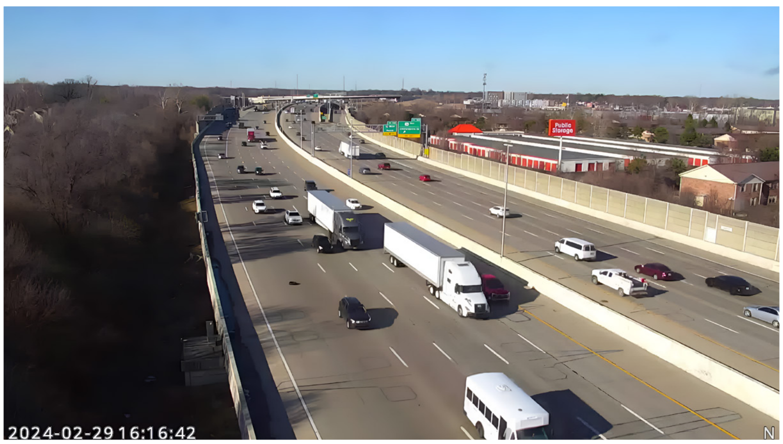
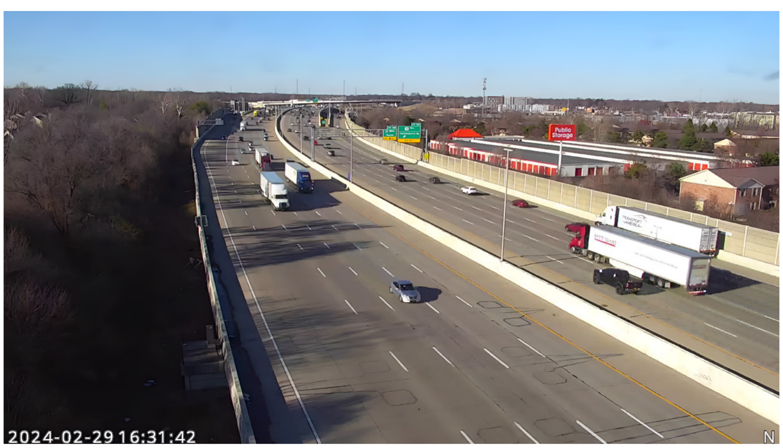
JOINT TRANSPORTATION
RESEARCH PROGRAM

INDIANA DEPARTMENT OF TRANSPORTATION
AND PURDUE UNIVERSITY

Anomaly Detection in Traffic Patterns Using the INDOT Camera System





**Lauren Christopher, Stanley Chien, Yaobin Chen,
Mei Qiu, William Reindl, Liya Koshy**

SPR-4855 • Report Number: FHWA/IN/JTRP-2024/30 • DOI: 10.5703/1288284317778

RECOMMENDED CITATION

Christopher, L., Chien, S., Chen, Y., Qiu, M., Reindl, W., & Koshy, L. (2024). *Anomaly detection in traffic patterns using the INDOT camera system* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2024/30). West Lafayette, IN: Purdue University. https://doi.org/10.5703/1288284317778

AUTHORS

Lauren Christopher, PhD
Associate Professor of Electrical and Computer Engineering
Elmore Family School of Electrical and Computer Engineering
Purdue University in Indianapolis
(317) 828-0340
lachrist@purdue.edu
Corresponding Author

Stanley Chien, PhD
Professor of Electrical and Computer Engineering
Elmore Family School of Electrical and Computer Engineering
Purdue University in Indianapolis

Yaobin Chen, PhD
Director of TASI, Professor of Electrical and Computer Engineering, Chancellor's Professor
Elmore Family School of Electrical and Computer Engineeringy
Purdue University in Indianapolis

Mei Qiu
William Reindl
Liya Koshy
Graduate Researchers
Elmore Family School of Electrical and Computer Engineering
Purdue University in Indianapolis

JOINT TRANSPORTATION RESEARCH PROGRAM

The Joint Transportation Research Program serves as a vehicle for INDOT collaboration with higher education institutions and industry in Indiana to facilitate innovation that results in continuous improvement in the planning, design, construction, operation, management and economic efficiency of the Indiana transportation infrastructure. https://engineering.purdue.edu/JTRP/index_html

Published reports of the Joint Transportation Research Program are available at http://docs.lib.purdue.edu/jtrp/.

NOTICE

ACKNOWLEDGMENTS

This work was supported by the Joint Transportation Research Program (JTRP) SPR-4855, administered by the Indiana Department of Transportation (INDOT) and Purdue University. The authors would like to thank all members of the INDOT Study Advisory Committee (SAC), including Shuo Li, Project Advisor (PA), Nathan Shellhamer, the Business Owner (BO), and Dan McCoy, Jack Gallagher, Michelle Witkowski, Karen Stippich and Drew Sorenson, for their guidance and advice throughout the project period. Special thanks go to Dr. Darcy Bullock, Director of JTRP, and his staff at Purdue University for their administrative support and service. The authors also thank SAC members for their technical support and constructive suggestions.

Technical Report Documentation Page

1. Report No. FHWA/IN/JTRP-2024/30	**2. Government Accession No.**	**3. Recipient's Catalog No.**
4. Title and Subtitle Anomaly Detection in Traffic Patterns Using INDOT Camera System		**5. Report Date** October 2024 **6. Performing Organization Code**
7. Author(s) Lauren Christopher, Stanley Chien, Yaobin Chen, Mei Qiu, William Reindl, and Liya Koshy		**8. Performing Organization Report No.** FHWA/IN/JTRP-2024/30
9. Performing Organization Name and Address Joint Transportation Research Program Hall for Discovery and Learning Research (DLR), Suite 204 207 S. Martin Jischke Drive West Lafayette, IN 47907		**10. Work Unit No.** **11. Contract or Grant No.** SPR-4855
12. Sponsoring Agency Name and Address Indiana Department of Transportation (SPR) State Office Building 100 North Senate Avenue Indianapolis, IN 46204		**13. Type of Report and Period Covered** Final Report **14. Sponsoring Agency Code**

15. Supplementary Notes

Conducted in cooperation with the U.S. Department of Transportation, Federal Highway Administration.

16. Abstract

The Transportation and Autonomous Systems Institute (TASI) of Purdue University Indianapolis (PUI) and the INDOT Traffic Management Center worked together to develop a system that monitors traffic conditions using INDOT CCTV video feeds. Computer vision-based traffic anomaly detection has been studied for the past 20 years, and a thorough state-of-the-art analysis was produced in a recent survey paper. Although AI has contributed to improving anomaly detection, several major challenges remain, such as tracking errors, illumination, weather, occlusion handling, camera pose, and perspective. In addition, the lack of real-life datasets makes the effectiveness of anomaly detection techniques unclear. This project builds on previous research by using automatic anomaly detection and AI algorithms to identify anomalous behavior of the short- and long-term variations of traffic patterns. The research team designed the new system, including the hardware and software components; the existing INDOT CCTV system; the database structure for traffic data extracted from the videos; and a user-friendly web-based server for showing the anomalies automatically.

17. Key Words anomaly detection, highway traffic, camera	**18. Distribution Statement** No restrictions. This document is available through the National Technical Information Service, Springfield, VA 22161.

19. Security Classif. (of this report)	**20. Security Classif. (of this page)**	**21. No. of Pages**	**22. Price**
Unclassified	Unclassified	18	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

EXECUTIVE SUMMARY

Introduction

Safety is an important design aspect for highway and surface street systems. Today's access to 24-7 data from surveillance cameras can help inform safety and operational improvements to transportation infrastructure. TASI used Deep Learning AI Computer Vision methods to automatically collect data, such as counting the number of vehicles in each lane (SPR-4638) and producing origin-destination counts for weaving vehicles in complex weaving areas between interchanges (SPR-4738). This project builds on previous projects by adding automatic anomaly detection and AI algorithms to identify anomalous behavior of the traffic patterns of short and long-term variations. This research will inform transportation infrastructure designers of detailed traffic behavior and automatically identify outlier cases that may indicate operational and safety issues that must be addressed before they result in congestion or crashes. The AI algorithms work from the tabulated data (car counts per lane over time) available from previous projects with INDOT.

Computer vision-based traffic anomaly detection has been studied for the past 20 years, and a thorough analysis was produced in a recent survey paper (Santhosh et al., 2021). Although AI has contributed to improving anomaly detection, Santhosh et al. (2021) noted that several major challenges remain, even with AI. Tracking errors, illumination, weather, occlusion handling, camera pose, and perspective all contribute to errors. In addition, the lack of real-life datasets makes the effectiveness of anomaly detection techniques unclear.

The Transportation and Autonomous Systems Institute (TASI) of Purdue University and the Traffic Management Center and Traffic Engineering Division of INDOT worked together to develop a system that monitors traffic conditions based on the INDOT CCTV video feeds. The proposed system uses traffic flow data to train an AI model to identify the difference between input data and reconstructed data and then use that information to identify anomalies.

Findings

The research team designed the hardware and software components; the existing INDOT CCTV system; the database structure for traffic data extracted from the videos; and a user-friendly web-based server for showing the anomalies automatically.

The preliminary prototype of some system components was implemented using previous JTRP research (2018–2023). These prior works provided the structure of the automatic traffic lane flow extraction from the video feeds. The specific goal of this JTRP project was to develop a day and week duration anomaly detection system and implement it on INDOT's premises. The system implemented at INDOT had the following features.

- **Automatic Detection**
  Vehicle-detection by lane in multi-region of interests (ROI) road segments and lane flow rate data was captured. The artificial intelligence object detection method, an improved version of YOLOv4, was used in the prior project. Compared to the original, the improved version of YOLOV4 used transfer learning to fine-tune performance using data generated from INDOT videos. This version had a similar vehicle detection performance in daytime, but detection accuracy at night increased from 60% to about 90% in the ROI. This reused the multi-ROI automatically selected regions of interest because it achieved over 90% car and truck detection accuracy. Since the positioning of each camera can change, it was necessary to make a set position reference in the anomaly design to gather data at the camera set position. The knowledge of the camera set position made the anomaly metric consistent for each camera. The vehicles in each lane were counted, and the frames were timestamped. Each lane's flow rate (vehicles/30-seconds) was captured based on vehicle counting and the camera frame timestamps.
- **Web-Based Graphical User Interface (GUI)**
  A web-based Graphical User Interface (GUI) was developed with suggestions from INDOT. The GUI automatically read data from the camera flowrate database and displayed information on the output webpage. The GUI displayed three types of information: (1) the location of all installed cameras on the Google Map, (2) the video of the selected anomaly time and location, and (3) an anomaly timeline. In addition, this GUI supported the recording and processing of data for anomaly detection.
- **Video and Data Capture**
  Six cameras were available to record data in this research. It was decided that the 15-minute video segments were the best choice for analysis. Recordings of 15-minute videos began in February 2024 and produced more than 10,000 hours of data. The video data was saved only when the camera was in a predetermined set position (direction and zoom scale matched the INDOT-identified set position). Like in the previous project, the lane flow was extracted from the videos and then stored in a database. The videos were then saved for later analysis of anomalies.
- **Anomaly Detection: Develop and Train Anomaly Detectors (VAE's) 15 Minute Day/Night Segments**
  Because each camera had a unique lane flow profile, a Variational Auto-Encoder (VAE) was trained on the lane flow data for each camera using that camera's historical data. The day and night data were handled in separate VAEs. Each day or night data sample was a vector containing each lane's flow rate (number of vehicles by each lane) for every 30 seconds in the 15 minutes. The VAE learned the common patterns in the data by iteratively comparing the input to the output reconstruction over many historical data samples. The learned day or night VAE was then frozen (not continuous learning). The anomaly was detected when the reconstructed output of the VAE was statistically different from the new input data. This effectively identified short-term anomalies.
- **Anomaly Detection: Develop and Train Anomaly Detectors (VAE's) Weekly Day/Night Segments**
  Longer-term anomalies were found by aggregating the 15-minute data into weekly anomaly detection. The above 30-second data samples were summed over the 15-minute interval (one summed sample per 15-minute), and the data for day or night was used to train two weekly VAEs (day and night). The anomaly was detected when the reconstructed output of the VAE was statistically different from the new input data.

Implementation

The research team will install the system on computers in INDOT for road traffic monitoring operations. A detailed user instructional document will be written.

TABLE OF CONTENTS

1. INTRODUCTION 1

2. SYSTEM ARCHITECTURE AND DEVELOPMENT 1
   2.1 Laneflow Detection and Recorder 1
   2.2 Data Acquisition: 15-Minute Environment Learning Change Videos 1
   2.3 Vehicle Flow Rate, Adaptations for Anomaly Detection 1
   2.4 Vehicle Object Detection and Vehicle Count Accuracy Re-Evaluation 3
   2.5 Database Interface for Anomaly Detector. 3

3. ANOMALY DETECTION 5
   3.1 Model Development 5
   3.2 Anomaly Detection 6

4. GUI INTERFACE 7
   4.1 The Anomaly GUI 8

5. CONCLUSIONS AND FUTURE WORK 10

REFERENCES 10

LIST OF TABLES

Table 3.1 Computational Performance with Recorded 15-Minute Video 7

LIST OF FIGURES

Figure 2.1 Anomaly detection system structure 2

Figure 2.2 Heatmap used for checking the set-point road location in the image 2

Figure 2.3 The new database section contains several tables in addition to the original laneflow information 4

Figure 2.4 Example data table extracted from flow rate database 5

Figure 3.1 Variational Auto Encoder for detecting anomalies 6

Figure 3.2 Example of training AI architectures ablation study 15-minute models day/night (best performance in red) 6

Figure 3.3 (a) Upper screen shot shows no vehicles on the right two lanes. (b) The lower screen shot shows a truck slowed down and parked on the roadside 7

Figure 4.1 GUI anomaly detection setup screen 8

Figure 4.2 Setting time duration for anomaly results 9

Figure 4.3 Server availability popup screen 9

Figure 4.4 GUI with anomaly detections 9

Figure 4.5 Expanded version of anomaly (red) and actual (red/green) vs. reconstructed (black) 10

1. INTRODUCTION

The Indiana Department of Transportation (INDOT) uses about 600 digital cameras along highways in Indiana to monitor highway traffic conditions. The videos from these cameras are currently observed by human operators looking for traffic conditions and anomalies. However, it is time-consuming for the operators to scan through all video data from all the cameras in real-time. Therefore, the main objective of this research was to develop an automatic and real-time system and implement the system in INDOT to detect anomalies automatically.

The Transportation and Autonomous Systems Institute (TASI) of the Purdue University Indianapolis, the Traffic Management Center and Traffic Engineering Division of INDOT have worked together to conduct this research to develop a system that monitors the traffic conditions based on the INDOT CCTV video feeds. The proposed system performs traffic flow measurement and anomaly detection based on traffic flow statistics and learned features. The goal was to develop a system and prepare for future implementation.

The research team designed the system that includes the hardware and software components, and using the currently existing INDOT CCTV system, creating the database structure for traffic data extracted from the videos, and providing a user-friendly web-based server for automatically showing the anomalies. The specific work in this 2023–2024 JTRP project is to develop automated anomaly detection and implement it in INDOT.

2. SYSTEM ARCHITECTURE AND DEVELOPMENT

In the current project, the system architecture was revised from the previously delivered code, to reduce the complexity of the installation process at INDOT. Encapsulating the main software blocks into Docker containers (available at https://www.docker.com/) is a main feature of the new architecture. Figure 2.1 shows the overall anomaly detection system structure for the implemented system. The entire system has four subsystems. The first subsystem is the laneflow and recorder portion of the implementation running on the field computers (in green). The second subsystem is the webserver and database server (in red) implemented on the central computer. The third subsystem is the anomaly detection server (in yellow), and the final subsystem is the video storage computer (in blue). The anomaly detection system was further developed and updated, as described in this document.

2.1 Laneflow Detection and Recorder

The input to a computer is a stream of video frames from the cameras. INDOT provides the IP address of over 500 cameras on the road. Since the cameras in the INDOT system have been purchased over many years in multiple batches, the frame per second rate and camera video resolution varies significantly. The flow rate varies from 6 to 60 fps (frames per second). The resolution varies from 640×480 to 2048×1536. The frame rate setup of the cameras is affected by the network speed at the installation location and method. Therefore, the software was developed to adapt to all possible frame rates and resolutions of camera videos.

The outputs of each laneflow computer are the real-time road flow data abstracted from video streams, which includes the following:

1. number of lanes and lane locations,
2. traffic direction of each lane, and
3. the vehicle flowrate in each lane.

This output is inserted into a database every 30 seconds. The anomaly detection system uses the vehicle flowrate (laneflow) in each lane.

2.2 Data Acquisition: 15-Minute Environment Learning Change Videos

The prior project's environment learning is used to find lanes, lane boundaries, lane directions, and the region. The steps are as follows.

1. Vehicle detection (in the whole frame)
2. Road boundary detection
3. Multiple Region of Interest (ROI) determination and reference line detection for each ROI
4. Lane center detection at the reference line in each ROI
5. Lane detection in each ROI based on vehicle clustering
6. Lane direction detection and lane boundaries determination in ROI
7. Traffic status tracking for each lane (no car, few cars, many cars, jam)
8. Camera viewing angle change detection
9. Camera direction detection
10. Database interface

This environment learning work is described in detail in the prior project reports and papers (Qiu et al., 2020, 2021). The difference between the prior environment learning and this anomaly project is that the environment learning is trained once to the set position and re-used without re-learning when the camera returns to the set position. This was done to make the lane numbers and laneflow consistent in each 15-minute segment recorded.

2.3 Vehicle Flow Rate, Adaptations for Anomaly Detection

Although the capability of recording videos was possible in the previous projects, the project did not have a requirement for an automated, synchronized, timed recording. This anomaly project description requirement required saving anomaly videos for later analysis; therefore, the new system has some changes. As in prior projects, the laneflow is sampled every 30 seconds and stored in the database. Then, all the detailed data is available to download in a spreadsheet.

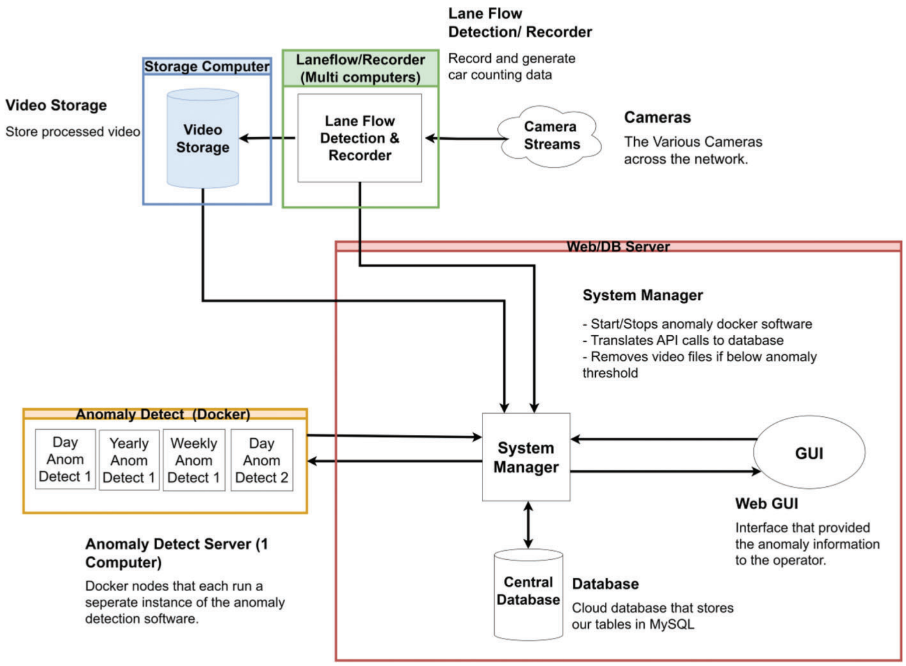


Figure 2.1 Anomaly detection system structure.

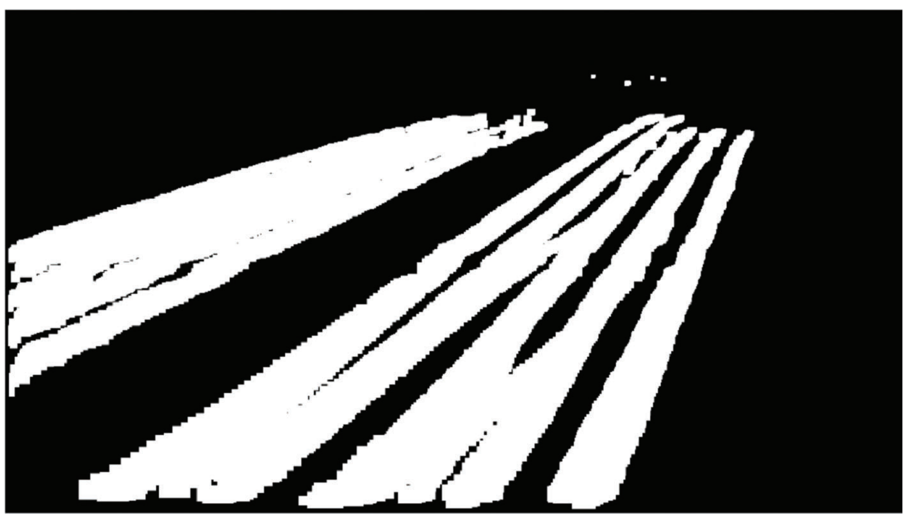


Figure 2.2 Heatmap used for checking the set-point road location in the image.

2.3.1 Checking Set Position

The first change for the anomaly detector is to test the camera's set position for the recording of the video. While recording the 15-minute video, the system automatically checks if the camera's set position is in the frame (the same as the pre-determined set position) or not (at the 7.5-minute mark). If the set position is not matched, then the video recording is stopped. If the set position is correct, then the recording system saves the video output of the laneflow result (with lane markings and flow rate on screen) for each 15-minute interval. The correct position is determined by generating lane heatmaps (Figure 2.2) in every video. The heatmaps are generated from multiple cars driving the same path on the road generating these white lines, which are lanes. A known good video's heatmap is saved and every subsequent video is checked against this known good heatmap. The similarity between them is determined by subtracting the original heatmap with the new heatmap. If the number of pure white pixels in the resulting heatmap is greater than 40% of the number of white pixels in the original heatmap, it is considered out of position.

2.3.2 Number of Skip Frames Set at Constant = 4

The original system adapted to the hardware resources dynamically by skipping more or fewer frames adapting to the varying throughput of the object detection (TASI transfer learning fine-tuned YOLOv4) and tracking (DeepSort). Because the data to the anomaly detection is structured by the precise time window of 15-minutes in 30-second increments, this dynamic adaptation was not usable. Therefore, the number of skipped frames is set to a constant value of 4 for writing laneflow data to the database that is later usable by the anomaly detector. The video data capturing method was improved by changing the data passing through disk files through RAM; hence, the frame skip can always be less than or equal to 4 frames. This setting retains the quality of object detection performance and has some speed benefits compared to dynamic skipping. As GPU hardware speed improves or a new YOLO version is improved, this setting can be lowered.

2.3.3 15-Minute Automated Recording

The 15-minute videos are synced to the hour using a consistent time reference. When the 15-minute interval is completed and the anomaly data is available, the system stores the output video corresponding to the anomaly.

2.3.4 Adapting Storage for Weekly Anomalies

Because the weekly anomalies will rely on the whole week before marking each anomaly segment (15 minutes), the data store, therefore, will be a minimum of 7 days of 15-minute segments. When the weekly is processed, then the deletion of non-anomaly video segments is performed. This phase of the project deliverable is to detect 15-minute daily and weekly anomalies. For future work to address anomaly detection longer than weekly, using only the laneflow data is recommended, without keeping the corresponding 15-minute videos, since the aggregated numbers are larger than any 15-minute segment.

2.4 Vehicle Object Detection and Vehicle Count Accuracy Re-Evaluation

The vehicle detection performance is improved with transfer learning under different lighting and weather conditions.

- Day time detection rate (>95%).
- Evening time (before it becomes totally dark) detection rate: 87%–98%.
- In dark-lit conditions, the detection rate is in the 60%–87% range.
- In dark unlit and viewing the vehicle taillight direction, the accuracy can be 60%–90%.

The vehicle detection does not work well under the following conditions.

- The detection rate during the daytime decreased to the 10% range when the images were not stable (vibrating).
- The detection rate decreases if the camera angle is oblique even during the daytime.
- The detection rate is extremely low in the unlit dark condition.

2.5 Database Interface for Anomaly Detector

The database is specifically designed for anomaly detection and then added as a new part to the database developed to the previous project. The new database section contains several tables in addition to the original laneflow information as seen in Figure 2.3. Each table is described in detail in the following pages.

2.5.1 Video Table

The video table contains the information for each of the stored 15-minute video segments. The attributes of the table include the camera's ID, the date/time, the result of the anomaly detection on this time segment, the video file path, and the designation of day or night.

2.5.2 Camera Table

This is like the previous project, and contains the information for each camera, including status, timestamp, mile marker and other ID information.

2.5.3 Road Mile Marker Table

This is like the previous project and contains the information for the road name, the lane numbers, the view direction, and links to the camera ID.

2.5.4 Lane Flow Rate Table

This is like the previous project, and contains the lanes, the flowrate per lane and video linked information. A data table can be extracted that contains this information, as seen in Figure 2.4.

2.5.5 Anomaly Model Table

The anomaly model table contains the information for each anomaly detection model, and it contains the key information for each of the trained anomaly detection models. The name of the model describes the model type. As described in Section 3 of this report, it is important to retain the comparison of the input vehicle count vs. the output vehicle reconstructed data over time. This allows users to quickly see the difference in flow rate, at which time, that causes the out-of-distribution anomaly. The output GUI graphs are in number of vehicles vs. time for each lane. In this project, we have developed and trained 4 models for each camera (as indicated by model name).

1. *15-Minute Day Model*
   This model takes data from the laneflow result in 15-minute intervals between 6:00:00 AM–5:59:30 PM (EST). The data is structured of 30 samples by N lanes, where N is the number of lanes in the laneflow data table.
2. *15-Minute Night Model*
   This model takes data from the laneflow result in 15-minute intervals between 6:00:00 PM–5:59:30 AM

(EST). The data is structured of 30 samples by N lanes, where N is the number of lanes in the laneflow data table.

3. *1-Week Day Model*
   This model takes data from the laneflow result in the 15-minute intervals between Monday through Sunday night times: 6:00:00 AM–5:59:30 PM (EST). The data is structured of 30 samples by N lanes, where N is the number of lanes in the laneflow data table. In this model, the 30 samples per lane are summed to form one sample per 15-minute interval. For the 12 hours, the data is then 336 (= $4 \times 12 \times 7$) samples by N lanes.
4. *1-Week Night Model*
   This model takes data from the laneflow result in the 15-minute intervals between Monday–Sunday night times: 6:00:00 PM–5:59:30 AM (EST). The data is structured of 30 samples by N lanes, where N is the number of lanes in the laneflow data table. In this model, the 30 samples per lane are summed to form one sample per 15-minute interval. For the 12 hours, the data is then 336 (= $4 \times 12 \times 7$) samples by N lanes.

The data table stores the unique model ID (which is used as the file name of the model in the model folder and stored at a fixed file path on the anomaly server), the trained parameters to run that model, the date/time/author of the trained model, the time information about the model, the revision status of the model, and the anomaly threshold used with this model. The data table is structured such that a new model on the same road could be retrained if traffic conditions change (such as for construction), and a construction model anomaly detector could be run and compared to the normal model anomalies. In this

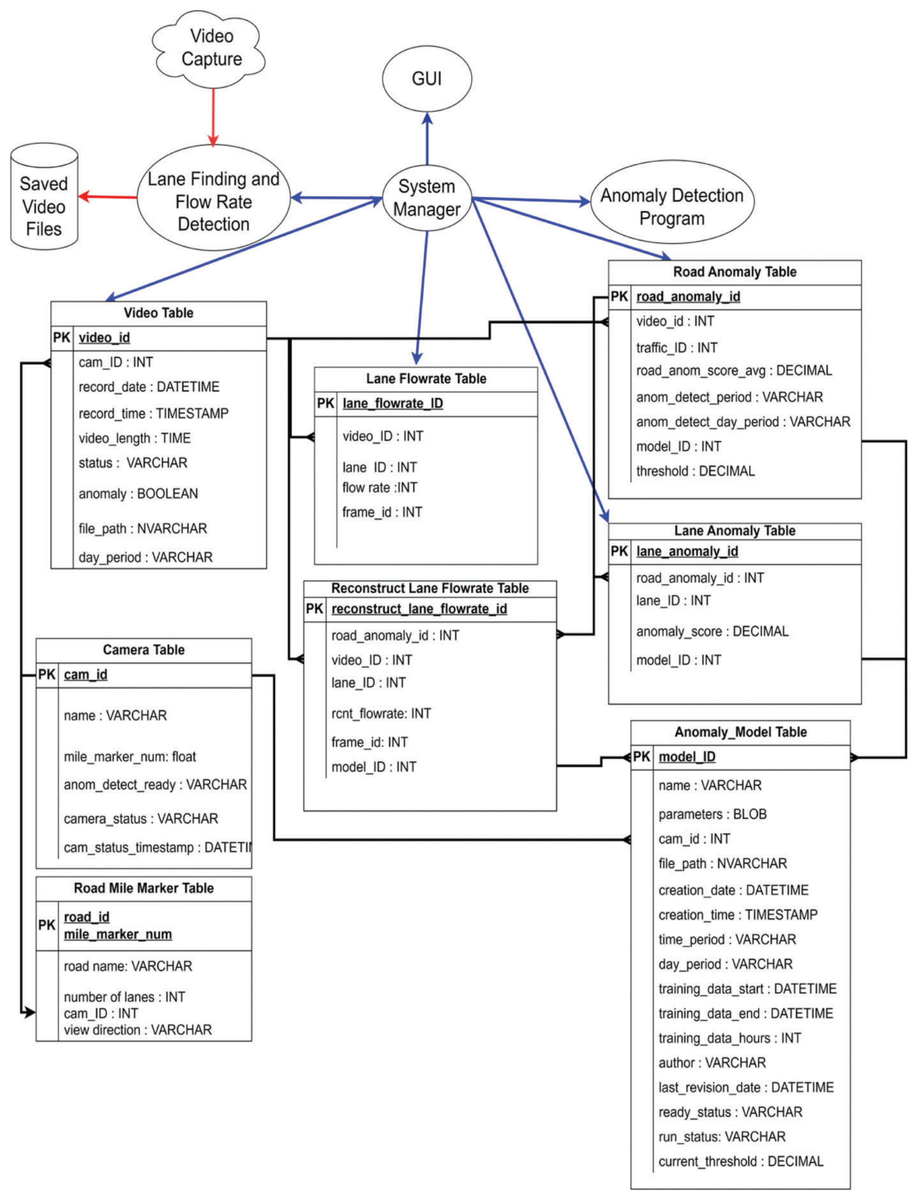


Figure 2.3 The new database section contains several tables in addition to the original laneflow information.

video_id	record_datetime	hour_label	30min_label	frameID	Lane:-5	Lane:-4	Lane:-3	Lane:-2	Lane:-1	Lane:1	Lane:2	Lane:3	Lane:4	Lane:5	Lane:6
video_9155	22:12:02	22	44	901	8	9	15	13	14	8	12	18	21	13	11
video_9155	22:12:02	22	44	1801	12	14	14	11	12	15	11	17	17	12	6
video_9155	22:12:02	22	44	2701	7	12	7	13	14	15	21	20	26	18	4
video_9155	22:12:02	22	44	3601	5	7	7	11	8	16	11	17	15	10	5
video_9155	22:12:02	22	44	4501	9	9	12	15	14	21	21	23	16	10	6
video_9155	22:12:02	22	44	5401	8	15	11	12	7	9	20	20	23	8	11
video_9155	22:12:02	22	44	6301	5	8	9	7	8	8	7	5	12	6	11
video_9155	22:12:02	22	44	7201	13	7	7	14	14	31	16	13	18	12	5
video_9155	22:12:02	22	44	8101	16	12	13	14	17	14	18	14	24	12	10
video_9155	22:12:02	22	44	9001	9	10	12	8	5	28	21	18	20	10	5
video_9155	22:12:02	22	44	9901	5	15	12	11	15	16	14	15	11	11	6
video_9155	22:12:02	22	44	10801	7	7	7	5	5	11	18	12	16	16	7
video_9155	22:12:02	22	44	11701	17	9	10	11	11	16	14	15	12	16	13
video_9155	22:12:02	22	44	12601	6	12	7	7	10	13	10	18	22	9	12
video_9155	22:12:02	22	44	13501	9	10	7	8	8	18	23	15	24	8	15
video_9155	22:12:02	22	44	14401	15	9	11	9	9	10	11	15	13	19	14
video_9155	22:12:02	22	44	15301	10	9	10	14	15	16	15	12	15	9	8
video_9155	22:12:02	22	44	16201	7	12	14	13	14	15	13	27	22	9	7
video_9155	22:12:02	22	44	17101	7	8	9	9	6	15	8	17	16	8	7
video_9155	22:12:02	22	44	18001	13	15	11	8	5	25	16	19	25	14	6
video_9155	22:12:02	22	44	18901	8	13	5	11	12	6	14	16	10	9	12
video_9155	22:12:02	22	44	19801	7	8	11	9	7	8	7	10	17	7	3

Figure 2.4 Example data table extracted from flow rate database.

project, each camera needs its own anomaly detection models. Model automatic generation and retraining is outside of the scope of this project but is envisioned in the future.

2.5.6 Road Anomaly Table

This table collects the data (input, reconstructed, anomaly), which will be provided to the GUI with the N-lanes averaged.

2.5.7 Lane Anomaly Table

This table collects the data (input, reconstructed, anomaly), which will be provided to the GUI with the N-lanes individual.

3. ANOMALY DETECTION

3.1 Model Development

Anomaly detection is a key contribution in this research. The system uses a Variational Auto-Encoder (VAE) structure, trained on the lane flow data for each given camera using historical data from that camera. The day and night data are handled in separate VAEs, as noted in Section 2.5.5. Each day or night data sample is a vector containing each lane's flow rate (number of vehicles by lane) for the 15-minute day/night patterns, or for the 1-week day/night patterns. The VAE learns these common patterns during training by comparing the input to the output reconstruction and iterating over many historical data samples. Common patterns are within the learned distribution of the data (in-distribution), while uncommon samples of data are where the reconstruction does not closely match the input, causing a large difference, and statistically out-of-the-normal distribution. The anomaly score is generated by normalizing the max difference on the interval selected. The anomaly threshold uses this score (set on a 0 to 1 scale, where closer to 1 is the most anomalous). The farther the vehicle flow rate is from the reconstructed flow rate of the distribution, the more likely it is to be considered an anomaly. The learned day or night VAE is then frozen (not continuous learning). The anomaly is then detected when the reconstructed output of the VAE is statistically different from the new input data.

The VAE structure is shown in Figure 3.1. In this figure, the input laneflow data is X, the output reconstruction is X'. The Mean Squared Error (MSE), or reconstruction loss L_R, is one component of the training error function. The other component of the training error function is the clustering loss, L_C, which is a statistical function which will tend to drive the latent vector, z, toward a random variable distribution (over many samples). The encoder and decoder functions are designed to lower the dimensionality of the data by linear and non-linear combining (convolutions) and force the latent variable, z, into a salient feature set. A key idea in the VAE is that it trains to minimize the total loss, therefore tends to create reconstructions that are statistically like the input. After sufficient training, the VAE can reconstruct with low error in normal cases. The anomaly is detected where the difference between the input and the reconstruction is dissimilar and is statistically out-of-distribution from the learned distribution.

This project experimented with the structure of each of the four VAE models during the training phase to find the best architecture. The historical laneflow training data was used to test each model, and the models that produced the lowest error reconstruction with respect to the input data were chosen. The number of hidden layers in the encoder and decoder is proportional to the model's complexity. The choice of the size of the latent variable, z, is where the model can reconstruct well, but does not over-fit (memorize data). A good model will have low complexity and low MSE. An example of the testing of various architectures is seen in Figure 3.2. In these experiments it was determined that the lowest MSE and complexity tradeoff for the 15-minute model was a latent vector size of 64 and 2 hidden layers each in the encoder/decoder of size 64. The overlapping of data during training was done to increase the training data set size. In the final testing, the window size is non-overlapped and set to the data for 15-minutes. It was also determined that there should be separate models trained for day/night, as the best performance/complexity tradeoff was found for separate models.

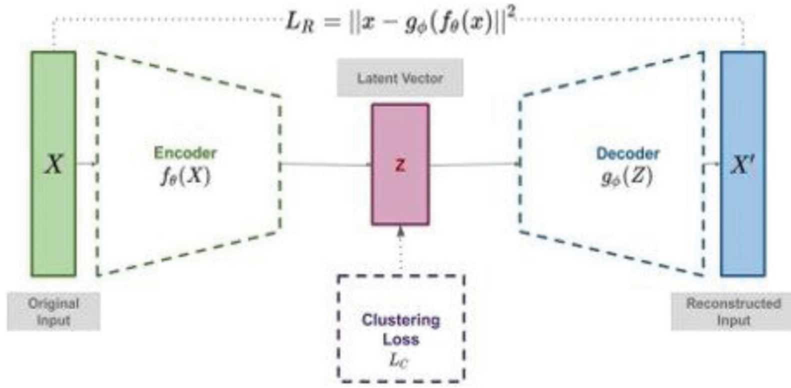


Figure 3.1 Variational Auto Encoder for detecting anomalies.

Cam ID	Data Time	Latent_Dim	Filters in hidden layers in both encoder and decoder	Window size & input size: overlapped window sampling for training	Test sample size(non-overlap window sampling)	Average MSE of all test data	Average MSE of lane 1	Average MSE of lane 2	Average MSE of lane 3	Average MSE of lane 4	Average MSE of lane 5	Detected anomaly sample index in test data
15	day	64	[64, 64]	30, [30,11]	28	1.07	0.7	0.93	0.81	0.93	1.57	24, 25
15	day	64	[64,64,64]	30, [30,11]	28	1.07	0.71	0.92	0.8	0.92	1.56	24,25
15	day	32	[64, 64]	30, [30,11]	28	1.09	0.8	0.93	0.86	0.95	1.6	25, 26
15	day	32	[64,64,64]	30, [30,11]	28	1.08	0.79	0.9	0.92	0.93	1.59	25
15	night	64	[64,64]	30, [30,11]	29	0.61	0.48	0.58	0.56	0.54	0.78	21, 26
15	night	64	[64, 64, 64]	30, [30,11]	29	0.62	0.46	0.57	0.58	0.55	0.77	21, 26
15	night	32	[64, 64]	30, [30,11]	29	0.65	0.48	0.61	0.61	0.63	0.89	8,26
15	night	32	[64, 64, 64]	30,[30,11]	29	0.66	0.51	0.61	0.61	0.6	0.89	8,26
15	day+night	64	[64, 64]	30, [30,11]	44	0.57	0.75	0.52	0.48	0.45	0.45	2, 24
15	day+night	64	[64, 64, 64]	30, [30,11]	44	0.59	0.79	0.6	0.45	0.56	0.46	0,2,24,40
15	day+night	32	[64, 64]	30, [30,11]	44	0.6	0.69	0.48	46	0.48	0.6	0,2,26,44
15	daty+night	32	[64, 64, 64]	30, [30,11]	44	0.59	0.78	0.53	0.48	0.47	0.49	0,2

Figure 3.2 Example of training AI architectures ablation study 15-minute models day/night (best performance in red).

In Table 3.1, the red text highlights the best MSE for these cameras.

Once the model architecture is chosen, and the weights of the convolutional layers are trained, the model is used to detect anomalies. The anomaly score uses the absolute difference between the input and reconstruction, divided by the maximum: $\frac{1}{max|X - X'|}|X - X'|$. The anomaly is normalized, so at 0: $X = X'$; and at 1: the anomaly is maximized. The anomaly threshold does not affect the underlying data, it identifies which data points are considered anomalous (above the threshold). The score is compared to the threshold to display the red colored anomalous result in the resulting graphic. The graphic presented to the user also shows the actual and the reconstructed values: X and X' as part of the bar chart in Figure 4.4. This allows the user to evaluate the anomaly and whether it is produced from a too high or too low value of lane flow compared to the learned, reconstructed distribution.

An example of detected anomaly by the algorithm is as follows; At some time before 16:30:35 PM, right two lanes on the right side of the road (up direction) had few cars (Figure 3.3a), which was abnormal. Later it was found that a truck slowed down and parked on the roadside (Figure 3.3b).

3.2 Anomaly Detection

In production, the user needs to define the camera input. The anomaly detection system will automatically learn the road layout, check for the match of the camera view to the preset direction, identify optimal areas (ROIs) for the best vehicle detection, and determine lane positions for real-time lane-wise vehicle counting. To detect anomalies, the system uses 15 minutes of video for lane-wise vehicle count generation, which is then processed by the camera's well-trained VAE model for reconstruction and error calculation. The reconstruction errors are normalized to a range of 0–1 using the maximum error from the training dataset. Finally, based on a predefined anomaly threshold for the camera, the system provides an anomaly detection result of "true" or "false."

Existing Limitations

1. *Anomaly Model Availability:* The anomaly detection model for the selected camera must be available already.
2. *Camera Quality and Positioning:* The accuracy of road layout learning and vehicle detection heavily depends on the quality and positioning of the cameras. Poor resolution, incorrect angles, or obstructions can significantly impact performance.
3. *Environmental Conditions:* Adverse weather conditions, such as rain, fog, or snow, can affect the visibility and accuracy of vehicle detection and lane counting.
4. *Lighting Variability:* Variations in lighting, especially at night or in low-light conditions, can hinder the system's ability to accurately detect and count vehicles.
5. *Traffic Density:* High traffic density and occlusions can cause difficulties in tracking individual vehicles, potentially leading to counting inaccuracies.

Figure 3.3 (a) Upper screen shot shows no vehicles on the right two lanes. (b) The lower screen shot shows a truck slowed down and parked on the roadside.

TABLE 3.1
Computational Performance with Recorded 15-Minute Video

Major Function	CPU Time
Laneflow	7.5 minutes per 15-minute recording
Anomaly Calculation	75 millisecond per 15-minute recording
Database/Web Server	Less than 1 millisecond depending on network performance

6. *Dynamic Changes:* Frequent and significant changes in camera angles or zoom levels require the system to re-learn the environment, which can temporarily disrupt real-time monitoring and counting.
7. *Computational Resources:* Real-time processing requires substantial computational power. Limited resources may affect the system's ability to maintain real-time performance, especially with high-resolution cameras or in dense traffic scenarios.
8. *Training Data Variability:* The VAE model's effectiveness depends on the diversity and representativeness of the training data. Insufficiently varied training data may reduce anomaly detection accuracy.
9. *Threshold Sensitivity:* The predefined anomaly threshold may need adjustments for different cameras and environments, requiring manual tuning to avoid false positives or negatives.
10. *System Scalability:* Scaling the system to monitor many cameras simultaneously may require significant infrastructure and could pose integration challenges.

The timing of running the program on the Lambda computer with one CPU and one GPU (Intel Xeon 5218/Nvidia Quadro RTX 5000) is shown in Table 3.1.

4. GUI INTERFACE

The goal of the System Manager and Database System is a unified suite of tools that INDOT operators can use to analyze traffic patterns. As such we've designed a landing page, seen in Figure 3.3, where operators can select which tool they wish to use by clicking the desired project button. As the toolset expands, we can group different systems into various categories or have the most crucial ones moved to the top by clicking favorites. More buttons will be added as past projects are added into the new web platform.

4.1 The Anomaly GUI

The main GUI structure is divided into two pages. The first takes input from the user on which camera, model, and time period to view and on which computer to run the anomaly software. This Setup Screen is shown in Figure 4.1. After this, the user goes to the next screen in Figure 4.1. The user follows these steps to setup and start the anomaly detection in Figure 4.1.

Step 1. Select the camera from the right-hand side camera list.

Step 2. Select from the drop-down menu in the red box on the left which model they want to run (e.g., 15 minute-day, 15 minute-night, etc.), or start data collection.

Step 3. Select which of the four possible actions they want to perform.

- 3.1 "Run Laneflow" starts accumulating the laneflow measurement alone. Program will resume from last unprocessed video file in time period by checking database for entry.
- 3.2 "Run Anomaly" starts the anomaly software as a long-term background process on a range of already collected laneflow data for the selected camera and model. The time range of data is automatically selected given the model selected (year model gets an entire years' worth of data to the present time for the anomaly model). Only processes data not already in database.
- 3.3 "Run Realtime" runs both laneflow and anomaly software in real-time so the user can immediately see anomalies populate in the results page, as in Figure 4.3.
- 3.4 "View Historical" simply queries the database for a selected time period and displays it to the user; no processes run in the background. A popup will ask for the correct time selection (shown below). Users can select from a predetermined amount of time (whole day, week or year) or specifically set the timespan to view.

If the selected actions 3.1, 3.2, or 3.3 are started by the System Manager, the box on the right of the button corresponding to the clicked button will change from "Ready" to "Success," else it will change to "Error." The success execution of the selected action 3.4 can be viewed in the task manager by clicking the task manager button in system utilities.

The data availability part of the window shows the percentage data missing for running the day, or week or year model. To determine the percentage of already processed data, we query the database for data starting from exactly 1 year, 1 week, or 1 day back to the present time, as shown in Figure 4.2. Then in Figure 4.3, the data availability is calculated, the percentage indicates how much of the data is unprocessed.

Users can view currently running programs by clicking the task manager button (Figure 4.1) in the system utilities tab. This initiates a popup screen that lists all connected servers currently running the Docker container management software and their tasks, as seen in Figure 4.3. The top boxes in red are the server names with the text just below indicating the current available resource capability of that server. For example: anomaly calculation uses fewer resources than lane flow, hence the lower capacity. Users can select which server they want to view by clicking the server boxes; the currently selected one will be outlined in blue. The table below the resources shows each individual Docker container running on the selected server. One container can run 1 action at a time (laneflow or anomaly), but multiple containers can run on a server simultaneously if the capacity is available. Figure 4.3 shows that container 1 is currently running laneflow counting on Camera 15 with the expected finish time being 15 minutes from now. These readouts will update as the program runs.

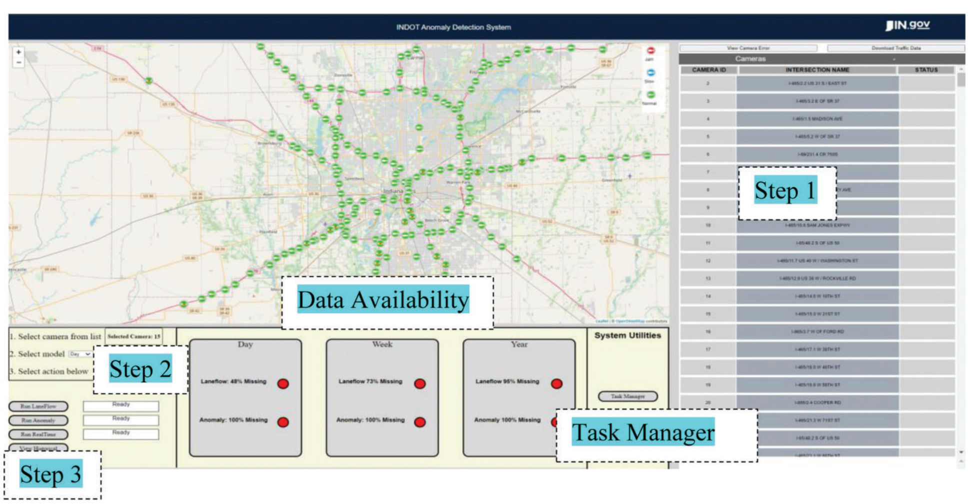


Figure 4.1 GUI anomaly detection setup screen.

Figure 4.2 Setting time duration for anomaly results.

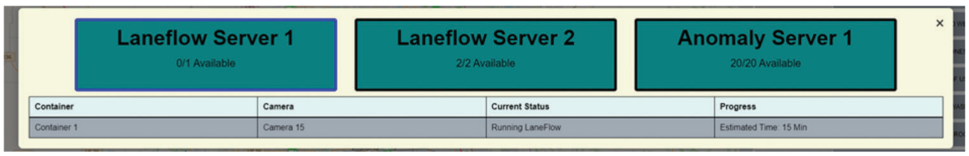


Figure 4.3 Server availability popup screen.

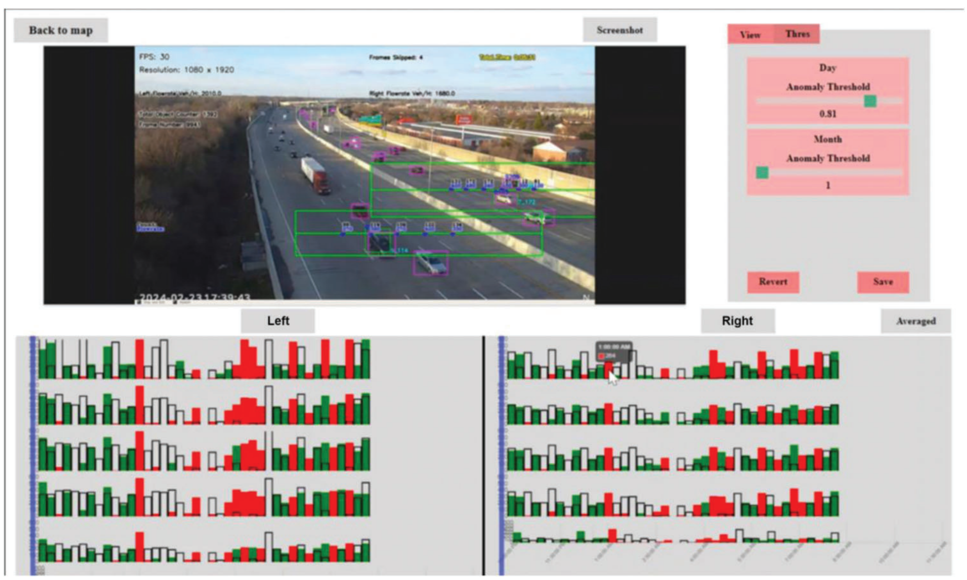


Figure 4.4 GUI with anomaly detections.

If the user selects *run real-time* or the *view historical button* in the main page (Figure 4.1), they are redirected to the results page (Figure 4.4), which displays the anomaly information. Each horizontal bar chart represents a single lane from the video (shown in the viewport in the middle of the screen). The left bar charts correspond to the lanes on the left side of the highway and the right bar charts corresponds to the lanes on the right side of the highway from the camera's point of view. Figure 4.5 shows an expanded view of anomaly (red) and actual (red/green) vs. reconstructed (black).

- The solid green/red bar colors are the true car count data collected from the video.
- The black outlined bars are the reconstructed car count data from our VAE model.
- The red outlined high bars in the chart indicating the anomaly.

The user can change what data is considered anomalous by modifying the anomaly threshold in the right menu. The anomaly score is generated by normalizing the max difference on the interval selected. The anomaly threshold uses this score (set on a 0 to 1 scale, where closer to 1 is the most anomalous) using the anomaly formula.

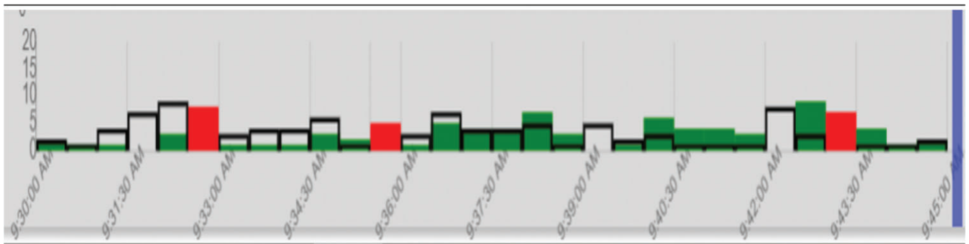


Figure 4.5 Expanded version of anomaly (red) and actual (red/green) vs. reconstructed (black).

$\frac{1}{max|X - X'|}|X - X'|$. X is the actual laneflow, and X' is the reconstruction from the VAE. The $\frac{1}{max|X - X'|}$ is used to normalize the anomaly score (0 to 1) over the selected interval of time.

5. CONCLUSIONS AND FUTURE WORK

This project has developed an anomaly traffic monitoring system for monitoring traffic conditions using the INDOT CCTV video feeds automatically in real time. More specifically, an AI-based deep learning algorithm, Variational Auto Encoder (VAE), was used for detecting anomalies. The system has been successfully tested during daytime, dawn, dusk, and on well-lit roads at night. A database was designed as the central place to gather and distribute the information generated from all camera videos. The traffic conditions generated from each camera video feed can be uploaded to the database in real time. The webpage GUI was developed to extract the information from the database and display the anomalies observed from each camera. The system will be installed at INDOT.

REFERENCES

Qiu, M., Chien, S., Mulay, A. A., Christopher, L., Ding, Z., Chen, Y., Sturdevant, J., & Cox, E. (2021). *Intelligent highway lane center identification from surveillance camera video* [Conference session]. IEEE International Conference on Intelligent Transportation, Indianapolis, IN, USA.

Qiu, M., Christopher, L., Chien, S. Y.–P., & Chen, Y. (2020). Intelligent highway adaptive lane learning system in multiple ROIs of surveillance camera video. *IEEE Transactions on Intelligent Transportation Systems*, *25*(8), 8591–8601.

Santhosh, K. K., Dogra, D. P., & Roy, P. P. (2021, November). Anomaly detection in road traffic using visual surveillance: A survey. *ACM Computing Surveys*, *53*(6), 1–26.

About the Joint Transportation Research Program (JTRP)

On March 11, 1937, the Indiana Legislature passed an act which authorized the Indiana State Highway Commission to cooperate with and assist Purdue University in developing the best methods of improving and maintaining the highways of the state and the respective counties thereof. That collaborative effort was called the Joint Highway Research Project (JHRP). In 1997 the collaborative venture was renamed as the Joint Transportation Research Program (JTRP) to reflect the state and national efforts to integrate the management and operation of various transportation modes.

The first studies of JHRP were concerned with Test Road No. 1—evaluation of the weathering characteristics of stabilized materials. After World War II, the JHRP program grew substantially and was regularly producing technical reports. Over 1,600 technical reports are now available, published as part of the JHRP and subsequently JTRP collaborative venture between Purdue University and what is now the Indiana Department of Transportation.

Free online access to all reports is provided through a unique collaboration between JTRP and Purdue Libraries. These are available at http://docs.lib.purdue.edu/jtrp.

Further information about JTRP and its current research program is available at http://www.purdue.edu/jtrp.

About This Report

An open access version of this publication is available online. See the URL in the citation below.

Christopher, L., Chien, S., Chen, Y., Qiu, M., Reindl, W., & Koshy, L. (2024). *Anomaly detection in traffic patterns using the INDOT camera system* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2024/30). West Lafayette, IN: Purdue University. https://doi.org/10.5703/1288284317778