San Jose State University

Department of Electrical Engineering

EE104, Spring 2023, Pham

Laboratory Assignment #6

# Objectives

This lab allows you to use CNN (Convolutional Neural Network) to recognize color images and use different methods to improve the accuracy.

# Grading

Refer to the section **Your Deliverables** for grading criteria.

# Bibliography

I would like to acknowledge the Python open-source community and respective suppliers for making the material available.

References:

* Keras: <https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529>
* CIFAR-10 dataset: <https://www.cs.toronto.edu/~kriz/cifar.html>
* Baseline code: <https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/images/cnn.ipynb#scrollTo=WRzW5xSDDbNF>
* OpenAI

# Download, Installation, and Licensing

## Install necessary Python packages

**From Google Colab: (It is the requirement for this lab to run from Google Colab)**

You will need to PIP INSTALL the followings:

!pip install tensorflow

You may need to install the followings. Check it out yourself!

!pip install keras

!pip install h5py

!pip install Matplotlib

!pip install numpy

# Graphic Files

1. CIFAR-10 Data Set: <https://www.cs.toronto.edu/~kriz/cifar.html>

The CIFAR-10 dataset is an image collection dataset created by the Canadian Institute for Advanced Research. This dataset contains 60,000 32x32 colored images in 10 different classes: airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks.

1. Given set of images for testing purposes.

# Tutorial Python Programs

1. You will perform the steps outlined from this tutorial: [**https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/images/cnn.ipynb#scrollTo=WRzW5xSDDbNF**](https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/images/cnn.ipynb#scrollTo=WRzW5xSDDbNF) **.** This baseline code achieved a test accuracy of over 70%.
2. You will modify the base code from the tutorial in step 1, and add your own code as needed using different techniques that you learned from the lecture notes to improve the accuracy. You will earn points as detailed in the Your Deliveries section. Hint: This website will give you a lot of different techniques and samples that you can use to improve your accuracy. <https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/>
3. You will add your own code to test with the given images.

# Your Deliverables

Once you learn the process and the code associate with each step in the process, you will be able to customize the program to do the followings.

|  |  |  |
| --- | --- | --- |
| **Program or Requirement** | **Use Case** | **Max Score** |
| Demonstration Video | You must submit a demonstration video or your score for this lab will be zero |  |
| Check in your README file, documentation, and codes to Github | You will put all your work on your GitHub account and submit the same code to Canvas and share access to your GitHub with your instructor for grading purpose. | \_\_\_\_\_ / 5 |
| **CNN - Baseline + Increasing Dropout + Data Augmentation + Batch Normalization** + Your own method | To achieve > 90% accuracy and recognize successfully 90% of the given test images.  Video recording note: Because it will take a long time to run this test, you must submit a recording showing the result. The recording can be done by Zoom or any screen video capturing software. *Trim out the irrelevant contents and only submit the last few minutes showing you achieved your highest accuracy.*  Achieve > 90% accuracy: Earn full 30 points  Achieve > 87% accuracy: Earn 25 points  Achieve > 82% accuracy: Earn 20 points  Achieve > 77% accuracy: Earn 15 points  Achieve > 74% accuracy: Earn 10 points  Achieve > 70% accuracy: Earn 5 points  Else: 0 point | \_\_\_\_\_ / 30 |
| CNN - Challenge test | Recognize 3 unrecognizable images from the TEST IMAGES section below (5 points each)  You will use these files provided on Canvas:  **CNNbaseline.py** – Use to run the baseline that yields 70% accuracy, then add your code to achieve 90% or higher accuracy  **test\_impage.py** – Use to test the images in the TEST IMAGES section below. | \_\_\_\_\_ / 15 |
| Game Development –Flight | Leverage the base code from either chapter Big Quiz or Happy Garden. Add two of your own Hacks and Tweaks from the list below:   * Big Quiz: Take the hint, More Questions, nicer blend of color using Dash of Color (10 points each) * Happy Garden: More fangflowers, more enemies, rain in the garden (10 points each) | \_\_\_\_\_ / 20 |
| Hello World to OpenAI | Using Python, perform the instructions from this website as a prerequisite:  <https://platform.openai.com/docs/quickstart>  You will follow the instruction from the file openai-quickstart.txt downloaded from Canvas.  Run the baseline as is first, then make the following changes:   1. Change the picture of the website from a dog to something else.(4 points) 2. Change the title from “Name my pet” to something else more generic applicable to things and humans as well. (3 points) 3. Change the code to generate 4 names instead of 3 names. (5 points) | \_\_\_\_ / 12 |
| Hello World to ChatGTP | Run the given ChatGTP code, then make one change at a time to the followings 6 criteria and document any output differences after each change (3 points each)  engine=model\_engine,  prompt=prompt,  max\_tokens=1024,  n=1,  stop=None,  temperature=0.5, | \_\_\_\_ / 18 |
|  | **TOTAL** | **100%** |

See next page for TEST IMAGES.

That’s all for this lab. Hopefully you found it useful and increase your interest in the Python world! See you in the next lab.

# TEST IMAGES

Below are suggestions that you can use to test the accuracy of your code.

## 'airplane',

* <https://cbsnews2.cbsistatic.com/hub/i/r/2017/03/27/73bd41ff-5703-48ca-995c-131d1b3572b4/thumbnail/640x335/10f4b442d725b8fa79d3e2dbf286ba76/air-force-one-two-planes.jpg>
* <https://upload.wikimedia.org/wikipedia/commons/thumb/f/f0/Another_Airplane%21_%284676723312%29.jpg/1024px-Another_Airplane%21_%284676723312%29.jpg>
* <https://upload.wikimedia.org/wikipedia/commons/thumb/d/d3/Voyager_aircraft.jpg/1200px-Voyager_aircraft.jpg>
* <https://static.wikia.nocookie.net/lostpedia/images/f/f7/Kateandplane.jpg/revision/latest/top-crop/width/360/height/450?cb=20061109012445> (Use for Challenge test)
* <https://www.zdnet.com/a/img/resize/071727877ee9884b60edd728253d2baadcb3985f/2021/02/23/19631992-64df-4af9-a288-a0cb4112e682/bombardier-globaleye-jet.jpg?width=1200&height=900&fit=crop&auto=webp> (Use for Challenge test)

## 'automobile',

* <https://sniteartmuseum.nd.edu/assets/166204/original/ferrari.jpg>
* <https://www.kbb.com/articles/wp-content/uploads/2020/04/00-2020-bmw-8-series-gran-coupe.jpg>
* <https://images.all-free-download.com/images/graphiclarge/classic_jaguar_210354.jpg> (Use for Challenge test)
* <https://hips.hearstapps.com/hmg-prod.s3.amazonaws.com/images/devel-motors-sixteen-1540564064.jpg> (Use for Challenge test)
* <https://amsc-prod-cd.azureedge.net/-/media/aston-martin/images/default-source/models/valkyrie/new/valkyrie-spider_f02-169v2.jpg?mw=1980&rev=-1&hash=92E23C911BDE23D418D37F9187844B7C> (Use for Challenge test)

## 'bird',

* <https://ichef.bbci.co.uk/news/976/cpsprodpb/67CF/production/_108857562_mediaitem108857561.jpg> (Use for Challenge test)
* <https://upload.wikimedia.org/wikipedia/commons/5/53/Weaver_bird.jpg> (Use for Challenge test)
* <https://images.all-free-download.com/images/graphiclarge/flying_bird_201952.jpg> (Use for Challenge test)
* <https://www.allaboutbirds.org/news/wp-content/uploads/2009/04/WKingbird-James.jpg>
* <https://www.pestworld.org/media/560900/istock_000027713740_large.jpg?preset=pestFeature360>

## 'cat',

* <https://static.toiimg.com/thumb/msid-67586673,width-1070,height-580,overlay-toi_sw,pt-32,y_pad-40,resizemode-75,imgsize-3918697/67586673.jpg> (Use for Challenge test)
* <https://wagznwhiskerz.com/wp-content/uploads/2017/10/home-cat.jpg> (Use for Challenge test)
* <https://res.cloudinary.com/dnkxl7hbd/images/f_auto,q_auto/w_400,h_664/v1610516304/Lanai-Cats-we-need-you/Lanai-Cats-we-need-you.png> (Use for Challenge test)

## 'deer',

## 'dog',

## 'frog',

## 'horse',

* <https://upload.wikimedia.org/wikipedia/commons/0/03/American_quarter_horse.jpg>
* <https://upload.wikimedia.org/wikipedia/commons/f/f0/White_horse.jpg>

## 'ship',

## 'truck'

# Laboratory Hand-In Requirements

Once you have completed a working design, prepare for the submission process. You are required to upload YouTube videos to demonstrate your working solutions. You are also required to submit an archive of your project in the form of a ZIP file. Use 7-Zip option to create the ZIP file. Name the archive lab#\_yourlastname\_yourfirstname.zip. Refer to Lab 1 for detail instructions.

You will submit your zip file to the instructor through Canvas by the due date and time. You also need to submit the same to GitHub to earn the credit. If your program is not completely functional by the due date, you should demonstrate and turn in what you have accomplished to receive partial credit. See the syllabus for the late penalty guideline