PROGRESS REPORT(PROJECT DIARY) :

Name : Richeek Das

Project : Gestures for 3D Space

DATE	PROGRESS
21.03.2020	Completed first 14 OpenCV tutorials from here : • https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_table_of_contents_imgproc/py_table_of_contents_imgproc.html#py-table-of-content-imgproc And completed the video series : • OpenCV - sentdex And implemented a few projects on github utilising most of those techniques : • https://github.com/sudoRicheek/PyShapes (A pip installable python3 package which utilises OpenCV) • https://github.com/sudoRicheek/Meme-makeawwr (An meme-maker app with inbuilt editing features which utilises OpenCV and Tkinter)
23.03.2020	Completed Pandas introduction from the video series : • Data Analysis with Python and Pandas Tutorial Introduction Completed Pandas introduction from the official documentation : • Getting Started • Basic Functionality • Introduction to Data Structures(DataFrames) Additional : Revised Numpy and Matplotlib functionalities from their official documentations: • Overview — Matplotlib 3.1.2 documentation • NumPy Documentation Further plans for Pandas : To complete the first few Kaggle micro-courses from : • Learn Python, Data Viz, Pandas & More | Tutorials (By 30.03.2020)

27.03.2020	Built a simple blogging website from scratch, utilising Django 3.0 as a backend framework and HTML and CSS for front-end, as a project for DevCom Recruitment. Github link : https://github.com/sudoRicheek/The-Blogger You can see the test run as well as interact with the web-page here : SudoRicheek Blogger Resources used : The official documentation : • https://docs.djangoproject.com/en/3.0/ (Django 3.0 official docs) For CSS : • https://www.w3schools.com/css/ (A lot of these CSS tutorials) Some video resources : • https://youtu.be/JT80XhYJdBw Things learnt : • Django 3.0 • CSS
30.03.2020	Completed an Introductory Kaggle Machine Learning Microcourse to gain required skills to start off with Kaggle's Intermediate Machine Learning Microcourse. Resources used : • https://www.kaggle.com/learn/intro-to-machine-learning Things learnt : • Core ideas in machine learning. • Learnt and built my first ML models.
31.03.2020	Completed an Intermediate Machine Learning Microcourse from Kaggle. Resources used : • https://www.kaggle.com/learn/intermediate-machine-learning Things learnt : • Learnt to handle missing values, non-numeric values and data leakages. • Learnt to make my models more accurate and useful, by using techniques like Extreme Gradient Boosting. • Learnt to write cleaner and more organized codes by using Pipelines to bundle pre-processing and modelling steps.

01.04.2020	Went through the mathematical aspects of OpenCV Feature Detection and Description, so as to learn about various algorithms governing it(with prime focus on SIFT, FAST and ORB techniques) Resources used : • Harris Corner Detection • Shi-Tomasi Corner Detection • Scale Invariant Feature Transform(SIFT) • Speeded-Up Robust Features(SURF) • FAST algorithm for Corner Detection(FAST) • Binary Robust Independent Elementary Features(BRIEF) • Oriented FAST and Rotated BRIEF(ORB) Learnt about different types of Feature Matchings from here : • https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_matcher/py_matcher.html Started analysing the algorithms from the following codes : • https://github.com/rkbvikrant/Hand-Gesture-Recognition-Python-OPENCV • https://github.com/aakashjhawar/hand-gesture-recognition
02.04.2020	Used the pure image-processing techniques offered by OpenCV to implement a preliminary model of Finger Count Recogniser, using the direct video feed of the webcam. This model has been kept simple, with OpenCV as its only dependency, so as to study the comparative improvements in the future models which will be utilising CNNs and other Machine Learning techniques. Link to the Github repo(contains both of the .ipynb and .py files) : • https://github.com/sudoRicheek/Finger-Count-Recogniser-OpenCV Resources used: • Techniques in OpenCV learnt so far, from the links listed under progress tabs of 21.03.2020 and 01.04.2020 .
03.04.2020	Started with the following Machine Learning Book for understanding the basics of ML: • Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow by Aurélien Géron The online code repos accessed while studying this book for interactive development, can be found here : • https://github.com/ageron/handson-ml2 • https://github.com/rasbt/python-machine-learning-book-3rd-edition Note : A few books on linear algebra as well as articles on probability should be referenced for understanding the

	mathematics working behind the various ML algorithms.
04.04.2020	Created a Jupyter Extension using JavaScript from scratch(Made it purely out of my own interests; and grossly it has no actual impact on the real 'project' of this Progress Report) Name of extension : DropdownExporter Function : A small and lightweight Jupyter Extension for exporting the .ipynb file to PDF, HTML, Markdown, LaTeX or Python script in a single-click. Link to the GitHub repo : • https://github.com/sudoRicheek/DropdownExporter Things learnt : • Basics of JavaScript • The mechanics of a Jupyter Notebook API
06.04.2020	Created a YouTube Scraper to build a Playlist Downloader with Selenium and BeautifulSoup4 as dependencies.(I am listing this as a progress, but this has no pre-planned impact on the actual project) Link to the GitHub repo : • https://github.com/sudoRicheek/YTPlaylistDownloader (Feel free to check it out) Things learnt : • BeautifulSoup4
09.04.2020	Completed the first 4 chapters from the book to get the basics of Scikit-Learn : • Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow by Aurélien Géron Started with this book for understanding the basics of Deep Learning and its implications in Python3: • Deep Learning with Python by FRANÇOIS CHOLLET Started with this book for understanding the mathematical preliminaries governing Neural Networks : • Pattern Recognition and Machine Learning by Christopher Bishop As a side note : I took up a Summer of Science project under MnP club IITB alongside this SoC project. Topic : Neural Networks and Deep Learning. And the topic is pretty much heavily related to this ongoing SoC project :)

	Github repo where I'll keep the notes/implementations, that I'll make throughout the SoS reading project : • https://github.com/sudoRicheek/NeuralNetworks-And-Deep Learning-SoS

10.04.2020	Made a spam-classifier using the following dataset of Apache SpamAssassin :
	• https://spamassassin.apache.org/old/publiccorpus/
	It included :
	• Downloading and unzipping data using tarfile and urllib
	• Handling emails objects in Python using email module
	• Splitting the data into test and train sets.
	• Getting the structure features of the emails and then using those features to get an idea of possible data refinements.
	• Carried out some simple refinements in the email data, like:
	  ○ "removing punctuations"
	  ○ "translating HTML emails into plain-text"
	  ○ "removing URLs"
	  ○ "removing NUMBERS"
	• Wrote a data preparation pipeline to convert each email into a feature sparse vector, which indicates the presence or absence of each possible word.
	• Finally used a RandomForestClassifier after testing with a wide range of other classifiers.
	
	Final results :
	• Average Cross Validation Score : 0.9737
	• Accuracy : 0.985
	Plans :
	• To test, train and refine the spam classifier further on more difficult datasets and real life scenarios :)
	
	#This was an Exercise problem of "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow by Aurélien Géron"

12.04.2020	Uploaded the first set of notes on the "behind the scenes" of ML Algorithms for the SoS project, you can view it here : • https://github.com/sudoRicheek/NeuralNetworks-And-Deep Learning-SoS
17.04.2020	Created a Realtime Chat Application using a Django REST API Backend, an Angular Frontend and a SOCKET.IO middleware server for real time communication between the users. Link to the GitHub repo : • https://github.com/sudoRicheek/The-Angular-Django-Socket-Messenger Things learnt : • Django REST API Backend Framework

	• TokenAuthentication for GET and POST requests. • Implementation of WebSockets using SOCKET.IO • Angular 9 for TypeScript based Front-End development. Resources used : • For Django REST API : ○ https://www.django-rest-framework.org/ • For Angular 9 (Their documentation is awesome !) : ○ https://angular.io/docs • For SOCKET.IO : ○ Docs | Socket.IO ○ Various blogs implementing real-time communication through socket.io
19.04.2020	Went through Chapter 9 of Python Machine Learning by Sebastian Raschka & Vahid Mirjalili, for understanding the basics behind hosting and embedding a ML model into a Web App. Created a repo for hosting the implementations/notes related to the project(Ya I know it's quite late to do this now, but I never needed one before now). Link : • https://github.com/sudoRicheek/Gestures-For-3D-Space
22.04.2020	Started with Stanford CS231 assignments. • The First Assignment deals with implementation of *kNN*, *SVM*, *Softmax*, and a simple *Neural Network* without any ML libraries. • The Second Assignment helps to get acquainted with *Backpropagation*, *Batch Normalisation*,*Dropout*, *CNNs*, and *deep learning frameworks*. • The Third Assignment deals with implementation of *RNNs*, *LSTMs*, and *GANs* You can find the assignment solutions here : •
24.04.2020	Completed the first five chapters from : • Deep Learning with Python by FRANÇOIS CHOLLET Understood the mathematical preliminaries of neural networks from first 4 chapters of this book : • Pattern Recognition and Machine Learning by Christopher Bishop
25.04.2020	Started with an ITSP project, X-Ray Anomaly Detection using CNN. Here's the flow sheet of ideas :

	• We will be training a Convolutional Neural Network using present state-of-the-art techniques for implementing multiclass classification of 14 different types of anomalies in Chest-X-Rays. • We plan on using the **CheXpert dataset**, a large dataset consisting of 224,316 chest radiographs from 65,240 patients. • Then we will want to have a **Django REST API** backend for establishing a communication between our trained Neural Network and our Frontend framework, so as to provide fast and seamless service, and reduce report latency. • Thirdly, we will select a **frontend framework(preferably Angular/ReactJS)** of our interest to design our web app and make the usage simple, fast and minimalistic. • Finally, we will select a cloud hosting service such as **AWS/Azure** for deploying our app and to make it accessible to anyone with a working internet! (I am mentioning this because, what I have learnt so far from my SoS will definitely help me with this project and this practical implementation surely adds to my SoS report :) Some of the research papers which helped us initially to get the basic idea : • CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison • [1911.06475] Interpreting chest X-rays via CNNs that exploit disease dependencies and uncertainty labels • Interpreting chest X-rays via CNNs that exploit disease dependencies and uncertainty labels • Deep Convolutional Neural Networks for Chest Diseases Detection • Detecting Anomalies in X-ray using CNN. - Analytics Vidhya Link to our GitHub Organization where we will keep the project files: • https://github.com/PerXeptron/
26.04.2020	Started with this paper : • Visualizing and Understanding Convolutional Networks You can find the summary of my understanding of the paper over here : • De-Convolutional Networks Summary

29.04.2020	Wrote an OpenCV code to easily and quickly create my own hand-gesture image datasets for this project. It implements background subtraction and binary thresholding and saves both the original colour image with 3 channels and the processed single channel image. Link to the GitHub repository, hosting the code along with the instructions: • https://github.com/sudoRicheek/Dataset-Creator-For-Hand-Gestures
30.04.2020	Created a dataset containing 2750 hand-gesture images of 5 different classes, with 550 images in each class. It has both the background subtracted, binary thresholded image as well as the original 3 channel colour image.
02.05.2020	Tested a bunch of models through Transfer Learning in pre-trained convnets like(VGG, ResNet50, DenseNet121) on the dataset that I had created yesterday. You can find my observations and the best model among them, over here : • https://github.com/sudoRicheek/Gestures-CNN-Model-Creation-And-FineTuning The best model among them : • Gestures_CNN_4_fine_tuned_1.h5
04.05.2020	Created a very very preliminary Django WebApp for using the fine tuned CNN classifier that we had initially trained. Here you can upload the preprocessed gesture image and it will show you the prediction it makes for that gesture image. It's a pretty basic webapp but I can't upload it to github, since the size of the classifier is 180MB+ :(
08.05.2020	Made an on-the-go gesture recognizer using a continuous video input through the webcam feed. To check the practical implications of this model, I added some basic fun-features like using gestures to play and pause videos, change current working application window and volume up-down features as well. You can check out the source code here : • Gesture Application Model Note : • I have used the Keras CNN model built on 02.05.2020 • It also uses PyAutoGUI, OpenCV and Keras as dependencies.

16.05.2020	Created a repository for hosting the notebooks for CNN model creation for our ITSP X-Ray Anomaly Detection : • https://github.com/PerXeptron/X-Ray-Anomaly-Detection-Models Things learnt : • Learnt a lot about effective collaboration using GitHub Organizations.
18.05.2020	Made a PythonQt5 Based Desktop App for deploying an Image-Upscaling Super Resolution Algorithm, based on Generative Adversarial Networks with cross platform feasibility. My main aim is to deploy state-of-the-art Super Resolution Techniques for upscaling low resolution images with better-than-eye performance. It is a quite fun project to work on and it'll take quite a bit of time to refine it :) Link to the GitHub Repo : • https://github.com/sudoRicheek/Image-Enhancer Things learnt and learning : • Cross Platforming using PyQt5 • Generative Adversarial Networks • Recent interesting work on Super Resolution Techniques, especially using GANs.
22.05.2020	Started with this paper : • Real-Time Hand Gesture Detection using CNN You can find the summary of my understanding of the paper over here : • A Review on Sliding Windows Approach to Real-Time Gesture Detection
28.05.2020	Well bye :) I am ending my progress report for this Summer Of Code 2020 project here. We will be now shifting our focus to the final problem statements for this project and you can take a look at the SRS document for the one I am doing, over here : • SRS Document One-Shot Learning