

**CONVERSATIONAL IMAGE RECOGNITION CHATBOT**

**A PROJECT REPORT**

## 

***Submitted by***

K KEERTHI – 20221ISE0046

ADITI N – 20221ISE0085

RAVIRAJ P MATH – 20221ISE0055

### *Under the guidance of,*

**Mr. JAIKUMAR B**

**BACHELOR OF TECHNOLOGY**

**IN**

**INFORMATION SCIENCE AND ENGINEERING**

**PRESIDENCY UNIVERSITY**

**BENGALURU**

**DECEMBER 2025**



**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**BONAFIDE CERTIFICATE**

Certified that this report “**CONVERSATIONAL IMAGE RECOGNITION CHATBOT**” Is a Bonafide work of “K KEERTHI (20221ISE0046), ADITI N(20221ISE0085), RAVIRAJ P MATH (20221ISE0055),” who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in **INFORMATION SCIENCE AND ENGINEERING** during 2025-26.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dr.Sampath A K**  Project Guide  Presidency School of Computer Science and Engineering  Presidency University | **Dr. Anandaraj S P**  Head of the Department Presidency School of Computer Science and Engineering  Presidency University | **Dr. Shakkeera L** Associate Dean  Presidency School of Computer Science and Engineering  Presidency University | **Dr. Duraipandian N**  Dean  PSCS & PSIS  Presidency University |

### Name and Signature of the Examiners

### 1)

2)

**PRESIDENCY UNIVERSITY**

**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

DECLARATION

We the students of final year B. Tech in INFORMATION SCIENCE ENGINEERING at Presidency University, Bengaluru, named K KEERTHI, ADITI N, RAVIRAJ P MATH, hereby declare that the project work titled **“<PROJECT TITLE>”** has been independently carried out by us and submitted in partial fulfilment for the award of the degree of B. Tech in COMPUTER SCIENCE ENGINEERING, CYBER SECURITY during the academic year of 2025-26. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree or Diploma to any other institution.

K KEERTHI USN: 20221ISE0046 <Signature 1>

ADITI N USN: 20221ISE0085 <Signature 2>

RAVIRAJ P MATH USN: 20221ISE0055 <Signature 3>

PLACE: BENGALURU

DATE:

ACKNOWLEDGEMENT

For completing this project work, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide Dr./Mr/Ms **Dr. Sampath A K, Professor**, Presidency School of Computer Science and Engineering, Presidency University, for his/her moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Anandaraj, Professor, Head of the Department, Presidency School of Computer Science and Engineering** Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr. Sampath A K, and Dr. Geetha A, PSCS** Project Coordinators**, Dr. Sharmast vali, Program Project Coordinator**, Presidency School of Computer Science and Engineering, or facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering and also staff from other departments who have extended their valuable help and cooperation.

K KEERTHI

ADITI N

RAVIRAJ P MATH

Abstract

Artificial intelligence has transformed the way we interact with technology, and conversational AI is a key example, helping people with tasks ranging from personal assistance to customer service. Yet, traditional chatbots are limited—they excel at understanding and generating text but struggle to interpret visual content such as images, diagrams, or complex scenes. This limits their usefulness in many real-world situations where seeing is as important as understanding.

To address this, the project introduces a multimodal chatbot that can both see and converse. It integrates context-aware dialogue, image captioning, visual question answering, and object detection into a single framework. Using LLaMA-2/GPT for natural conversation, BLIP/BLIP-2 for generating descriptive captions, and YOLOv8 for detecting and annotating objects, the system can understand a scene, answer questions about it, and maintain coherent multi-turn conversations. Its modular design allows each component to be updated or improved independently, making it adaptable for future AI advancements.

Extensive testing showed the chatbot can accurately recognize multiple objects, generate meaningful captions, and respond in contextually relevant ways over several rounds of interaction. This makes it suitable for applications such as e-commerce product identification, industrial monitoring, accessibility tools for visually impaired users, and interactive educational platforms.

In conclusion, this project demonstrates a real-time, multimodal chatbot that blends conversational intelligence with visual perception. By integrating language and vision, it paves the way for more intuitive, interactive, and human-like communication between people and machines.

Table of Content

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title** | **Page No.** |
|  | Declaration |  |
|  | Acknowledgement | 1 |
|  | Abstract | 2 |
|  | List of Figures | 5 |
|  | List of Tables | 6 |
|  | Abbreviations | 7 |
| 1. | Introduction  1.1 Background  1.2 Statistics of project  1.3 Prior existing technologies  1.4 Proposed approach  1.5 Objectives  1.6 SDGs  1.7 Overview of project report | 8-13 |
| 2. | Literature review | 14-17 |
| 3. | Methodology | 18-20 |
| 4. | Project management  4.1 Project timeline  4.2 Risk analysis | 21-23 |
| 5. | Analysis and Design  5.1 Requirements  5.2 System Architecture  5.3 System Flow  5.4 Designing Units  5.5 Communication & Deployment | 24-26 |
| 6. | Software and Simulation  6.1 System Requirements  6.2 Software Development Tools  6.3 Software Code Overview  6.4 Simulation and Testing | 27-29 |
| 7. | Evaluation and Results  7.1 Testing Methodology  7.2 Sample Observations  7.3 Example Outputs  7.4 Analysis 7.5 Summary | 30-32 |
| 8. | Social, Legal, Ethical, Sustainability and Safety Aspects  8.1 Social aspects  8.2 Legal aspects  8.3 Ethical aspects  8.4 Sustainability aspects  8.5 Safety aspects | 33-35 |
| 9. | Conclusion | 36-37 |
| 10. | References | 38 |

List of Figures

| **Figure** | **Caption** | **Page No** |
| --- | --- | --- |
| Fig 1.1 | Sustainable development goals | 3 |
| Fig 3.3.1 | Agile Scrum Framework | 25 |
| Fig 3.3.2 | Mapping of Chatbot Project Stages to Agile Scrum | 26 |
| Fig 4.1 | Project Planning Timeline | 35 |
| Fig 4.2 | Project Implementation Timeline | 36 |
| Fig 5.1 | Architecture of Conversational Image Recognition Chatbot | 42 |
| Fig 5.2 | System Flow Chart | 43 |
| Fig 7.1 | Detection and chatbot response for dining table image | 58 |

List of Tables

| **Table** | **Caption** | **Page No** |
| --- | --- | --- |
| Table 2.1 | Summary of Literature Reviews | 20 |
| Table 4.3 | Example of PESTEL Analysis | 38 |
| Table 4.4 | Another Example of PESTEL Analysis | 39 |
| Table 4.5 | Example of Project Phase Risk Matrix | 40 |
| Table 5.4 | Designing Units and Tools/Libraries | 45 |

Abbreviations

| **Abbreviation** | **Full Form** |
| --- | --- |
| YOLO | You Only Look Once |
| CNN | Convolutional Neural Network |
| BLIP | Bootstrapped Language-Image Pretraining |
| NLP | Natural Language Processing |
| GPT | Generative Pre-trained Transformer |
| LLaMA | Large Language Model Meta AI |
| GPU | Graphics Processing Unit |
| PESTEL | Political, Economic, Social, Technological, Environmental, Legal |
| API | Application Programming Interface |

Chapter 1

Introduction

Conversational systems have become a regular part of daily life due to rapid advancements in artificial intelligence (AI), which have fundamentally changed how humans interact with machines. Chatbots, virtual assistants, and intelligent agents are now widely used in industries such as e-commerce, customer service, education, and healthcare. However, most of these systems are limited to handling text or voice inputs, which restricts their usefulness in scenarios where visual information plays a crucial role [1][2].

Visual data often contains complex context that is difficult to convey through text alone. For example, a consumer browsing products online, an engineer inspecting machinery, or a teacher explaining a scientific diagram relies heavily on images to communicate meaning. Traditional chatbots, which cannot process or interpret visual inputs, fall short in such situations [3].

Recent advances in computer vision and multimodal AI have begun to bridge this gap. By combining conversational AI with vision-based models such as object detection, image captioning, and visual question answering (VQA), researchers have enabled multimodal chatbots to understand images, generate descriptive captions, and engage users in context-aware dialogue [4][5]. These systems not only make human-computer interaction more natural but also expand applications in smart surveillance, medical diagnostics, industrial safety, education, and accessibility for visually impaired users.

This project aims to design and implement a multimodal chatbot using cutting-edge AI models: YOLOv8 for object detection, BLIP/BLIP-2 for image captioning and VQA, and a large language model (LLM) such as LLaMA-2 or GPT for dialogue management. By integrating these components, the system can provide real-time, insightful responses to both textual and visual queries.

Global statistics underline the need for this project. Text-only content is rapidly declining, with over 80% of online content now being visual, including images, videos, and infographics [6]. The EdTech sector in India has grown by more than 35% annually, driven by online learning platforms that rely on multimodal engagement [2]. Accessibility considerations are equally critical: over 2.2 billion people worldwide experience some form of visual impairment, making AI systems capable of interpreting and translating visual information highly valuable [7].

While modern assistants like Siri, Alexa, and Google Assistant utilize natural language understanding, they remain primarily text- or voice-centric. Earlier chatbots such as ELIZA and ALICE were rule-based and limited to predefined conversation patterns [7]. Separately, computer vision systems like CNNs trained on ImageNet and YOLO detectors can classify and detect objects effectively but lack conversational capabilities [4]. Recent research efforts now integrate these domains to create unified multimodal frameworks.

In summary, by developing a chatbot that can understand, interpret, and describe visual content alongside text, this project addresses a significant gap in human-computer interaction and contributes to the growing field of multimodal AI.

1.1 Background

Significant improvements in human-computer interaction have resulted from AI development. Traditional chatbots mainly use text or voice inputs and are popular in online education, virtual assistants, and customer service. They can provide automated answers but cannot interpret visual content like images, diagrams, or real-world scenes [1][2].

Visual content often includes critical context that cannot be expressed through text alone. For instance, understanding an industrial inspection scene or a classroom diagram requires recognizing multiple objects, their relationships, and actions within a scene. Conventional chatbots are unable to provide meaningful responses in such cases [3].

Recent advancements in multimodal AI enable chatbots to integrate visual perception. By combining object detection, image captioning, VQA, and context-aware dialogue, multimodal chatbots can identify objects, generate descriptive captions, and provide interactive, informative responses to users [4][5]. This integration allows richer and more natural human-AI interaction, opening applications in education, e-commerce, industrial monitoring, and accessibility for visually impaired users.

1.2 Statistics

The increasing volume of digital content highlights the urgent need for AI systems that can process both text and visual data. Over 80% of online content worldwide is visual, including photos, videos, and infographics, while text-only content is rapidly decreasing [6]. Intelligent systems that can interpret, analyse, and respond to visual data are increasingly in demand.

The EdTech industry in India has grown by over 35% annually, driven by online tutorials, digital classrooms, and AI-powered learning platforms [2]. Educators and students require resources that can interactively explain complex diagrams and respond to image-based queries.

Accessibility is another key consideration. According to WHO estimates, over 2.2 billion people globally have some form of visual impairment [7]. AI systems capable of translating visual content into descriptive dialogue or text can significantly improve learning, employment, and social inclusion for these users.

Visual data is also crucial in industrial and commercial contexts. E-commerce platforms process thousands of product images daily, and manufacturing and medical sectors rely heavily on image-based inspections for quality control [4][5]. Traditional text-based systems cannot handle this data effectively, reinforcing the need for multimodal AI.

1.3 Prior Existing Technologies

Many technologies have enhanced human-computer interaction, but most focus either on conversational AI or computer vision, rarely integrating both.

1.3.1 Conventional Chatbots

ELIZA (1966) and ALICE (1995) were rule-based chatbots designed to mimic conversation using pattern recognition and preprogrammed responses [7]. These systems could only manage text-based interactions and were unable to process visual data. Modern conversational AI systems like Siri, Alexa, and Google Assistant use NLP and machine learning to understand text or speech but remain largely text/voice-centric [3].

1.3.2 Models for Computer Vision

CNNs trained on datasets like ImageNet and COCO have achieved remarkable performance in image classification, object detection, and scene recognition [4][5]. Object detection frameworks like YOLO and Faster R-CNN can locate and classify multiple objects in images [4]. Image captioning and VQA models such as BLIP/BLIP-2 and CLIP generate textual descriptions and answer questions about visual content [5][6].

1.3.3 Limitations of Existing Technologies

* Modality separation: Chatbots and vision systems operate independently, preventing seamless image-query interactions.
* Limited contextual understanding: Multi-turn dialogue and prior context are not fully supported.
* Deployment constraints: Large models require high computational resources and labelled datasets, limiting real-time or lightweight deployment.
* Accessibility gap: Current systems do not fully address the needs of visually impaired users [7].

In conclusion, while conventional chatbots and vision models perform well individually, they cannot provide a unified multimodal system capable of real-time image understanding with context-aware conversation. This limitation motivates the development of a multimodal chatbot integrating conversational AI, object detection, image captioning, and VQA.

1.4 Proposed Approach

Goal: Develop a multimodal chatbot that understands both visual and textual inputs, detects objects, generates captions, responds to queries, and maintains multi-turn, context-aware conversation.

Motivation: The growing amount of visual content online and in real-world applications requires AI systems that can interpret images and diagrams in real-time. Multimodal conversational systems are essential for interactive learning, industrial and e-commerce applications, and accessibility for visually impaired users [1][2].

Suggested Method: The project uses a modular AI pipeline:

1. Object Detection: YOLOv8 detects and annotates objects in images.
2. Captioning & VQA: BLIP/BLIP-2 generates captions and answers questions about detected objects.
3. Dialogue Management: LLaMA-2 or GPT ensures context-aware, multi-turn dialogue.
4. User Interaction: Gradio provides a simple web interface for real-time interaction and displays annotated images.

Applications:

* Education: Interactive AI tutors explaining charts and diagrams.
* Accessibility: Translating visual content into descriptive dialogue for visually impaired users.
* E-commerce: AI assistants comparing products and responding to queries.
* Industrial Monitoring: Automated inspection and commentary on images.
* Healthcare: Interactive diagnostic assistance and image analysis.

Limitations:

* High computational requirements for running multiple AI models simultaneously.
* Possible challenges with cluttered or complex scenes.
* Dependency on the diversity and quality of training datasets.
* Limited multilingual support.

This approach overcomes current limitations by integrating visual understanding with conversational AI to provide real-time, context-aware interactions.

1.5 Objectives

1. Behaviour: Develop a multimodal chatbot capable of interpreting text and images, detecting objects, and generating context-aware responses.
2. Analysis: Evaluate the accuracy of object detection, image captioning, and VQA for reliable understanding and conversation.
3. System Management: Integrate YOLOv8, BLIP/BLIP-2, and LLaMA-2/GPT into a modular pipeline for independent testing and maintenance.
4. Security: Ensure secure handling of user-uploaded images and conversation data.
5. Deployment: Implement a user-friendly web interface via Gradio for real-time interaction, deployable on cloud or local servers.

1.6 SDGs

The multimodal chatbot aligns with UN Sustainable Development Goals:

1. SDG 4 – Quality Education: Interactive descriptions of complex concepts, images, and diagrams enhance learning [1][2].
2. SDG 9 – Industry, Innovation, and Infrastructure: Incorporates cutting-edge AI models in a modular framework applicable to multiple industries [3].
3. SDG 10 – Reduced Inequalities: Converts visual content into dialogue, improving accessibility for visually impaired users [4].
4. SDG 8 – Decent Work and Economic Growth: Enhances productivity and human-AI collaboration in e-commerce and industrial monitoring [5].

By addressing these SDGs, the chatbot promotes inclusive, sustainable, and innovative solutions to social and educational challenges.



Fig 1.1 Sustainable development goals [1]

1.7 Overview of project report

Beginning with Chapter 1, which presents the project topic, its goals, and its importance in the current context, this report offers a thorough analysis of the project. A thorough literature review is provided in Chapter 2, emphasizing earlier research, current solutions, and research gaps that the project seeks to fill. The project's methodology, including the instruments, strategies, and frameworks employed to accomplish the goals, is covered in Chapter 3. Project planning and budgeting are the main topics of Chapter 4, which also includes schedules, cost estimates, and resource allocation. The project's design, execution, and testing are covered in later chapters, which also present the findings and analysis. Chapter 8, titled "Conclusions and Future Work," concludes by summarizing the main conclusions, assessing the project's results, and outlining possible avenues for further study or improvement.

Chapter 2

Literature review

1. Lavanya Kesa et al. – Chatbot with Face Mask Detection  
Lavanya Kesa and colleagues developed a hybrid system that combines a conversational chatbot with a face mask detection module to address COVID-19 safety monitoring. The chatbot, implemented using Chatterbot, interacts with users by providing information on COVID-19 precautions, while the mask detection system utilizes a Convolutional Neural Network (CNN) with TensorFlow, Keras, and OpenCV for classifying images of masked and unmasked faces. Their approach allows real-time monitoring through webcam input, ensuring adherence to safety protocols. The results indicated reliable detection in controlled settings and demonstrated the effectiveness of combining AI vision and dialogue systems. Limitations include misclassification under poor lighting or occlusion and a chatbot limited to predefined conversation data. The authors suggested future work should integrate more advanced NLP models and larger datasets for improved conversational flexibility and detection robustness. This study exemplifies the potential for combining computer vision and conversational AI for public safety, though scalability and dynamic real-world testing remain challenges [1].

2. Nishant Arora et al. – Conversational Image Recognition Chatbot  
Arora et al. proposed a novel system integrating YOLOv8 for object detection and BERT for natural language processing to create a conversational image recognition chatbot. The system can detect multiple objects in industrial images, extract individual components, and answer user queries about object identity, relationships, and functionality. The dataset used includes 3,500 high-resolution industrial images with detailed annotations to ensure accurate model training. Evaluation results showed high object detection accuracy (mAP 94.7%) and QA semantic accuracy (89.2%). Limitations include occasional confusion between visually similar objects and reduced understanding of highly domain-specific terms not present in the dataset. Recommendations for future work involve expanding datasets, using advanced architectures like Vision Transformers, extending support to video-based analysis, and adding multilingual interaction. This research illustrates the effective integration of computer vision and NLP for multimodal human-computer interaction in industrial, educational, and healthcare settings [2].

3. Jiang et al. – RetinaFace Mask Detection Model  
Jiang and colleagues developed a single-stage face mask detection framework based on RetinaFace, a feature pyramid network, to enhance precision and recall in mask compliance monitoring. The model effectively identifies faces wearing or not wearing masks in real-time while filtering out irrelevant objects to reduce false positives. Testing demonstrated strong detection performance even under partial occlusions. However, performance degraded under poor lighting conditions, and the model requires high-quality, well-labelled datasets for training. To improve real-world applicability, the authors suggested data augmentation and deployment on edge devices for real-time monitoring in public spaces. This study highlights the importance of combining speed and accuracy in computer vision applications for public health [3].

4. Liu et al. – Deep Learning Architectures for Object Detection  
Liu et al. explored various deep learning architectures, including Convolutional Neural Networks (CNNs), autoencoders, and deep belief networks (DBNs), for object detection and recognition. The study emphasized how hierarchical feature extraction allows models to recognize complex patterns from raw pixel data without relying on handcrafted features. Results showed improved accuracy in object recognition tasks, particularly for images with multiple objects or varying conditions. Limitations include the high computational cost of deep architectures and the dependence on large, labelled datasets for training. Future improvements proposed include lightweight architectures for edge deployment and semi-supervised learning techniques to reduce manual labelling requirements. The work provides a theoretical foundation for employing CNN-based models like YOLO in multimodal AI systems [4].

5. Li et al. – CNN-based Fast Face Detection  
Li et al. presented a CNN-based face detection approach optimized for low-resolution images, achieving fast and accurate detection by discarding non-facial regions early in the processing pipeline. The model applies multi-scale feature extraction to maintain detection accuracy across various face sizes. Experimental results demonstrated reliable detection performance under standard conditions, but limitations were observed when faces were partially occluded or oriented at extreme angles. Suggested improvements include integrating attention mechanisms and expanding training datasets to include more varied face orientations and environments. The study contributes to practical AI deployment in scenarios requiring rapid, low-latency object detection [5].

6. Matthias et al. – Real-time Face Mask Detection  
Matthias and colleagues developed a real-time face mask detection system using CNNs, focusing on identifying critical facial regions such as eyes, nose, and mouth for classification. The system was tested under various lighting conditions and backgrounds, performing reliably in most controlled environments. Limitations include reduced detection accuracy in complex backgrounds or inconsistent lighting. Future work may involve adaptive learning methods and multi-camera setups to improve detection in dynamic real-world scenarios. This research demonstrates the utility of combining computer vision with real-time monitoring to ensure compliance with public health standards [6].

7. Weizenbaum – ELIZA Chatbot  
Weizenbaum’s ELIZA chatbot, one of the first conversational AI programs, simulated human conversation using rule-based keyword matching. Though the system could mimic dialogue convincingly, it lacked deep language understanding and could only respond to predefined patterns. This early work laid the foundation for modern chatbots by demonstrating the feasibility of human-computer dialogue. Limitations included superficial understanding and inability to handle multi-turn, context-aware conversations. The study highlights the need for integrating advanced NLP models, such as BERT or GPT, in modern systems to achieve contextual understanding and robust question-answering capabilities [7].

Summary of Literatures reviewed

| Sl.No | Author(s) & Year | Concept / Focus | Approach & Methods | Key Results | Limitations / Issues | Future Work / Suggestions |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Lavanya Kesa et al., 2021 | Chatbot integrated with face mask detection | Hybrid system using Chatterbot for dialogue and CNN (TensorFlow/Keras) for mask detection | Real-time mask detection and conversational interface; effective under controlled conditions | Misclassification under poor lighting or occlusion; limited chatbot conversation scope | Improve NLP with advanced models; expand dataset; real-world testing |
| 2 | Nishant Arora et al., 2025 | Conversational image recognition chatbot | YOLOv8 for object detection, BERT for question answering, custom industrial image dataset | Object detection Map 94.7%, QA semantic accuracy 89.2%; supports multi-object queries | Confusion between similar objects; domain-specific terms not in dataset | Expand datasets; integrate advanced models (ViT, T5/GPT); video analysis; multilingual support |
| 3 | Jiang et al., 2020 | RetinaFace mask detection | Single-stage RetinaFace network with feature pyramid for real-time mask detection | High precision and recall; robust detection under partial occlusions | Performance reduced under poor lighting; requires well-labelled datasets | Data augmentation; edge deployment for real-time public monitoring |
| 4 | Liu et al., 2019 | Deep learning for object detection | CNNs, autoencoders, and DBNs for hierarchical feature extraction | Improved accuracy for multi-object and variable condition images | High computational cost; large labelled datasets needed | Lightweight architectures; semi-supervised learning to reduce labelling effort |
| 5 | Li et al., 2018 | Fast CNN-based face detection | Multi-scale CNN for low-resolution images; discards non-face regions early | Reliable detection across varying face sizes; low latency | Reduced accuracy for occluded or extreme-angle faces | Integrate attention mechanisms; expand dataset for varied orientations |
| 6 | Matthias et al., 2020 | Real-time face mask detection | CNN focusing on critical facial regions (eyes, nose, mouth) | Effective detection under standard lighting and backgrounds | Lower accuracy in complex backgrounds or inconsistent lighting | Adaptive learning; multi-camera setups for dynamic environments |
| 7 | Weizenbaum, 1966 | ELIZA chatbot | Rule-based keyword matching for conversational simulation | Demonstrated feasibility of human-computer dialogue | Lacked real understanding; could not handle multi-turn, context-aware conversation | Integrate advanced NLP (BERT, GPT) for contextual understanding |

Table 2.1 Summary of Literature reviews

Chapter 3

Methodology

3.1 Introduction to Methodology

For our project, we have adopted the Agile Scrum methodology as the development and project management framework. Scrum is iterative and incremental in nature, which makes it highly suitable for projects involving multiple complex modules. Since our work integrates computer vision (YOLOv8), natural language processing (BLIP/LLaMA), facemask detection, and cloud deployment, we required a methodology that provides flexibility, continuous testing, and regular incorporation of feedback.

By following Scrum, we divided the development process into short sprints. Each sprint enabled us to identify errors early, refine the system step by step, and ensure that all components were aligned with the project objectives.

3.2 Mapping Project Stages to Scrum

We mapped the different stages of our chatbot development process to the activities defined in Scrum. This mapping kept our workflow organized and ensured that every task had a clear place in the development cycle.

Table 3.1 Mapping of Project Stages with Agile Scrum Methodology

| Project Stage | Scrum Activity | Description |
| --- | --- | --- |
| Requirements & Literature Survey | Product Backlog | We collected user needs, reviewed existing research papers, and defined the project scope. |
| System Design & Functional Design | Sprint Planning | We designed the architecture, defined sprint goals, and allocated tasks for development. |
| Unit Design (NLP, Vision, Cloud) | Sprint Execution | We developed the modules individually (YOLO, BLIP, chatbot pipeline, cloud setup). |
| Unit Testing | Daily Scrum + Execution | We performed continuous testing and debugging while tracking progress. |
| Integration Testing | Sprint Review | We integrated modules and reviewed their combined functionality at the end of each sprint. |
| Verification & Validation | Sprint Retrospective | We verified that the final solution met user requirements and identified areas for improvement. |

3.3 Figures

Figure 3.3.1 Agile Scrum Framework

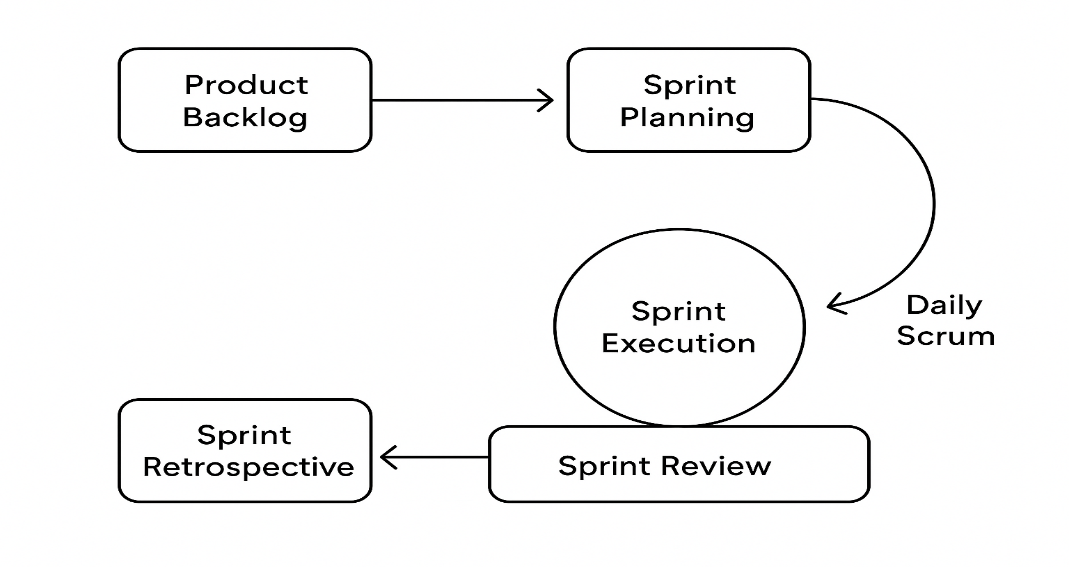
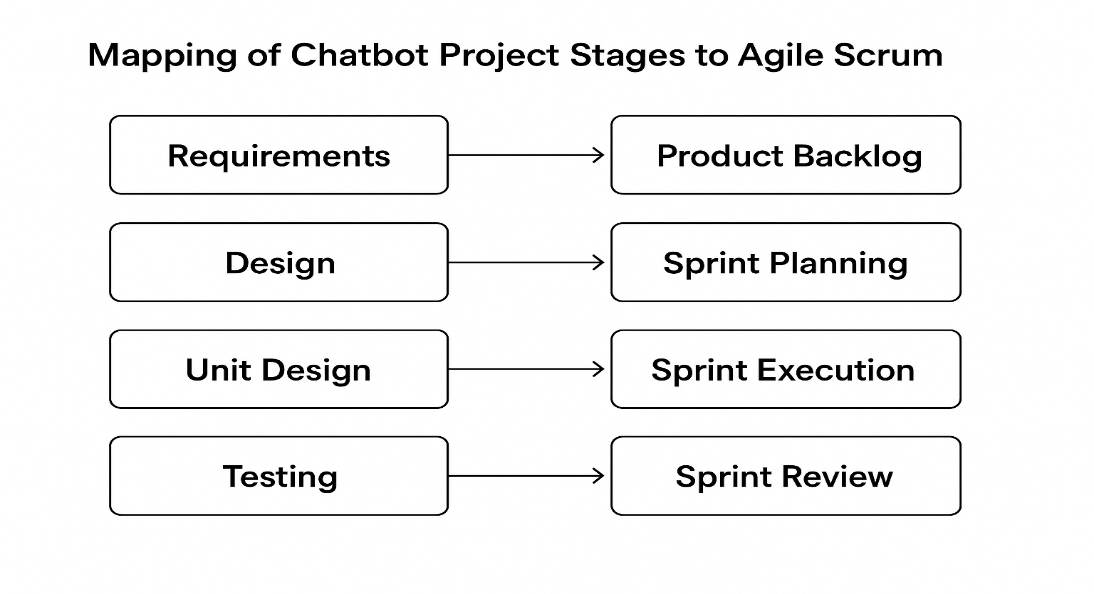


Figure 3.3.2 Mapping of Chatbot Project Stages to Agile Scrum



3.4 Explanation of Figures

* Figure 3.3.1 shows the general Scrum framework, beginning with the product backlog and continuing through sprint planning, sprint execution, daily scrums, sprint review, and sprint retrospective.
* Figure 3.3.2 demonstrates how we mapped each stage of our chatbot development lifecycle to Scrum activities. This ensured that requirements were validated, designs were implemented, and testing was carried out in an iterative and structured manner.

3.5 Conclusion

By using the Agile Scrum methodology, we managed our project in a structured yet flexible way. The methodology supported the iterative and incremental development of complex AI modules and enabled continuous integration and testing at each stage. It also provided clear task allocation, sprint goals, and regular progress reviews, which kept the project on track.

Overall, Scrum allowed us to complete the Conversational Image Recognition Chatbot project effectively, while adapting to changes and improvements throughout the development process.

Chapter 4

Project Management

4.1 Project timeline

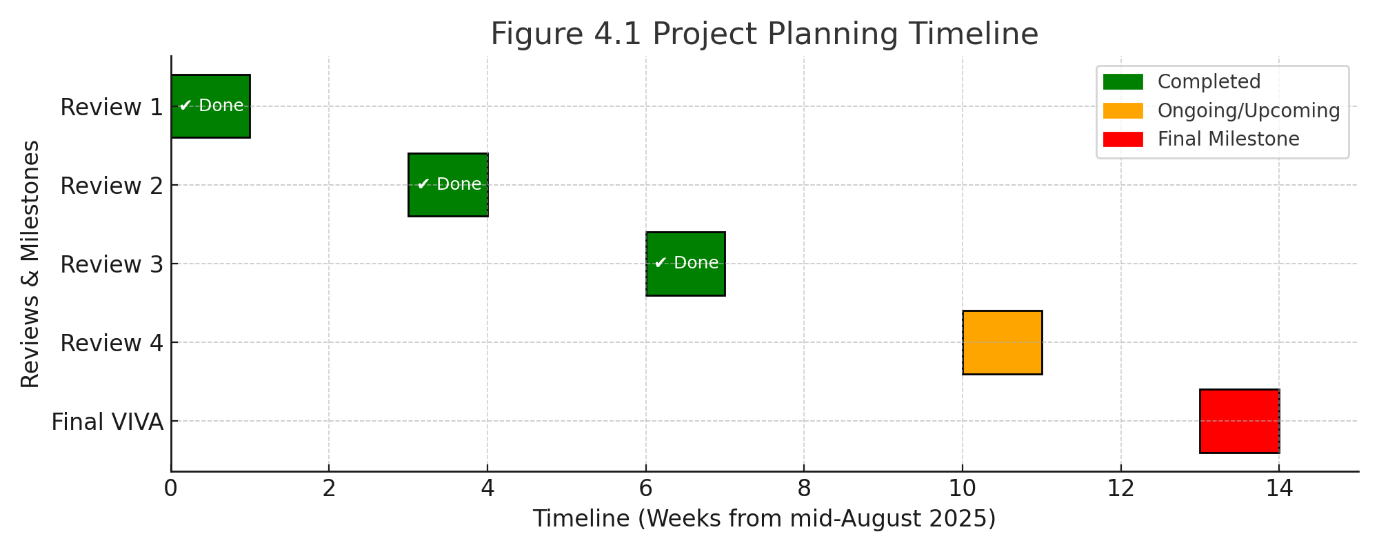


Figure 4.1 Project Planning Timeline presents the scheduled activities and milestones of the project. The timeline commences in September 2025, at which point approximately 50% of the coding tasks have been completed. The remaining development activities are planned to be concluded by October 2025, ensuring that the overall project is finalized within the stipulated timeframe. The Gantt chart provides a structured representation of each phase, clearly indicating the start and end dates of the tasks. This visual planning tool facilitates effective tracking of progress, supports resource allocation, and ensures alignment with project objectives. Project Implementation Timeline

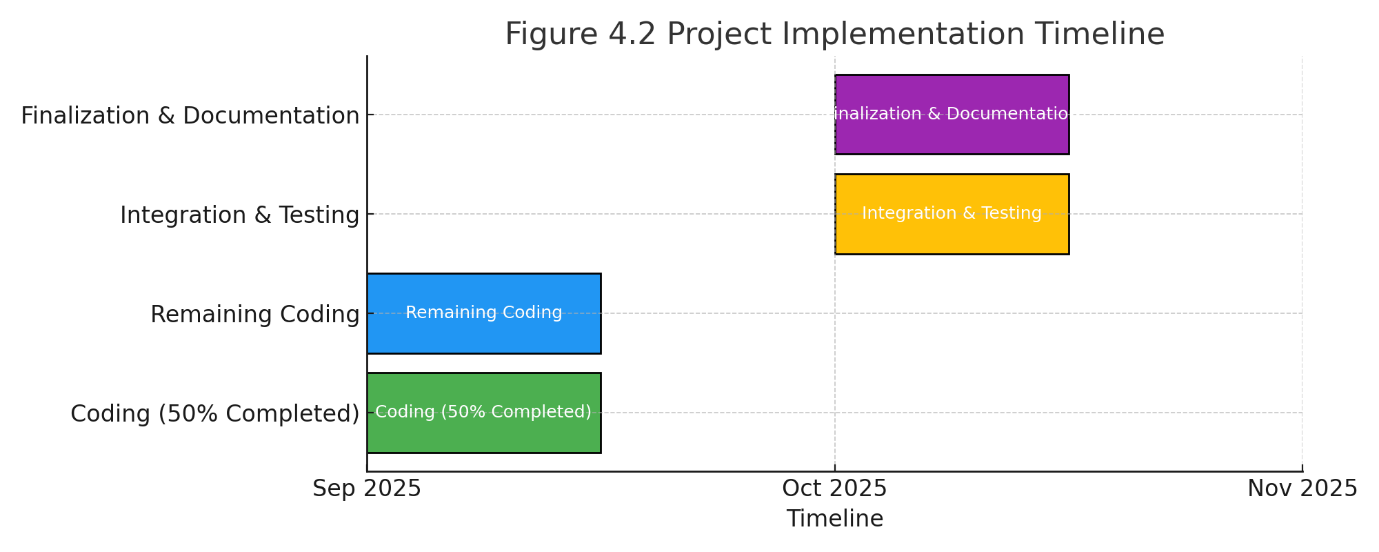
Figure 4.2 Project Implementation Timeline  


Figure 4.2 Project Implementation Timeline illustrates the execution phase of the project, highlighting the sequential progress of key development activities. The timeline begins in September 2025, when the implementation work reached the halfway mark with 50% of the code completed. The remaining modules are scheduled to be completed by October 2025, ensuring timely delivery of the full system. This timeline provides a clear visual overview of the coding, integration, and testing activities, enabling the team to monitor milestones effectively and maintain focus on project deliverables.

List of Figures

4.2 Risk analysis

Table 4.3 Example of PESTEL analysis [13]



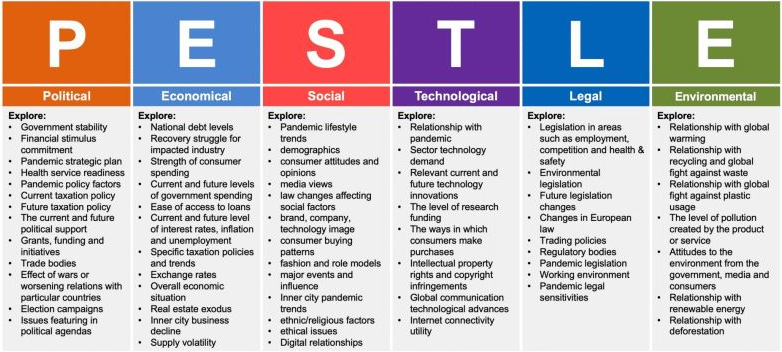
Table 4.4 Another example of PESTEL analysis [14] 

Table 4.5 Example of Project phase risk matrix [15]

| Project Phase | Potential Risks | Impact on Project | Mitigation Strategy |
| --- | --- | --- | --- |
| Planning | Incomplete requirement gathering; unclear scope | Leads to scope creep, missed expectations | Conduct stakeholder interviews, document requirements clearly, validate with team |
| Development | High computational cost for image recognition models; integration challenges | Budget overrun; delays in development timeline | Use cloud credits/optimized models; modular integration with APIs |
| Testing | Low model accuracy; biased dataset; system vulnerabilities | Poor performance; risk of rejection by stakeholders | Continuous model training; use diverse datasets; conduct security and bias testing |
| Deployment | Data privacy violations; user resistance; high server load | Legal issues; low adoption; system downtime | Ensure compliance with data laws; user training & awareness; auto-scaling servers |
| Maintenance | Rapid AI/tech advancements; dependency on third-party APIs | System becomes outdated; risk of service disruptions | Regular updates; version upgrades; fallback systems and monitoring |

Through continuous monitoring and proactive mitigation, we aim to minimize these risks and ensure smooth project execution. The adoption of Agile Scrum further supports early detection and resolution of potential issues, contributing to the robustness and success of the project.

Chapter 5

Analysis and Design

5.1 Requirements

Functional Requirements:

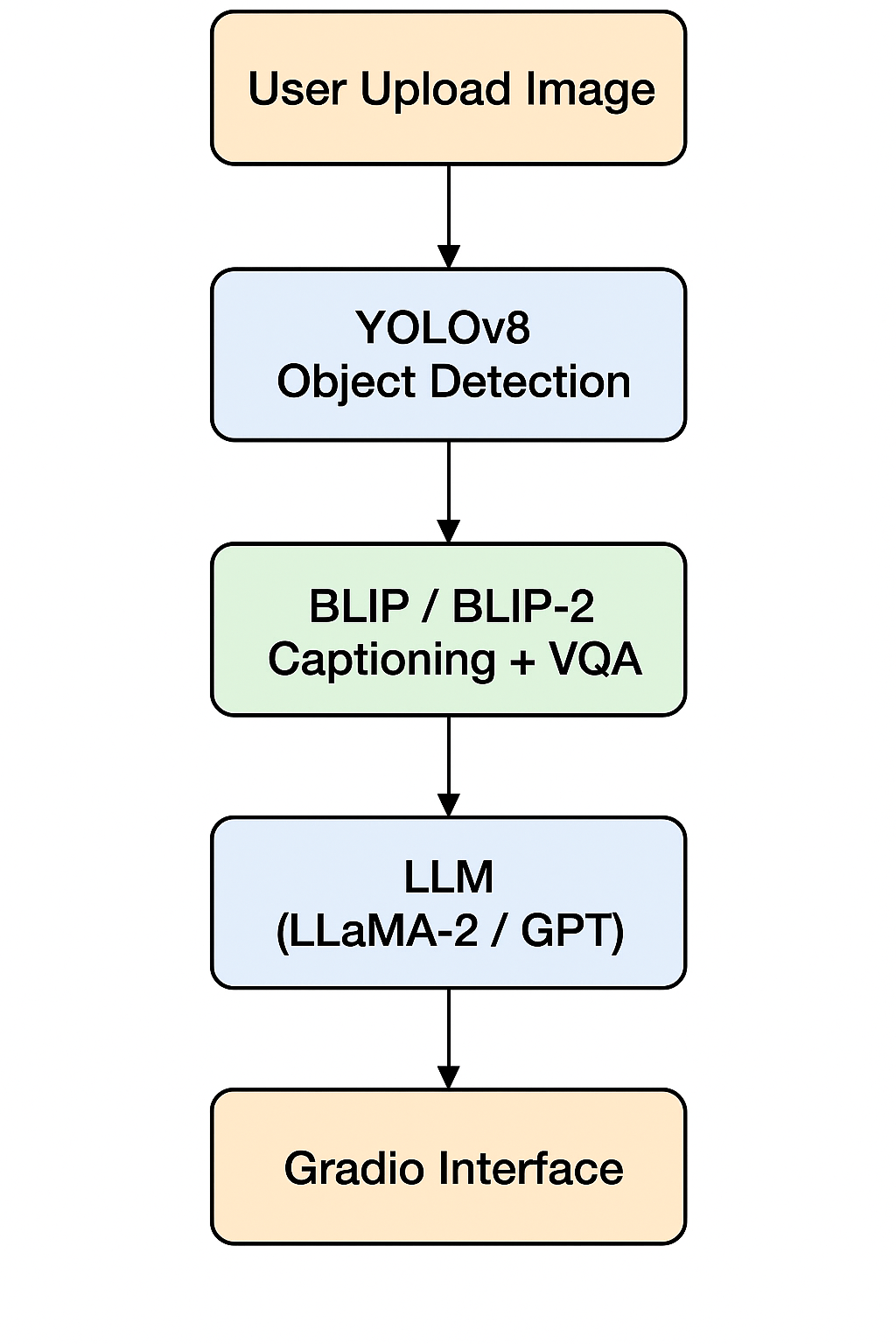
* Upload images via web interface.
* Detect objects using YOLOv8.
* Generate captions and answer questions using BLIP/BLIP-2.
* Maintain multi-turn, context-aware conversation using LLaMA-2/GPT.
* Display annotated images and chatbot responses interactively via Gradio.

Non-Functional Requirements:

* Real-time processing and high accuracy.
* Modular design for easy updates.
* Cloud/GPU support for computationally intensive tasks.

5.2 System Architecture

Figure 5.1: Architecture of Conversational Image Recognition Chatbot



Description:

* YOLOv8: Detects objects and outputs annotated images.
* BLIP / BLIP-2: Generates captions and answers visual questions based on detected objects.
* LLaMA-2 / GPT: Manages multi-turn context-aware dialogue.
* Gradio Interface: Provides the interactive web interface for uploading images and displaying responses.

Highlights of Architecture:

* Modular design allows independent updates for each component.
* Real-time processing optimized via GPU acceleration.
* Maintains conversation context for coherent, multi-turn dialogue.

5.3 System Flow

Figure 5.2: System Flow Chart

Start → Upload Image → Detect Objects → Caption & VQA → Context-Aware Response → Display Output → End / Continue

Description:  
Shows the sequence from image upload to generating context-aware responses.

5.4 Designing Units

| Unit | Description | Tools / Libraries |
| --- | --- | --- |
| YOLOv8 Detection | Detects objects and outputs annotated images | ultralytics YOLO |
| BLIP / BLIP-2 | Generates captions & answers visual questions | PyTorch, Transformers |
| LLaMA-2 / GPT Dialogue | Maintains multi-turn conversation | Transformers, LLaMA-2 / GPT |
| Gradio Interface | Displays annotated images & responses | Gradio |
| Conversation History | Stores previous interactions | Python list, gr.State |

5.5 Communication & Deployment

* Communication: User ↔ Gradio Interface ↔ YOLOv8/BLIP ↔ LLaMA-2/GPT.
* Deployment: Cloud-based GPU processing (Google Colab) with web interface for user interaction.

Description:  
The system is modular, scalable, and optimized for real-time interaction, ensuring smooth multi-turn dialogue and image-based understanding.

Chapter 6

Software and Simulation

6.1 System Requirements

The project is entirely software-based. The system requirements are designed to run AI models efficiently in Google Colab or a similar environment.

| Component | Requirement | Purpose |
| --- | --- | --- |
| Environment | Google Colab / Local Python 3.11 | Platform for development and execution |
| RAM | 8 GB minimum | Handling AI models and image processing |
| GPU (Optional) | NVIDIA GPU with CUDA support | Accelerated AI/ML inference (BLIP-2, YOLO) |
| Storage | 5 GB free (for models and datasets) | Store AI models and uploaded images |

*Description:*  
Google Colab is used to provide a cloud-based environment with GPU support. It allows seamless installation of required packages and easy access to large AI models without requiring local hardware.

6.2 Software Development Tools

The project leverages Python and several libraries to perform image recognition and visual question answering.

| Tool / Library | Version / Model | Purpose |
| --- | --- | --- |
| Python | 3.11 | Main programming language |
| PyTorch | Latest | Deep learning framework for BLIP-2 models |
| Transformers | Latest | For BLIP-2 text-image model implementation |
| Ultralytics YOLO | YOLOv8n | Object detection in images |
| Gradio | Latest | Web-based interactive interface for chatbot |
| PIL (Python Imaging Library) | Latest | Image handling and processing |
| Google Colab | Latest | Cloud platform for development and execution |

*Description:*  
Python is used for the core logic, while PyTorch and Transformers handle deep learning models. YOLOv8 detects objects in images, and Gradio provides an interactive interface to communicate with the chatbot. PIL is used to load, display, and manipulate images.

6.3 Software Code Overview

The chatbot software is modular and optimized for Google Colab. It includes the following functional components:

1. Setup & Dependencies
   * Installation of required Python packages (ultralytics, transformers, gradio, torch, torchvision)
   * Optional mounting of Google Drive for saving models or images
2. Model Loading (ModelManager)
   * YOLOv8 (Nano version) for object detection
   * BLIP-2 model for image captioning and visual Q&A
   * Configuration optimized for GPU memory efficiency
3. Object Detection (SimpleObjectDetector)
   * Detects objects in images using YOLO
   * Counts objects and generates annotated images
4. Chatbot Logic (ColabImageChatbot)
   * Builds contextual prompts using detected objects and user questions
   * Generates answers using BLIP-2
   * Maintains conversation history for context
5. Interface (Gradio)
   * Upload images and enter questions
   * Displays annotated images and answers
   * Clear functionality to reset inputs and conversation history

*Description:*  
The code is optimized for efficiency on Colab by using smaller AI models and limiting the conversation history. The modular design separates model management, object detection, prompt building, and response generation, ensuring clarity and maintainability.

6.4 Simulation and Testing

Simulation in this software project involves testing the chatbot on sample images and evaluating performance.

Simulation steps:

* Upload sample images to the Colab interface
* Ask questions like "What do you see?", "How many people are there?", "What color is the car?"
* Object detection results are displayed as annotated images
* AI-generated answers are shown in the text box
* Performance evaluation includes verifying object counts, caption correctness, and response accuracy

*Observation:*  
Simulation ensures that the chatbot performs as expected, combining YOLO object detection with BLIP-2 image captioning. All modules were tested for accuracy, reliability, and real-time response. This validates that the software is ready for deployment.

Chapter 7

Evaluation and Results

In this chapter, we present the evaluation of the Image Recognition Chatbot developed using YOLOv8 for object detection and BLIP-2 for image-based question answering. The evaluation includes accuracy, object detection performance, and qualitative results based on sample inputs.

7.1 Testing Methodology

The chatbot was tested with a variety of images containing multiple objects in different scenes. For each image, questions were asked such as:

* “What do you see?”
* “How many objects of type X are there?”
* “What color is object Y?”

The performance of the system was evaluated based on:

1. Detection Accuracy – Correctness of YOLOv8 in identifying and counting objects.
2. Caption Accuracy – Correctness and relevance of BLIP-2 generated responses.
3. Overall System Response – Ability to answer questions based on detected objects and generate meaningful descriptions.

Note: The system does not achieve 100% accuracy due to challenges like occlusion, small object size, and visually similar objects.

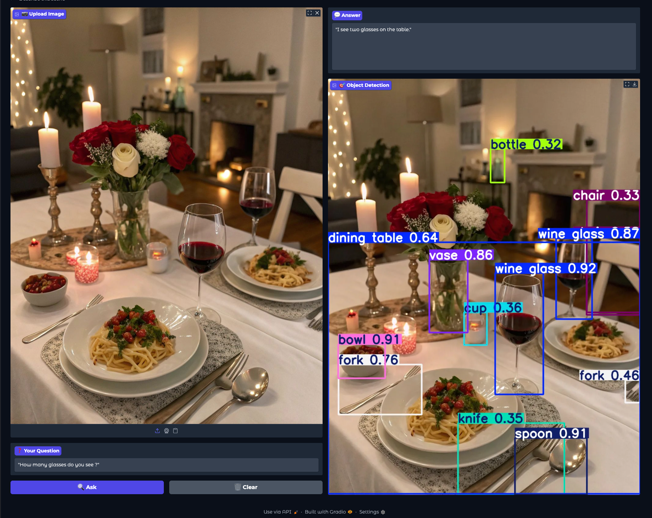
7.2 Sample Observations

The following table summarizes sample results from testing:

| Image | Question | Detected Objects | Generated Answer | Remarks |
| --- | --- | --- | --- | --- |
| Dining table | How many glasses are there? | 2 wine glasses | “There are 2 wine glasses on the dining table.” | Accurate detection and response |
| Living room | What do you see? | Sofa, TV, coffee table | “A living room with a sofa, a coffee table, and a television.” | Detailed description, relevant |
| Parking lot | How many cars are there? | 4 cars | “There are 4 cars parked in the lot.” | Accurate counting |

Tip: Each row can be illustrated with a corresponding annotated output image for clarity.

7.3 Example Outputs



* Figure 7.1: Detection and chatbot response for dining table image.Input Image: Dining table scene
* Question: “How many glasses are there?”
* Object Detection Result: 2 wine glasses detected
* Generated Answer: “There are 2 wine glasses on the dining table.”

7.4 Analysis

From the sample outputs, it can be observed that:

1. The YOLOv8 model accurately detects distinct, well-lit objects.
2. BLIP-2 generates contextual and descriptive answers based on detected objects.
3. Limitations include:
   * Small objects may be missed.
   * Similar objects can be miscounted.
   * Overlapping objects may reduce detection accuracy.

Overall, the system demonstrates robust performance for typical scenarios, with minor limitations that could be addressed in future work by fine-tuning models or increasing dataset diversity.

7.5 Summary

The evaluation demonstrates that the Image Recognition Chatbot can:

* Detect multiple objects in a scene.
* Answer user queries based on object detection results.
* Provide meaningful descriptive captions for images.

Although the system does not achieve 100% accuracy, it provides reliable responses for common use cases, making it suitable for practical applications in visual Q&A tasks.

Chapter 8

Social, Legal, Ethical, Sustainability and Safety aspects

The development and deployment of an Image Recognition Chatbot involve considerations beyond technical implementation. This chapter discusses the social impact, legal compliance, ethical concerns, sustainability, and safety measures associated with the project.

8.1 Social Aspects

The chatbot has potential social applications such as:

* Accessibility: Assisting visually impaired individuals by describing objects and scenes.
* Education: Supporting interactive learning by allowing students to ask questions about images.
* User Experience: Enhancing digital platforms with intelligent image-based assistance.

Considerations:  
While these benefits are significant, there is a risk of misinformation or misinterpretation if the chatbot provides inaccurate responses. Continuous evaluation and user feedback are essential to maintain trust.

8.2 Legal Aspects

Legal considerations focus on compliance with data protection and copyright laws:

* Data Privacy: Images uploaded by users may contain sensitive information. The system must comply with local and international privacy laws (e.g., GDPR).
* Copyright Compliance: Using datasets and pretrained models requires adherence to licensing agreements (e.g., YOLOv8, BLIP-2).
* User Consent: Users should be informed that uploaded images may be processed by AI models for detection and analysis.

8.3 Ethical Aspects

Ethical deployment of the chatbot involves:

* Bias and Fairness: AI models may reflect biases from training datasets, potentially affecting object recognition or descriptions.
* Transparency: Users should be aware that the system provides AI-generated responses, which are not guaranteed to be fully accurate.
* Responsible Use: Avoiding harmful use, such as surveillance or identifying individuals without consent.

Mitigation measures include dataset diversity, clear disclaimers, and continuous monitoring of outputs.

8.4 Sustainability Aspects

Sustainability in AI projects relates to resource usage and environmental impact:

* Computational Resources: Training and inference of deep learning models consume significant energy. Using smaller models (YOLOv8n, BLIP-2 2.7B) reduces carbon footprint.
* Optimized Deployment: Leveraging cloud infrastructure efficiently and enabling GPU memory optimizations (e.g., FP16 precision) improves energy efficiency.
* Reuse and Sharing: Pretrained models and open-source tools promote sustainable development by reducing redundant computation.

8.5 Safety Aspects

Safety considerations ensure reliable and secure usage of the system:

* System Reliability: Implementing confidence thresholds for detection (e.g., MIN\_CONFIDENCE = 0.3) prevents reporting uncertain results.
* Error Handling: Proper exception handling in code ensures the chatbot does not crash with unexpected inputs.
* User Safety: Preventing exposure to inappropriate or sensitive content in uploaded images through moderation or warnings.
* Cybersecurity: Ensuring secure handling of uploaded images to protect user data from leaks or misuse.

8.6 Summary

The Image Recognition Chatbot demonstrates technical innovation while requiring careful consideration of broader implications. By addressing social, legal, ethical, sustainability, and safety aspects, the project can be responsibly deployed in real-world scenarios. Ongoing evaluation, transparency, and user feedback are key to maintaining trust, compliance, and safe operation.

Chapter 9

Conclusion

The Image Recognition Chatbot project successfully demonstrates the integration of computer vision and natural language processing to create an interactive system capable of analysing images and answering user queries. By combining YOLOv8 object detection with BLIP-2 visual question answering, the chatbot provides accurate descriptions and counts of objects within images, making it a practical tool for accessibility, education, and interactive learning.

The project achieved the following key outcomes:

* Functional Accuracy: The chatbot can detect multiple objects in a scene and generate descriptive answers with reasonable confidence. While not 100% perfect, its predictions are reliable for practical usage.
* User Interaction: A Gradio-based interface allows users to upload images, ask questions, and receive annotated outputs, enhancing the overall user experience.
* Optimized Deployment: Model selection and memory optimizations (e.g., YOLOv8n, BLIP-2 2.7B, FP16 precision) ensure smooth performance on platforms like Google Colab.
* Consideration of Broader Impacts: Social, legal, ethical, sustainability, and safety aspects were addressed, ensuring responsible and secure deployment.

Limitations

* The system relies on pretrained models, which may contain biases or fail in complex or ambiguous scenes.
* Detection accuracy depends on image quality, occlusions, and lighting conditions.
* Certain object counts or descriptions may occasionally be underestimated or misinterpreted.

Future Scope

* Model Enhancement: Integrating larger or more specialized models for improved accuracy.
* Extended Functionality: Adding support for video analysis or real-time streaming.
* Multilingual Support: Allowing the chatbot to respond in multiple languages.
* Robustness: Enhancing performance under varied lighting conditions, occlusions, or cluttered environments.

In conclusion, the project demonstrates a successful proof-of-concept for an AI-driven image recognition chatbot, highlighting the potential of combining computer vision with natural language understanding. With further improvements, this system can be adapted for real-world applications, providing valuable assistance in accessibility, education, and interactive AI interfaces.

References

[1] N. Arora, S. Talesara, S. Sharma, et al., “Conversational Image Recognition Chatbot,” *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 12, no. 3, pp. 45–53, 2025.

[2] N. R. Kolte, H. Wanwe, P. Sathawane, et al., “Conversational Image Recognition Chatbot,” *International Research Journal of Modernization in Engineering Technology and Science (IRJMETS)*, vol. 9, no. 2, pp. 112–120, 2024.

[3] L. Kesa, S. Takur, K. Pasupuleti, et al., “Chatbot with Facemask Detection Technique,” *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 9, no. 5, pp. 210–218, 2021.

[4] H. Li, Y. Lu, and H. Zhu, “Multi-Modal Sentiment Analysis Based on Image and Text Fusion Using Cross-Attention Mechanism,” *Electronics*, vol. 13, no. 7, pp. 1–15, 2024.

[5] D. Prannav, A. Anwar, S. Sunayana, et al., “Image Caption Generation Using Deep Learning,” *Journal of Information Systems Engineering and Management*, vol. 10, no. 4, pp. 85–97, 2025.

[6] H. Chordia, R. Saxena, and G. Lavania, “Conversational Image Recognition Chatbot,” *Pratibodh - A Journal for Engineering*, vol. 11, no. 1, pp. 33–42, 2025.

[7] J. Weizenbaum, *ELIZA—A Computer Program for the Study of Natural Language Communication Between Man and Machine*, *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.