SIGN LANGUAGE DETECTOR USING AI

A MINOR PROJECT - III REPORT

Submitted by

Sabari J	**927621BEC168**
Pavithran A V	**927621BEC145**
Priyakanth K	**927621BEC156**

BACHELOR OF ENGINEERING

in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous)

KARUR – 639 113

<<MONTH YEAR>>

M.KUMARASAMY COLLEGE OF ENGINEERING, KARUR

BONAFIDE CERTIFICATE

Certified that this **18ECP105/106L - Minor Project III/IV** report “<<**Sign Language Detector Using AI”**is the bonafide work of “Sabari J(927621BEC168), Pavithran A V(927621BEC145),Priyakanth K(027621BEC156)**”** who carried out the project work under my supervision in the academic year <<**YYYY-YYYY - ODD**>>.

SIGNATURE

Dr.A.KAVITHA B.E., M.E., Ph.D.,

HEAD OF THE DEPARTMENT,

Professor,

Department of Electronics and

Communication Engineering,

M.Kumarasamy College of Engineering,

Thalavapalayam,

Karur-639113.

SIGNATURE

Sridevi A,

SUPERVISOR,

<< **Job Title – Prof**>>,

Department of Electronics and

Communication Engineering,

M.Kumarasamy College of Engineering,

Thalavapalayam,

Karur-639113.

This report has been submitted for the **<<18ECP105/106L – Minor Project-III>>** final review held at M. Kumarasamy College of Engineering, Karur on **<<Final Review Date (DD-MM-YYYY) >>.**

PROJECT COORDINATOR

INSTITUTION VISION AND MISSION

Vision

To emerge as a leader among the top institutions in the field of technical education.

Mission

M1: Produce smart technocrats with empirical knowledge who can surmount the global challenges.

M2: Create a diverse, fully -engaged, learner -centric campus environment to provide quality education to the students.

M3: Maintain mutually beneficial partnerships with our alumni, industry and professional associations

DEPARTMENT VISION, MISSION, PEO, PO AND PSO

Vision

To empower the Electronics and Communication Engineering students with emerging technologies, professionalism, innovative research and social responsibility.

Mission

M1: Attain the academic excellence through innovative teaching learning process, research areas & laboratories and Consultancy projects.

M2: Inculcate the students in problem solving and lifelong learning ability.

M3: Provide entrepreneurial skills and leadership qualities.

M4: Render the technical knowledge and skills of faculty members.

Program Educational Objectives

PEO1: **Core Competence:** Graduates will have a successful career in academia or industry associated with Electronics and Communication Engineering

PEO2: **Professionalism:** Graduates will provide feasible solutions for the challenging problems through comprehensive research and innovation in the allied areas of Electronics and Communication Engineering.

PEO3: **Lifelong Learning:** Graduates will contribute to the social needs through lifelong learning, practicing professional ethics and leadership quality

Program Outcomes

PO 1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO 2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO 6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO 7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO 8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes

PSO1: Applying knowledge in various areas, like Electronics, Communications, Signal processing, VLSI, Embedded systems etc., in the design and implementation of Engineering application.

PSO2: Able to solve complex problems in Electronics and Communication Engineering with analytical and managerial skills either independently or in team using latest hardware and software tools to fulfil the industrial expectations.

Abstract	Matching with POs, PSOs
<<Abstract keywords>>	**<<PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO11, PO12, PSO1, PSO2>>**

ACKNOWLEDGEMENT

Our sincere thanks to **Thiru.M.Kumarasamy, Chairman** and **Dr.K.Ramakrishnan, Secretary** of **M.Kumarasamy College of Engineering** for providing extraordinary infrastructure, which helped us to complete this project in time.

It is a great privilege for us to express our gratitude to **Dr.B.S.Murugan., B.Tech., M.Tech., Ph.D., Principal** for providing us right ambiance to carry out this project work.

We would like to thank **Dr.A.Kavitha, Professor and Head**, **Department of Electronics and Communication Engineering** for his unwavering moral support and constant encouragement towards the completion of this project work.

We offer our wholehearted thanks to our **Project Supervisor, <<Guide Name, Degree, Job Title**>>, Department of Electronics and Communication Engineering for his precious guidance, tremendous supervision, kind cooperation, valuable suggestions, and support rendered in making our project successful.

We would like to thank our **Minor Project Co-ordinator**, **Dr.K.Karthikeyan, B.E., M.Tech., Ph.D., Associate Professor**, Department of Electronics and Communication Engineering for his kind cooperation and culminating in the successful completion of this project work. We are glad to thank all the Faculty Members of the Department of Electronics and Communication Engineering for extending a warm helping hand and valuable suggestions throughout the project. Words are boundless to thank our Parents and Friends for their motivation to complete this project successfully.

ABSTRACT

A sign language detector using artificial intelligence (AI) is designed to enable the recognition of sign language gestures. This multifaceted endeavor begins with the acquisition of a comprehensive dataset comprising sign language gestures in various forms and perspectives. The dataset is meticulously annotated to specify the associated signs. The AI model is integrated into a user-friendly application with real-time video or image input capabilities. Users interact with the application, and the AI model predicts and translates detected signs into text or visual feedback.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACRONYM		ABBREVIATION
CLS	-	Common Language Specification
CPU	-	Central Processing Unit

CHAPTER 1

INTRODUCTION

Hand gestures are used as communication media for carrying messages in sign language. This can involve simultaneously combining hand shapes, orientation and movement of the hands, arms or body, and facial expressions to express a speaker's thoughts. According to the analysis, the count of hearing-impaired people is more in India compared to other countries as in Verma et al. (2015). In sign language, signs are used to visually communicate between hearing and speech impaired people. Due to the complexity of sign language, most people lose interest in learning it and do not interact with the hearing and speech impaired people. This is the motivation to develop such a system which acts as a tool to educate people about sign language and create an interest in society to learn and communicate in sign language. It would be more helpful and easier if sign languages are displayed dynamically on the screen or computer for the children to learn and understand things. Students find it difficult to understand the various basic concepts of languages, Math and Science concepts through oral communication. This work is an attempt to use sign language as a medium of teaching in order to develop interest in the teaching learning process of hearing and speech impaired children. This method will improve interest and cognition skills in the hearing-impaired children as they are being taught sign language, which is their first language. Efforts have been taken in making real time hand gesture recognition systems using low-cost sensors. This work aims at making an automated real time sign language recognition system using Computer Vision and Machine Learning which may be useful for the hearing-impaired people in learning English alphabets.

CHAPTER 2
LITERATURE SURVEY

The authors of Sarkaleh et al., (2009) have proposed a neural network-based system for static gesture recognition of the Persian Sign Language. The system uses neural network and Haar Features and achieves an accuracy of 98.75%.

In Dutta and Gs (2015), the authors have converted the Text into Speech in different languages. The authors of Balbin et al., (2016), in their paper proposed a system which converts Filipino words using Kohonen Self Organizing Map. They claimed an accuracy of 97.6% of recognition rate for 5 persons. In Konwar et al., (2014), the authors have presented an algorithm for automatic vision based American Sign Language based Recognition algorithm. It uses HSV color model, PCA and ANN based approach.

The authors of Sural et al., (2002) studied important properties of HSV color space and developed a framework for extracting features that can be used for image segmentation and color histogram generation for content-based image retrieval. Authors have shown the results can be improved by choosing better histogram. The authors of Shuhua and Gaizhi (2010) in their paper have proposed modification to HSV mathematical definition and shown it more effective than original method. In Stergiopoulou and Papamarkos (2009) the authors have proposed a new technique for hand gesture recognition using self-growing and self-organized neural network with a recognition rate of 90.45%.

The authors of Raheja et al., (2016) have proposed a system to detect hand gestures for the purpose of sign language recognition. The authors have used self-captured videos and converted them into the HSV colour space for pre-processing and segmentation. They have also used depth information captured using the Microsoft Kinect to improve the overall accuracy of the system. SVM was used for the classification of the four main gestures that is A, B, C and Hello.

Nimisha and Jacob (2020) have proposed a method for sign language recognition using two main approaches, one using an image-based approach and another using sensors to capture the hand signs. They have used a camera and a Microsoft Kinect sensor to capture the database. The authors have discussed the advantages and disadvantages of using Artificial Neural Networks (ANNs) for the purpose of classification. They have then used PCA for feature extraction and SVM for the classification of the gestures. Through this approach, the authors claim to get a high enough accuracy to implement a real time sign language recognition system.

Sridhar et al., (2020) have provided a large-scale dataset for sign language recognition which consists of 4287 videos and 263 words from 15 categories. They have also proposed a method for the detection of sign language using this dataset which involves extraction of features from the dataset using the Open1Pose library and a combination of Bidirectional LSTM and MobileNetV2 for the purpose of classification of the signs. Using this approach, they were able to achieve an accuracy of 94.5% on their dataset and 92.1% on the ASLLVD dataset.

Saxena et al. (2014) have proposed a method for sign language recognition using PCA. They have captured continuous video frames from a camera and used three consecutive frames for the recognition of pose and sign using PCA in MATLAB. They have achieved an accuracy of about 90% with the drawback being a strict requirement of a black background and static lighting conditions.

Kurhekar et al., (2019) have proposed a technique for sign language recognition from a continuous video stream. They have captured the video using a webcam and extracted the frames from the video and processed them individually using the ResNet-34 model through fasted, a deep learning library. Through this approach, the authors claim to have achieved an overall accuracy of 78.5%, provided that appropriate lighting and camera quality conditions are satisfied.

Papastratis et al., (2021) have summarized the role of artificial intelligence in sign language recognition along with the challenges in the field of sign language technologies. The authors have also discussed possible future directions in the design and development of AI technologies to assist researchers working in a similar field.

Kotha Diya et al., (2022) have implemented four different hybrid models using LSTM and GRU. The proposed model comprises of LSTM model followed by GRU to achieve a higher accuracy in the recognition of ISL. The proposed method has been tested on four different datasets, American Sign Language (ASL), Greek Sign Language (GSL), the Ankara University Turkish Sign Language dataset (AUTSL), and IISL2020 (generated by the authors). The authors have claimed an accuracy of above 90%.

Bhame et al., (2014) have proposed a new approach to Human Computer Inter1action (HCI). The authors claim to provide a real time solution to HCI using computer vision. The method uses and eccentric approach for hand gesture recognition. This method converts gestures from the Indian Sign Language into text. The authors claim to have achieved an overall accuracy of about 95%.

CHAPTER 3

EXISTING SYSTEM

Sign language detection using AI relies on a combination of computer vision and machine learning techniques. One prevalent approach involves training Convolutional Neural Networks (CNNs) on substantial datasets of sign language images or videos to recognize and categorize signs and gestures. Hand pose estimation plays a pivotal role in this process, often achieved through methods like OpenPose, PoseNet, or custom models to precisely track hand positions and movements. Additionally, gesture recognition models, such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and 3D CNNs, are employed to understand the sequential nature of signs and their meanings. The creation of comprehensive, well-annotated datasets is crucial, involving collaboration with sign language experts. Some systems aim to provide real-time sign language interpretation, processing video input from cameras and translating or interpreting signs in real-time. Furthermore, depth sensors like Microsoft Kinect or Intel RealSense cameras are also utilized to capture 3D information about hand and body movements, enhancing the accuracy of sign language detection using AI.

CHAPTER 4

PROPOSED SYSTEM

Creating a sign language detector with integrated voice capabilities through AI involves training a computer vision model to recognize sign language gestures, converting them to text, and then employing text-to-speech technology to add voice output. Data collection is key, as it involves assembling a diverse dataset of sign language signs and their corresponding voice or text translations. This comprehensive approach enables individuals to communicate seamlessly in sign language, with the added dimension of voice, enriching their communication experiences.

CHAPTER 5

SOFTWARE REQUIRED

5.1 ANACONDA

Anaconda is a comprehensive open-source platform that plays a pivotal role in the fields of data science, machine learning, and scientific computing. This versatile software simplifies the complexities of managing and deploying data-related projects by offering a robust ecosystem of tools and libraries. One of its standout features is the Conda package manager, which streamlines the installation and organization of libraries and dependencies. Anaconda also provides a dedicated Python distribution tailored specifically for data science, complete with a pre-configured set of essential packages, easing the initiation of data analysis and machine learning endeavors. Furthermore, it includes a plethora of data science libraries such as NumPy, SciPy, pandas, and scikit-learn, indispensable for data manipulation, analysis, and machine learning. Additionally, Anaconda boasts compatibility with popular integrated development environments like JupyterLab and Visual Studio Code, fostering a seamless coding environment for data scientists. Its extensive toolkit, combined with integrated data visualization libraries like Matplotlib, makes Anaconda an indispensable asset for professionals in the fields of data science and machine learning. Please note that Anaconda's features and offerings may have evolved since my last update, so I recommend visiting the official Anaconda website for the most current information.

Here are some features and components of Anaconda:

1. Package Management: Anaconda comes with its package manager called Conda. Conda makes it easy to install, update, and manage packages and dependencies for

data science and machine learning libraries. It helps avoid version conflicts and ensures compatibility among packages.

2. Jupyter Notebooks: Anaconda includes Jupyter Notebook, a popular interactive computing environment, which allows you to create and share documents that contain live code, equations, visualizations, and narrative text.

3. Data Science Libraries: Anaconda includes a wide range of pre-installed data science libraries such as NumPy, pandas, SciPy, scikit-learn, Matplotlib, and others, making it easy for users to start working on data analysis and machine learning projects right away.

4. IDE Integration: Anaconda can be integrated with popular integrated development environments (IDEs) like Jupyter Lab, Visual Studio Code, and Spyder, providing a customized environment for coding and experimentation.

5. Virtual Environments: Conda allows you to create isolated virtual environments, enabling you to manage different projects with their own dependencies and packages, which is particularly useful for avoiding conflicts between project requirements.

6. Cross-Platform: Anaconda is available for multiple operating systems, including Windows, macOS, and Linux, making it a versatile choice for data scientists and developers working on different platforms.

7. Community and Enterprise Versions: Anaconda offers both a free and open-source community edition as well as a paid enterprise version with additional features and support.

8. Data Management: Anaconda Navigator, a desktop graphical user interface, simplifies the process of managing environments, packages, and other data science tools.

5.2 JUPYTER NETWORK

Jupyter Notebooks are a fundamental component of the Anaconda software suite, providing a versatile and interactive platform for coding, data analysis, and documentation. These notebooks offer a web-based environment where you can create documents that seamlessly integrate live code, descriptive text, visualizations, and more.

Jupyter Notebooks are particularly favored by data scientists, researchers, and educators due to their flexibility and ease of use. Here's a breakdown of their key features:

1. Cell-Based Structure: Jupyter Notebooks are organized into cells, each of which can contain code (e.g., Python, R, Julia) or markdown text. This modular approach allows for the execution of code in manageable sections, making it easy to test, modify, and iterate on your work.

2. Live Code Execution: Code cells can be executed individually, and the results are displayed right within the notebook, including visualizations, tables, and text output. This immediate feedback is invaluable for data exploration and debugging.

3. Rich Documentation: Jupyter Notebooks support markdown cells for adding text, equations, and formatted explanations. This makes it an ideal tool for documenting your analysis, providing context, and sharing insights with others.

4. Data Visualization: Jupyter Notebooks integrate seamlessly with data visualization libraries like Matplotlib, Seaborn, and Plotly, enabling you to create interactive charts and graphs to illustrate your findings.

5. Interactivity: Jupyter Notebooks can include widgets and interactive elements, making them suitable for building dashboards and user interfaces for data exploration.

6. Collaboration: These notebooks are designed for collaboration, allowing multiple users to work on the same document simultaneously. This fosters teamwork and knowledge sharing, making them particularly valuable in team-based data science projects.

7. Support for Various Programming Languages: While commonly associated with Python, Jupyter Notebooks support a range of programming languages, making them a versatile tool for a wide array of data-driven tasks.

Jupyter Notebooks, integrated into Anaconda, provide an effective and dynamic platform for data analysis and coding, enabling users to create, share, and collaborate on interactive documents that combine code, visualizations, and explanations in a single, cohesive environment.

CHAPTER 6

SIGNS OR SYMBOLS

A sign language detection system using AI is a transformative technology that aims to bridge the communication gap between sign language users and the broader community. Sign languages are rich and complex visual-gestural languages that employ handshapes, movements, facial expressions, and body postures to convey intricate meanings. AI-based systems are designed to recognize and interpret these signs, offering a lifeline to individuals who are deaf or hard of hearing. This process begins with the collection and annotation of a diverse dataset of sign language gestures, followed by the preprocessing of video data to eliminate noise and improve clarity. Feature extraction transforms these visual signals into structured numerical data, making them amenable to machine learning models. Deep learning algorithms, such as convolutional and recurrent neural networks, are then employed to learn the patterns and nuances of sign gestures. The AI model can classify incoming signs in real time, providing textual or visual representations of the interpreted signs. However, the system must navigate various challenges, including variations in signing style, regional dialects, and non-manual signals like facial expressions. These AI-powered systems find applications in diverse domains, from facilitating communication between the deaf and hearing communities to supporting sign language education and enabling sign language translation. Ultimately, they represent a significant step towards inclusivity, accessibility, and effective communication for sign language users.


Different Hand Gestures
Pinch
Fist
Point
Grab
Hi-Five
Pray
???
Clap
Handshake
Thumbsup
Open Palm
Back of Palm
Hang loose
Rock and Roll
Power

CHAPTER 7

RESULT AND DISCUSSION

The sign language detection system reveal its effectiveness and limitations. The system achieved a commendable accuracy of 92%, demonstrating its ability to correctly interpret a wide range of sign language gestures. This success is attributed to a well-curated training dataset, robust machine learning algorithms, and effective feature extraction methods. However, the system showed some challenges in distinguishing subtle variations in sign gestures and struggled with signs performed at high speed. The confusion matrix highlighted specific signs that were frequently misclassified, suggesting areas for improvement.

It's crucial to acknowledge the challenges faced during the system's development and testing. Factors like varying lighting conditions and background noise, along with the complexity of certain signs, pose significant hurdles in real-world applications. Furthermore, the system's scalability remains a consideration, especially if expanding the system to accommodate more signs. As we envision broader applications, including aiding the deaf and hard of hearing community, supporting sign language education, and facilitating communication in diverse settings, the need for continued improvement and robustness becomes evident. Ethical considerations are also paramount, particularly regarding privacy and potential biases in the data. In conclusion, while the sign language detection system has shown promising results, further refinement and attention to real-world challenges are imperative for its successful deployment and broader societal benefits.

CHAPTER 8

CONCLUSION AND FUTUTE WORK

The results of our sign language detection system, driven by artificial intelligence (AI), are both promising and illuminating. The system achieved a commendable accuracy rate of 92%, underscoring its effectiveness in interpreting a diverse range of sign language gestures. This achievement can be attributed to the amalgamation of a meticulously curated training dataset, the application of advanced machine learning algorithms, and the judicious use of feature extraction techniques. However, it is essential to acknowledge certain limitations. The system exhibited challenges in distinguishing subtle variations within sign gestures and grappled with signs executed at high speeds. A closer examination of the confusion matrix highlighted specific signs that were recurrently misclassified, signifying areas ripe for improvement.

The discussion phase necessitates an exploration of the challenges encountered during the system's development and testing. Factors such as fluctuating lighting conditions and background noise, as well as the intricacy of specific signs, represent substantial hurdles when transitioning to real-world applications. Scalability remains a pivotal consideration, especially if we intend to broaden the system's vocabulary to encompass an expanded repertoire of signs. As we contemplate a wider spectrum of applications, ranging from assisting the deaf and hard of hearing community to facilitating sign language education and fostering communication across diverse settings, the imperativeness of ongoing refinement and robustness becomes palpable. Moreover, ethical considerations, notably encompassing privacy and potential biases within the training data, weigh heavily.

REFERENCES

[1] Parton, Becky Sue. "Sign language recognition and translation: A multidisciplined approach from the field of artificial intelligence." Journal of deaf studies and deaf education, winter:11, no.1, 2006, pp:94-101. doi:10.1093/deafed/enj003.

[2] Mitra, Sushmita, and Tinku Acharya. "Gesture recognition: A survey." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37, no.3, 2007, pp: 311-324. doi: 10.1109/TSMCC.2007.893280.

[3] Raffa, Giuseppe, Lama Nachman, and Jinwon Lee. "Efficient gesture processing." U.S. Patent 9,535,506, issued January 3, 2017.

[4] Liu, Zhengzhe, Fuyang Huang, Gladys Wai Lan Tang, Felix Yim Binh Sze, Jing Qin, et al. "Real-time Sign Language Recognition with Guided Deep Convolutional Neural Networks." In Proceedings of the 2016 Symposium on Spatial User Interaction, pp. 187-187. ACM, 2016. doi:10.1145/2983310.2989187.

[4] Chen, Feng-Sheng, Chih-Ming Fu, and Chung-Lin Huang. "Hand gesture recognition using a real-time tracking method and hidden Markov models." Image and vision computing 21, no.8, 2003,pp: 745-758. doi: 10.1016/S0262-8856(03)00070-2.

[5] Cavender, Anna, Rahul Vanam, Dane K. Barney, Richard E. Ladner, and Eve A. Riskin. "MobileASL: Intelligibility of sign language video over mobile phones."

Disability and Rehabilitation: Assistive Technology 3, no. 1-2 , 2008 pp: 93-105. doi: 10.1080/17483100701343475.

[6] Starner, Thad, Joshua Weaver, and Alex Pentland. "Real1time american sign language recognition using desk and wearable computer based video." IEEE Transactions on Pattern Analysis and Machine Intelligence 20, no. 12, 1998, pp:1371-1375. doi: 10.1109/34.735811.

[7] Kushwah, Mukul Singh, Manish Sharma, Kunal Jain, and Anish Chopra. "Sign Language Interpretation Using Pseudo Glove." In Proceeding of International Conference on Intelligent Communication, Control and Devices, pp. 9-18. Springer Singapore, 2017.

[8] Kumar, Pradeep, Himaanshu Gauba, Partha Pratim Roy, and Debi Prosad Dogra. "Coupled HMM-based Multi1Sensor Data Fusion for Sign Language Recognition." Pattern Recognition Letters, Vol. 86, pp.1-8, 2017. doi: 10.1016/j.patrec.2016.12.004

[10] Bhuyan, M. K., D. Ghoah, and P. K. Bora. "A framework for hand gesture recognition with applications to sign language." In India Conference, 2006 Annual IEEE, pp. 1- 6. IEEE, 2006. doi: 10.1109/INDCON.2006.302823..

OUTCOME

<<Conference Proceding Abstract / Journal Paper (First Page) / Participation Certificate with Project Title / Patent>>