Smart Desktop Assistant using Digital Image Processing.

Adarsh Kumar Yadav(1DT17CS003) **Aman Kulshrestha(1DT17CS009)**

Anand Mohan Singh(1DT17CS010) **Hemant Dhakad(1DT17CS039)**

Students, Department of Computer Science, DSATM, Bangalore, India

Under the Guidance of

Dr. C. Nandini

Vice Principal & H.O.D. , Department of Computer Science, DSATM, Bangalore, India

ABSTRACT

Smart desktop assistants are very helpful in various computer applications and also smart desktop assistants have the ability to interact with machines efficiently through human computer interaction(HCI). In this paper a survey of applications of smart desktop assistants is presented. Applications like control the media player, write by using hand gestures, play games like sudoku and also OCR. There are various issues of the smart desktop assistants and challenges faced in this project. Review methods of these problems and their solutions presented as well. Summary of research results of applications, their methods, databases, and comparison between old and new phases are also given. Advantages and disadvantages of the smart desktop assistants are explained finally.

Introduction:-

Almost everyone uses a machine, be it a laptop or a personal computer for performing everyday work that includes from doing something as simple as making a grocery list or making PPT to developing some high tech softwares for a big firm.
However we are still stuck to using traditional and old methods of inputs which includes keyboard and mouse, which can and will cause a serious health problem in the long run.

To deal with this we have devised a virtual keyboard and virtual mouse with subtle hand gestures and made ML recognize our gestures so that we can control our computer remotely, which will help us in controlling the computer without using the standard inputs that we talked about earlier.

But for doing this we need to make a standard gesture for people to adopt easily like a standard keyboard and mouse.So, we have made an assistant which will make the lives of humans much easier when it comes to using the machine.
Smart Desktop Assistant is a project intended to make the use of personal computer easier and the use of gestures is more effective.It includes functionalities like handling the machine by your gesture and also controlling the applications and playing games using the same.

Smart Desktop Assistant utilizes Image processing for three applications
1) Virtual Air Writing: To write on the screen without using any physical contact with the computer.
2) OCR: To recognize the handwritten characters and change them into machine characters
3) Media control using gestures: To use hand gestures for controlling the operations of the VLC media player.
4) Game: To be able to play the game using hand gestures.

Proposed System:-

CNN
Hand gestures are broken down by a convolutional neural network into the following major components -
Identifying hands in the picture
Focusing on each finger despite external factors, such as light, angle, pose, etc.
Identifying unique features
Comparing all the collected data with already existing data in the database to match a hand with a name.
A similar process is followed for scene labeling as well.

II. Prior and Related work

In 1974, Ray Kurzweil started the company Kurzweil Computer Products, Inc. and continued development of omni-font OCR, which could recognize text printed in virtually any font.

In 1977 Professor Ramakant Nevatia of USC published the first journal paper (we think) on color edge detection, in which he extended the Hueckel operator, developed 4 years previously, to color images

Work by Hubel and Wiesel in the 1950s and 1960s showed that cat visual **cortices** contain neurons that individually respond to small regions of the visual field.

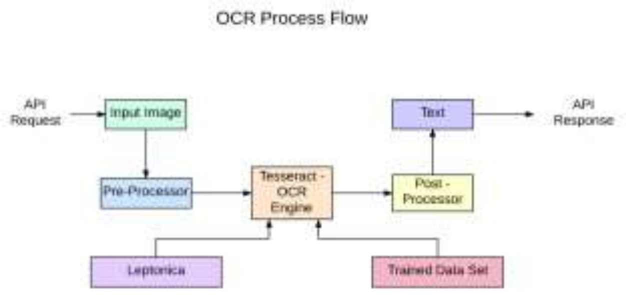
III. Methodology

KNN

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.
In the sudoku image, most of the time, similar data points are close to each other. The KNN algorithm hinges on this assumption being true enough for the algorithm to be useful. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph.

OCR

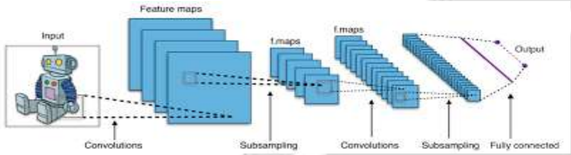
With OCR, a huge number of paper-based documents, across multiple languages and formats can be digitised into machine-readable text, that not only makes storage easier (saving a bomb on space, fireproofing, pest-control etc), but also makes previously inaccessible data available to anyone at a click.
The applications of OCR go way beyond digitisation of documents. Combined with the power of big data, artificial intelligence and low-cost internet, OCR brings the digital revolution to many who have not enjoyed its fruit so far.



CNN:

Convolutional neural networks have been one of the most influential innovations in the field of computer vision. They have performed a lot better than traditional computer vision and have produced state-of-the-art results. These neural networks have proven to be successful in many different real-life case studies and applications, like:

Image classification, object detection, segmentation, hand gestures, face recognition;
Self driving cars that leverage CNN based vision systems;
Classification of crystal structure using a convolutional neural network;
The image shows you that you feed an image as an input to the network, which goes through multiple convolutions, subsampling, a fully connected layer and finally outputs something.



KNN:-

K Nearest Neighbor algorithm falls under the Supervised Learning category and is used for classification (most commonly) and regression. It is a versatile algorithm also used for imputing missing values and resampling datasets. As the name (K Nearest Neighbor) suggests it considers K Nearest Neighbors (Data points) to predict the class or continuous value for the new Datapoint.

The algorithm's learning is:

1. Instance-based learning: Here we do not learn weights from training data to predict output (as in model-based algorithms) but use entire training instances to predict output for unseen data.

2. Lazy Learning: Model is not learned using training data prior and the learning process is postponed to a time when prediction is requested on the new instance.

3. Non -Parametric: In KNN, there is no predefined form of the mapping function.

K-NN algorithm uses the distance equations to determine if the input belongs to a classification. During the training phase, it does not compute anything, but just stores all the training data information for reference during the fitting phase. During that phase, it calculates the distance between every training data input parameter with the given input data and tries to isolate the ones with the least distance between the input. After this, it takes the first N closest training data and finds classification which is the most frequent amongst the N closest training data. The function for the distance can be set to any appropriate functions like the Manhattan distance. In the experiments, the Euclidean distance was used for the comparison.

$$D(p,q) = \sqrt{\sum_{i=1}^{n}(qi - pi)^2}$$

Algorithm

Hand Segmentation Methods

According to Nobuyuki Otsu, segmentation is done to convert gray scale image into binary image so that only two object in image which one is hand and other is background. Otsu notes again, this algorithm is used for segmentation purposes and gray scale images are converted into binary images consisting of a hand or background. A very good segmentation is needed to select an adequate threshold of gray level for extract hand from background for example there is no part of hand should have background and background also should not have any part of hand. In general, the selection of an appropriate segmentation algorithm depends largely on the type of images and the application areas. The Otsu segmentation algorithm was tested and found to give good segmentation results for the hand gesture. Otsu algorithm is nonparametric and unsupervised method of automatic threshold selection.

As each frames contains equal value of colour in a pixel, the pixel is then considered as eliminated or removed. These are applied to other pixels within each frame, and each frame is compared with respect to time. As any of the pixels between two frames do not match with each other, the pixel at the first frame retains its contain. Otsu's thresholding method involves iterating through all the possible threshold values and the measurement of spread for the pixel level for each side of the threshold is calculated for example pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum.

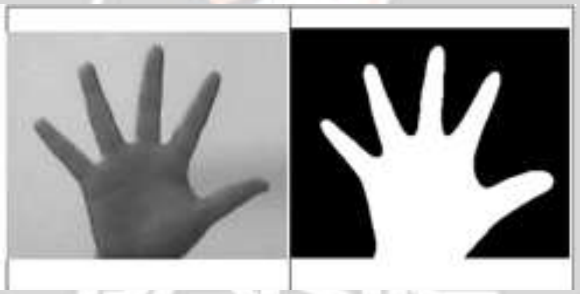


Figure 1

shows how the hand image is obtained from the real-time video in a grayscale filter and then the image is converted for Otsu's algorithm that shows the thresholded image.

Figure 1. Comparison between gray scale image and thresholded image

The limitation of the Otsu's thresholding method is that as I previously mentioned, this algorithm only considers a bimodal image where a bimodal image is an image whose histogram has two peaks. For that image, Otsu's does take a value in the middle of those peaks as a threshold value. Then, it automatically calculates a threshold value from image histogram for bimodal image. This only work when capturing a frame from video and converting the RGB format into grayscale filter.

IV. EXISTING SYSTEM

This paper aims to explore the existing options for hand gesture recognition in a common context. Most people nowadays own a laptop with a front-facing camera. If we could tap into this, we could possibly bring a more natural method of interaction to the masses. Moreover, as virtual reality devices become more common, the laptop camera may also become a viable complementary interaction device, capturing a field of view separate from the virtual reality device. In the existing system most of the gesture recognition is done using Arduino which is not a cost-effective way.

Acknowledgement
Anand Mohan Singh, Hemant Dhakad, Aman Kulshreshtha, Adarsh Kumar Yadav would like to thank Dr. C. Nandini, HOD,Department of CSE, DSATM for her constant encouragement,guidance and support throughout the course of the research
paper, without whose efforts this would not have been possible.

VII.CONCLUSION

Using OpenCv we were able to perform air writing using color detection, and we also successfully solved sudoku problems using KNN.
While performing OCR, we got 99.8 % accuracy, which means more than 99 times out of hundred we were able to recognise the character correctly.

VIII. SOURCES AND REFERENCES

1. www.superdatascience.com/opencv-facedetection/
2. “Controlling Multimedia Applications Using Hand Gesture Recognition” by Neha S. Rokade, Harsha R. Jadhav, Sabiha A. Pathan , Uma Annamalai.
3. “OpenCV-Python Tutorials Documentation” by Alexander Mordvintsev & Abid K.
4. “Enhanced Look Based Media Player With Hand Gesture Recognition” by Harshada Naroliya, Tanvi Desai, Shreeya Acharya, Varsha Sakpal.
5. Viola and Jones, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001
6. A Vision based Hand Gesture Interface for Controlling VLC Media Player,Siddharth Swarup Rautaray,Indian Institute of Information Technology,Allahabad,Anupam Agrawal,Indian Institute of Information Technology,Allahabad