International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Writing for the Specially-Abled using Object Detection

Aditya Sharma[1], Ketan Kumar Jain[2], Sushant Adlakha[3]

[1,2,3]*Student, Dept. of Computer Engineering, National Institute of Technology Kurukshetra, Haryana, India*

---***---

Abstract - Image processing involves analysis and manipulation of the digital image to extract useful information, from iris recognition, microscopic image processing to satellite image interpretation, image processing has been widely used in different domains. Unfortunately, some literate people have lost their hands due to unfavorable incidents, or due to congenital reasons, such people require bionic arms for writing, which costs over 3000 dollars. In this paper, we present the first step to provide a solution to this problem at a viable cost, which involves making use of computer vision to track the movement of marker worn on the amputated limb of the specially-abled person.

Key Words: Image Processing, Object Detection and Tracking, Human-Computer Interaction

1. INTRODUCTION

Globally many people have lost their hands due to one reason or another, might be because of accidents or even by congenital conditions. These people, even after acquiring the knowledge to read and write, become dependent on other people for writing-related works.

Multiple solutions to the problem exist in the market like a mechanical prosthetic arm, but it lies in the range of $3,000-$5,000. A functional prosthetic arm with a split hook at the end might cost $10,000. A myoelectric prosthetic arm with a human-like, functioning hand might cost $20,000-$30,000 or more [1]. Even covers for prosthetic arms are in the range of $200 to $300. Because of financial constraints, it is still not feasible for the masses. Another solution to the problem can be thought of using speech recognition technology, but still the problem might exist while working in loud and noisy environments, and in parallel certain section of the population who have lost ability to speak would not be able to make use of it.

Using non-wearable technology, which primarily involves the use of computer vision, provides a better solution to the problem in comparison to wearable technology, at a viable cost along with ease of use and increased feasibility.

With the advancement in technology, this solution can be tackled if the limb movements of the user could somehow be captured. One such potential solution may exist in the field of image processing, where a marker can be worn on the amputated limb of the specially-abled and tracked using image capturing systems. This approach allows the specially-abled people to write in thin air without making use of any external mousepad or trackpad. The motion of the marker attached to their limbs could be detected by computer vision. Though writing in this approach is a bit cumbersome in comparison to traditional handwriting because of the lack of an existing reference plane, which is paper in case of conventional handwriting.

2. RELATED WORK

Riso [2] discussed the developmental status of bionic arms and what could be added to provide kinesthetic sensibility. It also suggested that the user could be made aware of hot and cold surfaces via the use of thermal sensors.

Dong et al. [3] proposed a new approach for controlling muscle driven system, where computational time is a factor of the number of joints in the bionic arm, instead of being a factor of the number of muscles, which would allow adding a higher number of muscles in the bionic arm as per the requirement.

Chandan et al. [4] implemented real-time object detection using Single Shot Detector Algorithm, which is based on VGG-16 architecture. The model trained was capable of identifying 21 different objects like a train, bus, cycle, dog, etc., with confidence levels of above 95%. This provides a possibility of integrating the model with CCTV cameras to detect arms and ammunition.

Byoko et al. [5] used the HOG (Histogram of Oriented Gradients) algorithm for face recognition. They found that the OpenCV library performs better for face recognition in comparison to the Dlib library.

Shang et al. [6] provided a way for faster and accurate detection of face and eyes. AdaBoost algorithm for face detection was used in conjunction with Hough change circle detection to find the relative position of the pupil on the face. It could further be used to determine whether the user is paying attention to the object or not.

Wang et al. [7] proposed an approach that allows color-based motion capture in different environment settings. They made use of a few sets of web-cameras with a multi-colored t-shirt to serve as a landmark for motion tracking and capture. The approach was significantly simpler and economical as compared to the traditional motion capture methods.

Palekar et al. [8] proposed an approach for automating the logging of vehicles for traffic regulation, parking systems, etc. by tracking their number plates with the help of OpenCV for image processing and Tesseract, a Python library, for optical character recognition. The obtained results were then refined by using neuro-fuzzy logic to remove the unnecessary text.

3. PROPOSED APPROACH

The work on this specific topic starts from the basic idea of recognition of a distinct colored object in the image frame, referred to as a marker. Image processing libraries in python like OpenCV, SimpleCV, Pillow, etc., can be used for the purpose. Once we are able to detect the specific location of the marker in the image frame, we can assume that the center of our marker shall serve as a writing tip. We could then select the thickness of the writing tip, and keep on repeating the step of finding the marker in forthcoming image frames. In this way, we would be able to detect the motion of the marker. This movement is then recorded in an image file, and further Google Vision API can be used to convert images to text. A flowchart of the entire workflow is shown in Figure 1.

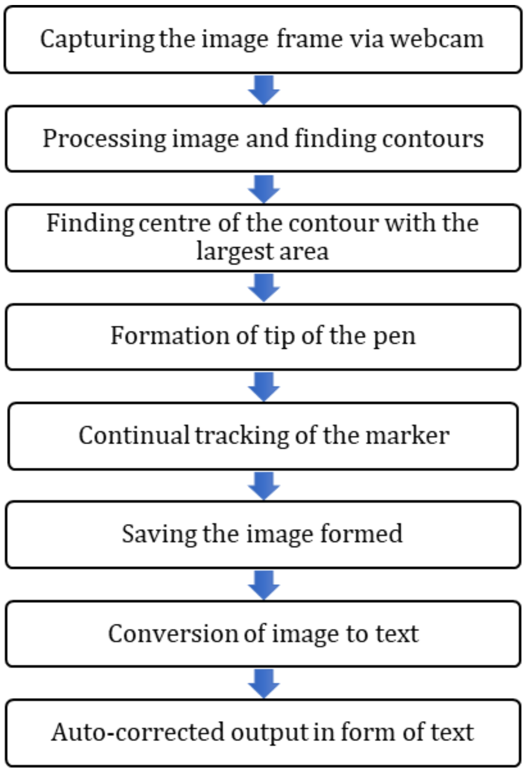

Figure 1 Steps in generating text from the input image

4. EXPERIMENTATION

The basic idea of the application begins with the recognition of the marker object. A variety of objects like a round bottle cap, flat tip of a pen, small ball, taped finger, etc. were tested to justify the usability for the target users. Trying to minimize the additional hardware resources, the input is taken from the web-camera of a laptop. The approach targets a particular color in a defined range, i.e., covering most of the possible shades of that color from the image frame. This is used to generate a color mask that can actively filter out only the areas with the desired color objects.

Some noise is also captured by internal factors of the camera like the varying electricity, heat, and sensor illumination levels or by interference due to improper positioning of the camera under the light source. This can be removed to some extent by applying some structural operations to the captured image called morphological operations [9]. Primarily, the generated mask is eroded to remove all rough edges by erosion, which computes a local minimum over the area of the image kernels and removing the area that lies beyond it. This, however, also removes a part of original image data, which might be needed in further computation. Thus, to compensate for the area removed by erosion, dilation is applied which computes the maximal pixel value overlapped by a kernel and replaces the image pixel in the anchor point position with that maximal value, thereby making the picture slightly bigger than the eroded image. These operations, when combined, help to reduce noise in the image significantly and yields a better-quality image for further processing. The same is represented in Figure 2(a) and 2(b) with different sized marker objects.

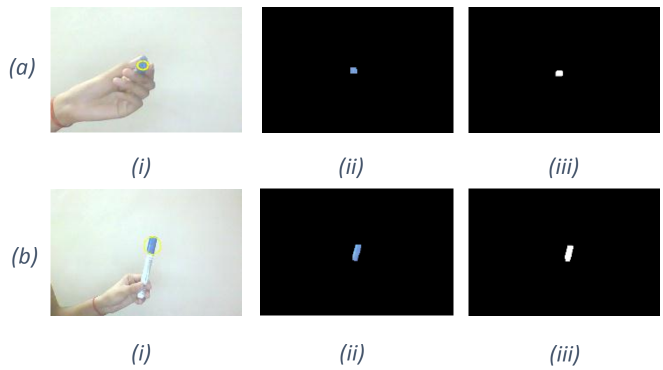

Figure 2 Different types of marker objects (i) and their corresponding filter masks (ii) and (iii) for: (a) small round tip, (b) broad flat end

Then, the contours (a group of pixels with similar colors) in the image are recognized. Out of all the contours selected in the image, the required contour must have the largest area among other contours. The center of the selected contour is ascertained, and then an arbitrary radius was fixed, which represented the tip of the writing marker. The center of the writing marker is concentric with contour selected. This process is repeated till the user indicates the end of the current word.

As the image frame progresses, the movement of contours on the frame is recorded. After the removal of background objects and noise, an image corresponding to user's movement is generated. This is presented to the user by projecting it onto the user's screen. This is as shown in Figure 3.

(i) (ii) (iii)

Figure 3 Tracking of the marker position
(i) Original image visible to the user
(ii) Background subtraction
(iii) Color Inversion

Thus, continuous capture and simultaneous display of recorded data gives an impression to the user that they are able to write in the air. Such kind of markers can be worn on a finger or amputated hand. Since the size of the marker does not hold correspondence to the size or width of rendered handwriting, various kinds of markers can be utilized for the purpose.

Once the user has spelled a word, they may indicate the end of the word or discard it with a gesture. The visual input captured so far is processed or rejected depending on the user's actions. If the user wishes to go ahead with the word, then the corresponding frame is translated using Google Vision API in the background. Meanwhile, the user is free to enter more words or phrases. The response from the API is captured and is parsed to the auto-correction module provided by autocorrect, a Python library. The refined text output is appended to the terminal output or the text file based on the preference of the user.

5. CONCLUSION AND FUTURE SCOPE

There are some fields that require writing records where specially-abled people are not able to compete with other people because of their disability. This paper represents our one such attempt to empower them. Since it's a computer vision-based application, appropriate lighting conditions are must, in case of improper lighting conditions, the threshold values of the color of marker to be detected can be adjusted. In the future, the system would be capable of recognizing certain gestures which would control the various computer actions like opening an application or file, scroll through media, etc. This could potentially substitute the need for input devices like a keyboard or mouse, which would further empower specially-abled people to use a computer system more effectively.

REFERENCES

[1] D. K. Blough, S. Hubbard, L. V. McFarland, D. G. Smith, J. M. Gambel, and G. E. Reiber, "Prosthetic cost projections for servicemembers with major limb loss from vietnam and OIF/OEF," *J. Rehabil. Res. Dev.*, vol. 47, no. 4, pp. 387–402, 2010.

[2] R. R. Riso, "Strategies for providing upper extremity amputees with tactile and hand position feedback - Moving closer to the bionic arm," *Technol. Heal. Care*, vol. 7, no. 6, pp. 401–409, 1999.

[3] H. Dong, N. Figueroa, and A. El Saddik, "Muscle force control of a kinematically redundant bionic arm with real-time parameter update," *Proc. - 2013 IEEE Int. Conf. Syst. Man, Cybern. SMC 2013*, pp. 1640–1647, 2013.

[4] G. Chandan, A. Jain, H. Jain, and Mohana, "Real Time Object Detection and Tracking Using Deep Learning and OpenCV," *Proc. Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2018*, no. September, pp. 1305–1308, 2018.

[5] N. Boyko, O. Basystiuk, and N. Shakhovska, "Performance Evaluation and Comparison of Software for Face Recognition, Based on Dlib and Opencv Library," *Proc. 2018 IEEE 2nd Int. Conf. Data Stream Min. Process. DSMP 2018*, pp. 478–482, 2018.

[6] L. Shang, C. Zhang, and H. Wu, "Eye focus detection based on OpenCV," *2019 6th Int. Conf. Syst. Informatics, ICSAI 2019*, no. Icsai, pp. 855–858, 2019.

[7] R. Wang, "Practical Color-Based Motion Capture," *Proc. - SCA 2011 ACM SIGGRAPH / Eurographics Symp. Comput. Animat.*, pp. 139–146, 2011.

[8] R. R. Palekar, S. U. Parab, D. P. Parikh, and V. N. Kamble, "Real time license plate detection using openCV and tesseract," *Proc. 2017 IEEE Int. Conf. Commun. Signal Process. ICCSP 2017*, vol. 2018-January, pp. 2111–2115, 2018.

[9] B. G. Batchelor and F. M. Waltz, "Morphological image processing," in *Machine Vision Handbook*, Springer, London, 2012, pp. 801–870.