HAMARA PARYATAN KATAI JAHAR:
Data Geeks, 10th May 2019, Machine learning

TEAM LEADER: Akshay Bengani

TEAM MEMBERS:

1. Akshay Bengani
2. Piyush Agarwal
3. Abhinay Bhatt
4. Mudit Jain
5. Priyal Vyas
6. Sandeep Jangir
7. Harsh Vijay
8. Smriti Khanwani
9. Bhawna Goyal
10. Chetan Gaur
11. Pradyumn Sharma

INTRODUCTION:

Our idea is to provide real-time monitoring of headcount, Gender count in different areas of Heritage locations. By this data, we will show count on a map of the location, this will help users to plan accordingly where to go 1st because they know that which area or block of a place is filled with tourists and which are empty. Adding a feature of Gender count will help boys to schedule where they want to go.
We will be creating a web portal showing these graphs on a website and this web portal will be used on a big display outside the main gate. We will also be building an Android application showing the same map and data in realtime to the users.

Key Benefits:

1. Showing headcount and availability of space in different locations
2. In case of emergency, the fire department and rescue teams can monitor where are the number of people and help to rescue them.
3. Calculating daily data Analysis will help us to note which areas need improvement.

TECHNOLOGY USED:

1. Python CGI & Socket Programming
2. Open CV
3. Keras
4. AWS EC2
5. Docker
6. HTML / CSS / JS / AJAX / JQUERY / Bootstrap
7. MSVCRT (Windows) Getch (Linux)
8. Android Studio (JAVA & XML)

MODULES OF PROJECT:

OpenCV(CV2) -> Opencv stands for Open Source Computer Vision. It is a library of programming functions mainly aimed at real-time computer vision. OpenCV application areas include Facial recognition, Gesture recognition, mobile robotics.

TensorFlow -> Tensorflow is a free and software library for dataflow and differential programming across a range of tasks. It is a symbolic maths library and is also used for machine learning applications such as neural networks.

KERAS -> Keras is a high-level neural network API, written in Python and capable of running on top of Tensorflow, CNTK or Theano.

AWS-EC2 -> Amazon elastic compute cloud(Amazon EC2) provides scalable computing capacity in the AWS cloud. Using amazon Ec2 eliminates your need to invest in hardware upfront, so you can develop and deploy applications faster.

DOCKER-CONTAINERS ->A container is a standard unit of software that packages up the code and all its dependencies so the applications run quickly and reliably from one computing environment to another.

PYTHON CGI -> The CGI(common gateway interface) is a set of standards that defines how information is exchanged between the web server and custom services.

SOCKET PROGRAMMING -> It is a programming schema in which sockets are used and manipulated to create a connection between software.

HTML -> It is a standard markup language for creating web pages.

CSS -> Cascading style sheet is style sheet language used for describing the presentation of a document written in a markup language like HTML.

JAVASCRIPT -> It is a scripting language used to enable programmatic access to objects within other applications.

BOOTSTRAP -> It is a most popular HTML, Javascript framework for developing a responsive and mobile-friendly website.

AJAX -> It is a set of web development techniques used many web technologies on the client-side to create asynchronous web applications.

JQUERY -> It is the fast, small and feature-rich javascript library.

MSVCRT(windows) -> This module gives you access to a number of functions in the Microsoft visual runtime library.

GETCH(Linux) -> It provides a function to read one single char from standard input without waiting for a newline.

PROJECT DESCRIPTION

- This project helps the user to identify which place is more crowded and according to this, the user can decide where to go first.
- It will help the government to decide where more peoples are coming and where not. So, it will help them to decide the areas of improvement
- This will tell where the girls are more and boys will automatically prefer to go there.
- It will help the rescue teams to identify the more crowded places and act accordingly.

FUTURE PLANS

- To improve the accuracy of gender detection
- To integrate street dog and animals detection with this project
- Our plan is to provide services to all the heritage sites across Rajasthan and further in India.
- To detect abnormal behavioural patterns in crowd scenarios.

INSTALLATION REQUIREMENTS AND GUIDELINES

1. Since this project is for human detection as such we need a set of cameras to run this project. To run this system you need some packages pre-installed
    - Python
    - Python-dev tools
    - OpenCV
    - Keras
    - Tensorflow
    - Numpy, Pandas, Matplotlib
    - MariaDB
2. Run the detection.py file in order to start the human detection system.
3. Run the project.py file to start listening for the count.
4. The whole project is accessible to users via our website, it is located in UI/index.html file.

SYSTEMS REQUIRED TO RUN THIS PROJECT

Basically, we need 2 systems for the project execution

1. One system is hosted on AWS containing the project.py listener and also containing the website files.
2. The second system will be installed at the heritage place security server room, which will be used to connect to cameras and run the human detection system.

PROCESS FLOW OF THE PROJECT

1. The detection.py file will use an algorithm to detect human beings using the video footage provided by the video cameras attached to the system.
   Note This system requires huge computation power
2. The file will generate a count and send this via a socket to another system.
3. On the other hand, the system hosted on AWS will receive the count via a socket within the project.py file.
4. The file will apply colours on the map which will represent the density of the crowd in a particular place of the Installed site.
5. The website login page is hosted on the AWS /var/www/html and it uses the user login credentials to create a docker container with a random available port and redirects the docker container as a website to the user.
6. As such every user accessing the website is isolated in the docker container.
7. The website contains an iframe of the map which refreshes itself every 10 seconds to update the density plot on it.
8. The map is an image which is dynamically updated by the project.py as per the change in density.

ROLES OF TEAM MEMBERS:

1. Akshay Bengani (Team Management, Resource Allocation and Troubleshooting)
2. Piyush Agarwal (Socket Programming and Heat Map Troubleshooting)
3. Abhinay Bhatt (Web Designer and Front End Developer)
4. Mudit Jain (Human Detection and Model Troubleshooting)
5. Priyal Vyas (Documentation and Team Entertainment)
6. Sandeep Jangir (Not present)
7. Harsh Vijay (Human and Gender Detection)
8. Smriti Khanwani (Map Architect)
9. Bhawna Goyal
10. Chetan Gaur (Human Detection, Gender Detection)
11. Pradyumn Sharma (On-site help and Troubleshooting)