**Introduction**

In recent years, the importance of road safety has gained significant attention due to the increasing number of accidents caused by poor road infrastructure. Among these challenges, unmarked or poorly maintained road bumps and humps pose a serious risk to drivers and passengers alike. These unexpected irregularities can lead to vehicle damage, traffic congestion, and even severe accidents, especially at night or in adverse weather conditions. To address this problem, our project, **Bump Safe Vision**, proposes a machine learning-based solution for the automated detection and mapping of road bumps and humps.

**Bump Safe Vision** aims to leverage computer vision and machine learning techniques to detect bumps in real-time using camera-based inputs from vehicles or smartphones. By processing images through trained models, our system identifies the presence and location of road bumps with high accuracy. Once detected, these bumps are geotagged using GPS data and marked on a digital map interface, with the goal of integrating this information into widely-used navigation platforms like Google Maps. This integration would allow drivers to receive advance warnings and adjust their driving behaviour accordingly, improving both safety and comfort.

The proposed system has strong applications in smart city development, urban planning, autonomous driving, and transport infrastructure maintenance. By creating a scalable and cost-effective solution that can be deployed via dashcams, mobile devices, or municipal survey vehicles, **Bump Safe Vision** contributes to building safer, more intelligent roads. The project also aligns with the larger vision of using Artificial Intelligence (AI) and the Internet of Things (IoT) to enhance public safety and transportation efficiency.

**Identification of Need**

India, like many other developing countries, faces persistent challenges with road infrastructure maintenance, particularly in urban and semi-urban areas. Road bumps and humps are commonly used to control vehicle speed and enhance pedestrian safety. However, many of these are poorly designed, unmarked, or not visible due to faded paint or environmental conditions such as rain and fog. This lack of visibility and standardized placement increases the risk of accidents, vehicle wear and tear, and reduced ride comfort—especially for two-wheelers and elderly passengers.

Despite advancements in road safety technologies, there is still no widely available system for automatically detecting and mapping road surface irregularities in real-time. Traditional manual inspections are time-consuming, labor-intensive, and lack the scalability needed for large-scale deployment. Additionally, existing navigation systems like Google Maps do not provide real-time alerts for road conditions such as unexpected bumps, which can be crucial for both regular and commercial drivers.

With the rapid adoption of smartphones, dashcams, and affordable AI technologies, there is a growing opportunity to develop low-cost, intelligent systems capable of recognizing and reporting such road hazards. The **need** for a solution like **Bump Safe Vision** arises from this technological gap—a system that not only detects road bumps using machine learning and computer vision but also geotags them for integration into navigation platforms.

**Problem Statement**

Road safety remains a critical issue in India and many other countries where infrastructure development struggles to keep pace with rapid urbanization. Among the various challenges that compromise road safety, the presence of **unmarked or poorly maintained road bumps and humps** is particularly hazardous. These irregularities, intended to control speed, often become **threats** when they are not clearly visible or standardized. Drivers encountering sudden or hidden bumps may lose control, damage their vehicles, or even cause accidents—especially during night-time travel, adverse weather conditions, or in unfamiliar areas.

Traditional solutions for identifying and maintaining road bumps rely heavily on **manual inspections** or **citizen reports**, which are both time-consuming and inefficient for large-scale application. Moreover, navigation systems like Google Maps and Waze, while advanced in route guidance, lack **real-time road surface information**, leaving drivers unaware of such critical road features. This gap in real-time, location-specific road quality data is a significant limitation in the current transportation ecosystem.

Given the widespread availability of cameras in smartphones and vehicles, and the growing power of machine learning and computer vision technologies, there is an opportunity to build a scalable, automated system that can detect and report road bumps accurately. However, there are several challenges involved—such as differentiating road bumps from shadows or potholes, managing variations in lighting, angles, and road textures, and integrating this data with geolocation and mapping platforms.

**Preliminary Investigation**

Before initiating the design and development of the **Bump Safe Vision** system, a comprehensive preliminary investigation was conducted to understand the current landscape of road safety solutions, existing technologies, and the feasibility of implementing a bump detection system using machine learning and computer vision.

The investigation began with a review of current road monitoring practices used by municipal authorities. It was found that most cities rely on **manual surveys** or **citizen complaints** to identify road defects like potholes and unmarked speed breakers. These methods are not only inefficient but also lack real-time responsiveness, especially in densely populated or rapidly expanding urban areas.

Further research was carried out on existing technologies used for road anomaly detection. Some commercial systems use **LiDAR, accelerometers, or vehicle-mounted sensors**, but these setups are expensive and not feasible for widespread deployment, especially in developing countries. On the other hand, the increasing availability of **camera-equipped smartphones and vehicles** provides a low-cost, scalable alternative when combined with machine learning and image processing.

We also explored relevant academic literature and case studies on object detection models such as **YOLO (You Only Look Once)** and **SSD (Single Shot Detector)**, which are capable of real-time object recognition with high accuracy. These models can be trained to recognize road bumps based on visual features in captured images or video frames. Additionally, the use of **GPS modules** for geotagging enables detected bumps to be mapped with precision, making the data useful for navigation systems and road maintenance authorities.

To validate the feasibility, an initial dataset of road images with bumps and humps was collected from various online sources and tested using pre-trained models. Early results indicated that with appropriate data augmentation and model tuning, reliable detection was achievable. This confirmed the technical viability of the proposed solution and guided the decision to move forward with model training, field data collection, and system integration.

**System Analysis**

**1. Business Need**

India’s roads often suffer from inconsistent infrastructure quality, especially in Tier-2 and Tier-3 cities. A major challenge lies in the lack of real-time, data-driven systems to identify and map **road anomalies**, particularly **unmarked or poorly maintained speed bumps and humps**. These obstacles, while meant to improve safety, often have the **opposite effect** when they are not clearly visible or documented in navigation systems. This results in **vehicle damage**, **sudden braking**, **driver discomfort**, and even **road accidents**, leading to increased maintenance costs for vehicle owners and reduced trust in local infrastructure.

From a business perspective, municipal corporations and urban development authorities **lack a scalable, low-cost, and automated solution** to monitor road quality. Manual inspections are time-consuming and labor-intensive, often leading to **delayed maintenance** and **inefficient resource allocation**. On the other hand, ride-sharing platforms, logistics companies, and commercial vehicle fleets face significant risks due to unpredictable road conditions.

With growing investments in **smart cities and intelligent transportation systems (ITS)**, there is a **clear need** for a system that can:

* Detect and map road irregularities in real-time.
* Improve navigation systems with detailed road condition data.
* Assist civic bodies in planning timely maintenance.
* Enhance safety and experience for everyday commuters.

**2. Objective**

The primary objective of the **Bump Safe Vision** project is to design and develop an **automated, real-time road bump detection system** using **machine learning and computer vision**, and to integrate the detected bumps with **GPS-based mapping** tools like **Google Maps**. This will provide drivers with advance warnings and help local authorities maintain road quality more efficiently.

Specific objectives include:

* To collect and prepare a dataset of road images and video clips showing road bumps and humps under various conditions.
* To develop a computer vision model (e.g., using YOLO or SSD) capable of accurately detecting road bumps in real-time.
* To implement a GPS module to geotag the detected bumps.
* To visualize detected bumps on a digital map interface.
* To make the solution lightweight and scalable for deployment via smartphones, dashcams, or survey vehicles.

**3. Proposed System**

The **proposed system** is a modular application comprising four main components:

1. **Data Collection Module**: Captures road video footage using smartphones or vehicle dashcams. This data includes frames with various road surface conditions.
2. **Detection Module**: Uses a trained machine learning model to process images and detect road bumps and humps. This model is based on convolutional neural networks (CNNs), particularly real-time detectors like YOLOv5 or SSD.
3. **GPS Integration**: Geotags each identified bump using GPS coordinates from the device capturing the footage.
4. **Visualization & Mapping**: Displays the detected and geotagged bumps on a digital map interface. The goal is to integrate this data with Google Maps or similar platforms using APIs for user alerts and navigation.

The system will be tested in real-world scenarios and optimized for efficiency, minimizing false positives and negatives.

**4. Benefits**

The implementation of **Bump Safe Vision** brings a wide range of benefits to various stakeholders:

**For Drivers and General Public:**

* Improved **driving safety** by getting real-time alerts about upcoming road bumps.
* Enhanced **ride comfort** and reduced risk of accidents or vehicle damage.
* Better route selection based on road condition data.

**For Government and Municipal Authorities:**

* Real-time and crowdsourced data to identify **problematic road zones**.
* Reduced dependence on **manual inspections**.
* Improved planning and budgeting for **infrastructure maintenance**.

**For Businesses and Logistics:**

* Smarter route planning and fewer delays due to **unexpected road conditions**.
* Lower maintenance and operational costs.
* Competitive advantage through **fleet safety optimization**.

**Environmental and Economic Impact:**

* Reduced fuel consumption from smoother driving patterns.
* Data-driven decision-making leading to **long-term infrastructure sustainability**.

1. **Feasibility Study**

To ensure the success and practicality of implementing **Bump Safe Vision**, a feasibility study was conducted considering **operational**, **technical**, and **economic** aspects. The goal is to validate whether the project can be realistically developed, deployed, and sustained with available resources and technology.

**1. Operational Feasibility**

Operational feasibility assesses whether the proposed system will function effectively in real-world environments and be accepted by its intended users. The **Bump Safe Vision** system is designed to work with widely available devices such as **smartphones, dashcams, or vehicle-mounted cameras**, making it easily deployable. It can operate in real-time or batch mode, depending on use case (e.g., live navigation alerts vs. post-drive analysis).

The system also offers value to a wide range of users:

* **Drivers** benefit from bump alerts and safer routes.
* **Government bodies** receive reliable data for road maintenance.
* **Logistics and transportation companies** reduce risks and costs.

With no major changes required in user behavior and a simple user interface for map visualization, the solution is **highly operable** and fits into existing workflows.

**2. Technical Feasibility**

Technically, the project leverages **well-established technologies**:

* **Computer Vision and ML frameworks** (e.g., YOLOv5, OpenCV, TensorFlow/PyTorch).
* **GPS modules** or smartphone location services for geotagging.
* **Mapping APIs** (like Google Maps) for data visualization.

Hardware requirements are minimal: a standard smartphone camera or low-end dashcam is sufficient for data collection. ML models can be optimized for **real-time inference on edge devices**, or run on cloud infrastructure for higher scalability.

Moreover, the development tools and libraries are **open-source** and well-documented, reducing technical complexity and increasing project viability. Thus, from a development and deployment standpoint, the system is **technically feasible**.

**3. Economic Feasibility**

The system is designed to be **cost-effective**:

* No need for specialized hardware (uses existing cameras and phones).
* Open-source software eliminates licensing costs.
* Cloud storage and computing, if needed, can be scaled based on demand.

Initial costs involve **model training**, **system integration**, and **basic infrastructure** for data storage and visualization. However, since the core system can be built using free tools and frameworks, the overall development cost remains low.

In the long term, the system can save **millions in road repairs, accident claims, and vehicle damage** costs by helping drivers avoid unexpected hazards and enabling better infrastructure planning.

Therefore, the project is **economically feasible**, especially for urban safety initiatives, smart city projects, and scalable transportation solutions.

**Tools and Platform**

The development of the **Bump Safe Vision** system relies on a combination of modern tools, libraries, and platforms that support efficient machine learning, computer vision, geolocation, and user interface development. The selection of these tools ensures that the system is scalable, cost-effective, and compatible with real-world deployment scenarios.

**1. Programming Language**

* **Python**  
  Python is the primary programming language used for this project due to its extensive support for machine learning, data science, and computer vision libraries. It enables rapid prototyping and integration of different components.

**2. Machine Learning & Computer Vision Libraries**

* **OpenCV**  
  Used for image and video processing tasks such as frame extraction, preprocessing, and visualization.
* **YOLOv5 / YOLOv8 (You Only Look Once)**  
  A real-time object detection model used to detect road bumps and humps with high accuracy and low latency.
* **TensorFlow / PyTorch**  
  Frameworks used for training and deploying deep learning models depending on compatibility and speed.

**3. Data Annotation Tools**

* **LabelImg**  
  An open-source image annotation tool used to manually label images with bounding boxes for training the object detection model.

**4. Development Environment**

* **Jupyter Notebook**  
  For experimentation, data visualization, and model training/testing.
* **Google Colab**  
  Cloud-based Python notebook environment with GPU support for faster model training without local system limitations.
* **Visual Studio Code (VS Code)**  
  Integrated development environment used for writing, testing, and debugging code.

**5. Geolocation and Mapping**

* **Google Maps API / Leaflet.js**  
  Used to visualize the GPS coordinates of detected bumps on a digital map. This helps users navigate roads more safely and assists authorities in road maintenance.
* **GPS Module / Smartphone GPS**  
  Collects real-time location data of detected road anomalies.

**6. Platform**

* **Windows or Linux OS**  
  The system is platform-independent but primarily developed on a Windows machine. It can also be deployed on Linux servers or Android devices for mobile applications.
* **Cloud (Optional)**  
  Google Cloud Platform (GCP) or Amazon Web Services (AWS) may be used for model hosting, storage, and scalable API deployment if needed for production.

**7. Version Control and Collaboration**

* **Git & GitHub**  
  Used for source code management, version control, and collaboration.

**GANTT CHART**

**Software Requirement Specification (SRS)**

**Project Title**: Bump Safe Vision

**Domain**: Road Safety, Machine Learning, Computer Vision, GIS

**1 Document Convention**

* Bold text represents section titles.
* Italics represent emphasis on key elements.
* Bullets (•) are used for listing features or points.
* Code or parameters are written as inline code.

**2 Intended Audience and Reading Suggestion**

This document is intended for:

* Project Guides and Reviewers – to assess scope, feasibility, and structure.
* Developers and Engineers – for system design, implementation, and integration.
* Researchers/Students – seeking to understand ML applications in road safety.

**3 Project Scope**

Bump Safe Vision is an AI-based road hazard detection system that uses computer vision and GPS mapping to identify and mark speed breakers and potholes. The system detects anomalies from live or recorded road footage and geotags them on a map for driver awareness and government maintenance planning.

Key Goals:

• Reduce accidents and vehicle damage.

• Provide real-time alerts to drivers.

• Help city authorities plan road repairs.

**4 Functional Requirements**

Image/Video Input Handling – Accepts video or live feed from a dashcam/mobile.

Object Detection – Detects bumps/humps using ML model (e.g., YOLOv5).

Location Tagging – Uses GPS to fetch real-time coordinates.

Data Logging – Saves detection data (image, time, location) in a database.

Map Visualization – Shows hazard locations on a digital map (Google Maps).

User Notification – Sends alerts or displays markers near hazardous roads.

Manual Feedback – Allows users to confirm or deny detections.

**5 Non-Functional Requirements**

• Performance: Real-time processing with <1 sec delay.

• Reliability: 90%+ detection accuracy with minimal false positives.

• Scalability: Support for edge deployment or cloud hosting.

• Usability: User-friendly interface with minimal training.

• Portability: Can run on Android, PC, or embedded hardware.

• Security: Data should be stored securely; location sharing must be encrypted.

**6 Output Requirements**

• Detection Reports – List of timestamps and GPS-tagged images/videos.

• Mapped Interface – Interactive map with hazard markers.

• Log Files – JSON/CSV files for all detections.

• Alerts/Warnings – On-screen notifications or voice alerts (optional).

**7 Other Non-Functional Requirements**

• Maintainability: Code should follow modular design for easy updates.

• Extensibility: Should allow integration of additional features (e.g., pothole severity detection).

• Accessibility: Interface should support basic accessibility standards.

• Battery Optimization: Efficient for mobile use without heavy drain.

**8 Product Perspective**

Bump Safe Vision is a standalone system but can also function as a module in larger Smart City or Navigation systems. It relies on:

• Hardware (camera + GPS module or mobile phone).

• Software (ML model + location services + mapping API).

It is part of a broader ecosystem aiming to improve road safety through intelligent automation.

**9 Product Features**

• AI-based Road Bump Detection

• Real-Time Geolocation Tracking

• Map-based Visualization of Hazard Zones

• Open-Source and Low Cost

• User Feedback Loop for Accuracy Improvement

**10 User Classes & Characteristics**

Drivers (General Users)

• Require map-based interface, alert system.

• Use mobile-based version or in-car integration.

Government Authorities

• Use data exports for road maintenance planning.

• Need access to bulk reports, heatmaps.

Developers/Researchers

• Use source code, APIs, and logs for improvement or extension.

**11 Resources Used**

• Python (Programming)

• YOLOv5 / OpenCV (Object Detection)

• Google Coolab / Jupyter Notebook (Model Training)

• Labelling (Image Annotation)

• Google Maps API (Mapping)

• SQLite / Firebase (Storage & Logs)

• Web Interface (UI)

**12 Assumptions & Dependencies**

• A clear view of the road is available via camera.

• GPS signal is strong and accurate.

• ML model is trained on varied bump/hump images.

• Internet is required for map visualization and cloud logging.

• The environment (e.g., weather, lighting) does not severely affect visibility.

**Modules**

1. **Data Collection Module**

Responsible for gathering real and AI-generated images of roads with and without speed bumps from multiple sources like Kaggle, Gemini, and Copilot.

1. **Data Preprocessing Module**

Handles cleaning, resizing, augmenting, and normalizing images to improve the quality and consistency of the dataset for training.

1. **Data Annotation Module**

Involves labelling and tagging images manually or using automated tools to classify roads as "with speed bumps" or "without speed bumps."

1. **Model Training Module**

Focuses on developing and training a deep learning model (e.g., CNN) using the prepared dataset to recognize speed bumps in road images.

1. **Model Evaluation Module**

Evaluates model performance using metrics such as accuracy, precision, recall, and confusion matrix to ensure effective classification.

1. **Model Optimization Module**

Applies fine-tuning techniques like hyperparameter tuning, transfer learning, and regularization to improve model accuracy and efficiency.

1. **GPS Mapping Module**

Integrates GPS data with detected speed bumps and prepares the data for visualization on digital maps for better navigation insights.

1. **Web Application Module**

Develops a user-friendly web interface using Flask/React to allow users to upload road images and view detection results.

1. **Map Visualization Module**

Implements interactive maps using Google Maps API or Leaflet.js to display detected speed bumps with markers and other relevant data.

1. **Image Upload and Processing Module**

Enables users to upload road images via the web or mobile application and process them in real time to detect speed bumps.

1. **Security and Authentication Module**

Ensures secure access to the web and mobile application by implementing authentication methods such as JWT (JSON Web Tokens) and role-based access control (RBAC).

1. **Database Management Module**

Stores processed images, detection results, GPS coordinates, and user data securely using databases like MySQL, PostgreSQL, or Firebase for efficient retrieval and analysis.

Modules done by me

**9. System Design**

Designing a system like **Bump Safe Vision** requires a structured approach that ensures every component functions efficiently and integrates seamlessly. This section outlines the software process model chosen, the overall system design approach, and two complementary design methodologies: Top-Down and Bottom-Up design.

**9.1 Software Process Model**

For this project, the **Iterative Incremental Model** is adopted. This model is best suited for AI/ML-based systems where continuous refinement and feedback play a critical role.

**Why Iterative Incremental Model?**

* Allows **early development** of core features such as object detection.
* Each module (e.g., detection, GPS tagging, map visualization) is built incrementally.
* Continuous testing ensures **real-time improvements** and bug fixes.
* Perfect for **machine learning models**, which often need multiple training iterations.

**9.2 System Design Approach**

The system follows a **modular and layered design** to ensure flexibility, easy maintenance, and scalability. The approach breaks the application into independent functional units or modules that communicate through interfaces.

**Design Goals:**

* Maintain clear separation between logic layers.
* Enable parallel development of modules like detection, GPS tracking, and UI.
* Ensure that core AI models can be trained and replaced without disrupting the system.

**Major Layers:**

* **Input Layer** – Captures video/images from the camera.
* **Processing Layer** – Performs object detection and applies ML models.
* **Data Layer** – Handles GPS tagging, logging, and storage.
* **Interface Layer** – Displays results on a map or through alerts.

**9.3 Top-Down Design**

In this approach, the system is broken down from the top-level functionality into smaller and more manageable sub-modules. It focuses first on **what the system should do** before diving into how it is implemented.

**Application in Project:**

* Started by identifying the primary objective: detecting and tagging road bumps.
* Defined **core functionalities** such as input handling, object detection, GPS logging, map visualization.
* Each function was broken into sub-functions (e.g., detection → frame capture, model loading, confidence filtering).