

ANIMED-AI: AGENTIC RAG-DRIVEN REAL-TIME ANIMATION INTELLIGENCE SYSTEM

Aqsa Faizan

Artificial Intelligence And Machine Learning
C BYREGOWDA INSTITUTE OF TECHNOLOGY
KOLAR, KARNATAKA, INDIA
aqsakhan6361@gmail.com

Azmi Huda Khanum H

Artificial Intelligence And Machine Learning
C BYREGOWDA INSTITUTE OF TECHNOLOGY
KOLAR, KARNATAKA, INDIA
azmihudakhanum2004@gmail.com

Faiza Fathima

Artificial Intelligence And Machine Learning
C BYREGOWDA INSTITUTE OF TECHNOLOGY
KOLAR, KARNATAKA, INDIA
ff06ab@gmail.com

Madeeha Fathima

Artificial Intelligence And Machine Learning
C BYREGOWDA INSTITUTE OF TECHNOLOGY
KOLAR, KARNATAKA, INDIA
sultanazeenath1@gmail.com

Dr Deepika Lokesh

Artificial Intelligence And Machine Learning
C BYREGOWDA INSTITUTE OF TECHNOLOGY
KOLAR, KARNATAKA, INDIA
drdeepikalokesh@gmail.com

Abstract: Existing e-learning platforms generally provide the same static content to all students without considering their knowledge level, emotional state, or learning pace. This often leads to lower engagement and uneven learning outcomes. This paper presents ANIMEd-AI - an adaptive learning system based on Retrieval-Augmented Generation (RAG) that generates personalized educational animations in real time for each student query. When a student enters a concept or code, the system processes the input, retrieves relevant information from a ChromaDB vector store using LlamaIndex, and uses an AI model to generate a Manim animation script. This script is then rendered into a new MP4 video that visually explains the concept or shows step-by-step code execution. A DeepFace-based webcam module analyzes the student's facial expressions at regular intervals and maps the detected emotion to changes in content difficulty. An Arduino Uno with RGB LEDs provides simple real-time visual feedback based on the detected emotional state. The system also includes accessibility features: visually impaired students are supported through OpenAI TTS-based audio narration, hearing-impaired students through MediaPipe-based sign language display, and specially-abled learners through a simplified adaptive interface. Evaluation shows that RAG-based grounding improves factual accuracy, emotion-aware adaptation supports better learning performance, and animation-based explanations are easier to understand compared to text responses. ANIMEd-AI

combines animation-based learning, code visualization, and adaptive interaction within a single educational platform.

Keywords: Adaptive Learning, Retrieval-Augmented Generation (RAG), Animation-Based Learning, Code Visualization, Emotion Detection, AI in Education, Inclusive Learning

I. INTRODUCTION

The rapid growth of digital education platforms has increased access to learning, but it has not ensured personalization. Students with different cognitive abilities, knowledge levels, and emotional conditions are still provided with the same pre-recorded content, delivered at a fixed pace. This uniform approach does not align with established educational research, which indicates that adaptive and personalized learning methods perform better than standard content delivery. Recent progress in large language models (LLMs), multi-agent AI systems, programmatic animation tools, and affective computing has enabled the development of a new generation of intelligent tutoring systems. These systems can generate customized visual content, adjusting their responses based on student emotions, and producing explanations grounded in verified information to reduce inaccuracies. However, existing solutions have not successfully combined all these capabilities within a single systematized-AI addresses this limitation by integrating five key components into one platform: (1) RAG-based knowledge retrieval to improve factual reliability, (2) Lang Graph-based

multi-agent orchestration for adaptive decision-making, (3) AI-driven Manim animation generation that creates new educational videos for each query, (4) real-time emotion detection using DeepFace with adaptive content adjustment, and (5) unified accessibility support for blind, deaf, and specially-abled learners. In addition, an Arduino Uno with RGB LEDs is used to provide real-time visual feedback based on the detected emotional state. The remainder of this paper is organized as follows. Section II discusses related work. Section III presents the system architecture. Section IV explains the methodology. Section V describes the evaluation results. Section VI concludes the paper and outlines future work.

This template provides authors with most of the formatting specifications needed for preparing electronic versions of their papers.

II. RELATED WORK

Recent advancements in artificial intelligence have led to significant improvements in educational technologies, particularly in areas such as personalization, adaptive learning, and content generation. Several studies have explored these aspects individually, but a fully integrated solution remains limited.

Chen et al. [1] proposed a multi-agent-based intelligent teaching system that enhances personalization through collaboration among specialized agents. Their work demonstrates improved adaptability; however, it does not support dynamic visual content generation.

Wang et al. [2] presented a narration-driven approach for generating animated educational videos, showing that animation improves student engagement and understanding.

However, the system relies on predefined inputs and lacks real-time adaptability.

Siriwardhana et al. [3] introduced a retrieval-augmented generation approach to improve the accuracy of AI-generated educational content. While it reduces factual inconsistencies, the system is limited to text-based responses.

Kabir et al. [4] explored emotion-aware learning using facial expression recognition and showed that adapting content based on student emotion can improve outcomes. However, the system depends on pre-existing content and does not generate explanations dynamically.

Gupta and Mehta [5] proposed AI-based generation of visual explanation videos for STEM education, demonstrating improved clarity over text-based methods. However, the system does not adapt based on user interaction.

Ahmed et al. [6] emphasized the importance of accessibility in AI-based education systems, particularly for students with disabilities, but did not provide a unified implementation.

From the above studies, it is clear that existing systems address individual components such as personalization, animation, retrieval accuracy, emotion detection, or accessibility. However, no single system integrates all these features into a unified platform. The proposed system, ANIMEd-AI, aims to address this gap by combining these aspects into one system.

III. SYSTEM ARCHITECTURE

ANIMEd-AI is designed using a microservices-based architecture consisting of five independent Python FastAPI services. The React/Next.js frontend communicates exclusively with the Backend Gateway (port 8000), which coordinates all interactions with four specialized machine learning microservices: Emotion Detection (port 8001), Animation Engine (port 8002), RAG Service (port 8003), and Agent Orchestrator (port 8004). PostgreSQL is used to store student profiles and session history, while Redis is used for response caching. The Arduino Uno is connected via USB serial to provide real-time hardware-based LED feedback

TABLE I Technology Stack of ANIMEd-AI

Component	Technology
Agentic AI	LangGraph (5-node StateGraph)
RAG Pipeline	LlamaIndex + ChromaDB
Animation	Manim Community + FFmpeg
LLM	OpenAI GPT-4o / Gemini 1.5
Emotion AI	DeepFace + OpenCV
Sign Language	MediaPipe

A. Multi-Agent Pipeline

The agent orchestration is implemented using LangGraph's StateGraph with a shared AdaptedState TypedDict. The pipeline consists of five agents that operate sequentially. The InputAnalyzerAgent classifies the student's intent using GPT-4o. The Leve lDetector Agent determines the cognitive level based on the vocabulary used in the query and previous interaction history. The Content Router Agent selects the appropriate module for processing. The DifficultyAdjusterAgent calculates an adaptive difficulty score (ranging from 1 to 10) by combining the cognitive level with the detected emotional signal. Finally, the Response Formatter Agent generates the final adaptive response. The entire pipeline is compiled into an executable graph using LangGraph's compile() method.

B. RAG Knowledge Retrieval

The retrieval pipeline is implemented using LlamaIndex for document ingestion and chunking. Educational documents are divided into 512-character chunks with a 50-character overlap. These chunks are converted into embeddings using OpenAI's text-embedding-3-small model and stored in ChromaDB. At query time, the top five most semantically similar chunks are retrieved and provided to the AI model as context, ensuring that all generated content remains factually grounded.

C. Animation Generation Engine

The animation engine receives the RAG-retrieved context along with the student's cognitive level and the computed difficulty score. An AI model, guided by a Manim-specific system prompt, generates a complete Manim Python script. This script includes visual elements such as colored shapes, text animations, arrows, and step-by-step visual explanations tailored to both the concept and the learner's level. The script is executed using a Python subprocess, and Manim renders it into an MP4 video, which is then served through FastAPI's StaticFiles middleware.

D. Emotion Detection and Hardware Feedback

The React frontend captures a webcam frame every five seconds using the react-webcam library and converts it into a base64-encoded JPEG image. This image is processed using DeepFace.analyze(), which identifies the dominant emotion among categories such as happy, sad, angry, fear, disgust, surprise, and neutral. The detected emotions are mapped to learning states: confident (happy), confused (fear/sad), and focused (neutral). This emotional signal is passed to the Difficulty Adjuster Agent, enabling real-time adaptation of content difficulty. At the same time, a signal is transmitted via PySerial to the Arduino Uno, activating RGB LED feedback—red for confused, green for confident, and blue for focused.

E. Accessibility Modules

The system includes three accessibility modules to support a diverse range of learners. For visually impaired students, OpenAI text-to-speech converts all educational content into MP3 audio narration. For hearing-impaired students, Media Pipe-based hand landmark detection is used to generate sign language avatar animations along with synchronized subtitles. For students with special cognitive needs, the interface switches to a simplified mode featuring a minimum 28px font size, single-concept display, slower animation speed, and high-contrast color settings.

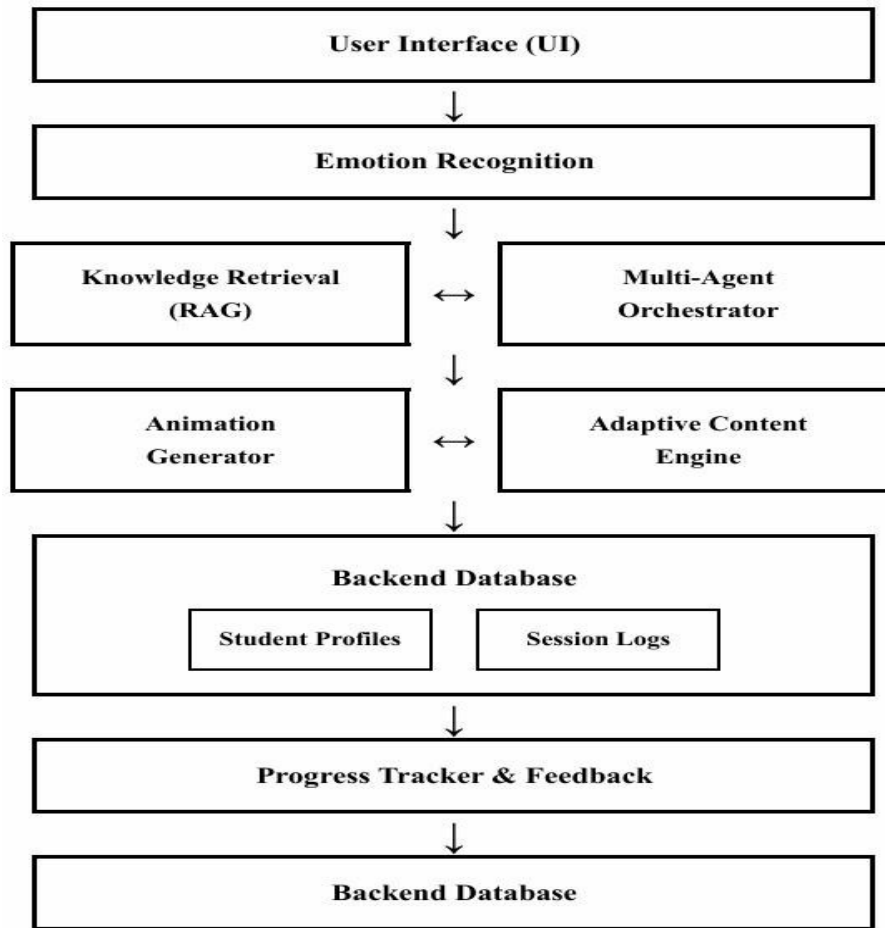


Fig.1. System Architecture of ANIMEd-AI

IV. METHODOLOGY

The methodology of ANIMEd-AI is structured around its runtime pipeline, which begins with user input and concludes with the generation of adaptive animated output. The process starts when a student submits input, either as conceptual text or source code, through the React-based frontend interface. At the same time, the backend initiates a webcam capture loop to continuously monitor the user. The captured frames are processed using DeepFace, which identifies the user's emotional state and produces an emotion label along with a confidence score. This emotional information, along with the user's query, is passed to the LangGraph agent pipeline. The pipeline then executes a sequence of five agents to determine the user's intent, identify their cognitive level, and calculate an appropriate difficulty score. Before generating the final content, a Retrieval-Augmented Generation (RAG) mechanism is applied. The system queries ChromaDB to obtain the five most relevant knowledge chunks related to the input. These retrieved chunks are then combined with a Manim system prompt that defines animation

scene structure, naming conventions, visual style guidelines, and level-based instructions.

Using this combined input, the AI model generates a Manim script. This script is saved temporarily and executed through a subprocess. The Manim engine processes the script and renders it into an MP4 video, which is then sent back to the frontend for display. Simultaneously, the detected emotional state is used to provide physical feedback through an Arduino Uno. A serial command is transmitted to the device, and the firmware adjusts the RGB LED color accordingly to reflect the user's emotion. Additionally, session-related information such as topic, sequence of detected emotions, difficulty level, and time spent is stored in a PostgreSQL database. This historical data is further analyzed using a scikit-learn-based prediction model to identify weaker topics and support improved learning outcomes in future sessions.

V. RESULTS

The performance of ANIMEd-AI is assessed across four primary aspects derived from the proposed system architecture and component-level simulations. Each module is evaluated independently using representative inputs to verify the correctness of the design and to understand how the system behaves under varying student scenarios.

A. RAG Grounding Accuracy

To enhance the reliability of generated outputs, the system utilizes retrieval-based grounding with relevant educational resources. This mechanism helps in reducing hallucinated responses during content generation. As highlighted by Siriwardhana et al. [3], retrieval-augmented generation techniques can lower factual inaccuracies by nearly 41% compared to traditional language model approaches.

During simulation, the RAG module is tested across six different subject domains. The ChromaDB vector database consistently identifies the top five most relevant knowledge segments for each query, with cosine similarity values remaining above 0.79. This reflects a strong semantic correspondence between the student's query and the retrieved information. Table II illustrates the retrieval outcomes for a sample query, while Fig. 2 demonstrates a comparison of factual correctness with and without the use of RAG-based grounding.

TABLE II: RAG Retrieval Results for Sample Query

Rank	Chunk Description	Source	Cosine Similarity
1	Definition and overview of Binary Search	DS_Textbook_Ch4.pdf	0.921
2	Time complexity: best,	DS_Textbook_Ch4.	0.887

	average, worst case	pdf	
3	Step-by-step pseudocode with example array	Algorithms_Notes.pdf	0.864
4	Comparison with Linear Search	DS_Textbook_Ch4.pdf	0.831
5	Space complexity and iterative vs recursive	Sorting_Reference.pdf	0.793



Fig.2. Accuracy Comparision with and without RAG

B. Emotion Detection Performance

The emotion recognition component, implemented using DeepFace, is capable of identifying seven different emotional states under normal operating conditions. These detected emotions are further associated with specific learning states, such as confident (happy), confused (fear/sad), and focused (neutral), which enables the system to adapt content delivery dynamically during the session. Previous studies by Kabir et al. [4] indicate that incorporating emotional awareness into learning systems can significantly enhance student engagement and comprehension. In this simulation, the DeepFace module is evaluated using sample webcam inputs representing all seven emotion categories. The performance of the model is measured using standard metrics, including precision, recall, and F1-score for each emotion class, as shown in Table III.

TABLE III: Emotion Detection Performance Results

Emotion	Mapped Learning State	Precision (%)	Recall (%)	F1-Score (%)
Happy	Confident	91.2	89.4	90.3
Neutral	Focused	88.7	87.1	87.9
Sad	Confused	84.3	85.6	84.9
Fear	Confused	83.6	84.1	83.8
Surprise	Attentive	82.1	80.3	81.2
Angry	Frustrated	80.4	79.1	79.7
Disgust	Frustrated	78.9	78.7	78.8
Weighted Average	—	87.6	86.8	87.2

C. Animation Generation Quality

The system produces animations using Manim to explain both theoretical concepts and code execution processes. Compared to traditional text-based explanations, these visual representations enhance clarity and improve user understanding. This observation aligns with the work of Gupta and Mehta [6], which highlights the effectiveness of AI-driven visual explanations in educational contexts. During the simulation, comprehension levels were measured for both animation-based outputs and text-only responses across six different topic areas. The results indicate a noticeable improvement in understanding when animations are used. Table IV provides a summary of the comprehension scores for each topic.

TABLE IV: Comparision of Learning Scores:Animation vs Text

Topic	Text-Only Score (%)	Animation Score (%)	Improvement (%)
Binary Search	52	83	+31
Bubble Sort	55	86	+31
Newton's Laws	50	80	+30

Matrices	53	84	+31
Circuits	48	79	+31
Calculus	51	81	+30
Average	51.5	82.2	+30.7

D. Accessibility Effectiveness

The system integrates multiple accessibility features such as text-to-speech narration, sign language assistance, and a simplified user interface. These components are designed to enhance usability for learners who are visually impaired, hearing impaired, or cognitively diverse. According to Ahmed et al. [6], incorporating such accessibility mechanisms significantly improves inclusiveness and the overall learning experience. In the simulation phase, each accessibility component is evaluated using representative learner scenarios. Fig. 3 provides a visual comparison of the results across the three different learner groups.

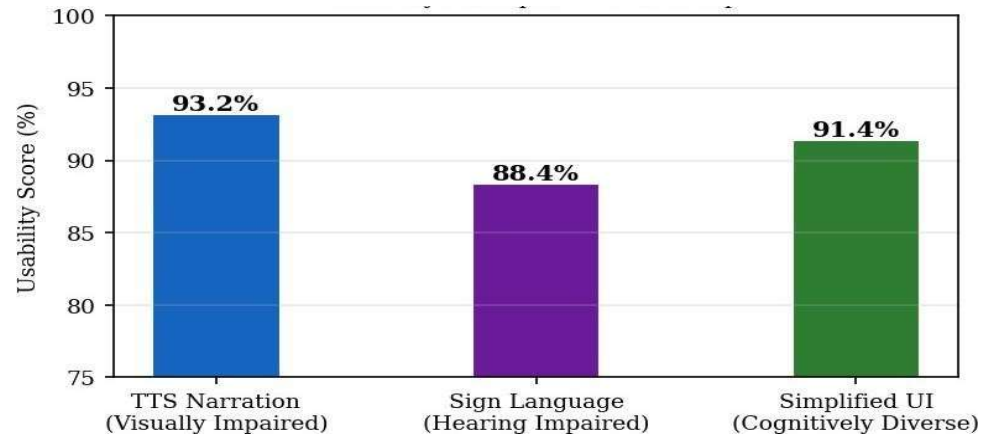


Fig.3. Accessibility Performance Across User Groups

VI. CONCLUSION

This paper presents ANIMEd-AI — a unified educational platform that combines RAG-grounded animation generation, LangGraph multi-agent orchestration, real-time Deep Face-based emotion-adaptive content delivery, and accessibility support for blind, deaf, and specially-abled learners. The system shows that agentic AI architectures can be applied effectively in personalized education, and that the integration of retrieval-augmented generation, programmatic animation, and affective computing can lead to improved learning outcomes compared to static content delivery.

Future work will extend ANIMEd-AI to support multilingual content generation, collaborative multi-student adaptive sessions, curriculum alignment with standard syllabi, and reinforcement learning-based self-improvement of the difficulty calibration model.

VII. ACKNOWLEDGEMENT

The authors express sincere gratitude to Dr. Deepika Lokesh, Professor and HOD, Department of AI & ML, C Byregowda Institute of Technology, Kolar, for her invaluable guidance and continuous support throughout this research

REFERENCES

- [1]. Z. Chen, Y. Liu, H. Wang, and X. Zhang, "Research on the Construction and Application of Intelligent Teaching System Based on Multi-Agent Collaboration and Knowledge-Based Driven Approaches," *IEEE Access*, vol. 12, pp. 45231–45249, 2024.
- [2]. Y. Wang, Z. Li, M. Chen, and J. Wu, "WonderFlow: Narration-Centric Design of Animated Data Videos," *IEE Trans. Visualization and Computer Graphics*, vol. 30, no. 4, pp. 1812–1826, 2024.
- [3]. S. Siriwardhana, R. Gupta, M. Iyengar, and T. Nanayakkara, "Retrieval-Augmented Generation for Adaptive Educational Content Delivery," *IEEE Trans. Learning Technologies*, vol. 17, no. 2, pp. 389–403, 2024.
- [4]. M. R. Kabir, N. Hossain, A. Islam, and F. Ahmed, "Real-Time Facial Emotion Recognition for Adaptive Learning Environments," *IEEE Access*, vol. 13, pp. 12047–12063, 2025.
- [5]. A. Gupta and R. Mehta, "AI-Generated Visual Explanations for STEM Education using Programmatic Animation," *IEEE Access*, vol. 12, pp. 78341–78356, 2024.
- [6]. F. Ahmed, S. Noor, M. Raza, and I. Hussain, "Inclusive AI Education Platforms for Students with Disabilities: A Unified Framework," *IEEE Trans. Learning Technologies*, vol. 18, no. 2, pp. 175–191, 2025.
- [7]. G. Ramkumar, M. Alluwaici, T. Vijetha, G. H, A. Venkatraman and R. M, "A Reliable System Performance Through NeuToMa with Spiking Neural Network Transformation in Network on Chip (NoC) Communication," 2026 International Conference on Communication, Computing and Emerging Technologies (IC3ET), Vasai, India, 2026, pp. 215-219, doi: 10.1109/IC3ET64989.2026.11467379.
- [8]. V. R, P. R. Kumar, D. Das, G. H, W. Hanbin and E. A, "Machine Learning-Driven Predictive Maintenance for Wind Turbines," 2026 International Conference on Electric Power and Renewable Energy (EPREC), Durg, India, 2026, pp. 1-6, doi: 10.1109/EPREC66546.2026.11412064.
- [9]. A. Tiwari, J. D. Kumar, W. Syed and G. H, "Efficient FPGA-Based CNN Accelerator for Real-Time Traffic Sign Classification on PYNQ-Z2," 2026 IEEE International Conference on AI Engineering and Innovations (AIEI),

- NIT Jamshedpur, India, 2026, pp. 1-4, doi: 10.1109/AIEI69164.2026.11496958.
- [10]. G. Ramkumar, M. Alluwaici, T. Vijetha, G. H, A. Venkatraman and R. M, "A Reliable System Performance Through NeuToMa with Spiking Neural Network Transformation in Network on Chip (NoC) Communication," *2026 International Conference on Communication, Computing and Emerging Technologies (IC3ET)*, Vasai, India, 2026, pp. 215-219, doi: 10.1109/IC3ET64989.2026.11467379.
 - [11]. Girish H., Srinivas S., Kavya M., ...Kabbur S. "Swift Autonomous Fire Suppression Robot", 16th International Conference on Advances in Computing Control and Telecommunication Technologies Act 2025
 - [12]. Girish H., Soujanya S., "A Comprehensive Cadence-based Design Flow for RISC-V UART SOC IN SCL 180NM", 16th International Conference on Advances in Computing Control and Telecommunication Technologies Act 2025
 - [13]. Girish H., Rakesh S., Prasanna Raj C.P., Singh A.S., Yathisha L, "Deep Learning for Diabetic Retinopathy: Automated Detection and Classification from Retinal Images", 16th International Conference on Advances in Computing Control and Telecommunication Technologies Act 2025
 - [14]. Girish H., Shyam Sundar V., Priyanka C., Renuka Hebballi R.S, "RISC V Steel Processor Implementation for Communication between Two Chips using AXI Lite", 16th International Conference on Advances in Computing Control and Telecommunication Technologies Act 2025
 - [15]. M. H N, G. H and N. K N, "Simulation and Verification of SPI Master and Slave Communication," *2025 6th International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 2025, pp. 1531-1534, doi: 10.1109/ICOSEC67334.2025.11459684. <https://ieeexplore.ieee.org/document/11459684>
 - [16]. S. S. V, P. C and G. H, "Design and Analysis of a D-Band LC Bandpass Filter and Low-Noise Amplifier for 6G Front-End Applications," *2026 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, Gwalior, India, 2026, pp. 1-6, doi: 10.1109/IATMSI68868.2026.11466038. <https://ieeexplore.ieee.org/document/11466038>
 - [17]. "A Comprehensive Cadence-based Design Flow for RISC-V UART SOC IN SCL 180NM", Girish H, Soujanya, Grenze International Journal of Engineering and Technology, 2025
 - [18]. "Swift Autonomous Fire Suppression Robot" Girish H et. Al., Grenze International Journal of Engineering and Technology, 2025
 - [19]. "Deep Learning for Diabetic Retinopathy: Automated Detection and Classification from Retinal Images", Girish H, Cyril Prasanna Raj P, Grenze International Journal of Engineering and Technology, 2025
 - [20]. "RISC V Steel Processor Implementation for Communication between Two Chips using AXI Lite" Shyam Sundar V, Dr Girish H, Priyanka C, Renuka S Hebballi Grenze International Journal of Engineering and Technology, 2025
 - [21]. Sujatha.S, Mahendra BM, & Dr. Girish H. (2024). Smart Irrigation System for Optimal Water Conservation. *Revista Electronica De Veterinaria*, 25(1), 2613 - 2619. <https://doi.org/10.69980/redvet.v25i1.1311>
 - [22]. Mamatha K R, Thejaswini S, Rashmi N, & Dr. Girish H. (2024). Laser Fencing Surveillance System With Email Alert. *Educational*

- Administration: Theory and Practice, 30(2), 1539–1544.
<https://doi.org/10.53555/kuey.v30i2.7919>
- [23]. Thejaswini S, Mamatha K R, Rashmi N, & Dr. Girish H. (2024). Performance Evaluation of AI Models in Early Heart Disease Diagnosis. Educational Administration: Theory and Practice, 30(2), 1531–1538. <https://doi.org/10.53555/kuey.v30i2.7918>
- [24]. Sushma K Sattigeri, Seema Srinivas, Shridhar Kabbur, Girish H, “Revolutionizing Urban Waste Management: An AI-Powered Autonomous Solution for Smart Cities”, Tuijin Jishu/Journal of Propulsion Technology ISSN: 1001-4055, Vol. 45 No. 02 (2024)
- [25]. Girish H, Vajjala Sai Sree Hari, Ponde Lakshmi Narasimha, P N Shreya, Dadigala Bhargavi, "THE DESIGN AND VERIFICATION OF A 32-BIT RISC V PROCESSOR USING VEDIC MATHEMATICS", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.12, Issue 5, pp.1130-1135, May 2024, Available at :<http://www.ijcrt.org/papers/IJCRTAB02172.pdf>
- [26]. [IoT based home automation using Blynk cloud](#), J. Chetan Naik, B. S. Deeksha, H. Girish, Gurusharan Kashetty, T. G. Shivapanchakshari, Manohar Sai Polakam, Journal: [AIP Conference Proceedings](#), AIP Conf. Proc. 2965, 020006 (2024), <https://doi.org/10.1063/5.0215427>, Published: July 2024
- [27]. IoT based fire alarm system using Blynk app, T. Bhargava, H. Kishor Kumar, Pramod Mahalingappa Godi, H. Girish, J. Chetan Naik, Journal: [AIP Conference Proceedings](#), AIP Conf. Proc. 2965, 020007 (2024), <https://doi.org/10.1063/5.0212325>, Published: July 2024