Group project report
Hand Gesture Recognition

—

Team Rush

27th November, 2018

Introduction

Our task in this project was to use **OpenCV** and **Python** to calculate the number of fingers being held up from an image captured from a webcam and using machine learning to achieve more accurate results.

Approach

We first collected images from the internet as well as taken photos of our hands with different number of fingers and in different orientations. In order to make it easy to process the data, we chose dark and static backgrounds.

The captured images are stored in different folders which determine the number of fingers (I.e. for images with two fingers the name of the folder is two and so on).

The images are first loaded into a list and a separate list is created to keep the actual number of fingers in these images, these two combined is our dataset.

We then first shuffled out dataset and separated this shuffled dataset into training data and test data in the ratio 66 **:** 34.

The next part of the code can be divided into 3 parts -

1. Preprocessing.
   ○ Convert the image into a binary image by Gaussian blurring and thresholding.
   ○ Find the hand contours from the binary image, then find the convex hull for each contours and detect the defects in the hull.
   ○ We do this for all of the training data and pass this data to the next part of the code.
2. Fitting the data into the model.
   ○ As a predictive model we are using KNN algorithm from Scikit-Learn library.
   ○ We fit two lists in the model, the first is the number of defects of the training images and an array of the actual number of fingers for the respective images.

- Then we predict the number of fingers for each of the test images. By comparing the prediction and the actual number of fingers in each image we can estimate the accuracy of the model.
3. Capturing from the webcam and predicting the number of fingers.
   - We set up the webcam to capture an image when the Space Bar is clicked.
   - After the image is captured we find the number of defects and we pass the number of defects to the KNN model to predict the number of fingers.

Accuracy of the results

We have collected a total of 146 images and with this, we get an accuracy of 26-50%.

Contributions

We worked as a group and everyone contributed in each part of the assignment, including the code, the report and the presentation.

Resources Used

1. Python Programming Language.
2. OpenCV Library - For the Image Processing.
3. Scikit-Learn Library - For Machine Learning models.
4. Numpy Python library.

Possible Improvements

- The simpliest improvement would be to provide a larger dataset to the model, which would make the prediction more accurate.
- We can also add images with varying backgrounds. This would depend on the application of this code.
- We could use more features, so that one feature could make up for the flaws of another feature.
    - One possible feature might be the percentage of white pixels ,which it becomes a valid measurement when you get a cropped image with only the hand from fingers to the wrist.
    - Another feature might be the number of white blobs left after you subtract the palm using the radius from the center of the palm to 60-70% of the maximum distance in the contour.
- We could try out other Supervised Learning models as well and find out which one gives better accuracy in predicting the number of fingers.

Conclusion

In this project , we have implemented hand gesture recognition using Python and OpenCV. We have explored Background Subtraction, Thresholding, Contour Extraction, Convex Hull and Convexity Defects on collected images. We have used the K-Nearest Neighbour model for prediction .To show the result in real time we are capturing an image from the webcam and we are using the model to predict the number of fingers shown.