Orange birds with wings and a heart with text

Description automatically generated with medium confidence

School of Consumer Intelligence and

Information Systems

Department of Applied Information Systems

4. Data Collection & Analysis

Research title: Harnessing Deep Learning For Predicting Traffic Congestion Using Image Classification

KENAN MALALE KASONGO – 220024976

SUPERVISOR: dr tebogo bokaba

Table of Contents

[4. Data Collection 3](#_Toc177625307)

[4.1 Introduction 3](#_Toc177625308)

[4.2 Dataset Selection 3](#_Toc177625309)

[4.3 Data Security and Accessibility 4](#_Toc177625310)

[4.4 Ethical Considerations 4](#_Toc177625311)

[4.5 Conclusion 5](#_Toc177625312)

[5. Data Analysis 5](#_Toc177625313)

[5.1 Data Preprocessing 5](#_Toc177625314)

[5.2 Data Distribution 6](#_Toc177625315)

[5.3 Model Architecture 6](#_Toc177625316)

[5.4 Model Training 7](#_Toc177625317)

[5.5 Model Accuracy 7](#_Toc177625318)

[5.6 Evaluation on Test Set 8](#_Toc177625319)

[5.7 Classification Report 8](#_Toc177625320)

[5.8 Confusion Matrix 9](#_Toc177625321)

[5.9 Identifying Key Features Using Integrated Gradients 10](#_Toc177625322)

[**5.10** Conclusion 14](#_Toc177625323)

# Data Collection

# Introduction

Data collection is an important component of this research, forming the empirical basis for developing a deep learning model to predict road traffic congestion through image classification (Abdullah et al., 2023). This section outlines the effective and thorough implementation of data collection methods.

# Dataset Selection

The dataset selected for this study is the **Traffic-Net** dataset, publicly available on Kaggle. This dataset was chosen due to its direct relevance to the research objectives, images, and balanced representation of various traffic conditions and incidents that impact road congestion.

**Dataset Composition**:

* **Total Images**: 4,400 images.
* **Categories**:
  + Accident
  + Dense Traffic
  + Fire
  + Sparse Traffic
* **Training Set**: 3,600 images, with 900 images in each category.
* **Test Set**: 800 images, with 200 images in each category.

The images encompass a variety of environmental conditions, including different weather scenarios and lighting conditions, enhancing the model's ability to generalize to real-world situations.

**Preprocessing Steps**:

* All images will be resized to **224x224 pixels**.
* Images will be **normalized** to a pixel value range of **[0,1]**.
* **Data augmentation**: random flips, rotations, and lighting adjustments will be applied to improve the model's generalization ability and performance on unseen data.

# Data Security and Accessibility

For added transparency, the data and any supporting resources, such as code or supplementary information, will be kept in a GitHub long-term repository. Access can be granted to any individual who requests it, and a link to the repository will be included in the final research documentation. This approach guarantees that the research may be expanded upon in the future and contributes to the body of scientific knowledge.

By utilizing GitHub, version control is maintained, allowing for tracking changes and collaborative opportunities. Data security is ensured through GitHub's secure infrastructure, and access permissions can be managed to control who can view or contribute to the repository. This strategy effectively balances the need for data security with the accessibility required for academic research.

# Ethical Considerations

Ethical compliance was maintained throughout the data collection process:

* **Privacy Protection:**
  + **Anonymity Assurance:** The images do not contain personally identifiable information, such as recognizable faces or license plates, thereby safeguarding individual privacy.
  + **Compliance with Regulations:** Adhered to the General Data Protection Regulation (GDPR) and institutional ethical standards.
* **Responsible Use:**
  + **Licensing Compliance:** The dataset is used strictly within the bounds of the Kaggle License Agreement, exclusively for educational and research purposes.
  + **Attribution:** Proper credit is given to the dataset creator in all reports and publications resulting from this research, adhering to academic standards and intellectual property rights.

# Conclusion

The data collection process was executed with meticulous attention to detail, ensuring effective implementation of methods that align with the research methodology. By following established protocols and minimizing errors, high data quality was maintained. The data was effectively secured and organized, ensuring both security and accessibility using a GitHub repository. This robust data foundation is essential for the subsequent development and evaluation of the deep learning model for predicting road traffic congestion.

# Data Analysis

This section focuses on the process and results of analyzing traffic images using a Convolutional Neural Network (CNN) model. The dataset comprises images from four categories: accident, dense traffic, fire, and sparse traffic. This analysis will delve into data distribution, preprocessing steps, model training, and the evaluation metrics used to assess the model's performance.

# Data Preprocessing

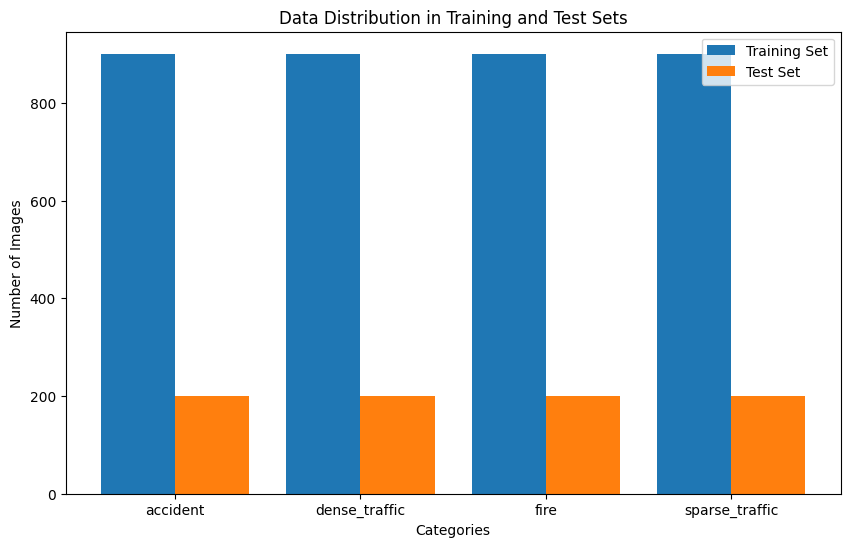
Preprocessing is an important step in preparing the raw image data for input into the CNN model (Yang et al., 2022). The following transformations were applied:

* **Rescaling**: All image pixel values were rescaled from their original range of [0, 255] to the range [0, 1]. This normalization reduces the range of values and helps the model converge faster during training.
* **Image Augmentation**: To improve generalization and reduce overfitting, several augmentations were applied to the training images, including:
  + **Random Zoom**: Up to 15% zoom was applied to randomly selected images.
  + **Horizontal and Vertical Shifts**: Up to 20% shifts were applied to simulate different camera angles.
  + **Shear Transformations**: Shearing within a 15% range was applied to distort the images slightly, further enhancing generalization.

All images were resized to **224x224 pixels** with 3 color channels (RGB) to match the CNN's input layer requirements. Each batch of training data contained 32 images.

# Data Distribution

The dataset contains images for both the training and test sets, evenly distributed across the four categories (accident, dense traffic, fire, and sparse traffic).



* **Training Set**: Each category contains 900 images.
* **Test Set**: Each category contains 200 images.

The equal distribution of images ensures that the model does not develop bias toward any one category and can learn to classify all traffic scenarios effectively.

# Model Architecture

The CNN model used for traffic classification consists of several convolutional layers designed to extract relevant features from the images, followed by fully connected layers that help interpret these features and make predictions.

The model architecture includes:

* **Convolutional layers**: These layers apply filters to the input image, helping the model learn to detect features such as edges, textures, and patterns.
* **Max-pooling layers**: These layers downsample the image, reducing the dimensionality and retaining essential features.
* **Fully connected (dense) layers**: These layers take the learned features and use them to make predictions.
* **Softmax output layer**: The final layer has four neurons, corresponding to the four traffic categories, and applies the softmax function to output the probability of each class.

# Model Training

The model was trained on the training set over **20 epochs**, with the loss function and accuracy being tracked for both training and validation data.

The graph below shows the **training loss** and **validation loss** over the epochs:

A graph showing a loss

Description automatically generated with medium confidence

* The **training loss** decreases consistently over time, indicating that the model is learning to minimize errors on the training data.
* The **validation loss** fluctuates slightly but also shows an overall downward trend, suggesting that the model is generalizing well, though some overfitting is present.

# Model Accuracy

In addition to loss, accuracy was tracked over the training period.

The graph below shows the **training accuracy** and **validation accuracy**:

A graph of a graph

Description automatically generated with medium confidence

* By the 20th epoch, the model reaches a training accuracy of **84.09%**, and the validation accuracy closely mirrors this, suggesting that the model performs well on unseen data.

# Evaluation on Test Set

After completing training, the model was evaluated on the test set:

* **Test Accuracy**: **84.09%**
* **Test Loss**: **0.4664**

This test set accuracy indicates that the model can generalize effectively to unseen traffic images.

# Classification Report

To gain further insights into the model’s performance, we generated a classification report, which provides precision, recall, and F1-score for each class.

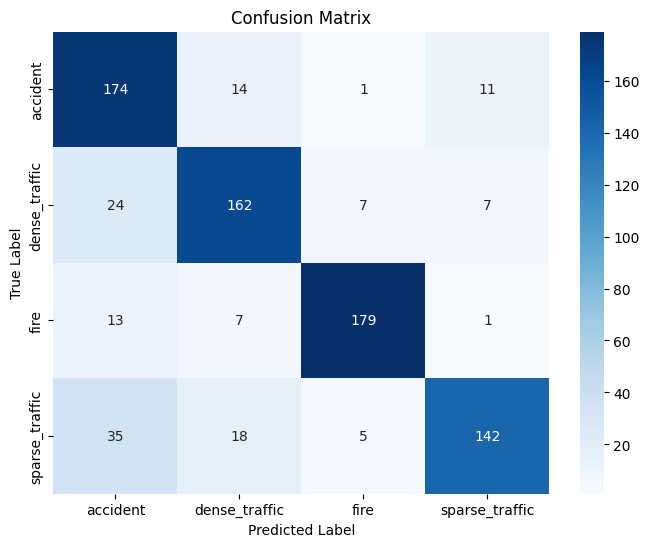
The classification report for all categories is shown below:

A blue and white rectangular boxes with white text

Description automatically generated

* **Precision**: The fire category has the highest precision at **0.93**, indicating that the model is very good at avoiding false positives for this class.
* **Recall**: The **fire** category also has the highest recall at **0.90**, meaning the model is very good at capturing most true positives for this class.
* **F1-score**: The fire class has the highest F1-score at **0.91**, balancing both precision and recall.

# Confusion Matrix

The confusion matrix provides a visual representation of the model's performance by comparing the true labels with the predicted labels. It is widely used in classification tasks to summarize the number of correct and incorrect predictions across different categories. By presenting the number of true positives, false positives, true negatives, and false negatives, the confusion matrix enables a deeper understanding of where a model's performance excels or fails. This insight is critical for improving model accuracy, precision, recall, and other performance metrics by identifying areas that require adjustment (Arias-Duart et al., 2023).****Below is the **confusion matrix** for the test set:

* The model correctly predicts **179 out of 200** images in the fire category.
* Some misclassifications occur between the accident and sparse traffic categories, with **35 instances of sparse traffic** being classified as accident.

# Identifying Key Features Using Integrated Gradients

One of the key objectives of this research is to identify which features the Convolutional Neural Network (CNN) relies on when classifying traffic images into categories such as accident, dense traffic, fire, and sparse traffic. To achieve this, the **Integrated Gradients** method was used to generate heatmaps that visualize the importance of each pixel in the image with respect to the model's decision. The following section discusses the use of Integrated Gradients and the insights gained from the heatmaps produced for each traffic category.

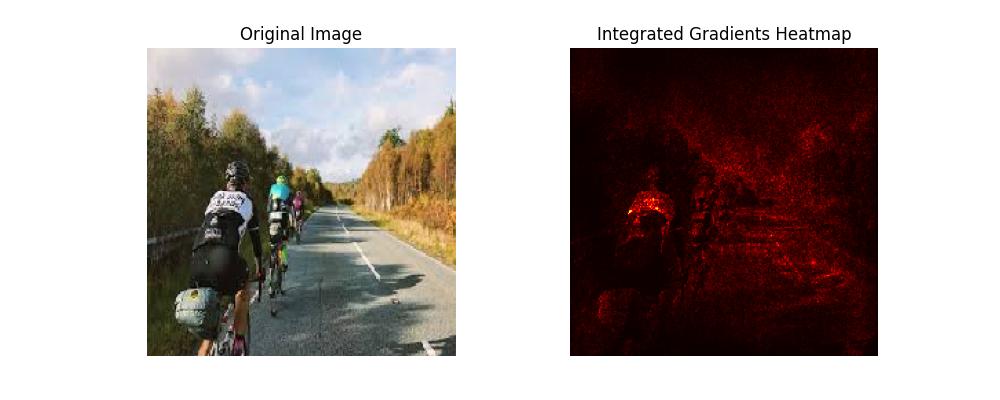
**Methodology**

Integrated Gradients is a technique that attributes the model's prediction to input features by comparing the image with a baseline (e.g., a black image) (Ma et al., 2023). It measures the contribution of each pixel by integrating the gradients along the path from the baseline to the input image. This method is particularly effective as it provides a smooth, continuous explanation of the model's predictions, highlighting which parts of the image played the most significant role in classification.

For each traffic image in the dataset, a corresponding heatmap was generated, showing the pixels that contributed the most to the classification. These heatmaps were then analyzed to identify common patterns and features the model focuses on in each traffic category.

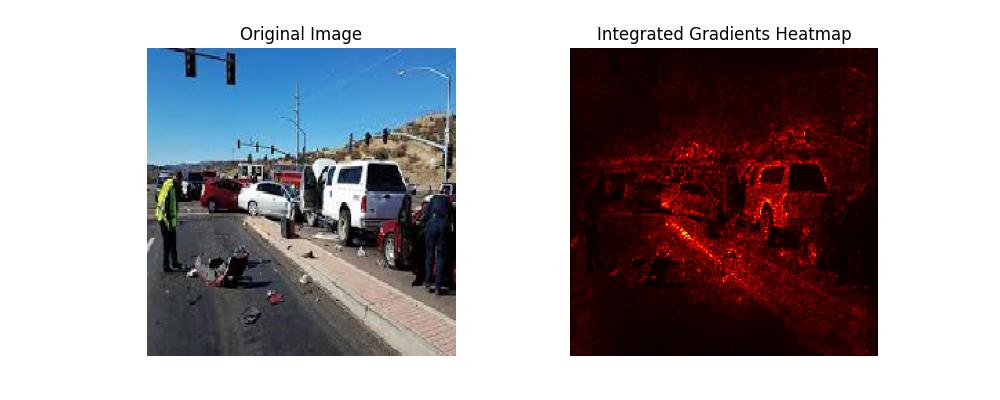
**Results and Insights**

1. **Sparse Traffic**



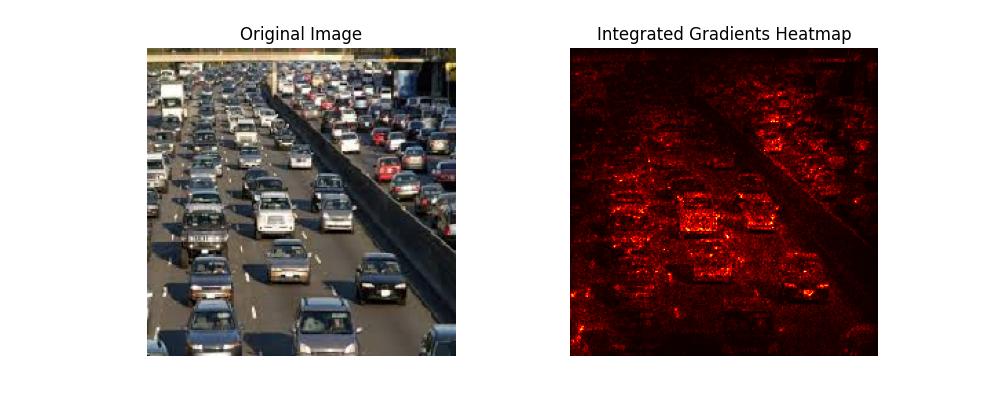
* + **Original Image**: The original image shows cyclists on a clear road with sparse traffic.
  + **Integrated Gradients Heatmap**: The heatmap reveals that the model primarily focuses on the road itself and the presence of the cyclists in the image. The surrounding foliage and background contribute little to the classification. The model detects the emptiness of the road and the presence of distinct objects (the cyclists) to determine that the traffic is sparse.

1. **Accident**



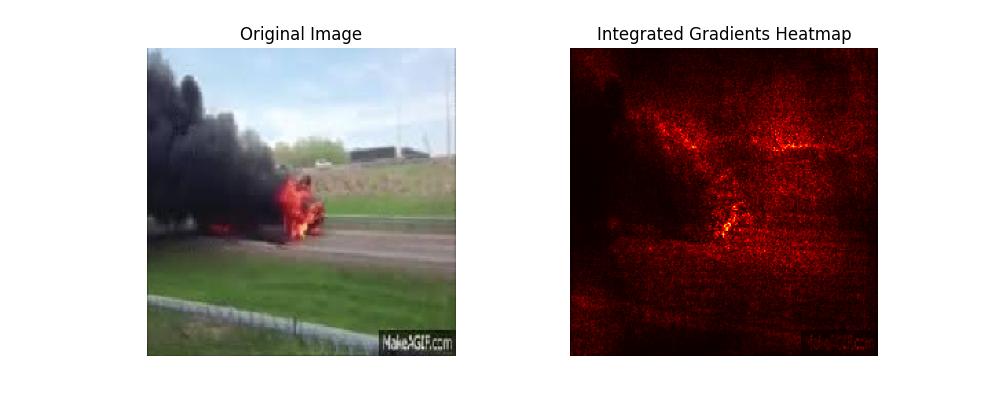
* + **Original Image**: The original image depicts a car accident scene with debris and multiple vehicles stopped on the road.
  + **Integrated Gradients Heatmap**: In this case, the heatmap highlights areas around the damaged vehicles and the debris on the road. The model appears to focus heavily on the cluster of cars and the disruption in the usual flow of traffic. This shows that the model uses visual cues like vehicle damage and traffic stoppage to identify accidents.

1. **Dense Traffic**



* + **Original Image**: The original image shows a large number of vehicles on a congested road.
  + **Integrated Gradients Heatmap**: The heatmap highlights the concentration of cars, with particular emphasis on the cars' outlines and their proximity to one another. The model effectively identifies dense traffic situations by focusing on the vehicles' clustering and the uniformity of the flow of cars.

1. **Fire**

****

* + **Original Image**: The image shows a vehicle on fire, with large flames and thick black smoke.
  + **Integrated Gradients Heatmap**: The heatmap reveals that the model strongly focuses on the bright flames and the thick smoke in the image. These are the key visual features that the model uses to distinguish the fire category from other categories. The high intensity of color in the heatmap around the fire suggests that the model relies heavily on detecting extreme visual cues like bright flames and smoke when making predictions for this class.

**Analysis**

The Integrated Gradients heatmaps provide valuable insights into the key features that the model relies on for classification:

* **Road and Object Presence**: For categories like sparse traffic and dense traffic, the model places emphasis on the road and the presence (or absence) of objects (vehicles, cyclists) on the road.
* **Clustering and Disruption**: In the case of accidents, the model is highly sensitive to disruptions in traffic flow, such as vehicle damage and debris.
* **Extreme Cues (Fire)**: For the fire category, the model focuses primarily on extreme visual cues like bright flames and dark smoke, indicating that it uses color and intensity as key differentiators.

These insights suggest that the model has learned to identify the most relevant features for traffic classification, focusing on objects, congestion, and specific events like fire or accidents. However, there are still some areas for improvement, as the model occasionally misclassifies between accident and sparse traffic, likely due to visual similarities in these categories.

Using Integrated Gradients, the key visual features that the model uses for traffic image classification have been successfully identified. These include the presence of objects like vehicles and cyclists, the level of congestion, and extreme events like fires. By understanding which features the model focuses on, we gain deeper insights into how the model makes its decisions, allowing for further refinement and improvement of the model. Future work could involve fine-tuning the model to address misclassifications between visually similar categories, such as accident and sparse traffic.

# Conclusion

The CNN model developed for traffic image classification achieves the following:

* **Test Accuracy**: The model reaches a test accuracy of 84.09%, indicating a strong ability to generalize to unseen data.
* **Precision, Recall, and F1-Scores**: The model demonstrates robust performance across all traffic categories, with particularly high precision (0.93), recall (0.90), and F1-score (0.91) in the fire category. This suggests that the model is especially effective in correctly classifying fire-related images.
* **Confusion Matrix**: The confusion matrix reveals the model's strengths in accurately classifying most images. However, some misclassifications are observed, particularly between the accident and sparse traffic categories, where 35 instances of sparse traffic were classified as accidents. This suggests that the model occasionally struggles to differentiate between visually similar scenes.
* **Key Features Identification (Integrated Gradients)**: Using Integrated Gradients, the key visual features that contribute to the model’s predictions were identified. For sparse traffic, the model focuses on the road and the presence or absence of vehicles; for accident scenes, it detects damage and debris; in dense traffic, the clustering of cars is emphasized; and for fire, the model highlights extreme cues such as flames and smoke. This analysis confirms that the model is learning to classify traffic images based on the most relevant visual features, but there is room for improvement in distinguishing between categories with overlapping visual elements.

Abdullah, S. M., Periyasamy, M., Kamaludeen, N. A., Towfek, S. K., Marappan, R., Kidambi Raju, S., Alharbi, A. H., & Khafaga, D. S. (2023). Optimizing Traffic Flow in Smart Cities: Soft GRU-Based Recurrent Neural Networks for Enhanced Congestion Prediction Using Deep Learning. *Sustainability*.

Arias-Duart, A., Mariotti, E., García-Gasulla, D., & Alonso-Moral, J. M. (2023). A Confusion Matrix for Evaluating Feature Attribution Methods. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 3709-3714.

Ma, W., Li, Y., Jia, X., & Xu, W. (2023). Transferable Adversarial Attack for Both Vision Transformers and Convolutional Networks via Momentum Integrated Gradients. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 4607-4616.

Yang, S., Xiao, W.-T., Zhang, M., Guo, S., Zhao, J., & Furao, S. (2022). Image Data Augmentation for Deep Learning: A Survey. *arXiv*, *abs/2204.08610*.