Development of a sign language interpretation glove with limited resources

Majid Azizi*, Elliot Bodin†, Fredrik Johansson‡,
School of Innovation, Design and, Engineering
Mälardalens University, Västerås, Sweden
Email: *mai20018@student.mdu.se, †ebn21011@student.mdu.se, ‡fjn20007@student.mdu.se

***Abstract*—This paper is a work of students exploring the possibility to develop a cost-effective system able to interpret sign language, with a focus on the sign language alphabetic characters. The proposed system consists of different sensors together with a microcontroller attached to a glove, which is then used to discern different finger orientation as sign language characters. The bend of the fingers is determined using a setup of an LED and a photo resistor attached within a flexible tube, and the separation of the fingers is determined using hall effect sensors and magnets. Two methods of classifications are presented, wherein the first method consists of a binary approach to the representation of the finger states. Each finger is determined to be in one out of a possible three different "bend states" and one out of two "separation states". The combination of the different states then corresponds to different characters. The second method consists of using the mean value and standard deviation in order to determine probabilities of the finger states and from there determine what character the states corresponds to. Results are satisfactory for a system intended as a prototype and a proof of concept. A lot of improvements can be made in both hardware and software, and these improvements are presented as future work.**

***Index Terms*—accelerometer, classification, esp32, hall sensor, photo resistor, sign language**

I. Introduction

Sign Languages (SLs) are commonly used by deaf and mute people to communicate. They consist of different hand signals and motions to convey meaning and form words. Most able people do not know any form of SL, which can cause a gap in understanding and a difficulty in communication between people who use SL and people who does not. Therefore, there are people who would benefit from a system that can translate SL into written or spoken languages. Such systems do exist today, but they are often very expensive or not very portable, or both. Some systems use a camera to interpret what hand gestures and signals mean, but this can be limiting in the form of portability, interruptions from the environment and limited operational space. Other systems use sensors attached to a glove in order to translate SL. These systems can be less limiting, but also has a tendency to be expensive since these systems commonly use flex sensors, which are expensive components, to approximate the bend of the fingers.

The project in this paper was carried out by students from Mälardalens University with the aim to study and learn about how to work with sensors and also to propose a possible solution to a system which is affordable and portable given a limited budget and timeframe. This system estimates the pose and orientation of both the hand as a whole and each individual finger with the help of a combination of different sensors attached to a glove. Since the budget of this project did not allow for flex sensors to be used, a different solution was explored. The bend of the fingers is determined by having an LED and a photodiode in a tube, causing the bend of the tube to correspond to the output value of the photodiode. Hall sensors and magnets are attached to the side of the fingers to determine finger separation. Lastly, an accelerometer is used in order to determine the orientation of the hand. These sensor values are read with the help of a microcontroller and the corresponding alphabetic SL character is then interpreted. The following sections will go over other possible solutions, give a more in-depth explanation of how the different parts of the developed system works, and will go over the experiment results and system validation.

1) Limitations: This project is created by 3 students with a total timeframe of 3 weeks, with a total budged of 300kr.

2) Research Questions: Some research questions considered during this project were if it's possible to create a sign language interpenetration glove with a low budget. Also, if the glove can distinguish between the different signs in the American sign language.

- Is it possible to create a sign language interpenetration glove with a budget of 300kr.
- Is it possible to distinguish between the different signs. If so, which?

II. Related Work

A. Optical glove controller using LDR

This project idea proposes a solution for controlling a motorized hand sculpture in a cost-effective manner. Instead of utilizing expensive flex sensors, the approach involves using light-dependent resistors combined with an LED which is connected through a flexible tube. As the fingers bend, the light intensity changes, affecting the voltage read by the Arduino. While the author does not specify whether the angle of finger bending can be quantified, the accompanying video suggests that it may be possible to determine finger curvature based on the voltage readings. [1] The idea of flexible tube is a cheaper alternative to flex sensors. Therefore, it will be taken into consideration during this project.

B. CNN and camera based sign language recognition

The Department of Computer Engineering in Visakhapatnam Institute of Technology [2] made a project to translate between multiple languages. The languages being Indian Sign Language(ISL), American Sign Language(ASL), American text and Indian text. their approach was to use a deep convolutional neural network (CNN) which was trained from the state of the art in classifying static ISL, with a dataset of 35,000 images. The CNN got a 99.72% accuracy on coloured images and 99.90% accuracy on gray scaled images. This model was then retrained to comprehend ASL. Their system works by first feeding live image data from a camera, extracting the hand from the image and then the processing was done. Worth noting, none of the actual algorithms used was ever presented, only the result was.

C. 3-D hand motion tracking glove using accelerometers

A paper published from Seoul, Korea in a collaboration between the Department of Biomedical Engineering and the Department of Computer Engineering [3] describes how they developed a 3-D hand motion tracking glove using accelerometers. They were able to create a 3-D model of a hand which displayed the orientation and motion of the glove. This was achieved by using three tri-axis accelerometers on a glove, computing the data with a controller and transmitting the data to a pc using Bluetooth. The accelerometers were attached to the tip of the long finger, the back of the hand and lastly, on the tip of the thumb. The system was able to track the orientation and motion of the hand and display it using the hand model through the kinematic chain theory. Three different gestures were performed and were recognized by the system. These gestures were rock, paper and scissor. The system achieved a 100% recognition rate over a total of 150 trials. Though the project was successful, faster computation times and more advanced recognition methods can be used to improve the system significantly.

An accelerometer would be helpful to be able to determine hand orientation in order to distinguish between similar signs such as "I" and "J". It could also be used to determine different gestures, which a lot of sign language is composed of.

III. State of the art

The University of Patras[4] investigated deep neural network (DNN) methods of solving sign language (SL) translation. They took into great consideration the SL structure. Everything from hand shape, position, orientation, movement and so on is taken into consideration when developing their DNN's. They use 6 different Large scale publicly available SLR datasets and created a new one to train and validate their system. In the end the university created/used 4 different DNN-based approaches; SubUNets, GoogLeNet + TConvs, I3D+BLSTM, 3D-ResNet+BLSTM. The results differed a lot depending on the signer. Where some best presented result were up to 80% accuracy and the lower end was around 60% accuracy. Most models converged in performance after 10 epochs, but were run around 30 epochs. In the end, the overall best-performing model was I3D+BLSTM, but depending on the environment, signer, wanted advantages and drawback, other methods might be more fitting for another case.

A machine learning algorithm such as DNN would classify which sign the glove tries to interpret. But because of the time frame, this is out of this project scope.

IV. Background

A. Sign language

Sign language is a fully developed visual-gestural language that serves as the primary mode of communication for deaf individuals and those with severe hearing loss. It is estimated that 70 million people around the world use sign language as their first language [5].

Conversational sign language and finger spelling (Alphabetic Sign Language) are two distinct components of signed languages, each serving different communicative purposes. Conversational sign language, such as American Sign Language (ASL), is a fully developed natural language with its own grammar, syntax, and vocabulary [6]. Through the use of a combination of hand shapes and movement, facial expressions, and body movements, users can express complex ideas, emotions, and social interactions. In contrast, finger spelling involves spelling out words letter by letter using hand shapes that correspond to the letters of a written alphabet, as can be seen in figure 1. Unlike conversational signs, finger spelling is less efficient for extended communication, as it is slower and more labour-intensive.

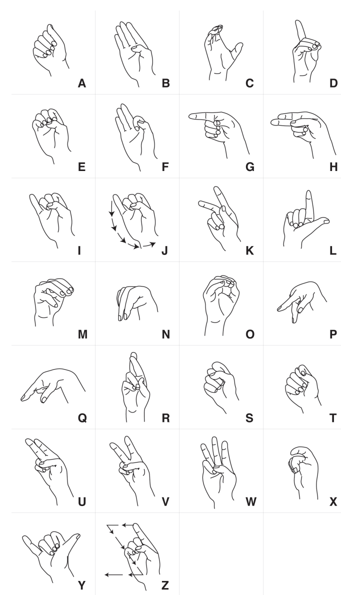


Figure 1. The Alphabetic Sign component of ASL, which was referred to when assigning finger orientation to characters during classification.
[7]

Due to certain limitations, such as project budget constraints and the complexity of implementing a full gestural-based ASL, the team has decided to initiate the project with a focus on developing the finger-spelling aspect of sign language.

B. Sensors and components

The objective of the project is to accurately classify all the alphabetic characters shown in Figure 1 using off-the-shelf components. As noted in section II, conventional approaches often require sensors that exceed the project budget. To address this challenge, an alternative approach inspired by the work in section II-A was considered as a replacement for the costly flex sensors. It involves the use of an LED and a photodiode, connected through a flexible tube. These sensors will be used to determine the shape of a finger, indicating whether it is straight, slightly bent, or fully bent. A photodiode functions by converting light energy (photons) to electrical current. When light with sufficient energy hits the photodiode, it excites electrons, creating electron-hole pairs within the diode's depletion region. An inherent electric field separates these pairs, resulting in a photocurrent [8]. A Light Emitting Diode (LED) produces light when an electric current flows through a semiconductor, specifically a p-n junction. The process is known as electroluminescence and occurs when electrons and holes recombine at the junction region, resulting in a release of energy in the form of photons [9]. The colour of the emitted light depends on the specific material that is used for the semiconductor.

It was determined that utilizing only a photodiode and an LED would not sufficiently interpret every alphabetic character presented in figure 1, as certain letters, such as "U" and "V", are very similar, differing primarily by the contact between the index and middle finger. To differentiate between such cases, a Hall effect sensor will be used. A Hall effect sensor works by detecting magnetic fields based on the principle of the Hall effect. When an electric current passes through a thin strip of conductive or semiconductive material inside the sensor, and a magnetic field is applied perpendicular to the direction of the current, a voltage, known as the Hall voltage, is generated across the material. The voltage generated is perpendicular to both the current and the magnetic field [10]. The Hall effect sensor will be used together with a magnet to determine one finger's position relative to another finger. Furthermore, certain letters are represented by similar finger formations, with the primary difference being the orientation of the entire hand. Additionally, letters such as "J" and "Z" require motions during their representation; a factor that cannot be captured by the current sensor setup.

To address these challenges, the integration of an accelerometer is needed, as it would provide the necessary motion detection capabilities that are required to improve the classification accuracy. An accelerometer measures the acceleration. It works by detecting the motion of a mass inside the sensor, which is then converted into an electrical signal that is proportional to the acceleration. The signal is then processed to accurately determine the direction and the magnitude of the acceleration [11]. To determine the orientation of the hand, a three-dimensional accelerometer is used in this project.

V. Method

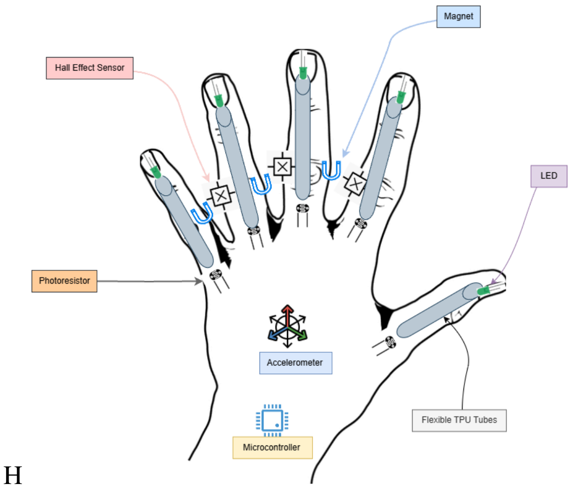
A. Hardware

A full list of the components used in this project is shown in table I.

The first step in creating the system was to create a very early prototype in order to test the components. For this, the group started by designing the flexible tubes in CAD for it to later be printed using a commercial 3D printer. The idea was to use TPU as filament because of its flexible properties, however, this also makes it challenging to print as a long tube. So instead, it was decided to print it in smaller parts and use paper straws to connect them. For the first assembly of the tubes, the connections were secured using heat shrink, but the stiffness of the heat shrink after they were heat-treated made it very hard to move the fingers. Instead, deeper fitting points were created where the straws would be inserted and relied on friction alone to keep the straws in place. This worked for the most part, except for when the straws started to deform and bend, leading to the joints disconnecting. For the photodiode and the LED, a cap was designed that the sensors were inserted into, with the cathode and anode pushed through the back of the cap, in an effort to limit the surrounding ambient light from reaching the photo resistor. CAD was used once again to create the fittings to hold the tubes, sensors, and magnets in place on the fingers. This part was constructed as a ring with a smaller ring on top of it to secure the tube on top of the finger. Two more slots were added, one on the left side of the finger for a Hall effect sensor to be attached, and another on the right side of the ring to hold the magnet. The components were tested again to ensure correct functionality, but unfortunately at this stage the accelerometer was connected incorrectly, resulting in shorting the board, rendering it useless. It was clear that this would affect the end result, limiting the letters the device could interpret, since the accelerometer was an important sensor for this project. All the remaining components were assembled and fitted onto the glove. Each finger containing an LED at the tip and a photodiode on the opposite side close to the knuckles, connected by the flexible tube. The middle, ring, and pinkie fingers have a Hall effect sensor facing the adjacent finger, while the index, middle, and ring fingers have a magnet directed towards the Hall effect sensor, as illustrated in Figure 2. The accelerometer was intended to be positioned on the back of the hand in a

Type	Unit	Quantity
Microcontroller	XIAO ESP32S3	1
Accelerometer	LIS3DH Triple-Axis Accelerometer	1
Light Sensor	Ambient Light Sensor 550 nm	5
Hall Effect Sensor	AH49FZ3-G1	3
LED	LED 574nm Green	5
Magnet	Magnet, Ferrite, 30N, 10 x 10mm	3
Gloves	ESD Protective Gloves	1
Resistor	NA	NA

Table I
BILL OF MATERIALS.



H

Figure 2. An early rough estimate of the component placement in reference to the hand.

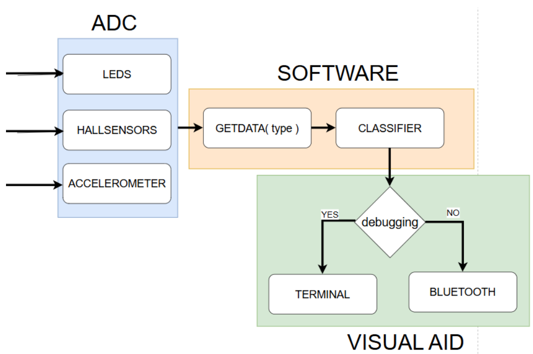


Figure 3. A flow chart of the system structure. There exist 4 threads. One for each ADC, which reads data in the blue locks. The last treads handle both the orange and green blocks. The orange block handles classifying data and the green block is the visual representation for the user, being in Bluetooth or terminal mode.

central location to accurately capture the overall orientation of the hand, with the microcontroller, an ESP32S3 [12], located beneath it, near the wrist. All the sensors were then connected to the microcontroller on a prototyping board.

B. Software

The system consist of 4 stages. The first stage consists of extracting the data from the gpio ports of the microcontroller, the second consists of calibration of the glove sensors, the third is the software responsible for how to extract a sign from raw data, and lastly, the fourth stage presents the resulting character.

The esp32s3[12] have a dual-core processor which supports multi-threading, which is used to read data while extracting information from the data flow. To have safe, non-corrupt data flow and no race conditions, FreeRTOS[13] was added to this project. FreeRTOS makes it easy to add queues, priority assignment and, most important, allows the use of a task manager.

1) Extracting data: Each sensor type (photo resistors, hall sensors and accelerometer) has each its own thread, queue and binary semaphore, where 100 data points are read each second. Before the data is sent to its specified queue, the tread takes a queue specific binary semaphore to stop all other access to that specific queue. Then, the data will be sent to that specific queue which the classifier can read from. After the data is sent, the semaphore is given back to the task manager. Each sensor type has its own queue, making it easy to separate them. This is represented in the blue square of figure 3.

To extract the data, which is to be sent to the classifier, from the different queues, a separate function called “GetData” was created. This function can be seen in the first sub-block in the orange square in figure 3. The function takes in a pointer to store the data in and also, which queue to take the data from. If the semaphore is free, it takes it and looks if the queue has data. If not, it gives the semaphore back and waits 10 CPU ticks. If there is data, the first data pointer is stored in the provided pointer. The function then clears the queue, gives back the semaphore and leaves the function.

2) Calibration: Every time the glove is used, and the system is initiated, a calibration step is needed in order for it to work as intended. The sensors can behave differently depending on things such as different users wearing the glove, how much light is present in the environment and circumstances involving the configuration of the hardware. The calibration consists of stages wherein the user is asked to pose their fingers in three basic poses; straight, slightly bent and fully bent. For each pose, n samples are taken of the photo resistor sensor values of each finger over t_s seconds. These samples are then used to calculate the mean μ and the standard deviation σ. The means and the standard deviations are calculated with equation 1 and equation 2 respectively,

$$\mu = \frac{\sum_{i=1}^{n} x(i)}{n} \tag{1}$$

$$\sigma^2 = \frac{\sum_{i=1}^{n} \mu - x(i)}{n-1} \tag{2}$$

where $x(i)$ is sample # i. This is done two times for all fingers in each pose, and then the mean of the two values are given as parameters to the classification step. Put more simply, samples are taken of each photo resistor when the fingers are configured in three different poses, and the means and standard deviations are calculated and saved for classification.

3) Classification: To classify, two different methods were created with two different approaches. But only one of them can be implemented at a time. The first method is a binary one, where each sensor provides a discrete case of which character state it can be in. This philosophy can be seen in image 4. For example, if the photo resistor placed on the pointer finger gives a result of “I am straight up”, then it can not be an A because then it should say “I am straight down” as can be seen in figure 1. When this method provides a character, the system is very certain that it’s the correct character, because all sensor needs to classify that specific character. The main drawback is that each sensor is dependent on each other. Meaning, if one sensor dies, the system will provide no result. Or if one of the sensor gives false readings, then the system will also provide no results. This is where the idea of the second method comes in.

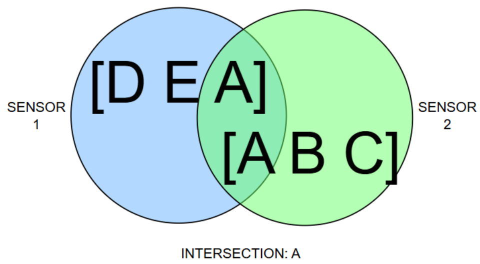


Figure 4. Visual representation of a binary approach in finding the intersecting character.Where the blue is result from sensor 1 and green is result from sensor 2

The second method provides a probability of being right. Where it will, instead of saying "I am straight up", provide "I can guarantee with 80% certainty that I am straight up". This is done by creating a normal distribution of each state each sensor can be in from the provided mean and standard deviation extracted from the calibration part. Then, each character in the alphabet is weighed depending on the total probability of it being true, as seen in figure 5. Because it is hard to extract the area under the normal distribution the z-score[14] and its representative table was used, since it's both simpler to implement and easier for the esp to calculate, reducing CPU power.

It is worth noting that the current weight is proportional to the probability in each state. Meaning that the total sum of all weights is greater than 1. Also, Bayer's theorem [15] would be optimal to implement but was never implemented, this limitation was because of time constraints.

4) Visual representation: For the user to know what character the system is interpreting from their hand sign, the final character needs to be displayed as a visual representation. This is done with the esp32s3 Bluetooth system. By sending the expected character via Bluetooth, it is possible for the user to see the character using a smart-phone. When the system is connected to a computer, more information can be presented with the use of the terminal. Everything from the probability of characters in the alphabet being true, to computation time and raw data can be displayed. If the system is in approach 1 (binary approach) the sensor states can be extracted (are they saying up, down) and it's corresponding intersection with the other sensors.

C. Validation and test method

To test the system total computation time, the FreeRTOS function "xTaskGetTickCount()" was used in the testing phase. This is done by taking the time after calibration, and taking the time after the character is presented and comparing the two times. If the time is too small, a mean value over multiple iteration will be needed.

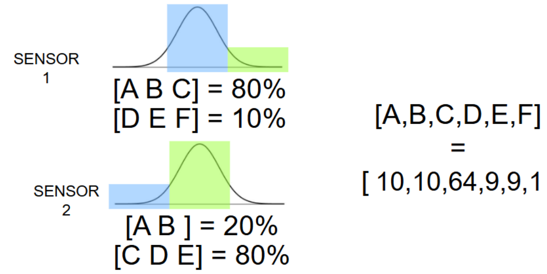


Figure 5. Visual representation of a percentage based approach in finding the most likely character. Where the top normal distribution is the result from sensor 1, where the blue part is the likelihood of the character being A, B or C. The green area is the likelihood of it being D, E or F. The second normal distribution, where the blue part is the likelihood of the character being A, B. The green area is the likelihood of it being C, D or E. The value to the right is the resulting weights where [A,B,C,D,E,F] have weights of [10,10,64,9,9,1]

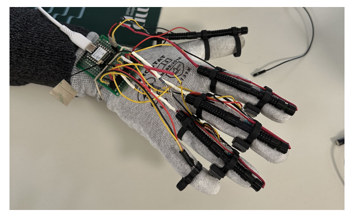


Figure 6. The glove, with all the parts and sensors assembled

More importantly, to test the accuracy of the system, two tests will be implemented. The first is trying to get a letter, with all the information the terminal can give. This is done in order to see if the system is able to discern between all the characters. Then for the second test, the program will go through all the letters and ask the user to create a specific letter with no information from the terminal. The users will be each member of the project group and an outsider which has no prior experience or familiarity with the system.

VI. Results

The mean execution time over 10 runs was 10 580 milliseconds per 1000 runs, or 10.6 millisecond per execution.

The binary method can be seen in table II. Where it can not find C,G,N,P,R,T,X or Z.

Where the probability based method seen in table III could not see N or T, and had a hard time finding unique solutions.

The final assembly of the hardware is shown in Figure 6.

VII. Discussion

The computation times mentioned in the results were calculated by running the system over 1000 iterations and taking the mean value. This was done because the system was too fast to only use pdTICKS_TO_MS(start - end) of a single iteration.

Some hand signs were not possible to perform because the configuration of the glove was rather bulky. Characters such as N and T were not possible to sign due to this fact. The disconnecting of the joints and deformation of the material

true	classified
A	A
B	B
C	
D	GLQ
E	E M S
F	F
G	
H	H
I	I J
J	I J
K	K P R
L	G L Q
M	E M S
N	
O	E M S
P	
Q	G L Q
R	
S	E M S
T	
U	U
V	V
W	W
X	
Y	Y
Z	

Table II
BINARY TABLE WITH TERMINAL AS HELP

true	classified
A	A
B	B
C	C
D	D
E	E M S
F	F
G	G L Q
H	H
I	I J
J	I J
K	K P R V
L	G L Q
M	E M S
N	
O	O
P	K P R V
Q	G L Q
R	K P R V
S	E M S
T	
U	U
V	K P R V
W	W
X	X
Y	Y
Z	Z

Table III
PROBABILITY TABLE WITH TERMINAL AS HELP

used led to the deterioration of the glove, and further validation of the system suffered as a consequence.

The probabilities calculated in method 2 of the classification part does not represent the true probabilities, since their sum add up to over 100%, but are still able to be used as a measure to determine the most likely character. The system as a whole was complete with close to no time left, and a lot of fine-tuning and polish was not carried out. Also, due to time constraints, the validation of the system was not explored thoroughly enough since more testing is needed in order to determine the accuracy and usability of the system. Another problem encountered was the fact that the accelerometer that was planned to be implemented in the system broke, and as a result, overall hand orientations were not used in order to determine hand signs.

VIII. CONCLUSION

This paper presented the development of a system that is able to discern multiple different alphabetic sign language characters given a limited budget and a limited development timeframe. The resulting system uses two different types of sensors, the photo resistor in combination with LEDs and the hall sensors in combination with magnets, in order to determine finger orientations. The project was deemed good enough as a proof of concept prototype and shows that the photo resistor and the LED is a possible replacement of flex sensors in order to estimate the bend of the fingers. This substitution is much cheaper and works well after some configuration and calibration. Due to the accelerometer burning up, it was not possible to discern hand orientation fully, and it was not possible to implement gesture recognition.

IX. FUTURE WORK

To improve the system in the future, multiple functions can be implemented. For instance, adding an accelerometer or gyroscope would make it possible to discern hand gestures and orientation. This would make it possible to implement more characters or words corresponding to different gestures. The system can also be improved by adding a writing functionality, wherein the user is able to save signed characters and write words. This can be combined with a word suggestion generator in order to create a faster and more efficient spelling system. There is also a lot of improvement to be done with the glove design. In creating a more flexible and compact design, the user would be more comfortable, and it would be easier to perform the different hand signs.

ACKNOWLEDGMENT

The students in this project would like to thank MDU for providing us with the funding in creating this project.

REFERENCES

[1] H. Pishro-Nik, "1.4.3 bayes' rule," August 24, 2014. [Online]. Available: https://www.probabilitycourse.com/chapter1/1_4_3_bayes_rule.php

[2] S. M. Antad, S. Chakrabarty, S. Bhat, S. Bisen, and S. Jain, "Sign language translation across multiple languages," in *ESIC 2024 - 4th International Conference on Emerging Systems and Intelligent Computing, Proceedings*. Institute of Electrical and Electronics Engineers Inc., 2024, pp. 741–746.

[3] J.-H. Kim, N. D. Thang, and T.-S. Kim, "3-d hand motion tracking and gesture recognition using a data glove," in *2009 IEEE International Symposium on Industrial Electronics*, 2009, pp. 1013–1018.

[4] N. Adaloglou, T. Chatzis, I. Papastratis, A. Stergioulas, G. T. Papadopoulos, V. Zacharopoulou, G. J. Xydopoulos, K. Atzakas, D. Papazachariou, and P. Daras, "A comprehensive study on deep learning-based methods for sign language recognition," 7 2020, better report then the indian verion. [Online]. Available: http://arxiv.org/abs/2007.12530http://dx.doi.org/10.1109/TMM.2021.3070438

[5] W. F. of the Deaf, "Our work," 2025. [Online]. Available: https://wfdeaf.org/our-work/
[6] K. Yin, T. Regier, and D. Klein, "American sign language handshapes reflect pressures for communicative efficiency," *arXiv preprint arXiv:2406.04024*, 2024.
[7] signsanddisplays, "American sign language alphabet," 2025. [Online]. Available: https://signsanddisplays.wordpress.com/2011/03/15/american-sign-language-alphabet-free-vectors/
[8] K. Murari, R. Etienne-Cummings, N. Thakor, and G. Cauwenberghs, "Which photodiode to use: A comparison of cmos-compatible structures," *IEEE sensors journal*, vol. 9, no. 7, pp. 752–760, 2009.
[9] C. M. Bourget, "An introduction to light-emitting diodes," *HortScience*, vol. 43, no. 7, pp. 1944–1946, 2008.
[10] M.-A. Paun, J.-M. Sallese, and M. Kayal, "Hall effect sensors design, integration and behavior analysis," *Journal of Sensor and Actuator Networks*, vol. 2, no. 1, pp. 85–97, 2013.
[11] I. A. Faisal, T. W. Purboyo, and A. S. R. Ansori, "A review of accelerometer sensor and gyroscope sensor in imu sensors on motion capture," *J. Eng. Appl. Sci*, vol. 15, no. 3, pp. 826–829, 2019.
[12] Spencer, "Getting started with seeed studio xiao esp32s3 series," Aug 14, 2024. [Online]. Available: https://wiki.seeedstudio.com/xiao_esp32s3_getting_started/
[13] freertos, "Real-time operating system for microcontrollers and small microprocessors," 2025. [Online]. Available: https://www.freertos.org
[14] H. Pishro-Nik, "4.2.3 normal (gaussian) distribution," August 24, 2014. [Online]. Available: https://www.probabilitycourse.com/chapter4/4_2_3_normal.php
[15] G. Payne, "Optical glove controller using ldr," 2023. [Online]. Available: https://hackaday.io/project/189165-optical-glove-controller-using-ldr