A Gesture-Based Tool for Sterile Browsing Of Radiology Images CNN and Open CV

Introduction:

Humans can recognize body and sign language easily. This is possible due to the combination of vision and synaptic interactions that were formed along brain development. In order to replicate this skill in computers, some problems need to be solved: how to separate objects of interest in images and which image capture technology and classification technique are more appropriate, among others.

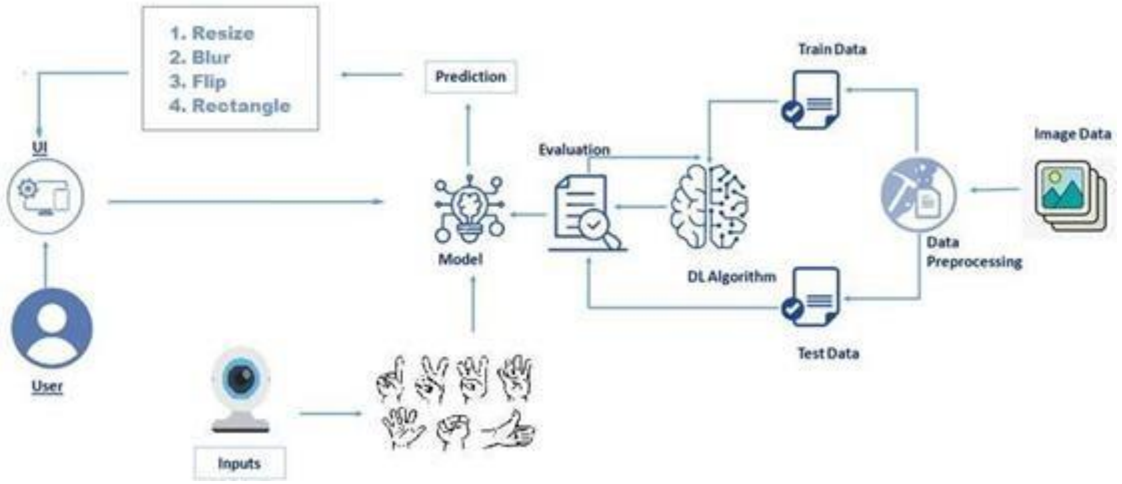
In this project Gesture based Desktop automation, First the model is trained pre trained on the images of different hand gestures, such as a showing numbers with fingers as 1,2,3,4. This model uses the integrated webcam to capture the video frame. The image of the gesture captured in the video frame is compared with the Pre-trained model and the gesture is identified. If the gesture predicts is 0 - then images is converted into rectangle, 1 - image is Resized into (200,200), 2 - image is rotated by -45∘, 3 - image is blurred, 4 - image is Resized into (400,400), 5 - image is converted into grayscale etc.

Project Objectives

- Know fundamental concepts and techniques of Convolutional Neural Network (CNN).
- Gain a broad understanding of image data.
- Know how to pre-process/clean the data using different data pre-processing techniques.

- Know how to build a web application using Flask framework.

Technical Architecture:



Overview:

1. Defining our classification categories
2. Collect training images
3. Train the model
4. Test our mode

Project Flow

- User interacts with the UI (User Interface) to upload the image as input
- Depending on the different gesture inputs different operations are applied to the input image.
- Once model analyses the gesture, the prediction with operation applied on image is showcased on the UI.

To accomplish this, we have to complete all the activities and tasks listed below:

- Data Collection.
  - Collect the dataset or Create the dataset
- Data Pre processing
  - Import the ImageDataGenerator library
  - Configure ImageDataGenerator class

    - Apply ImageDataGenerator functionality to Trainset and Testset
- Model Building
    - Import the model building Libraries
    - Initializing the model
    - Adding Input Layer
    - Adding Hidden Layer
    - Adding Output Layer
    - Configure the Learning Process
    - Training and testing the model
    - Save the Model
- Application Building
    - Create an HTML file
    - Build Python Code

Following software, concepts and packages are used in this project

- Anaconda navigator
- Python packages:
    - open anaconda prompt as administrator
    - Type “pip install TensorFlow” (make sure you are working on python64 bit)
    - Type “pip install opencv-python”
    - Type “pip install flask”

Deep Learning Concepts

CNN: a convolutional neural network is a class of deep neural networks, most commonly applied to analyzing visual imagery.

Opencv: It is an Open Source Computer Vision Library which are mainly usedfor image processing, video capture and analysis including features like face detection and object detection.

Flask: Flask is a popular Python web framework, meaning it is a third-partyPython library used for developing web applications.

Project Structure

- Dataset folder contains the training and testing images for training our model.
- We are building a Flask Application which needs HTML pages stored in the templates folder and a python script app.py for server side scripting
- we need the model which is saved and the saved model in this contentis gesture.h5
- The static folder will contain js and css files
- Whenever we upload a image to predict, that images is saved in uploads folder.

References :

1. Schultz M, Gill J, Zubairi S, Huber R, Gordin F. "Bacterial contamination of computer keyboards in a teaching hospital," *Infect Control Hosp. Epidemiol* 2003;4(24):302-303. [PubMed] [Google Scholar]

2. Nishikawa A, Hosoi T, Koara K, Negoro D, Hikita A, Asano S, Kakutani H, Miyazaki F, Sekimoto M, Yasui M, Miyake Y, Takiguchi S, Monden M. "Face MOUSe: A Novel Human-Machine Interface for Controlling the Position of a Laparoscope," *IEEE Trans. on Robotics and Automation* 2003;19(5):825-841. [Google Scholar]

3. Smith KR, Frank KJ, Bucholz RD. "The NeuroStation- a highly accurate, minimally invasive solution to frameless stereotatic neurosurgery," *Comput Med Imaging Graph* 1994;18:247-256. [PubMed] [Google Scholar]

4. Graetzel C, Fong TW, Grange S, Baur C. "A non-contact mouse for surgeon-computer interaction," *Technol Health Care* 2004;12(3):245-257. [PubMed] [Google Scholar]

5. Kuno Y, Murashima T, Shimada N, Shirai Y. "Intelligent Wheelchair Remotely Controlled by Interactive Gestures." *Proceedings of 15th International Conference on Pattern Recognition* 2000;4:672-675. [Google Scholar]