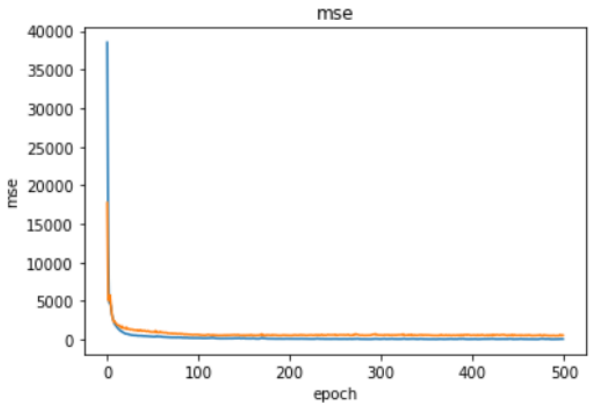
Model 1:File 1



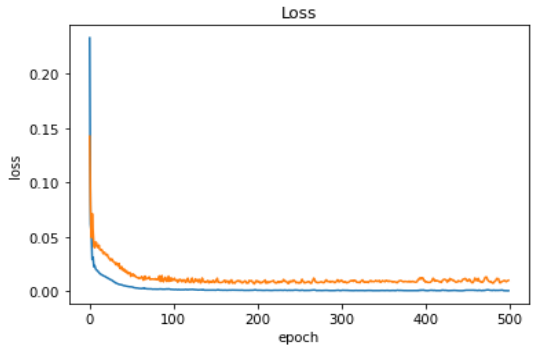


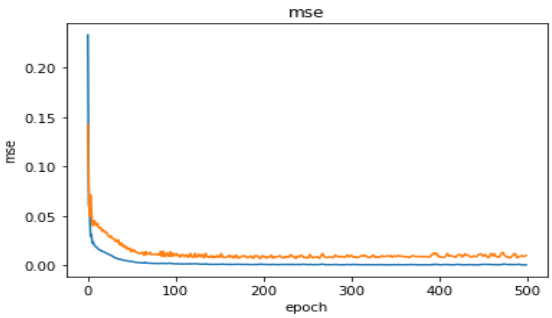
Model 2



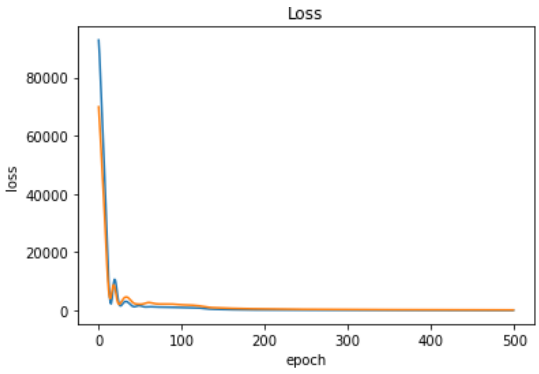


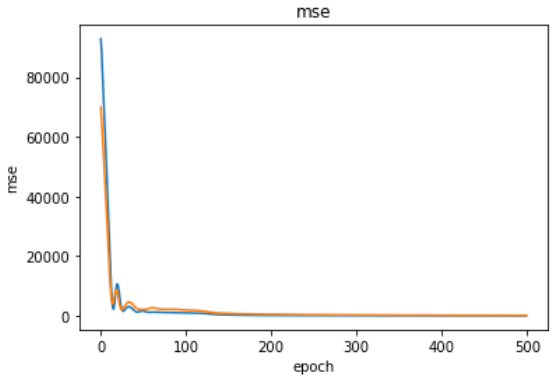
Model3:





Model 4





Evaluation of the Model:

The evaluations of the models are:

Model1: 4462.812307269679
Model2: 567.80850539711764
Model3: 3939.8130167977611
Model4: 204.03300419015648

By seeing the performance of these models, the model4 is having the lowest metrics. That can be considered as an better performed model among others.

Instructions for running your code on the data to reproduce your results:

A20516330_CollectedData_annotation
This is the file with annotations of all images.

file1: This is the model 1,
file1_test: This is model1 testing file
file2: This is the model 2,
file2_test: This is model2 testing file
file3: This is the model 3,
file3_test:This is model3 testing file
file4: This is the model 4.
File4_test: This is model4 testing file

Conclusions

We got good performance of the model 2, we can improve the model by training it with more dat a. Here we had less amount of data, we can build a model better training.