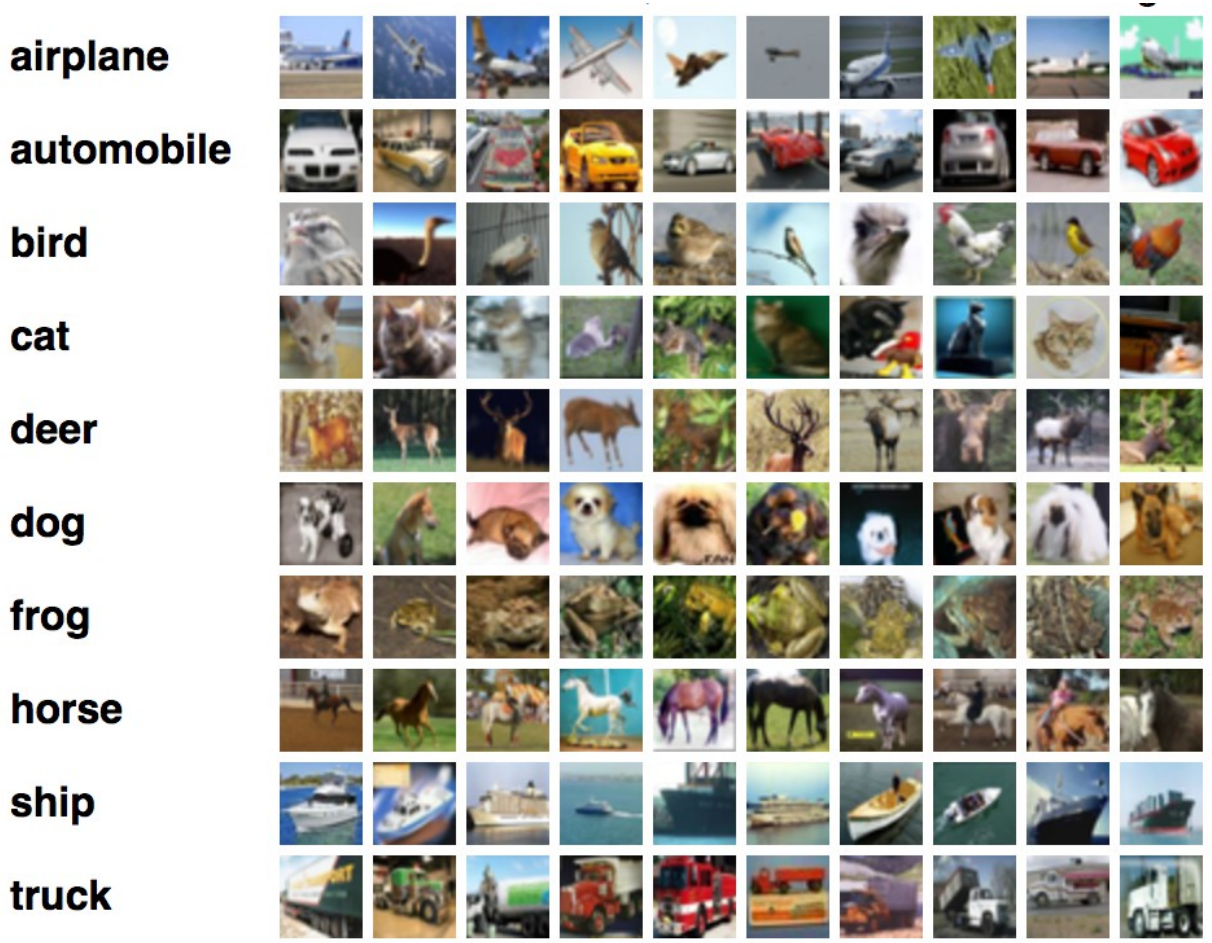
Cifar10 CNN

Eliran Amzaleg

Rotem Israeli

Yakir Arie

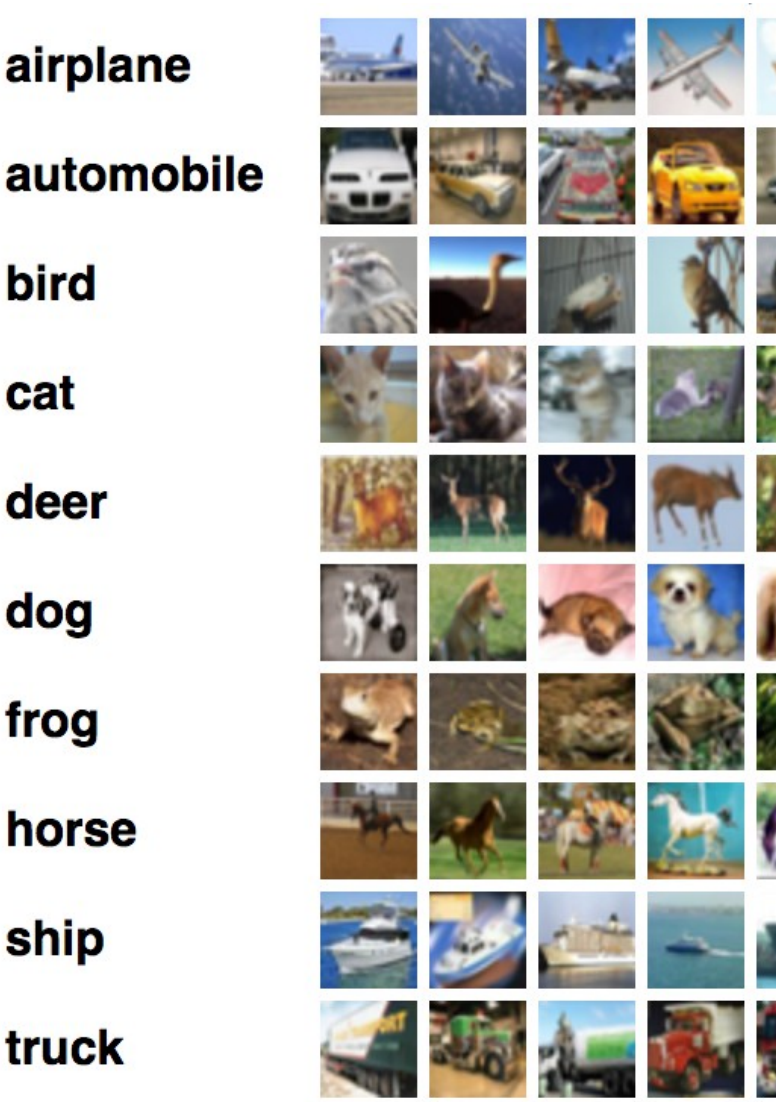


Abstract

After the previous assignments, we tried to build bette
classifier using cnn to be able to classify from 10 class
same dataset.

The convolutional layer consist of “filters” and it grate
images.

Using convolutional layers we were able to get almost
accuracy as previous assignment that classify 4 classe
used 10 classes in this project.



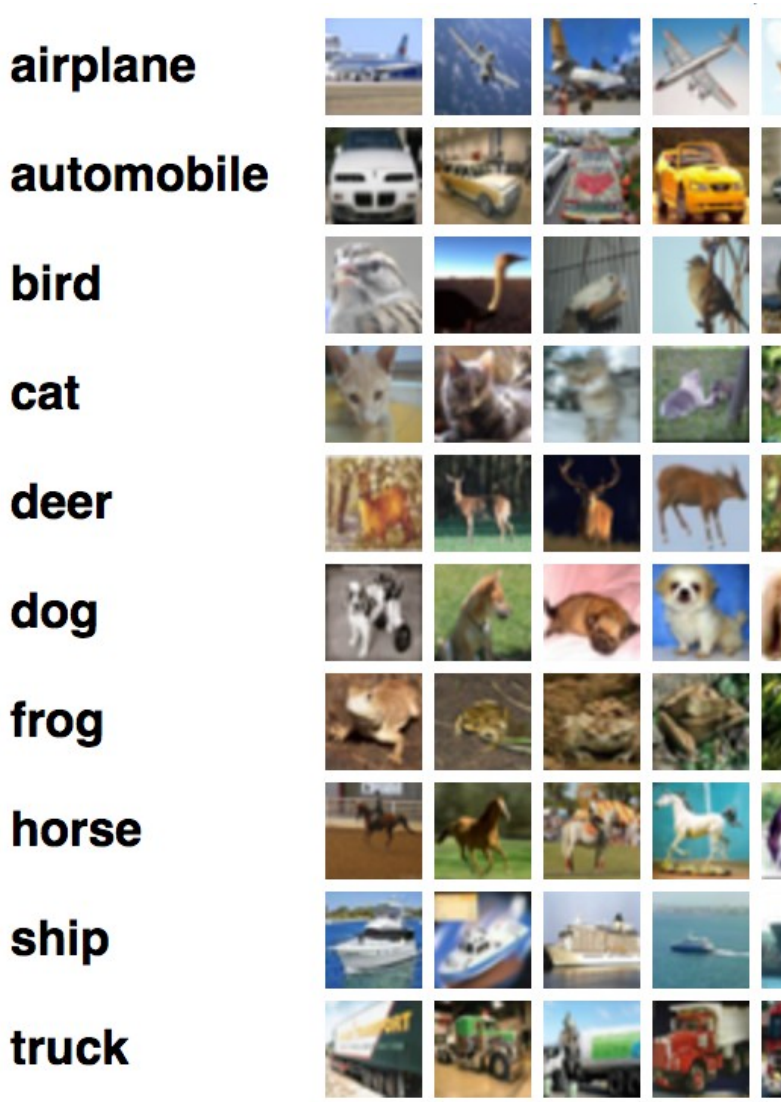
introduction

We tried to create a convolutional neural network to t
dataset which consist of 50k of images (10 of each cla

Each image is 32x32 with color.

The architecture of the model is two convolutional lay
two fully connected layers.

After 50 epochs the accuracy on validation set is 76%



Experiments Results

In The previous assignment we created neural network consist of only fully co

This network was train on cifar10 images but from only 4 classes and achieve

We tried to make convolutional neural network for better result, we created a 2 convolutional layers and it did 55% accuracy in classify the full 10 classes.

After adding fully connected layer the model was able to get to 70% accuracy

We added another fully connected layer and dropout in every layer and we go

After adding learning rate decay we got 76% in 30 epochs.

Conclusion

- Adding Convolutional layers can lead to better result in images dataset.
- Using dropout and fully connected layers in the end increase the accuracy
- Increase learning rate while training can help.

```
epoch: 1   train loss: 1.511   validation accuracy: 54.474%
epoch: 2   train loss: 1.141   validation accuracy: 61.635%
epoch: 3   train loss: 0.966   validation accuracy: 64.827%
epoch: 4   train loss: 0.819   validation accuracy: 68.523%
epoch: 5   train loss: 0.701   validation accuracy: 69.027%
epoch: 6   train loss: 0.589   validation accuracy: 74.046%
epoch: 7   train loss: 0.486   validation accuracy: 72.828%
epoch: 8   train loss: 0.398   validation accuracy: 74.634%
epoch: 9   train loss: 0.329   validation accuracy: 74.172%
epoch: 10   train loss: 0.277   validation accuracy: 74.844%
epoch: 11   train loss: 0.245   validation accuracy: 75.075%
epoch: 12   train loss: 0.209   validation accuracy: 74.508%
epoch: 13   train loss: 0.178   validation accuracy: 75.18%
epoch: 14   train loss: 0.165   validation accuracy: 75.201%
epoch: 15   train loss: 0.154   validation accuracy: 75.138%
epoch: 16   train loss: 0.140   validation accuracy: 75.369%
epoch: 17   train loss: 0.127   validation accuracy: 75.663%
epoch: 18   train loss: 0.123   validation accuracy: 75.999%
epoch: 19   train loss: 0.120   validation accuracy: 74.508%
epoch: 20   train loss: 0.112   validation accuracy: 75.789%
epoch: 21   train loss: 0.102   validation accuracy: 75.222%
epoch: 22   train loss: 0.103   validation accuracy: 75.684%
epoch: 23   train loss: 0.097   validation accuracy: 76.566%
epoch: 24   train loss: 0.092   validation accuracy: 75.411%
epoch: 25   train loss: 0.087   validation accuracy: 76.104%
epoch: 26   train loss: 0.085   validation accuracy: 74.928%
epoch: 27   train loss: 0.085   validation accuracy: 76.335%
epoch: 28   train loss: 0.078   validation accuracy: 75.705%
epoch: 29   train loss: 0.081   validation accuracy: 75.516%
epoch: 30   train loss: 0.081   validation accuracy: 75.936%
```