Adaptive Traffic Signal Control Based on Real-Time Road Traffic Image Segmentation Calculation

Abstract

The research presents a novel approach for adaptive traffic signals based on real-time road traffic image segmentation calculation. The proposed method utilizes advanced image segmentation mode—Segment Anything Model (SAM) to segment traffic images in real time. The segmented images clearly show items on the road, with which we can evaluate vehicle density. Based on this important index, the model can finally enable the adaptive adjustment of traffic signal timings based on the current traffic conditions. By integrating computer vision and traffic engineering principles, the system aims to optimize traffic flow, reduce congestion, and enhance overall transportation efficiency. The experimental results demonstrate the effectiveness and potential of the proposed adaptive traffic signal control system in improving urban traffic management.

Keywords：image segmentation adaptive traffic signals SAM

1 Introduction

1.1 Efficient traffic management and challenges

The increasing number of vehicles on the road implies that our life has become more and more convenient. At the same time, population explosion leads to various traffic problems, which greatly affects the quality and efficiency of people's commuting. In many countries, the most noteworthy traffic problem is traffic congestion, especially during rush hours or holidays. Therefore, it is very necessary to effectively reorganize and transform traffic management to control the vehicle density of high-flow sections such as intersections.

Efficient traffic management is crucial for maintaining smooth and safe flow at busy intersections. One key aspect of traffic management is the control of traffic signal timings at intersections, which directly impacts the flow of vehicles and the overall transportation efficiency. The traffic signal system is an important traffic tool used to dredge traffic flow, improve road compacity and reduce traffic accidents. Signals consist of illuminated signs that operate periodically at specific time intervals. However, with the increasing number of vehicles on the roads and the complexity of traffic patterns, the traditional fixed-time cycle mode of traffic signals is no longer suitable for dynamically changing road condition. Therefore, we propose a new traffic signal system to address these challenges.

1.2 Research Objectives

Traditional methods used to detect traffic conditions commonly depend on road sensors. In this context, the research presents a novel approach for adaptive traffic signal control based on real-time road traffic image segmentation calculation. Recent decades have witnessed blooming development of various language models, among which we chose SAM to segment targeted pictures. In the research conducted by Deng et al. [22], image segmentation has already been applied to accurately recognize the number and letters on license plates, contributing to the construction of intelligent parking lot. Our paper lays emphasis on the macroscopic condition of traffic by focusing on traffic density. To build the calculation model, we are devoted to solving the following problems:

1. How can the image data be transformed to numerical index?
2. What is the connection between traffic condition and traffic signal control?
3. How to evaluate the efficiency of the calculation system?

By leveraging advanced image processing techniques, the proposed system aims to analyze traffic images in real time and make adaptive adjustments to traffic signal timings, thereby optimizing traffic flow, reducing congestion, and improving overall transportation efficiency.

Adaptive traffic signal control based on this calculation system is foreseen to detect traffic conditions dynamically. Also, this model provides a platform for further research and is friendly to portability.

2 Review

Several comprehensive traffic signal adaptive patterns have been proposed in the past. Hawi et al. (2017) presented the design and implementation of a smart traffic light system using fuzzy logic and a wireless sensor network (WSN) [10]. This system, designed for four-way roundabouts, incorporates WSN to collect real-time traffic data. However, the system has not been validated under real road conditions.

Another dynamic traffic light system was proposed by Ashraff Mohd et al. (2017), using LoRaWAN for congestion monitoring [16]. However, this system is costly and requires large physical space, which may not be feasible for implementation on roads. Pratama et al. (2018) presented the design and implementation of a smart traffic light system that uses bilateral filtering image processing technology and special hardware to control the traffic signals [17]. The adaptive signal logic is based on input and output lane density. This approach introduced an innovative method of calculating traffic density through image processing in 2018, compared to previous studies. Traffic density is obtained by counting the number of black and white pixels in the processed image. This system had been tested in real world and yielded good results. Also, it proves that the image processing can be an important part in traffic control.

From the researches above, to build an adaptive traffic signal system, data collection and the selection of appropriate algorithms for signal control are essential.

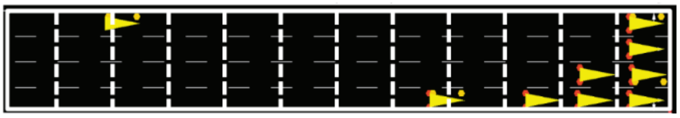
2.1 Traffic controlling

2.1.1 Common Traffic Condition Data Used in Adaptive Traffic Signal Control

Indexes used in traffic signal timing problems vary a lot, such as signal timing saturation and traffic volume ratio. Current fixed signal timing patterns are based on these data. Recent researches on adaptive traffic control have focused on obtaining both traditional and new data. Two important and relevant adaptive traffic control systems are described below.

Split Cycle Offset Optimization Technique (SCOOT) utilizes on-street detectors embedded in the road. When vehicles pass the sensors, SCOOT collects real-time traffic data and uses them to construct traffic flow models to estimate queue length [2].

InSync is composed of an IP video camera and a processor. It counts the number of vehicles and the delay experienced in order to optimize phasing. Queue length estimation plays a significant role in adaptive urban traffic signal control and has garnered attention in recent decades.



(a)

0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	0	1	1	1

(b)

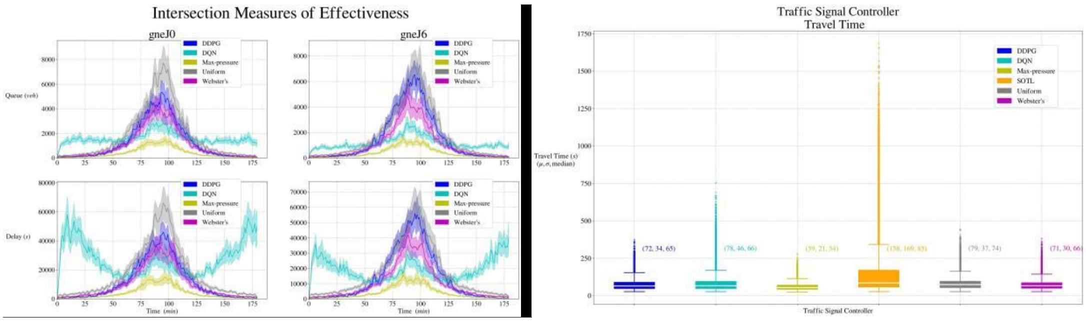
Figure 2-1: (a) snapshot of traffic at road. (b) matrix of vehicle position.

2.1.2 Review of traffic control algorithms at intersections

The process of obtaining traffic condition should also align with the traffic control algorithms. The data type of DQN adaptive traffic control is unique (Gao et al., 2017) [7]. Real-time traffic conditions are essential for DQN and related DPG learning. The input data must take the form of matrix (Figure 2-1). The research is conducted using SUMO, a traffic simulator. However, in the real world, acquiring such data may be time-consuming, and lane division needs to be accurate. Moreover, DQN can be inflexible in the long term unless it is frequently trained. Additionally, reinforcement learning is not suitable for this paper as image processing is typically not capable of accurately measuring length.

Another traffic signal control method is Max-Pressure (MP). In an urban network, with a predefined phase sequence and minimum/maximum green times for each phase, MP dynamically adjusts phase times for all intersections in real-time. It decides whether to extend the currently active phase or activate the next predefined phase at each time step. Consequently, the resulting signal cycles evolve dynamically [11]. MP demonstrates good efficiency. There are five mainstream algorithms: DDPG, DQN, Max-Pressure, Uniform, and Webster's. Depending on different situations, they produce varying outcomes. Genders et al. (2019) tested these algorithms using SUMO [8]. It appears that Max-Pressure yields the best results in terms of travel time and queue length (Fig2-2). Moreover, the queue length around the intersection can be easily obtained through image segmentation.

Overall, if a semantic algorithm can get the queue length in a relative short time with a high accuracy, it can be applied in the MP control system, and in a rapidly developing world, this is just the first step towards a highly efficient adaptive traffic signal system. The intersection needs a comprehensive semantic algorithm.



(a) (b)

Figure 2-2:(a) Comparison of traffic signal controller individual intersection queue and delay measure of effectiveness in units of vehicles and seconds(s). (b) Solid colored lines represent the mean and shaded areas represent the 95% confidence interval.

2.2 image segmentation

In recent years, with the continuous development of machine learning technology, it is more and more common that image segmentation is widely used in fields such as medical imaging, automatic driving and speech recognition. This part will analyze the application of deep learning in traffic.

2.2.1 Achievements of deep learning in the field of transportation in recent years

Deng et. al. (2017) [3] described a CNN-based semantic segmentation solution using fisheye camera which covers a large field of view. In the years since, the traffic image semantic segmentation plays a crucial role in automatic driving.

Wang et. al (2023) [19] studied a traffic image semantic segmentation algorithm based on UNET. A semantic segmentation algorithm based on UNET network model is proposed for getting better results in traffic images segmentation.

A simple and efficient geometry-sensitive energy-based loss function is proposed to Convolutional Neural Network (CNN) for multi-class segmentation on real-time traffic scene understanding (Feng et. al., 2023 [6]). The approach can be applied to different segmentation-based problems, such as urban scene segmentation and lane detection， which has been widely used in recent studies.

The most recent research result is (Kirillov et al., 2023 [12]) the Segment Anything (SA) project: a new task, model and dataset for image segmentation. SAM has pushed the boundaries of segmentation and significantly advanced the development of basic computer vision models.

2.2.2 Analysis and comparison of existing algorithms

In recent years, several architectural models of deep learning algorithms have been proposed, such as CNN, SSD, YOLO, SAM, etc., each model has its own advantages and disadvantages, particularities and ways of use.

Judging by its speed, R-CNN takes 40 to 50 s to calculate the result for several new images.YOLO uses a unified and fast object detection approach, which uses images in real-time at 45 frames per second (FPS) and an mAP of 63.4%. This model learns faster, surpassing other detection methods such as R-CNN and Faster R-CNN. Through the experiment, the SAM model takes 1.5-2.5 seconds to decompose an image on average. In contrast, YOLO is faster and more efficient, which is where the SAM model needs to improve.

In terms of accuracy, Models of two-stage algorithms provide more appropriate precision. By contrast, some models of one-stage DL algorithms, such as in the case with SSD and YOLO, did not achieve sufficient accuracy at the same time. But the SAM model has improved significantly in this respect. SAM contains 11 million diverse, high-resolution, permissioned images and 1.1 billion high-quality segmentation masks that have been verified by human ratings and numerous experiments to be of high quality.

From the detection area,Fast R-CNN presents the disadvantage of using the selective search feature to detect regions of interest in the images. SAM model overcame this difficulty and can accurately capture the position and outline of an object (i.e. in the form of a mask), thus distinguishing between various objects in the foreground and the background.

2.2.3 Why we choose the SAM model

After synthesizing the applicability of several current algorithms,our research selects the image segmentation technology based on SAM model.

First of all, Segment Anything Model it has great efficiency in segmenting the picture at the pixel level. This may provide so many high quality sementic pictures of the intersections, which may also be a great resource for the self-driving cars and the traffic management to use.

Secondly, SAM has demonstrated excellent zero sample performance in a variety of segmentation tasks, making it an immediate tool with low barriers to use and convenient enough to apply our research findings to more scenarios. And because of its high quality of generalization,it can handle the emergency,which is important in the traffic management.

In addition, by applying the SAM model to bounding box annotations, it can generate mask annotations for large video text datasets. This helps us to perform image segmentation of surveillance video to assess traffic flow.

Finally, the dataset introduced by SAM, as the largest dataset available, is able to adapt to changing traffic environments, providing a stronger guarantee for the accuracy of the experimental results.

Taking advantage of SAM model’s advancement, the video or picture of the road condition transmitted back from the monitoring can be segmented in real time. According to the results of image segmentation, we can quickly calculate the traffic flow information, and then judge the current road flow condition according to the existing traffic flow threshold - if the traffic flow is too dense or too low, we can flexibly extend or shorten the traffic light time at the intersection, so as to improve the traffic congestion problem on urban roads.

3 Methodology

3.1 Segment anything model (SAM)

3.1.1 Segment anything model architecture

In this section, we will describe the architecture of the Segment Anything Model (SAM). SAM has three components, illustrated in Fig.3-1: an image encoder, a flexible prompt encoder, and a fast mask decoder. It is based on Transformer vision models with specific trade-offs for real-time performance. We will describe these components in the follows.

Image Encoder. Motivated by considerations of scalability and effective pre-training methods, SAM employs a pre-trained Vision Transformer (ViT) with a minimal adaptation to handle high-resolution inputs. Specifically, SAM uses a Modified AutoEncoder (MAE) pre-trained ViT, optimized for processing high-resolution inputs. The image encoder, which runs once per image, is capable of processing high-resolution inputs efficiently and is applied before prompting the model.

Prompt Encoder. SAM utilizes two sets of prompts: sparse prompts (points, boxes, text) and dense prompts (masks). For sparse prompts such as points and boxes, positional encodings are combined with learned embeddings for each prompt type. Dense prompts, represented by masks, are embedded using convolutions and summed element-wise with the image embedding.

Mask Decoder. The mask decoder efficiently maps the image embedding, prompt embeddings, and an output token to generate a segmentation mask. Inspired by prior work, SAM employs a modified Transformer decoder block, followed by a dynamic mask prediction head. The modified decoder block incorporates prompt self-attention and cross-attention in two directions, updating all embeddings. After running two blocks, the image embedding is upsampled, and an MLP (Multi-Layer Perceptron) maps the output token to a dynamic linear classifier. This classifier computes the mask foreground probability at each image location.

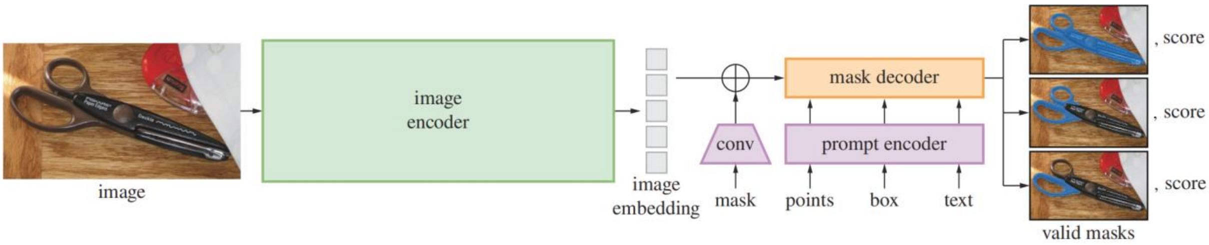


Figure 3-1: Segment Anything Model (SAM) overview. A heavyweight image encoder outputs an image embedding that can then be efficiently queried by a variety of input prompts to produce object masks at amortized real-time speed. For ambiguous prompts corresponding to more than one object, SAM can output multiple valid masks and associated confidence scores.

Handling Ambiguity. SAM addresses ambiguity in segmentation by modifying the model to predict multiple output masks for a single prompt. During training, backpropagation is performed only for the minimum loss over the masks. The model ranks masks by predicting a confidence score (estimated Intersection over Union-IoU) for each mask, allowing for the selection of the most appropriate segmentation result.

Efficiency. The overall design of SAM is heavily motivated by efficiency. Given a precomputed image embedding, the prompt encoder and mask decoder can run in a web browser on CPU in approximately 50 milliseconds. This runtime performance facilitates seamless, real-time interactive prompting of the model.

Losses and Training. SAM is supervised for mask prediction using a linear combination of focal loss and dice loss. Training for the promptable segmentation task involves using a mixture of geometric prompts, and an interactive setup is simulated during training to ensure SAM seamlessly integrates into the data engine.

This comprehensive structure enables SAM to perform a precise segmentation based on a variety of prompts while maintaining real-time efficiency.

3.1.2 Dataset description

The dataset used in this study consists of images capturing traffic flow at various intersections. These images were collected from publicly available datasets of surveillance cameras installed at different traffic intersections. Each image in the dataset represents the vehicular traffic at a specific time and location.

The dataset contains a total of 500 images and covers a diverse range of traffic scenarios, including peak hours and off-peak hours. This diversity enables the model to learn and adapt to different traffic patterns and conditions.

The dataset is intended to be used for the development and training of the segment anything model that can accurately estimate traffic flow and density. The ultimate goal is to utilize this model for real-time traffic signal control, allowing for efficient and dynamic traffic management.

Overall, the dataset provides a comprehensive and representative collection of traffic flow data, which can be leveraged for various applications in traffic engineering and management.

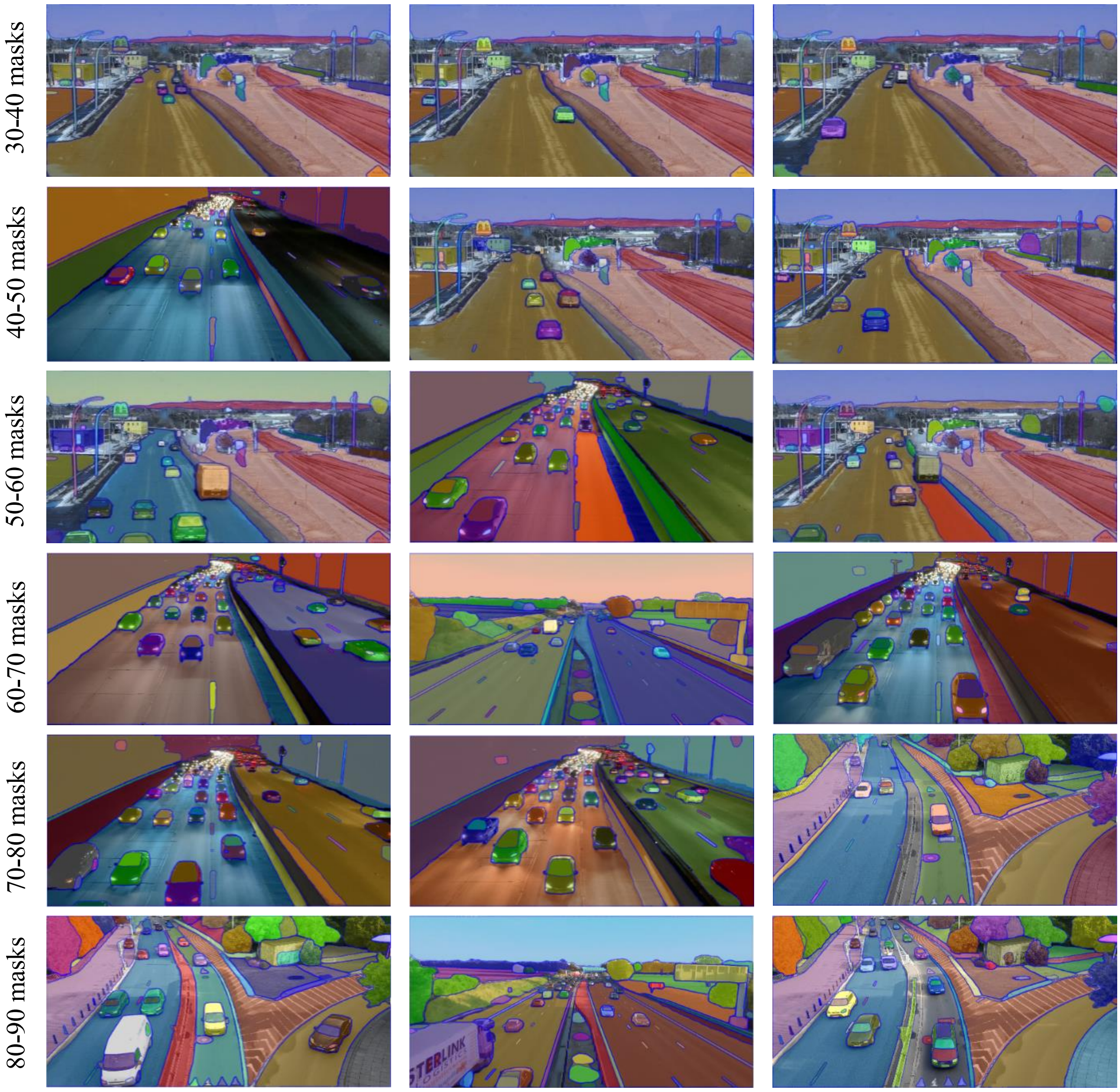


Fig 3-2: Here are segmented sample images from our self-created dataset for this project, featuring overlaid segmentation masks. This dataset comprises 500 diverse, high-resolution images. These masks were fully annotated automatically by SAM and have been verified through human ratings and multiple experiments, confirming their high quality and diversity. For visualization purposes, we have grouped the images based on the number of masks per image.

3.2 Metrics for evaluation of segmentation

In this section, we focus on comparing the performance and evaluation metrics of the image segmentation model SAM under conditions of limited sample training. The chosen task involves the adaptive analysis of real-time traffic flow, wherein the pre-trained SAM model is initially employed for image segmentation in specific traffic scenarios. Subsequently, leveraging both the previously trained data and the actual traffic flow values, a curve function representing the traffic flow is fitted. This function is then utilized to incorporate the segmented data, providing real-time insights into the traffic flow conditions. The subsequent content will particularly elaborate on the evaluation methods for segmentation performance.

Table3-1: The various performance metrics for image segmentation and their detailed explanations

Measure name	Formula
Intersection over Union(IoU)	$\frac{TP}{TP+FP+FN}$ *ration* of predicted and ground truth mask over the union of predicted and ground truth mask
Dice Coefficient(DC)	$\frac{2\cdot TP}{2\cdot TP+FP+FN}$ *ration* of correctly classified pixels compared to the total number of pixels in the image. It calculates the overall accuracy of the segmentation
Preision(P)	$\frac{TP}{TP+FP}$ *proportion* of true positive predictions relative to the total number of positive predictions
Recall(R)	$\frac{TP}{TP+FN}$ *proportion* of true positive predictions relative to the total number of ground-truth predictions
$\mathrm{F1-score(F1)}$	$\frac{2\cdot P\cdot R}{P+R}$ *harmonic* mean of precission and recall; meaningful when there is imbalance between positives and negative ground-truth *samples*

In the realm of real-time image segmentation for automatic traffic light control, assessing the temporal efficiency becomes paramount. The temporal aspect is crucial as it directly influences the responsiveness of the system in adapting to dynamic traffic scenarios. One pertinent metric for evaluating the real-time performance is the time elapsed from the capture of a road traffic image to the point where the segmented output, indicating the flow of vehicles, is generated.

Real-Time Responsiveness Metric. The Real-Time Responsiveness Metric (RTRM) quantifies the efficiency of the segmentation algorithm in terms of time. This metric is defined as the duration taken by the system to process the input image and produce the segmented output, specifically representing the vehicular flow. Mathematically, it can be expressed as: $\mathrm{RTRM} = T_{output} + T_{input}$, where:

T_{input} is the time taken from capturing the traffic condition image to uploading the image to the SAM model.

T_{output} is the time taken from the completion of segmentation by the SAM model to the output of traffic flow.

The lower the RTRM value, the higher the real-time responsiveness of the segmentation system. A reduced RTRM signifies that the algorithm can swiftly process the input data and promptly adapt the traffic light control strategy to prevailing road conditions. This metric serves as a vital indicator of the system's ability to meet the stringent temporal requirements of dynamic traffic management, ensuring seamless and adaptive traffic control.

By incorporating the Real-Time Responsiveness Metric into the evaluation framework, the efficacy of automatic traffic light control systems can be comprehensively assessed, offering insights into their responsiveness in real-world traffic scenarios.

3.3 Experimental setup

In this section, we employ the pre-trained SAM model for real-time segmentation of images at traffic intersections, obtaining segmentation results and real-time data. Subsequently, we conduct experimental evaluations of the performance metrics mentioned in Section 3.2 and analyze the real-time outcomes.

3.3.1 Experimental results

These image segmentation results are summarized in the Table 3-2, presenting key metrics including IoU (Intersection over Union), TP (True Positive), FP (False Positive), FN (False Negative), Precision (P), F1 Score, Dice Coefficient (DC), and Recall (R).

The mean IoU of 0.9836 indicates a high degree of overlap between predicted and ground truth segmentation. Precision (P), F1 Score, Dice Coefficient (DC), and Recall (R) also demonstrate consistent high values, around 0.9917 on average. The standard deviations (Std) are relatively low, suggesting stability and consistency across evaluations.

Examining the variability, the metrics' standard deviations indicate a consistent performance. The maximum and minimum values provide a range perspective. For instance, the model achieved a maximum IoU of 0.9965 and a minimum of 0.9590, showcasing the model's capability across different scenarios.

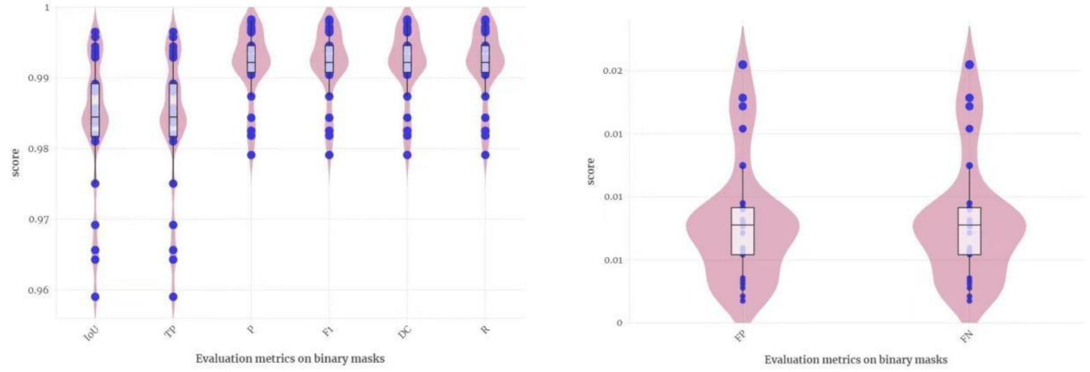
In conclusion, the results illustrate a robust and accurate segmentation performance, with high consistency and reliability across various evaluation scenarios.

Table 3-2:The average, standard deviation, maximum value, and minimum value of various performance evaluation indicators.

indicator	mean	std	max	min
IoU	0.9836	0.0100	0.9965	0.9590
TP	0.9836	0.0100	0.9965	0.9590
FP	0.0070	0.0038	0.0205	0.0012
FN	0.0094	0.0052	0.0312	0.0018
P	0.9929	0.0081	0.9981	0.9743
F1	0.9917	0.0094	0.9971	0.9689

DC	0.9917	0.0057	0.9968	0.9765
R	0.9905	0.0048	0.9921	0.9634

From Figure 3-3 (a), it can be observed that the differences between the lower and upper quartile of each metric are very small, indicating a high-quality data distribution with no significant deviation in numerical groups and no presence of outliers. Additionally, in Figure 3-3 (a), the median values are consistently high, directly implying a high accuracy of the segmented images, demonstrating strong adaptability and generalization capabilities. Moving on to Figure 3-3 (b), the small values for the upper quartile boundaries of FP and FN highlight the effectiveness of the model's segmentation capabilities, laying the foundation for subsequent traffic flow calculations. Furthermore, there are no instances of outlier occurrences in the data distributions, reflecting a high level of robustness in the experimental results.



(a) Box Plots of IoU,TP,P,F1,DC and R Metrics (b) Box Plots of FP and FN Metrics

Fig3-3: Box Plots of various Metrics

From the real-time segmentation time data (see Table 3-3), it can be observed that the entire real-time segmentation process takes approximately 2 seconds, achieving a relatively high temporal precision. This meets the requirements for real-time performance to a considerable extent. Moreover, across a substantial number of repeated experiments, there is minimal fluctuation in the duration of the two time intervals, indicating strong stability. Overall, it can be concluded that the system meets the latency requirements for assessing traffic road conditions in real-time.

Table 3-3:The processing time statistics for real-time image segmentation

indicator	mean	std	max	min
T_{input}(s)	0.836	0.791	0.981	0.683
T_{output}(s)	1.253	1.963	1.598	1.098

3.3.2 Comparative experiment

In this section, we will address how to convert the number of masks output by the SAM model into actual traffic flow. After repeated modeling, we provide the model function and validate the accuracy of the established model.

From Figure 3-4, it is evident that the number of masks output by the SAM model is not in a linear relationship with the actual traffic flow; instead, it follows a nonlinear functional form. Moreover, as the number of masks increases, the slope continuously decreases. In the curve fitting modeling, actual data points are closely clustered around the fitted curve, with no significant deviation points, reflecting the strong adaptability of the mathematical model.

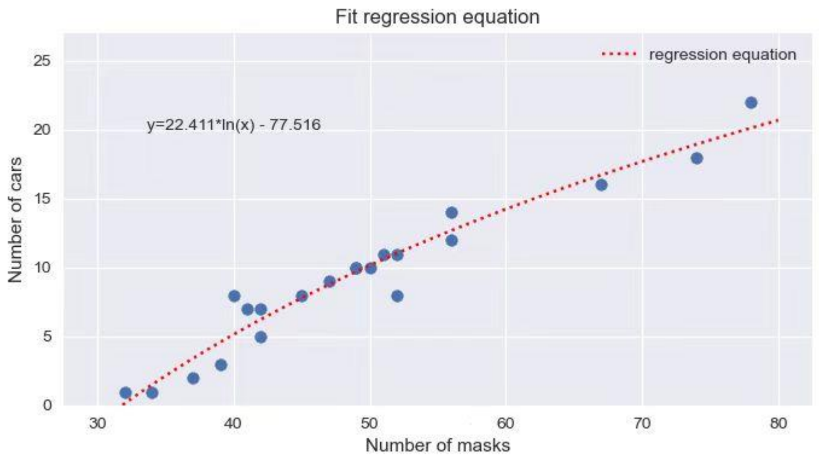


Fig 3-4: The fitted curve between the masks output by SAM and the actual traffic flow

Figure 3-5 clearly demonstrates the accuracy of the fitted curve after experimentation. The predicted traffic flow is nearly equal to the actual traffic flow, with a very small difference in the number of discrepancies. The error rate is also controlled within 13%. From a testing and validation perspective, this reflects the feasibility of the model.

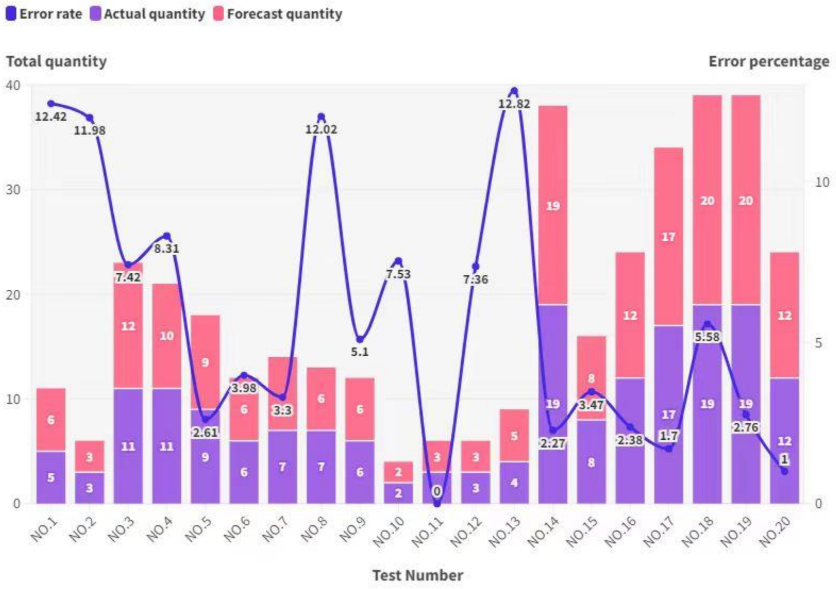


Fig 3-5: Summary of testing and validation data for the fitted curve

3.4 Data analysis

In the data analysis section, we will first conduct a detailed analysis of some data and phenomena observed in the above experimental results and experimental validations. This primarily includes an analysis of segmentation performance and the time delay differences in real-time control.

3.4.1 Segmentation performance analysis

In this section, we will conduct a detailed analysis of some underlying issues in the experimental results mentioned above:

1) Why does the slope of the fitted curve decrease as the number of masks increases?
2) Why is the error rate curve lower when the number of masks is higher?

After extensive analysis of a large volume of experimental data and a detailed examination of the SAM model, it is concluded that in scenarios with lower traffic flow, the captured traffic state scenes are relatively uniform. The trained SAM model tends to focus its attention on finer details rather than considering a complete entity. For instance, it may segment a car into more detailed components such as windows, body, and tires. Consequently, the rate of increase in traffic flow is smaller than the rate of increase in masks, leading to a decrease in the slope of the fitted curve as the number of masks increases. In scenes with a higher number of masks, the model tends to prioritize overall segmentation, resulting in more accurate predictions of traffic flow along the fitted curve.

3.4.2 Real-time performance analysis

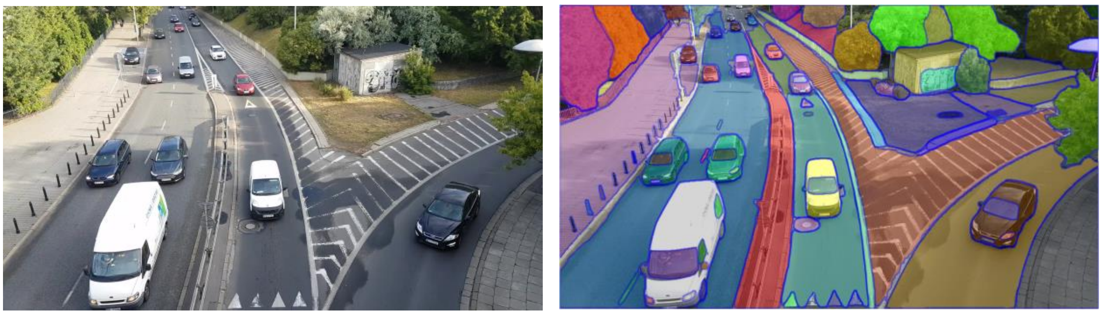
we will discuss the time-consuming issues in the real-time control process:

1) Why is the time taken from image segmentation to the output of traffic flow longer than the time taken from capturing traffic condition images to uploading them to the SAM model?

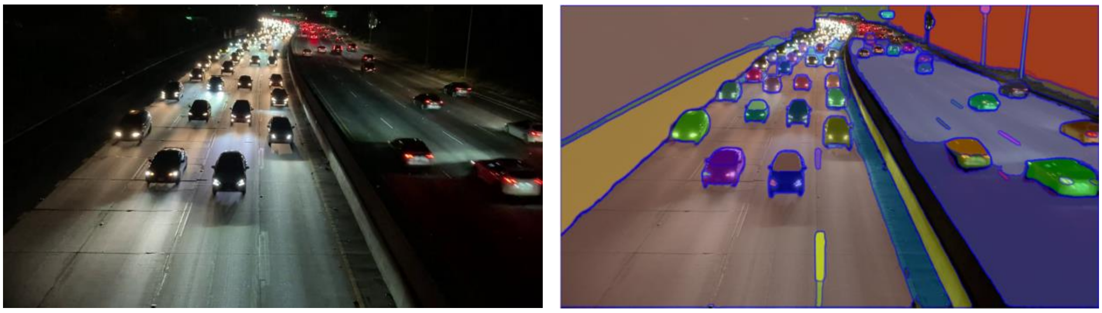
2) Why does the overall processing time increase as the number of masks within a certain range decreases?

After analyzing the data obtained from repeated experiments on the entire process from input to output traffic flow, it is observed that the reliability of hardware devices plays a more significant role during the image capture to SAM model input process. Additionally, since the volume of image data is relatively small, the time required is minimal during this phase. In the entire process of segmenting input images, extensive probabilistic calculations are performed on each pixel block of the image. Although the current computing speed has reached a relatively fast level, it still requires a certain amount of processing time. Therefore, the time consumed in this part is less than the first phase.

Furthermore, within a certain range of traffic flow, as the traffic flow decreases, the SAM model's region of interest becomes more refined. Attention is directed towards more intricate scenes that are harder to identify, leading to an increase in the overall processing time of the entire process.



(a) masks=91, cars=13



(b) masks=70, cars=36

Fig 3-6: Images (a) and (b) are schematic representations of masks and actual traffic flow before and after image segmentation, respectively. The comparison illustrates how the number of masks increases as traffic flow decreases within a certain range.

3.4.3 Areas for improvement

In this project, we only want to count the numbers of cars' masks. However, SAM not only generate masks for cars, but also create masks for other irrelevant objects, which greatly increases the difficulty of counting. So maybe a way to filter out non-car objects need to be found aiming to improve the methodology.

Iraklis Gianakis et.al point out that each mask can be classified into different categories based on geometric features. In the previous part SAM has been employed to extract the segmentation masks of different morphological features for an input image. Since the majority of cars have a rectangular shape, it is rational choice to filter out all the segmentation masks that are not rectangular. And then Canny filter can be applied to each one of the remaining masks, and the edges are fitted with rectangle. We predict that after this process, most of irrelevant objects can be filtered out like street lamps and street trees. But some masks of small-sized objects like rearview mirror tend to be approximate squares and therefore may mistakenly be classified into cars. To overcome this, a threshold needs to be added to remove certain masks whose area are smaller than a specific number of pixels.

Another way that might work to filter out irrelevant objects is introducing yolov8 (or any other detector) model into the project. We can use yolov8 to generate bounding boxes with classes and then automatically apply those classes to the masks generated by SAM. The number of cars will come out at the same time.

Further research and improvements need to be implemented to enable the methodology to effectively and accurately output the number of cars in the image.

4 Conclusion and Discussion

4.1 Discussion

4.1.1 Differences from traditional traffic detection models

As is mentioned in the conclusion, our model breaks the traditional framework of traditional traffic detection models which used detectors and other devices. Web-connected cameras can capture all essential elements of information, enabling convenient expandability to meet potential needs of analysis. Thus, the application prospect of adaptive traffic signal control based on real-time road image segmentation is promising. Also, traffic density derived from the segmentation of real-time photos can reflect traffic conditions more precisely, which exerts a guiding significance on signal setting.

4.1.2 Flaws and limitations existing in our model

There remains flaws and limitations in our model. When we look at the images processed by SAM, we can see that it tends to divide the entire car into multiple parts, resulting in increased masks. The increase of masks not only increases the error, but also prolongs the segmentation time. Moreover, in the case of data source, the scene we focus on is typical, but complex conditions in real life require classified discussion. In the case of framing, split pictures per second should be improved to serve a better real-time mechanism.

4.1.3 Suggestions

In the next stage of research, more detailed interference factors will be considered in our model. We will collect more data at various open webcam websites and carry out field investigation to enrich our database. The framing capability will be further improved. Our team will dig deep into our segmentation model to further polish the traffic signal control mechanism.

4.2 Conclusion

4.2.1 Summary of the purpose, methods and results of the study

In this paper, we conducted a research on the implementation methods of adaptive traffic signal control and found that image segmentation models have broad prospects for controlling dynamic traffic. Firstly, we collecting massive datasets of real-time traffic scenes and put them in the Segment Anything Model, from which we get the processed image with masks. To make the results clearer, we prepare validation sets. By comparing the training data with the actual data, it can be concluded that the segmentation accuracy of SAM on the road traffic flow is almost more than 98% and the error is less than 2%. Based on existing data, we get the connection between the number of masks output by SAM and the actual number of vehicles by testing quite a few regression functions, among which log-type fits best. According to the regression curve, we find that the error rate comparing the actual quantity and forecast quantity is within the acceptable rate,13%. In conclusion, the results of our evaluation metrics illustrate a robust and accurate segmentation performance, with high adaptability and generalization capabilities in various conditions, which reflects the feasibility of the model.

4.2.2 Significance of our model

SAM has a wide range of applications due to its segmentation properties, which enables the model to learn and adapt to different traffic patterns and conditions.

Our paper chose to lay emphasis on the macroscopic condition of traffic by focusing on traffic density. Through our research, we can let the combination of computer vision and traffic engineering have further development and application. We can use this powerful tool to dynamically regulate the signal of traffic lights, increasing the efficiency of people's commuting, avoiding traffic congestion, ensuring traffic safety in response to accidents by adjusting quickly and ultimately improving the bearing capacity of the road.

4.2.3 Merits of our model

For the advantages, this model can improve the performance of detectors in object detection, simplifying the operation, having a wide range of scenarios and decreasing the cost as no additional hardware or extra physical space is required. From the results of our experiment, the image segmentation model performs a high accuracy by providing high-quality segmentation masks and capturing clear position and outline of an object within an acceptable time frame. As an immediate tool with low barriers, the image segmentation model has the ability to adapt to different scenes and conditions, and can provide stable performance under different brightness and weather. Conventional methods with road sensors, on the other hand, may require different hardware configurations to suit different environments.

4.2.4 Applicable scene of our model

The usage of our model is simple. all we need to do is deploy the software with cameras at the traffic junction where the traffic lights can sense it, and then the video or picture of the road condition transmitted back from the monitoring can be segmented in real time. This method of combining image segmentation model with real-time traffic light control is in line with the development trend of smart cities. It can be integrated with other city management systems, such as intelligent traffic management systems, urban planning, and early warning systems, resulting in more comprehensive city management and planning.

5 References

[1] Ahmed, A., Outay, F., Farooq, M. U., Saeed, S., Adnan, M., Ismail, M. A., & Qadir, A. (2023). Real-time road occupancy and traffic measurements using unmanned aerial vehicle and fundamental traffic flow diagrams. Personal and Ubiquitous Computing, 27(5), 1669–1680. https://doi.org/10.1007/s00779-023-01737-w

[2] D. Robertson and R.D. Bretherton, "Optimizing networks of traffic signals in real time - the SCOOT method", IEEE Trans. on Vehicular Technology, vol. 40, no. 1, pp. 11-15, 1991.

[3] Deng, L., Yang, M., Qian, Y., Wang, C., & Wang, B. (2017). CNN based semantic segmentation for urban traffic scenes using fisheye camera. 2017 IEEE Intelligent Vehicles Symposium (IV), 231-236.

[4] Ee Heng Chen, Hanbo Hu, Zeisler, J., & Burschka, D. (2020). Pixelwise traffic junction segmentation for urban scene understanding. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), 8 pp.

https://doi.org/10.1109/ITSC45102.2020.9294654

[5] Ess, A., Mueller, T., Grabner, H., & Gool, L.V. (2009). Segmentation-Based Urban Traffic Scene Understanding. British Machine Vision Conference.

[6] Feng, Y., Lan, Y., Zhang, L., & Xiang, Y. (2023). Elastic Interaction Energy Loss for Traffic Image Segmentation. ArXiv, abs/2310.01449.

[7] Gao, J., Shen, Y., Liu, J., Ito, M., & Shiratori, N. (2017). Adaptive Traffic Signal Control: Deep Reinforcement Learning Algorithm with Experience Replay and Target Network. ArXiv, abs/1705.02755.

[8] Genders, W., & Razavi, S.N. (2019). An Open-Source Framework for Adaptive Traffic Signal Control. ArXiv, abs/1909.00395.

[9] Giannakis, I., Bhardwaj, A., Sam, L., & Leontidis, G. (2023). DEEP LEARNING UNIVERSAL CRATER DETECTION USING SEGMENT ANYTHING MODEL (SAM). arXiv. https://doi.org/10.48550/arXiv.2304.07764

[10] Hawi, R., Okeyo, G., & Kimwele, M. (2017). Smart traffic light control using fuzzy logic and wireless sensor network. 2017 Computing Conference, 450–460. https://doi.org/10.1109/SAI.2017.8252137 R. F. A .Mohd Nor, F. H. K. Zaman, S. Mubdi,

[11] J. Cao et al., "A Max Pressure Approach to Urban Network Signal Control with Queue Estimation using Connected Vehicle Data," 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 2020, pp. 1-8, doi: 10.1109/ITSC45102.2020.9294361.

[12] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W., Dollár, P., & Girshick, R.B. (2023). Segment Anything. ArXiv, abs/2304.02643.

[13] K. Singh, V. Varshney and M. Rajl, "Image Segmentation Role in Self Driving Car," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 861-864

[14] Kusumawardhana, D. B., Trilaksono, B. R., & Abdurrohman, H. (2022). Panoptic Segmentation Datasets for Rail-based Autonomous Vehicle Under Mixed-traffic Scenario. 2022 12th International Conference on System Engineering and Technology (ICSET), 19–24. https://doi.org/10.1109/ICSET57543.2022.10010755

[15] Li, L., Qian, B., Lian, J., Zheng, W., & Zhou, Y. (2018). Traffic Scene Segmentation Based on RGB-D Image and Deep Learning. IEEE Transactions on Intelligent Transportation Systems, 19(5), 1664–1669.

https://doi.org/10.1109/TITS.2017.2724138

[16] Nor, R. F. A. M., Zaman, F. H. K., & Mubdi, S. (2017). Smart traffic light for congestion monitoring using LoRaWAN. 2017 IEEE 8th Control and System Graduate Research Colloquium (ICSGRC), 132–137. https://doi.org/10.1109/ICSGRC.2017.8070582 B. Pratama, J. Christanto, M. T. Hadyantama and A. Muis.

[17] Pratama, B., Christanto, J., Hadyantama, M. T., & Muis, A. (2018). Adaptive traffic lights through traffic density calculation on road pattern. 2018 International Conference on Applied Science and Technology (iCAST), 82–86. https://doi.org/10.1109/iCAST1.2018.8751540

[18] Pujara, H., & Mvv, K. (2013). Image Segmentation using Learning Vector Quantization of Artificial Neural Network.

[19] Wang, C., Zeng, B., Gao, J., Peng, G., & Yang, W. (2023). A traffic image semantic segmentation algorithm based on UNET. Proceedings of SPIE - The International Society for Optical Engineering, 12610. https://doi.org/10.1117/12.2671074

[20] Venu, D., J, B., Saravanakumar, R., Cosio Borda, R., Abd Algani, Y.M., & Kiran Bala, B. (2022). End-to-end security in embedded system for modern mobile communication technologies. Measurement: Sensors.

[21] Wang, G., Meng, Y., Sahli, H., Yue, A., Chen, J., Chen, J., He, D., & Wu, B. (2016). Vehicles detection using GF-2 imagery based on watershed image segmentation. 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 3758-3761.

[22] Xiangyu Deng*, Wenjuan Qin, Ran Zhang, Yunping Qi (2019). Automatic segmentation algorithm of license plate image based on PCNN and DNN. Proc. SPIE 11321, 2019 International Conference on Image and Video Processing, and Artificial Intelligence, 1132102.

6 Contributions of team members

1. Abstract and Introduction sections were jointly completed by the group in the earlier proposal.

2. The review section was completed by *** and ***.

3. The methodology section was completed by *** and ***.

4. The conclusion and discussion section was completed by *** and ***.

5. Liu Yushun and Ma Yunwen are responsible for the overall cohesion and coherence of the entire paper, overseeing the comprehensive revision of the entire paper.