



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: <https://doi.org/10.22214/ijraset.2026.85277>

www.ijraset.com

Call: ☎ 08813907089

E-mail ID: ijraset@gmail.com

Smart Tourist Safety System: Integrating Machine Learning, Spatial Data, and Human-Centred Emergency Workflows

S. Jadesha Gowda¹, T. Dinesh Reddy², Sharath Bhoopal B J³, Chanabasava G⁴, Shashikanth Reddy⁵

Department of Computer Science and Artificial Intelligence, Ballari Institute of Technology and Management, Ballari, Karnataka, India

Abstract: Tourists visiting unfamiliar locations may face difficulties identifying nearby emergency services, communicating their location, and obtaining timely information about the status of an emergency request. The Smart Tourist Safety System (STSS) is designed to address these challenges through a software platform that combines location-aware services, emergency-service directories, machine-learning models, and responder communication workflows. The system includes district-level crime-risk estimation, audio-based distress classification, and natural-language intent recognition. It also uses geospatial calculations to support distance estimation and location-based processing.

According to the available project audit, the calibrated crime-risk model achieved 82.54% accuracy, while the TF-IDF and LinearSVC intent classifier achieved 84.92% test accuracy with a macro F1-score of 0.8437. The audio distress classifier reported 37.72% precision and 54.43% recall at a distress threshold of 0.80. At this threshold, the reported number of false positives decreased from 791 to 71. These results illustrate both the potential and the limitations of applying machine learning to tourist safety. In particular, prediction errors, incomplete geographic data, and differences between offline model artifacts and active application components must be considered. STSS is therefore presented as a developing decision-support platform rather than a proven replacement for emergency services. Further reproducible evaluation and real-world testing are required before its effectiveness can be established.

Keywords: Tourist safety, emergency response, machine learning, natural-language processing, audio classification, geospatial computing, location-based services.

I. INTRODUCTION

Tourism contributes to economic development and cultural exchange, but travelling in an unfamiliar environment can create practical safety challenges. Visitors may not know the location of the nearest police station or hospital, may have difficulty describing an emergency, or may be uncertain about whether their request has reached the appropriate responder. These problems can become more difficult when language barriers, limited connectivity, or unfamiliar local infrastructure are involved.

Many digital applications provide maps, contact directories, or emergency calling functions. However, these capabilities are often separated across different services. A tourist may need to switch between applications to identify a nearby facility, report an incident, share a location, and check whether assistance is being arranged. A more integrated approach can help reduce this complexity by bringing relevant information and communication functions into one platform.

The Smart Tourist Safety System (STSS) is developed to explore this approach. It combines tourist-facing interfaces, police and hospital directories, location-aware processing, natural-language query handling, and selected machine-learning components. The system is intended to help tourists access safety information and support communication between users and responders.

Machine learning is used for selected tasks, including district-level crime-risk estimation, audio distress classification, and tourist-query intent recognition.

Other operations, such as geographic distance calculation and point-in-polygon testing, rely on deterministic algorithms. Keeping these functions distinct is important because a learned prediction is uncertain, whereas a deterministic calculation follows a defined mathematical procedure, subject to the quality of its input data.

This paper describes the STSS architecture, its principal computational components, the available evaluation results, and the limitations identified in the project audit. It also discusses the importance of data quality, human confirmation, privacy, and reproducible testing in safety-related software.

The main contributions of this work are:

- 1) An integrated system design connecting tourist interfaces, emergency-service directories, and responder-oriented workflows.
- 2) A description of the machine-learning and geospatial methods used in the platform.
- 3) A summary of the reported model-evaluation results and their practical limitations.
- 4) An analysis of coordinate quality, runtime implementation status, and areas requiring further validation.

II. SYSTEM OVERVIEW

STSS is designed as a web-based application with a frontend, backend services, and a database. The project audit identifies a React and TypeScript frontend, a Node.js and Express backend, and MongoDB for data storage. Real-time communication is supported through Socket.IO where configured in the application.

The system can be understood through four principal layers.

A. User Interface Layer

The user interface provides access to tourist safety functions and relevant emergency information. Separate responder-oriented interfaces support police and hospital workflows. The purpose of these interfaces is to make the information required for reporting and handling an incident accessible through a consistent application.

B. Application and Processing Layer

The backend coordinates requests, processes application data, communicates with the database, and manages incident-related operations. It also connects the frontend to location-processing functions and the relevant computational components.

C. Machine-Learning and Geospatial Layer

The machine-learning components support specific prediction and classification tasks. The reported implementations include a Random Forest model for district-level crime-risk estimation, a two-dimensional convolutional neural network for audio classification, and a TF-IDF with LinearSVC classifier for tourist-query intent recognition. MiniLM sentence embeddings are also used for semantic matching.

Geospatial processing includes Haversine distance calculations and point-in-polygon checks. These methods support location-aware functionality but depend on the accuracy and completeness of the underlying coordinates.

D. Data and External-Service Layer

MongoDB stores application records and directory information. The application also uses external services, including OSRM routing, where configured. An optional Gemini API integration is reported in the project audit. Since external services may become unavailable or change their behaviour, the application should provide appropriate fallback handling.

Smart Tourist Safety System (STSS) — System Architecture

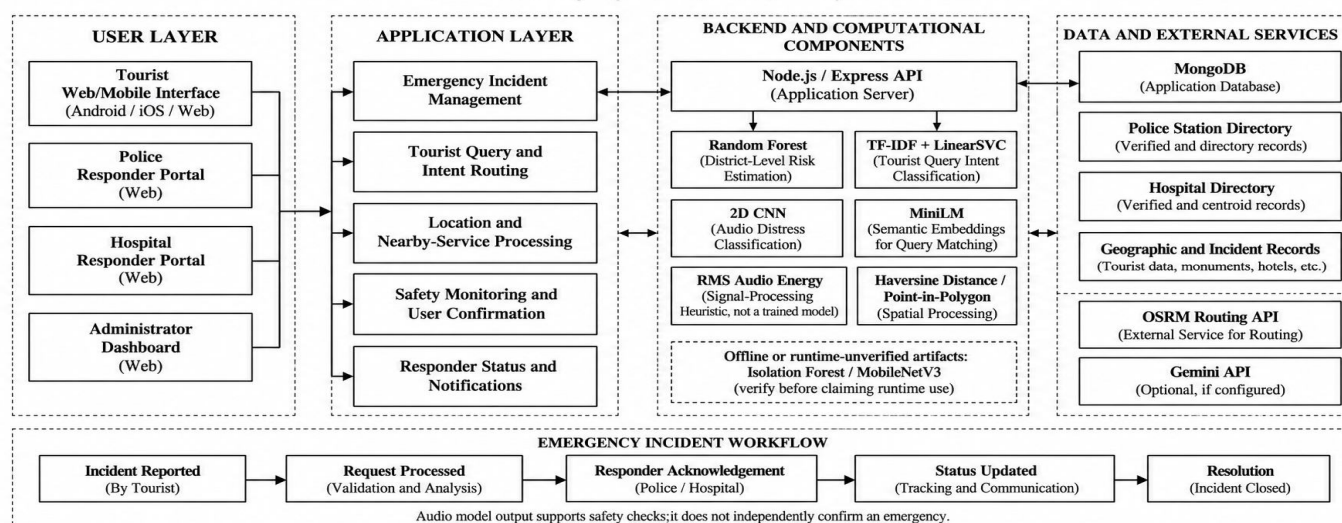


Fig. 1. High-level architecture of the Smart Tourist Safety System.

III. DATA SOURCES AND DATA QUALITY

The usefulness of a tourist safety platform depends partly on the quality of the information available to it. A directory may contain a facility name and telephone number but lack reliable geographic coordinates. Similarly, a coordinate may represent the centre of an administrative region rather than the actual location of a building.

The available project audit reports the following directory counts.

TABLE I. DIRECTORY COVERAGE REPORTED BY THE PROJECT AUDIT

Dataset	Total records	Facility-level or verified coordinates	Centroid or regional records	Missing or other
Police stations	3,208	1,228 with coordinates	Not separately reported	1,980 without coordinates
Hospitals	2,495	776 exact facility-level geocodes	1,508 administrative centroids	211 missing
Monuments	4,504	82 verified map coordinates	4,422 region-indexed	Not separately reported
Hotels	2,058	Coordinate breakdown unavailable	Not supplied	Not supplied

The figures indicate that coordinate coverage is an important limitation. In particular, an administrative centroid should not be treated as the precise location of a hospital or police station. If such a coordinate is used in a nearest-facility calculation, the calculated distance may not represent the actual travel distance to the facility.

For this reason, geographic records should ideally include their source, verification date, coordinate type, and confidence level. The interface should distinguish verified facility locations from approximate or missing coordinates instead of presenting all locations with the same level of certainty.

The project audit also reports district-level crime data associated with National Crime Records Bureau data for 2021–2023, an audio dataset involving ASVP_UPDATE and NIGENS, and 839 annotated tourist queries distributed across 24 intent categories. The exact dataset versions, licensing details, preprocessing steps, and split assignments should be documented in the final experimental record.

IV. METHODOLOGY

A. District-Level Crime-Risk Estimation

The project audit identifies a Random Forest model for estimating crime risk at the district level. The reported configuration uses 150 trees with a maximum depth of eight. The pipeline also applies Platt sigmoid calibration to the model output.

A Random Forest combines predictions from multiple decision trees. A simplified representation of its estimated class probability is:

where:

$$\hat{P}(y = 1 | x) = \frac{1}{T} \sum_{i=1}^T P_i(y = 1 | x)$$

- x represents the input feature vector.
- T is the number of trees in the forest.
- $P_i(y=1 \mid x)$ is the probability estimated by the i -th tree.
- $\hat{P}(y=1 \mid x)$ is the combined estimated probability.

The expression represents the general averaging principle used by a probability-producing forest. The exact implementation should be checked against the training code.

Calibration is intended to improve the relationship between predicted probabilities and observed outcomes. A commonly used Platt scaling expression is:

$$P(y = 1 | s) = \frac{1}{1 + \exp(As + B)}$$

where s is a model score and A and B are parameters fitted using calibration data. Their actual values should be taken from the trained implementation.

District-level risk estimates should be interpreted as broad contextual information. They do not establish that a particular street, facility, or individual is safe or unsafe. Geographic separation between training and evaluation data is especially important because nearby districts can share patterns that make random train-test splits overly optimistic.

B. Audio-Based Distress Classification

The project audit describes an audio-classification model based on a two-dimensional convolutional neural network (2D CNN). It operates on 64-band log-Mel spectrograms and predicts three classes: DISTRESS, NORMAL, and BACKGROUND.

The reported configuration includes audio sampled at 16 kHz, three-second windows, a short-time Fourier transform window of 1,024 samples, and a hop length of 512 samples.

A log-Mel spectrogram represents the distribution of signal energy across time and perceptually motivated frequency bands. These features provide the CNN with a structured representation of the audio signal.

A separate signal-processing measure, root-mean-square (RMS) energy, can be expressed as:

$$\text{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N x[n]^2}$$

where $x[n]$ is the audio sample at index n , and N is the number of samples in the analysis window.

RMS measures signal energy, but it does not determine whether a sound represents distress. It should therefore be distinguished from the trained audio classifier.

For safety, the audio model should be treated as a decision-support component. A model prediction alone does not confirm an emergency. Where appropriate, the application should request user confirmation or provide another means of checking the situation before escalating an alert.

C. Tourist-Query Intent Recognition

The natural-language component helps classify tourist queries into predefined intent categories. The main classifier reported in the audit uses term frequency-inverse document frequency (TF-IDF) features with sublinear term-frequency scaling and a Linear Support Vector Classifier (LinearSVC).

The reported dataset contains 839 labelled queries across 24 intent categories, divided into 586 training examples, 126 validation examples, and 126 test examples.

The audit reports a test accuracy of 84.92% and a macro F1-score of 0.8437. It also identifies Xenova/all-MiniLM-L6-v2 as a pretrained sentence-embedding model used for semantic matching.

The two approaches serve related but different purposes. TF-IDF with LinearSVC classifies queries using learned weights over text features, while sentence embeddings represent text as numerical vectors that can be compared for semantic similarity.

Classification performance can be described using the following metrics.

Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score

$$F1 = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Here, (TP) , (TN) , (FP) , and (FN) represent true positives, true negatives, false positives, and false negatives, respectively, for a particular binary class evaluation.

For multi-class intent recognition, macro F1 is calculated by computing the F1-score for each class and then taking their unweighted average. This approach gives each intent equal weight, regardless of the number of examples belonging to that intent.

D. Geospatial Processing

Geospatial functions help the application calculate distances and determine whether a point lies within a defined geographic area.

The Haversine formula estimates the great-circle distance between two points on a spherical Earth. For two locations represented by latitude and longitude in radians, define:

$$a = \sin^2 \left(\frac{\Delta\phi}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\Delta\lambda}{2} \right)$$

The distance is then:

$$d = 2R \operatorname{atan2} (\sqrt{a}, \sqrt{1-a})$$

where:

- (ϕ_1, ϕ_2) are the latitudes of the two locations.
- (λ_1, λ_2) are their longitudes.
- $(\Delta\phi = \phi_2 - \phi_1)$.
- $(\Delta\lambda = \lambda_2 - \lambda_1)$.
- (R) is the assumed radius of the Earth.
- (d) is the estimated surface distance.

The Haversine calculation provides an approximate straight-line surface distance, not the actual road distance or travel time. Road-network routing requires a separate routing service or routing algorithm.

Point-in-polygon processing can be used to check whether a coordinate falls inside a geographic boundary. This can support region-based queries, but its accuracy depends on the quality of the boundary data and the coordinate reference assumptions.

The audit also reports an Isolation Forest artifact for movement anomaly detection. However, it indicates that the active movement-monitoring implementation may use fixed kinematic thresholds instead. Similarly, the audit reports a MobileNetV3 heritage-recognition artifact while indicating that active heritage lookup may rely on keyword matching. These implementation details should be confirmed from the actual runtime code before the models are described as active components.

V. RESULTS AND DISCUSSION

The following table summarises the model-evaluation values reported in the project audit. They are reproduced as reported and have not been independently rerun for this paper.

Table II. Reported Model-Evaluation Results

Model or evaluation	Accuracy	Precision	Recall	F1-score or other metric
Calibrated crime-risk model	82.54%	77.50%	70.45%	F1 = 0.7381; Brier = 0.13524
Crime-risk model, five-fold state-grouped cross-validation	Not reported	Not reported	Not reported	ROC-AUC = 0.7137 ± 0.1468; F1 = 0.2635
Audio CNN, distress threshold 0.80	Not reported	37.72%	54.43%	71 reported false positives
Audio CNN, default threshold	Not reported	7.81%	84.81%	791 reported false positives
Intent classifier, held-out test	84.92%	86.07% macro	84.86% macro	Macro F1 = 0.8437

A. Crime-Risk Model Performance

The calibrated crime-risk model achieved a reported accuracy of 82.54%, with an F1-score of 0.7381 and a Brier score of 0.13524. These values provide an initial indication of model performance on the reported evaluation data.

However, the state-grouped cross-validation results were weaker, with a reported ROC-AUC of 0.7137 ± 0.1468 and an F1-score of 0.2635. This difference is important because evaluation across geographically separated groups can reveal whether a model generalises to regions that were not represented in its training data.

The results suggest that further evaluation is needed before the model can be used to support consequential decisions. Future tests should examine geographical generalisation, calibration, class-specific performance, and possible temporal or geographic data leakage.

B. Audio Classification and Threshold Selection

The audio classifier illustrates the trade-off between false alerts and missed distress events. According to the audit, using a distress threshold of 0.80 reduced false positives from 791 to 71.

The relative reduction is calculated as:

$$\text{False Positive Reduction} = \frac{791 - 71}{791} \times 100$$

$$\text{False Positive Reduction} \approx 91.02\%$$

Although this reduction is substantial, the reported recall at the 0.80 threshold was 54.43%. This means that the model failed to identify a considerable proportion of distress examples in that evaluation.

Threshold selection must therefore consider both types of error. A high threshold may reduce false alerts but also increase the number of missed distress events. A suitable threshold requires testing under realistic audio conditions and careful consideration of the consequences of false positives and false negatives.

For this reason, the audio classifier should not independently trigger an emergency dispatch. Human confirmation and clearly defined fallback procedures remain important safeguards.

C. Intent Classification Performance

The LinearSVC intent classifier achieved a reported accuracy of 84.92%, with a macro F1-score of 0.8437. This suggests that the classifier can distinguish among many of the labelled query categories in the reported test set.

Nevertheless, the performance may vary with spelling errors, short or ambiguous queries, different languages, and intents that have relatively few training examples. A confusion matrix and class-wise precision and recall would help identify the categories that require additional training data.

The audit also reports a separate 100-query benchmark in which LinearSVC achieved 96% accuracy and MiniLM achieved 85%. These results should be reported separately from the main test-set results unless the two evaluation protocols are confirmed to be comparable.

D. Directory Coverage and Practical Reliability

The directory statistics show that the availability of geographic coordinates varies across data collections. This limitation can affect nearby-service searches and location-based recommendations.

A distance algorithm cannot compensate for an incorrect or approximate input coordinate. Consequently, a facility directory should store coordinate provenance and make uncertainty visible to users. The system should also distinguish between a facility that is confirmed to exist at a particular location and a record that has only been associated with a broader administrative region.

These considerations are particularly important in a safety application, where a misleading location may waste time or direct a user away from the intended service.

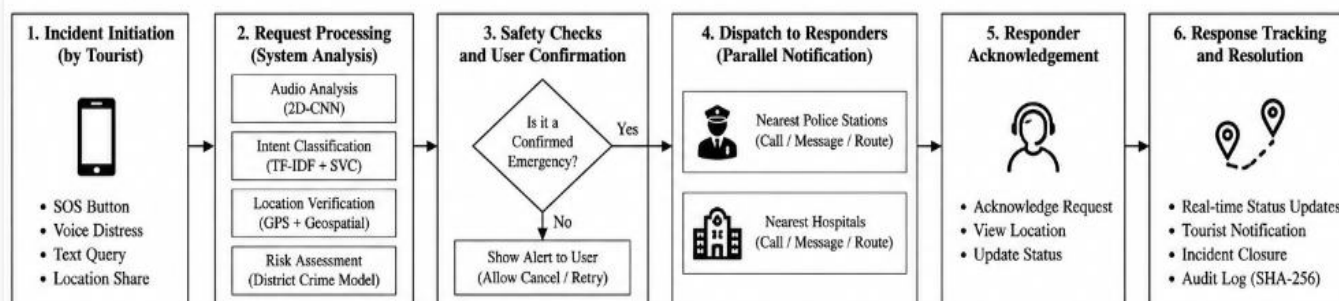


Fig. 2. Human-centred emergency incident workflow.

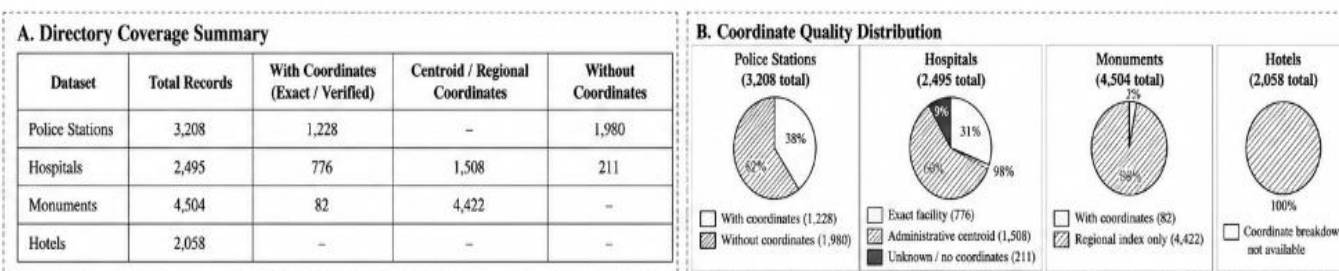


Fig. 4. Directory coverage and coordinate-quality summary reported by the project audit.

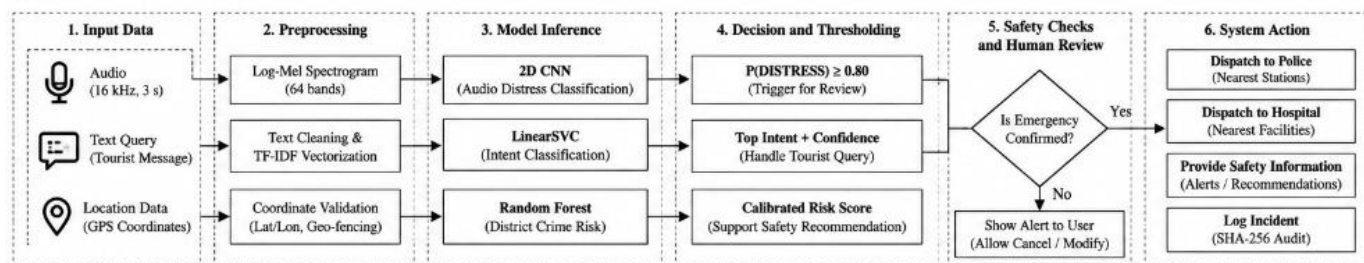


Fig. 3. Model inference, safety checks, and human review.

VI. EMERGENCY WORKFLOW, PRIVACY, AND DATA QUALITY

A safety-oriented system should distinguish between a prediction, a user-reported emergency, and an incident confirmed by a responder. These events represent different levels of evidence and should not be treated as interchangeable.

When an audio model detects a possible distress signal, the application should provide a suitable confirmation process where feasible. The user should receive understandable information about the alert, and the system should provide an appropriate cancellation or escalation path. A classifier output alone should not be treated as proof that an emergency has occurred.

Incident records should capture relevant status changes, including whether the request was received, whether a responder acknowledged it, and whether the incident was resolved. Recording the source of an event can also help distinguish user-triggered requests from automated signals and responder-generated records.

Location information requires careful handling. The system should not describe an approximate centroid as an exact facility location. Similarly, a missing coordinate should not be silently substituted with an arbitrary point. Each directory entry should ideally include a source, last-verified date, and coordinate-quality indicator.

Privacy is another important consideration because the application may handle location history, incident descriptions, contact information, and potentially audio data. Data collection should be limited to what is necessary for the stated function. Access permissions, retention periods, secure transmission, and deletion procedures should be documented and tested.

The project audit describes a SHA-256 hash-chained audit list stored in application memory. If this matches the deployed implementation, it should be described precisely as an in-memory hash-chained log rather than a blockchain or a tamper-proof distributed ledger. Its persistence and resistance to modification depend on the surrounding implementation and operational controls.

VII. LIMITATIONS AND FUTURE WORK

Several limitations need to be addressed before STSS can be considered ready for real-world safety deployment.

First, the model metrics in this paper come from the project audit and have not been independently reproduced. Reproducibility requires the exact datasets, preprocessing scripts, train-validation-test splits, model versions, random seeds, and evaluation outputs. Second, the reported crime-risk model performs less strongly under state-grouped cross-validation than in its reported test evaluation. Further experiments should examine geographical generalisation, temporal leakage, class imbalance, and probability calibration.

Third, the audio classifier presents a clear trade-off between false positives and recall. Evaluation should include realistic environmental noise, variation between recording devices, and conditions that differ from the training dataset. Thresholds should be selected using an explicit safety analysis rather than accuracy alone.

Fourth, incomplete and approximate coordinates limit the reliability of nearby-service searches. Directory records should be verified regularly, and their coordinate quality should be visible to both users and administrators.

Fifth, the actual runtime use of certain model artifacts must be confirmed. A model file being present in the repository does not establish that the application loads it or uses its predictions. The final system description should distinguish deployed models from experimental or offline artifacts.

Finally, the system needs usability testing with tourists and emergency responders. Testing should examine accessibility, ease of reporting, response-status clarity, unreliable network connections, and the consequences of incorrect or delayed information.

Future work should prioritise reproducible evaluation, improved directory verification, geographically robust model testing, privacy safeguards, and supervised field trials. These steps would provide stronger evidence about whether the platform can improve the practical experience of requesting and coordinating assistance.

VIII. CONCLUSION

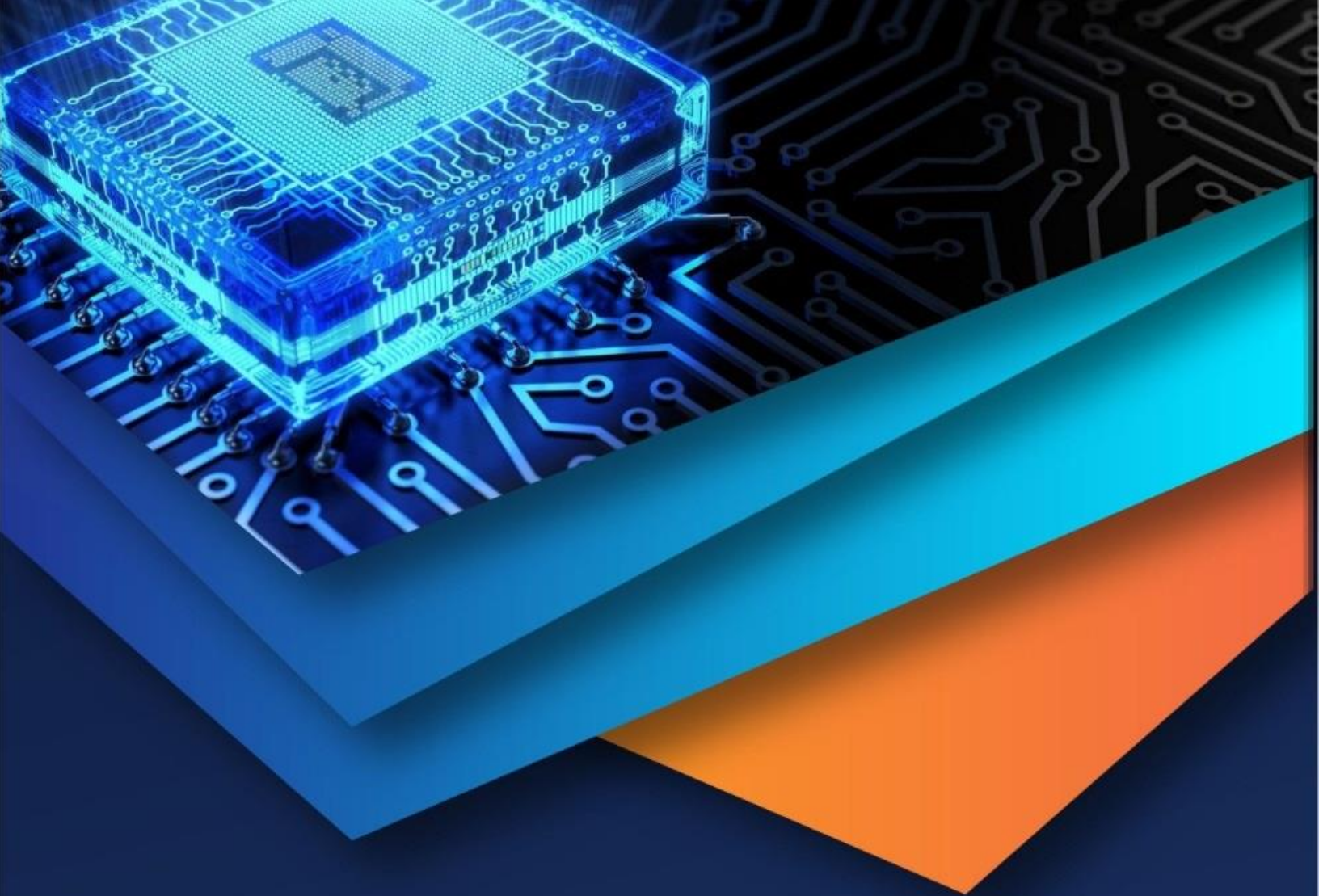
This paper presented the Smart Tourist Safety System as an integrated platform for tourist safety information, location-aware processing, machine-learning-assisted classification, and responder communication. The system combines trained models with deterministic geospatial calculations and application-level workflows.

The available project audit reports promising results for the intent classifier and calibrated crime-risk model, while the audio classifier demonstrates the difficulty of balancing false alerts against missed distress events. The audit also identifies limitations in coordinate coverage and differences between certain offline model artifacts and the active runtime.

These findings indicate that STSS is a developing research platform with potential for further investigation, rather than a system whose real-world safety effectiveness has already been established. Reliable deployment will require reproducible experiments, verified directory data, transparent communication of uncertainty, careful privacy controls, human confirmation for consequential alerts, and testing under realistic conditions.

REFERENCES

- [1] National Crime Records Bureau, Crime in India. Government of India. Verify the exact editions, publication years, tables, and URLs used.
- [2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of NAACL-HLT, 2019, pp. 4171–4186.
- [3] A. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems, vol. 30, 2017.
- [4] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proceedings of the IEEE International Conference on Data Mining, 2008, pp. 413–422.
- [5] J. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," in Advances in Large Margin Classifiers, 1999, pp. 61–74.
- [6] Project OpenStreetMap, OSRM: Open Source Routing Machine. Verify the documentation URL and access date.
- [7] Sentence Transformers, all-MiniLM-L6-v2 Model Card. Verify the exact model repository and version used by the project.
- [8] Add the primary references and licensing information for the exact ASVP_UPDATE and NIGENS dataset versions used in the audio experiments.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)