World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:12, No:6, 2018

Hand Gesture Detection via EmguCV Canny Pruning

N. N. Mosola, S. J. Molete, L. S. Masoebe, M. Letsae

Abstract**—**Hand gesture recognition is a technique used to locate, detect, and recognize a hand gesture. Detection and recognition are concepts of Artificial Intelligence (AI). AI concepts are applicable in Human Computer Interaction (HCI), Expert systems (ES), etc. Hand gesture recognition can be used in sign language interpretation. Sign language is a visual communication tool. This tool is used mostly by deaf societies and those with speech disorder. Communication barriers exist when societies with speech disorder interact with others. This research aims to build a hand recognition system for Lesotho's Sesotho and English language interpretation. The system will help to bridge the communication problems encountered by the mentioned societies. The system has various processing modules. The modules consist of a hand detection engine, image processing engine, feature extraction, and sign recognition. Detection is a process of identifying an object. The proposed system uses Canny pruning Haar and Haarcascade detection algorithms. Canny pruning implements the Canny edge detection. This is an optimal image processing algorithm. It is used to detect edges of an object. The system employs a skin detection algorithm. The skin detection performs background subtraction, computes the convex hull, and the centroid to assist in the detection process. Recognition is a process of gesture classification. Template matching classifies each hand gesture in real-time. The system was tested using various experiments. The results obtained show that time, distance, and light are factors that affect the rate of detection and ultimately recognition. Detection rate is directly proportional to the distance of the hand from the camera. Different lighting conditions were considered. The more the light intensity, the faster the detection rate. Based on the results obtained from this research, the applied methodologies are efficient and provide a plausible solution towards a light-weight, inexpensive system which can be used for sign language interpretation.

Keywords**—**Canny pruning, hand recognition, machine learning, skin tracking.

I. Introduction

ASSISTIVE visual technologies have enjoyed considerable use due to various technological expansions. Technological trends are exponentially rising. For example, the envisaged Internet of Things (IoT), enabled by the concepts of cloud computing, is almost being adopted by most of technology users and organizations. Visual-based systems implementing hand gesture recognition are used in applications involving HCI, automatic control devices such as drones, assistive communication in sign language interpretation etc. Technology is directly or indirectly evolving the lives of people. Hand gesture recognition computer systems have been used in HCI for input and output purposes. This eases interaction between computers and end-users through HCI, as the complexities associated with human needs continue to grow [1]. The idea behind hand gesture recognition techniques in HCI is meant to enable human beings to interact in a friendly manner and more efficiently with computer systems. Various HCI paradigms have been proposed to enhance this interaction, spanning voice recognition, face detection, hand detection etc. Collectively, these techniques can be classed as gestures. Various researchers have tried to define gestures. However, due to its wide use in different applications, it is still difficult to have a specific definition of gestures. In the quest to have a proper definition, [2] defined gestures as motion of parts of the body with the intention to relay a message or interact with other agents. This paper adopts this definition and bases itself on the gestures being expressive movements of parts of the body which convey a message or bears meaning. Gestures are classified under two categories, these are: dynamic and static gestures. Dynamic gestures change over time, whilst static gestures are observed over a short period of time. For example, a waving hand is a dynamic gesture whereas a stop sign is a static gesture. For use in communication, a hybrid communication model consisting of dynamic and static gestures is used. This hybrid model is referred to as gesture recognition, which is a process of detecting and recognizing continuous gestures from an input stream. Furthermore, gesture recognition techniques have been employed in various applications, e.g. in computer aided design (CAD) systems. CAD systems are a sub-field of HCI. These systems are useful when interpreting and manipulating multi-dimensional (e.g. 3D) data which may include gestures. CAD systems allow humans to interact with computers more efficiently. They allow humans to point or rotate 3D models through hand gestures. Another area of HCI where gesture recognition is increasingly being used is virtual reality (VR) systems. VR systems are applicable when a simulation of an environment, representing a physical/real or imaginary world, is created [3]. Most VR systems simulate an environment which is a visual perception/experience. Upon further manipulation of the environment and applying computer graphics concepts, the environment is displayed on a raster device, such as a monitor or stereoscopic display. Various fields also make use of VR systems, including medical imaging, gaming applications, haptic systems etc., and all of them make heavy use of gesture recognition. Moreover, hand gesture recognition techniques are widely adopted for sign language interpretation. Sign language is a visual communication tool meant to break down communication barriers between disabled societies. Sign

N. N. Mosola is a Lecturer in Computer Science and member of the Mathematics and Computer Science department, National University of Lesotho, P.O Roma 180 (corresponding author, phone: +266 6260 6670; e-mail: nn.mosola@nul.ls).

M. J. Sekese is a Lecturer in Computer Science and member of the Mathematics and Computer Science department, National University of Lesotho, P.O Roma 180 (e-mail: sj.molete@nul.ls).

L. S. Masoebe and M. Letsae are final year Computer Science students at the National University of Lesotho, P.O Roma 180 (e-mail: masoebesamuel@gmail.com, mokhutli@gmail.com).

language has been in existence since the advent of civilization and long before the existence of computers. Assistive technologies addressing the gap between able and disabled societies exist and are now being used to interpret sign language. A sign language interpreting system affords the opportunity for the deaf community to interact with non-signing persons, without the need of a human interpreter. Such systems are vital as sign language has become an integral part of communication, which is used extensively and globally [4]. These systems are developed to output speech, text, images, or a combination of these, to aid different kinds of societies.

Another use of hand gesture recognition is tele-operation of automated control systems. In this application area, hand gestures can be used to control unmanned aerial vehicles (UAVs), such as Parrot Bebop drones. Furthermore, such systems can be used in hazardous environments such as during a rescue mission, where human lives would be endangered if deployed in a disaster-struck environment. Being able to remotely operate an UAV could also be used for training simulations for human rescue teams. The emergence of intelligent driverless cars brings the concept of hand gesture recognition techniques for deployment in vehicular applications. Such systems create gestural interfaces for drivers to interact with vehicles effectively.

Although hand gesture recognition is being used and continuously researched, capturing and recognizing gestures requires expensive tools and complex hardware resources such as Microsoft Kinect [5], data gloves, etc. Hence, this paper proposes a low-cost solution which utilizes a laptop's webcam for capturing and recognition of gesture-based inputs. Just like many other gesture recognition systems, the proposed system adopts the basic processes of detecting and recognizing gestures. These are: image acquisition, pre-processing, and gesture recognition. The proposed solution is meant to be deployed on devices with low memory specifications such as smart phones. Hence, the solution is a portable light-weight system. Most research focuses on hand gesture recognition using a single camera, from a single point of view. This research contributes to the body of knowledge by using techniques of hand recognition to detect various orientations, which are important in sign language. While a significant number of researchers assume that a hand being recognized is upright and aligned with the plane of the camera, this paper focuses on hand gesture detection and orientations from different angles and under different light intensities.

The rest of this paper is structured as follows: section II discusses existing literature, section III introduces and discusses the proposed model solution, section IV discusses the results obtained.

II. Literature Review

Various research dedicated to hand gesture recognition has been conducted before. A glove-based hand recognition system equipped with fibre optic sensors which recognized about 14 alphabetical characters was developed [6]. This system served as a basis for more glove-based systems subsequently following it. For example, Power glove, Dexterous hand master, were all implemented based on the design of [6]. However, the mentioned systems are a single-hand detection system and were not able to capture and recognize dynamic gestures in real-time. These systems also required a lot of computing resources as they integrated sensors along the fingers of the glove.

In the quest to build a less resource-intensive hand gesture recognition system, a system which aims to improve the accuracy of captured hand gestures using less computing power was developed [7]. Furthermore, the proposed system uses a neural network to classify the captured gestures. Although neural networks are easily implemented, the training process takes a lot of time to complete. Moreover, occlusion may also make it difficult for a neural network implementation to be used for sign language translation. As with most neural network implementation, the system also encounters problems when a large dataset of hand gestures is used, in different environments [8]. This is because when there are more neural network inputs, the processing within the hidden layers becomes slow, while the network grows large [9]. Another neural network implementation for hand recognition was developed to transform Indian sign language into textual output, using the Leap motion device. The device detects data which come from the leap controller. The data may include pointing, waving, grabbing, etc. [10]. To advance the research conducted in [10], [11] proposed a hand gesture recognition system utilizing the leap motion controller and neural networks. The system was developed to detect hand movements. The movements and gestures were then converted to computer commands, which mapped the Australian sign language. The two systems are inherently faced with the disadvantages of neural network implementations for large datasets. The latter also has low accuracy.

Research conducted in [12] proposes a hand recognition system using Microsoft Kinect. The system recognizes American Sign Language (ASL). The system uses a depth camera, which is a sensor on the Kinect, for ASL alphabet detection. While the results of the system were impressive as it achieved 90% accuracy, the system is costly as it uses the Microsoft Kinect, which is an expensive piece of hardware. A hand recognition system aimed for Taiwanese sign language translation was proposed and developed. This system uses Hidden-Markov Models (HMMs) and a data glove. The system consists of a grammar generating the Taiwanese language (i.e. $G :\rightarrow L$). The grammar G is used to match the semantics for the sentences formed. The HMM used provides probabilistic estimates of a series of hand gestures and movements. Hence, the HMM increases the accuracy of the gesture recognition. However, HMMs can lead to severe over-fitting when large parameters are used, thus resulting in a very large transition matrix [13].

A feature extraction hand gesture recognition system for detecting Indian Sign Language (ISL) was proposed and developed in [14]. The system integrates Camshift tracking algorithm, which is based on the MeanShift tracking algorithm.

III. Proposed System

This section introduces the proposed hand gesture recognition system for Lesotho's Sesotho and English Language Interpreter, abbreviated as LESELI. The aim of the proposed LESELI system is to investigate how feasible it is to recognize different hand gestures from various orientations. The objective of the research is to demonstrate how an efficient hand recognition system can be without the use of any special, expensive hardware. Fig. 1 depicts the proposed system architecture.

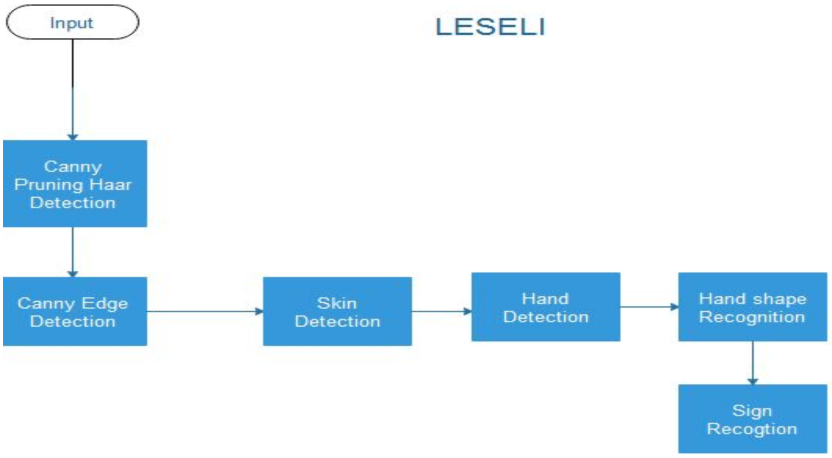


Fig. 1 Proposed model solution

The proposed system is developed according to the following functional requirements:

A. Hand Detection

The system must be able to automatically detect the position of the user's hand. This enables the system's hand tracking window to capture the region-of-interest from a captured image. The system uses the canny pruning Haar detection using cascade classifiers. This method is based on the work conducted in [15], where it was proven to be accurate and efficient. This method uses an image processing algorithm called canny edge detection for capturing an area of interest from a captured image. The canny edge detection is a multi-stage algorithm used to detect a wide range of edges in an optimal manner. The algorithm aims to achieve low error rates, meaning that a good detection of existing edges will be captured. Another attribute of the canny edge detection algorithm is good *localization*. This means that the distance between edge pixels detected and the real edge pixels must be minimized. The algorithm also achieves *good minimal response*. This means that only one detection response per edge is captured.

Implementation of the canny edge detection algorithm requires noise filtering. The next sub-section discusses the noise filtering method based on the Gaussian filter below.

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\left(i-(k+1)\right)^2+\left(j-(k+1)\right)^2}{2\sigma^2}\right); \quad 1 \le i,j \le (2k+1) \tag{1}$$

The performance of the system's detection engine is affected by the size of the Gaussian kernel. The bigger the size will be, the lower the detection will be. An increase of the Gaussian filter kernel increases the localization error used to detect the edges on an object. In most cases, a 5x5 matrix is used to fit the kernel. However, this varies according to various specific situations [16].

B. Gradients of a Detected Image

An edge in an image may point in a variety of directions. Thus, the Canny algorithm uses four filters to detect horizontal, vertical, and diagonal edges. The edge detection operator returns a value for the first derivative in the horizontal and the vertical directions (i.e. G_x and G_y) respectively. From this, the edge gradient and direction can be determined as follows:

$$G = \sqrt{{G_x}^2 + {G_y}^2} \tag{2}$$

The Canny pruning Haar detection utilizes a weak Haar-like classifier which is optimized on a set of training data. The method provides accurate detection in real-time. The next sub-section discusses the skin detection procedure.

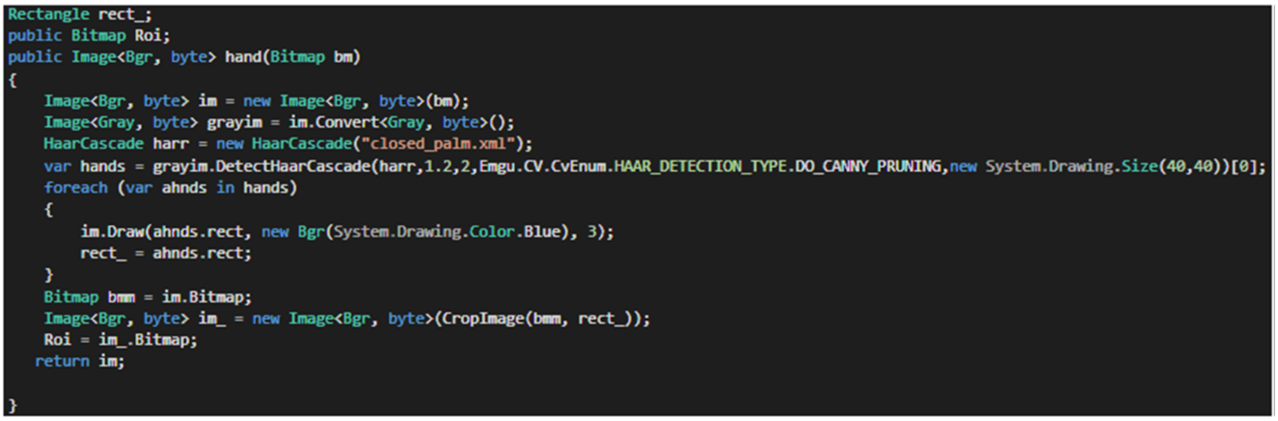
C. Skin Detection

The skin detection is used to mark skin pixels and remove non-skin background. This filters the unwanted noise in the environment. A robust skin detection process based on the algorithm in [17] is used. This process defines a set of meta-rules in RGB color model. This model is applied on a captured image to mark skin. The skin image may consist of holes, which need to be filled. The process of filling the skin is adopted from [18]. The next section discusses the results obtained.

IV. Discussion of Results

The results presented herein were obtained after executing the proposed system on a system running Microsoft Windows 8, installed on a HP laptop, with a built-in front camera. The system was developed using visual studio 2013, using C# as the programming language.

The system was tested with various experiments. Different users were asked to sit in-front of the laptop to capture their hand gestures Over 40 gestures were made, and the hand was detected. The screen-shots below depict the system executing and capturing hand gestures in real-time. The results demonstrate a positive and plausible achievement of hand gesture recognition. Fig. 2 shows the algorithm which performs hand detection, in C# programming language.



```
Rectangle rect_;
public Bitmap Roi;
public Image<Bgr, byte> hand(Bitmap bm)
{
    Image<Bgr, byte> im = new Image<Bgr, byte>(bm);
    Image<Gray, byte> grayim = im.Convert<Gray, byte>();
    HaarCascade harr = new HaarCascade("closed_palm.xml");
    var hands = grayim.DetectHaarCascade(harr,1.2,2,Emgu.CV.CvEnum.HAAR_DETECTION_TYPE.DO_CANNY_PRUNING,new System.Drawing.Size(40,40))[0];
    foreach (var ahnds in hands)
    {
        im.Draw(ahnds.rect, new Bgr(System.Drawing.Color.Blue), 3);
        rect_ = ahnds.rect;
    }
    Bitmap bmm = im.Bitmap;
    Image<Bgr, byte> im_ = new Image<Bgr, byte>(CropImage(bmm, rect_));
    Roi = im_.Bitmap;
   return im;

}
```

Fig. 2 Hand detection algorithm

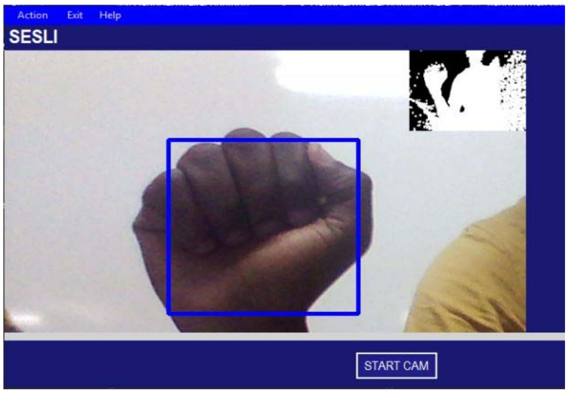


Fig. 3 Closed palm hand detection

Fig. 3 depicts the system being used to detect a closed palm using the Canny pruning Haar detector algorithm. The system uses the algorithm depicted in Fig. 2. The system tracks the hand's location. Upon finding the position of the hand, the system extracts the area of interest through a feature extraction done by the Canny pruning, which utilizes the Canny edge detection algorithm. Thereafter, the system creates a convex hull of the region highlighted with a blue square. The region highlighted as the convex hull is then cropped after being detected. Thus, the area of interest is cropped and stored as an image in a folder generated by the HaarCascade file written in xml. The hand detection algorithm is dependent on the distance of the user from the camera.

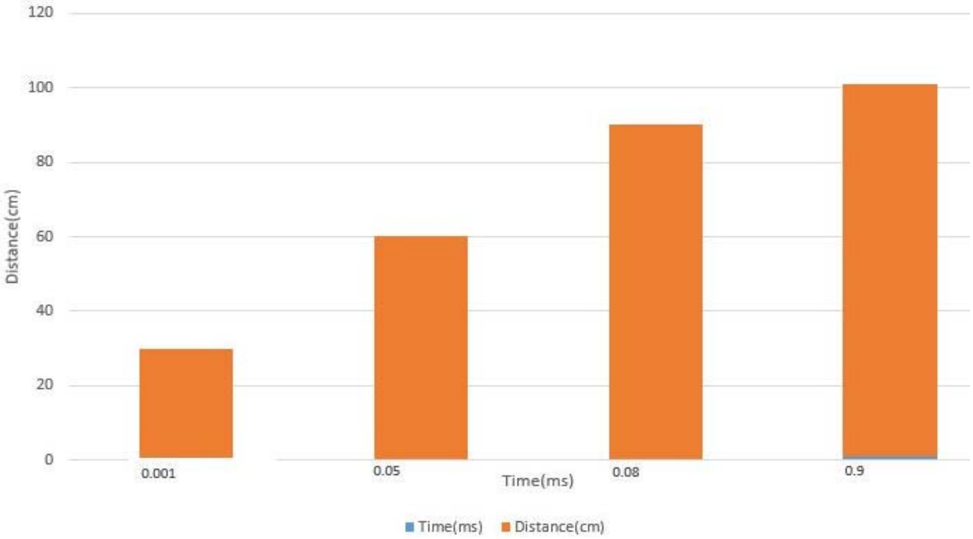


Fig. 4 Detection rate

It is important to note that the position and distance of the hand being detected affects the accuracy of the system. The system is more accurate when the hand is closer to the camera and upright. The system becomes less accurate when the hand is distant from the camera. Fig. 4 shows different times, in milliseconds (ms), when the system is used to detect a hand from 30 cm, 60 cm, 90 cm, and 100cm away from the camera. The results depict the idea that the distance factor has a bearing on the hand detection results obtained.

Fig. 5 depicts a skin detected image from a detected hand.

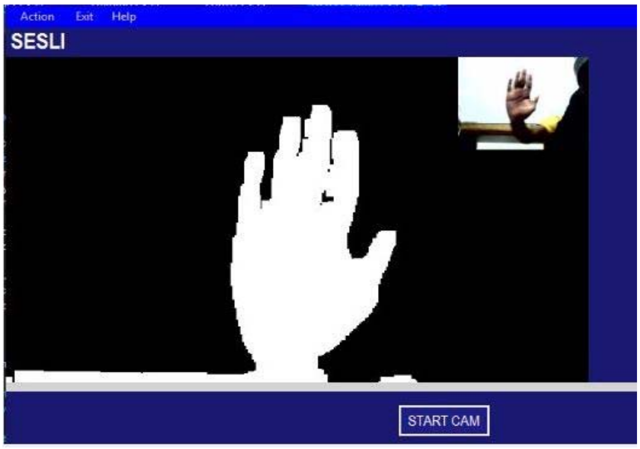


Fig. 5 Skin detection

The skin detected image in Fig. 4 was captured 30 cm from the camera.

REFERENCES

[1] H. Birk and T. B Moeslund, “*Recognizing Gestures from the Hand Alphabet Using Principal Component Analysis*”, Laboratory of Image Analysis, Aalborg University, Denmark, 1996.

[2] A. Wilson and A. Bobick, “*Learning visual behavior for gesture analysis*,” In Proceedings of the IEEE Symposium on Computer Vision, Coral Gables, Florida, pp. 19-21, November 1995.

[3] H. Grant, C. Lai, “*Simulation modelling with artificial reality technology (smart): an integration of virtual reality simulation modelling*”, Proceedings of the Winter Simulation Conference, 1998.

[4] R. George, K. Gerard J. Nigel, “*Hand Gesture Recognition for the Application of Sign Language Interpretation*”, Karunya University, Coimbatore.

[5] msdn.microsoft.com/enus/library/hh438998.aspx. (Accessed 13 Feb. 2018).

[6] K. Jacobs, M. Ghaziasgar, I. Venter, and R. Dodds, “*South African Sign Language Hand Shape and Orientation Recognition Using Deep Learning*”. Southern Africa Telecommunications Networks and Applications Conference, George, Western Cape, 2016.

[7] P. Premaratne and Q. Nguyen, “*Consumer electronics control system based on hand gesture moment invariant*”, IET Computer Vision, 1(1), 2007, 35-41.

[8] T. Khan and A. H Pathan, “*Hand Gesture Recognition based on Digital Image Processing using MATLAB*”. International Journal of Scientific & Engineering Research, Volume 6, Issue 9, September 2015.

[9] https://www.sciencedirect.com/science/article/pii/S0895435696000029, (Accessed 13 Feb. 2018).

[10] P. Karthick, N. Pratibha, V. B. Rekha, S. Thanalaxmi, “*Transforming Indian Sign Language into Text Using Leap Motion*”, International Journal of Innovative Research in science, Engineering and Technology, pp. 10906-10908,2014.

[11] L. E Potter, J. A. L Carter, “*The leap Motion Controller: A view on sign language*”, Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collabration.pp.1-4, 2013.

[12] R. Bowden, D. Windridge, T. Kadir, A. Zisserman, M. Brady, “*A linguistic Feature Vector for the Visual Interpretation of sign Language*”, 8th European conference on computer vision, Prague, Czech Republic, proceedings, part 1, pp: 390396, 2004.

[13] Z. Ghahramani, “*An Introduction to Hidden Markov Models and Bayesian Networks*”, International Journal of Pattern Recognition and Artificial Intelligence, vol. 15, no. 1, pp. 9–42, 2001.

[14] G. R. Bradski, “*Real time face and object tracking as a component of a perceptual user interface*”, in Applications of Computer Vision, 1998. WACV’98. Proceedings., Fourth IEEE Workshop on. IEEE, 1998, pp. 214–219.

[15] P. Viola, M. J. Jones. “*Robust real-time face detection*,” International Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004.

[16] https://en.wikipedia.org/wiki/Canny_operator.

[17] G. Gomezand E. Morales, “*Automatic feature construction and a simple rule induction algorithm for skin detection*,” in Proc. of the ICML workshop on Machine Learning in Computer Vision, 2002, pp. 31–38.

[18] M. Ghaziasgar, J. Connan, and A. B. Bagula, “*Enhanced adaptive skin detection with contextual tracking feedback*,” in Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), 2016. IEEE, 2016, pp. 1–6.