



From Pixels to Semantic Coordinates: Vehicle Recognition over Narrowband Internet of Things with Link-Adaptive Progressive Transmission

Jing Li 

Chongqing Technology and Business Institute, Chongqing 401520, China

Corresponding Author Email: lijinggzy0504@163.com

Copyright: ©2026 The author. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/ts.430405>

ABSTRACT

Received: 10 March 2026

Revised: 17 July 2026

Accepted: 20 August 2026

Available online: 31 August 2026

Keywords:

NB-IoT, traffic image processing, vehicle recognition, semantic coordinates, progressive transmission, edge-cloud collaboration

Traffic imagery is inherently high-dimensional and redundant, whereas Narrowband Internet of Things (NB-IoT) offers only an extremely narrow uplink bandwidth under a stringent low-power transmission budget. To resolve this fundamental conflict, this paper proposes a unified framework that integrates semantic coordinate encoding of vehicle identity, link-adaptive progressive transmission, and edge-cloud feedback refinement. Instead of transmitting raw images or generic feature maps, the edge device compresses each vehicle region into quantizable, rankable, and queryable semantic coordinates. The priority of semantic blocks is organized by a unified variable—the discriminative gain for vehicle identity—which in turn drives link-state-aware bit allocation. Upon receiving a subset of semantic blocks, the cloud performs progressive classification and re-identification, and triggers compact downlink queries based on classification uncertainty and gallery-matching ambiguity, thereby instructing the edge to re-encode difficult vehicles at higher resolution and return supplementary semantics. Experiments are conducted on UA-DETRAC, VeRi-776, VehicleID, and a self-collected NB-IoT traffic image dataset. Under a per-vehicle transmission budget of only tens to a few hundred bytes, the proposed method outperforms existing compressive transmission and edge metadata schemes on both vehicle classification and re-identification, while satisfying the prescribed low-power transmission constraint. These results show that shifting the transmitted object from visual data to vehicle identity semantic coordinates enables effective semantic compression of traffic images and reliable vehicle recognition under narrowband constraints, offering a viable path toward IoT-based traffic visual perception.

1. INTRODUCTION

Intelligent transportation systems have a sustained demand for vehicle detection, classification, and re-identification [1]. With the characteristics of wide coverage, low power consumption, and massive connectivity, Narrowband Internet of Things (NB-IoT) is suitable for city-scale traffic surveillance backhaul [2]. However, the effective uplink throughput of NB-IoT is limited and the payload available in a single transmission is small [3], and it is further affected by retransmission and link quality fluctuation [4]. Traffic images contain substantial texture, illumination, and background redundancy [5]; directly compressing pixels [6] or transmitting generic deep features [7] can hardly preserve vehicle identity discriminative information within the NB-IoT budget. Therefore, for NB-IoT-oriented traffic image processing and vehicle recognition, the key lies not in how to compress images, but in how to define and transmit the semantic information that is truly necessary for vehicle recognition [8]. This problem has the cross-cutting attributes of image processing, visual semantic coding, and Internet of Things communication, and is of important significance for low-power visual perception, edge intelligence, and vehicle re-identification [9].

Existing studies have not yet formed an effective closed loop on this problem. At the level of transmission object, schemes that perform lightweight detection at the edge and then send back metadata such as category, location, and timestamp have been widely adopted [10]; for example, in NB-IoT smart sensors, only the target type, direction, and timestamp are sent to a centralized platform. Although such schemes avoid the bandwidth bottleneck, the cloud completely loses the vehicle appearance semantics [11], and cannot support cross-camera vehicle re-identification and fine-grained attribute analysis [12]. If one instead transmits compressed images or intermediate feature maps, another dilemma arises [13]: codecs optimized for human visual perception, such as JPEG and H.264, irreversibly truncate high-frequency spatial components and chrominance information [14], destroying the fine-grained semantic priors required by machine vision backends [15]; directly transmitting intermediate continuous neural tensors often inflates the payload of a single frame to several megabytes [16], far exceeding the range that NB-IoT can actually carry. At the level of transmission strategy, most schemes adopt a fixed compression ratio or a fixed feature dimension [17], without being aware of the NB-IoT link quality and the available transmission budget [18]. Although semantic

communication has shown potential in data reduction [19], its decoding objective is usually anchored to image reconstruction quality or generic object recognition [20], and it does not sufficiently preserve vehicle identity specificity in the reconstructed image, limiting the feasibility of accurate vehicle recognition. Meanwhile, vehicle re-identification technology itself relies on high-dimensional appearance features for cross-camera matching [21], and such features are difficult to directly fit into a narrowband transmission budget [22]. At the level of cloud-edge collaboration, split inference reduces the front-end computation burden by deploying the early layers of the model at the edge and the remaining layers in the cloud [23], but the communication overhead of intermediate feature maps [24] still needs to be controlled by compression means such as quantized incremental masks. Such schemes focus on computation offloading rather than semantic selection; the cloud cannot use recognition uncertainty to reversely guide the edge to refine difficult vehicles on demand, and the limited uplink bits are allocated uniformly [25]. The above analysis indicates that it is necessary to make the vehicle identity discrimination requirement run through the whole process of coding, transmission, and feedback, so that the narrowband constraint is transformed from a passive limitation into a driving factor of semantic selection.

This paper proposes a unified framework of semantic coordinate encoding of vehicle identity, link-adaptive progressive transmission, and edge-cloud feedback refinement, and uses the discriminative gain for vehicle identity as a unified variable to drive semantic block ranking, bit allocation, and feedback refinement. At the edge, a dual-branch encoder is designed to compress the vehicle region into quantizable, rankable, and queryable semantic coordinates; through metric learning, quantization constraints, and an information bottleneck, identity discriminative information is preserved, and a single vehicle can be compressed to about 128 bytes. The transmission layer prioritizes the transmission of semantic blocks with high discriminative gain according to the link state and the available budget, and the cloud can perform progressive classification and re-identification after receiving part of the semantic blocks. The feedback layer triggers compact downlink queries based on classification uncertainty and gallery-matching ambiguity, guiding the edge to perform high-resolution re-encoding of difficult vehicles and send back supplementary semantics. Experiments are conducted on UA-DETRAC, VeRi-776, VehicleID, and a self-collected NB-IoT traffic image dataset; under a per-vehicle budget of tens to a few hundred bytes, the proposed method outperforms existing compressive transmission and edge metadata schemes on vehicle classification and re-identification tasks, and satisfies the prescribed low-power transmission budget constraint.

2. METHOD

2.1 Problem definition and overall idea

This paper studies the problem of traffic image processing and vehicle recognition in the NB-IoT narrowband uplink scenario. Let the edge device collect a traffic image $x_t \in \mathbb{R}^{H \times W \times 3}$ in the t -th time slot, and obtain a set of vehicle regions $\{(b_i, p_i)\}_{i=1}^{N_t}$ through a lightweight detector, where b_i is the bounding box of the i -th vehicle and p_i is the detection confidence. The available NB-IoT uplink budget is denoted as

B_t , and the link state is denoted as $Q_t = \{\text{SNR}_t, \text{RSRP}_t, \text{retrans}_t, P_t\}$, where SNR_t is the signal-to-noise ratio (SNR), RSRP_t is the reference signal received power (RSRP), retrans_t is the number of retransmissions, and P_t is the available application-layer payload budget of the current time slot. This problem does not aim at transmitting raw images, JPEG compressed frames, or generic deep feature maps; instead, each vehicle region is encoded into quantizable, rankable, and queryable semantic coordinates of vehicle identity q_i , and the cloud completes vehicle classification and re-identification relying only on q_i and the scene context s_i . The overall optimization objective is written as:

$$\begin{aligned} \min_{\theta, \phi, \pi, \psi} & \mathbb{E}[L_{\text{cls}} + \lambda_1 L_{\text{reid}} + \lambda_2 L_{\text{quant}} + \lambda_3 L_{\text{ctx}} + \lambda_4 R(B) + \lambda_5 L_{\text{fb}}] \\ \text{s.t.} & \sum_{i,t} b_{i,t} \leq B_{\text{total}}, b_{i,t} \leq B_{\text{pkt}}(Q_t), q_i = Q(E(x_t, b_i)) \end{aligned}$$

where, θ, ϕ, π, ψ correspond to the learnable parameters of the semantic encoding, saliency prediction, transmission policy, and feedback refinement modules, respectively; L_{cls} is the vehicle classification loss, L_{reid} is the vehicle re-identification loss, L_{quant} is the quantization loss, L_{ctx} is the context consistency loss, $R(B)$ is the transmission cost, L_{fb} is the feedback refinement loss, and λ_1 to λ_5 are the weights of the respective losses. $E(\cdot)$ is the edge semantic encoder and $Q(\cdot)$ is the quantizer; $b_{i,t}$ is the uplink bits actually occupied by the i -th vehicle, B_{total} is the total uplink budget, and $B_{\text{pkt}}(Q_t)$ is the number of bits that a single packet can carry under the current link state. This objective does not simply pursue classification accuracy, but maximizes the vehicle identity discriminative capability under the constraints of the NB-IoT transmission budget and link fluctuation.

The overall idea is to treat the NB-IoT bandwidth constraint as a semantic selection problem rather than a mere compression problem. What vehicle recognition truly needs is identity discriminative information such as vehicle type, color, brand, texture salient regions, and local structure, whereas background, illumination, and global texture redundancy contribute little to identity discrimination. To this end, this paper takes the discriminative gain for vehicle identity $\alpha_{i,k}$ as a unified variable that runs through the three stages of semantic encoding, progressive transmission, and feedback refinement. $\alpha_{i,k}$ denotes the gain of the k -th semantic block of the i -th vehicle for vehicle identity discrimination given the preceding semantic blocks; a higher value means that this semantic block should be transmitted or refined with higher priority. In the encoding stage, semantic blocks are organized according to $\alpha_{i,k}$; in the transmission stage, limited bits are allocated according to $\alpha_{i,k}$; in the feedback stage, the semantic direction that needs to be supplemented is determined according to the cloud recognition uncertainty. As a result, the vehicle identity discrimination requirement is no longer confined to the cloud classifier, but is advanced into edge encoding and link transmission decisions, so that the limited uplink budget is concentrated on the most discriminative semantic components.

2.2 Framework overview

The overall pipeline consists of four stages, including edge-side vehicle detection and Region of Interest (ROI) extraction, semantic coordinate encoding of vehicle identity and semantic

block organization, link-state-aware progressive transmission, and cloud-side progressive recognition and feedback refinement. As shown in Figure 1, this system constructs a complete closed-loop architecture from edge front-end processing to narrowband transmission, and then to cloud inference and downlink control. At the edge, a lightweight backbone network is first used to extract multi-scale features, and ROIAlign is adopted to obtain the region feature of each vehicle. Subsequently, the dual-branch semantic encoder outputs the appearance identity embedding and the scene context embedding, respectively. After the appearance identity

embedding undergoes normalization, quantization, and semantic block partitioning, the saliency prediction head estimates the discriminative gain α_{ik} of each semantic block, forming a semantic block priority queue sorted in descending order of discriminative gain. The transmission layer prioritizes the transmission of semantic blocks with high discriminative gain according to the NB-IoT link state Q_t and the currently available budget B_t , and adaptively allocates the quantization bit width, so that key vehicle attributes remain recognizable when the link degrades, and the re-identification accuracy is further improved when the link is in good condition.

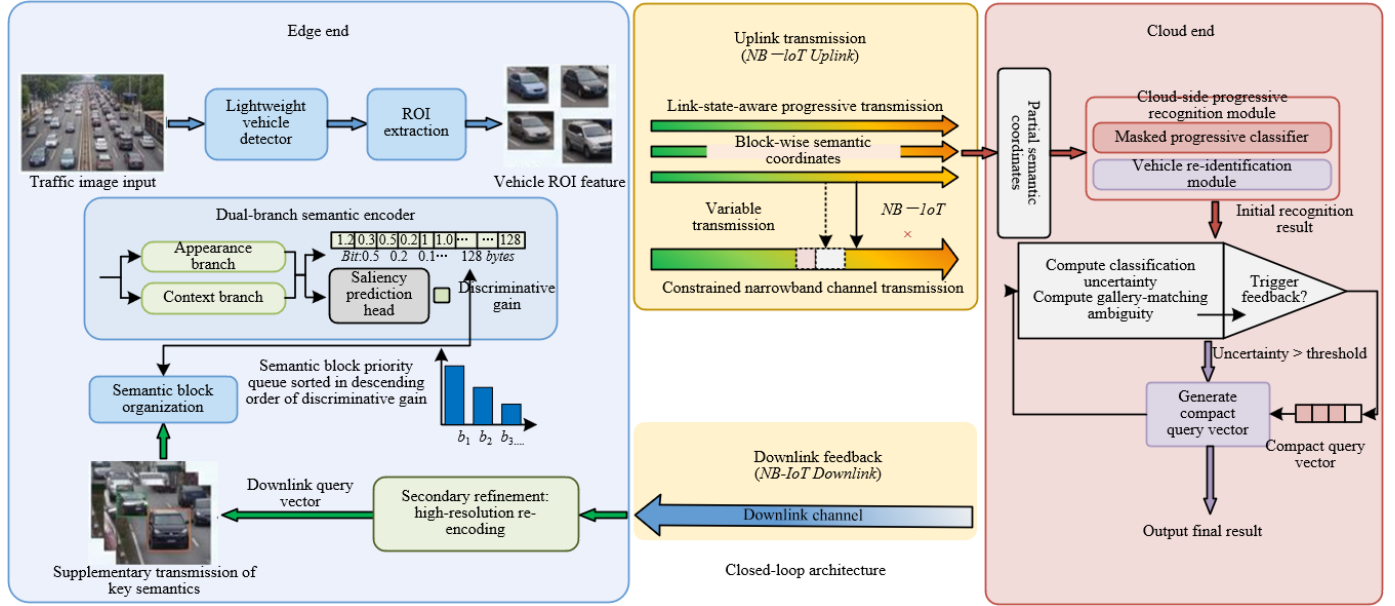


Figure 1. Overall framework of edge-cloud collaborative vehicle semantic progressive transmission and feedback refinement

After receiving part of the semantic blocks, the cloud performs masked progressive classification and re-identification, and computes the classification uncertainty and the gallery-matching ambiguity. If the classification uncertainty is high or the matching with the vehicle gallery is ambiguous, the cloud sends a compact query vector through the downlink narrowband channel, triggering the edge to perform high-resolution re-encoding of the specified vehicle ROI and to send back supplementary semantics. The cloud updates the vehicle recognition result after fusing the supplementary semantics. This process makes the uplink transmission not a one-time fixed output, but a dynamic adjustment according to the cloud recognition state. The three stages share the same discriminative gain variable α_{ik} , forming a closed loop among encoding, transmission, and feedback, and avoiding the situation where edge encoding and cloud recognition are independent of each other. Through this closed loop, the narrowband constraint of NB-IoT is transformed into a semantic selection mechanism, and the limited uplink bits preferentially carry the semantic information most needed for vehicle identity discrimination, thereby establishing a quantifiable, schedulable, and verifiable unified framework between traffic image processing and vehicle recognition tasks.

2.3 Semantic coordinate encoding of vehicle identity

The edge device has limited computation and power consumption, and a lightweight single-stage detector is

adopted for vehicle localization. The detector input resolution is set to 256×256 or 384×384 , and the backbone network uses MobileNetV3-Small or ShuffleNetV2 to reduce the number of parameters and memory access. After non-maximum suppression, the detection output retains vehicle boxes whose confidence is higher than the threshold. For each vehicle box b_i , ROIAlign is used to extract a fixed-size region feature $f_i \in \mathbb{R}^{7 \times 7 \times C}$ from the multi-scale feature pyramid. ROIAlign avoids quantization errors and preserves fine-grained structures such as vehicle edges, headlights, and license plate regions. To reduce the burden on the edge device, the detector and the semantic encoder share part of the shallow features and are separated only at the back-end branches. The detection stage does not pursue an extremely high mAP, but guarantees a high recall rate, so that the subsequent semantic encoding covers the main vehicle instances.

On the basis of the region feature, the dual-branch semantic encoder consists of an appearance branch E_{app} and a context branch E_{ctx} . The appearance branch takes the ROI feature f_i as input and outputs a d -dimensional identity embedding $a_i \in \mathbb{R}^d$. This branch consists of global average pooling, two fully connected layers, and one linear projection, with Gaussian Error Linear Unit (GELU) activation and LayerNorm used in between. To preserve local discriminative structures, lightweight channel attention is introduced before pooling, so that regions such as headlights, grille, and window contours obtain higher weights. The context branch extracts scene semantics such as lane position, driving direction, time period, weather, and camera ID, and outputs a d_c -dimensional context

embedding s_i . The context branch retains only low-dimensional discrete information such as camera ID, lane, and time period, which is counted into the transmission load together with the semantic data and serves as an auxiliary prior for cloud-side classification and re-identification. Figure 2 shows in detail the microscopic network structure and processing flow of the edge encoder, including feature extraction, scalar quantization, and the semantic block ranking mechanism driven by saliency prediction. The appearance

embedding is L_2 -normalized to obtain a unit hypersphere representation:

$$z_i = \frac{a_i}{\|a_i\|_2}$$

Normalization stabilizes the quantization error and makes cosine similarity equivalent to Euclidean distance, which facilitates vehicle gallery matching.

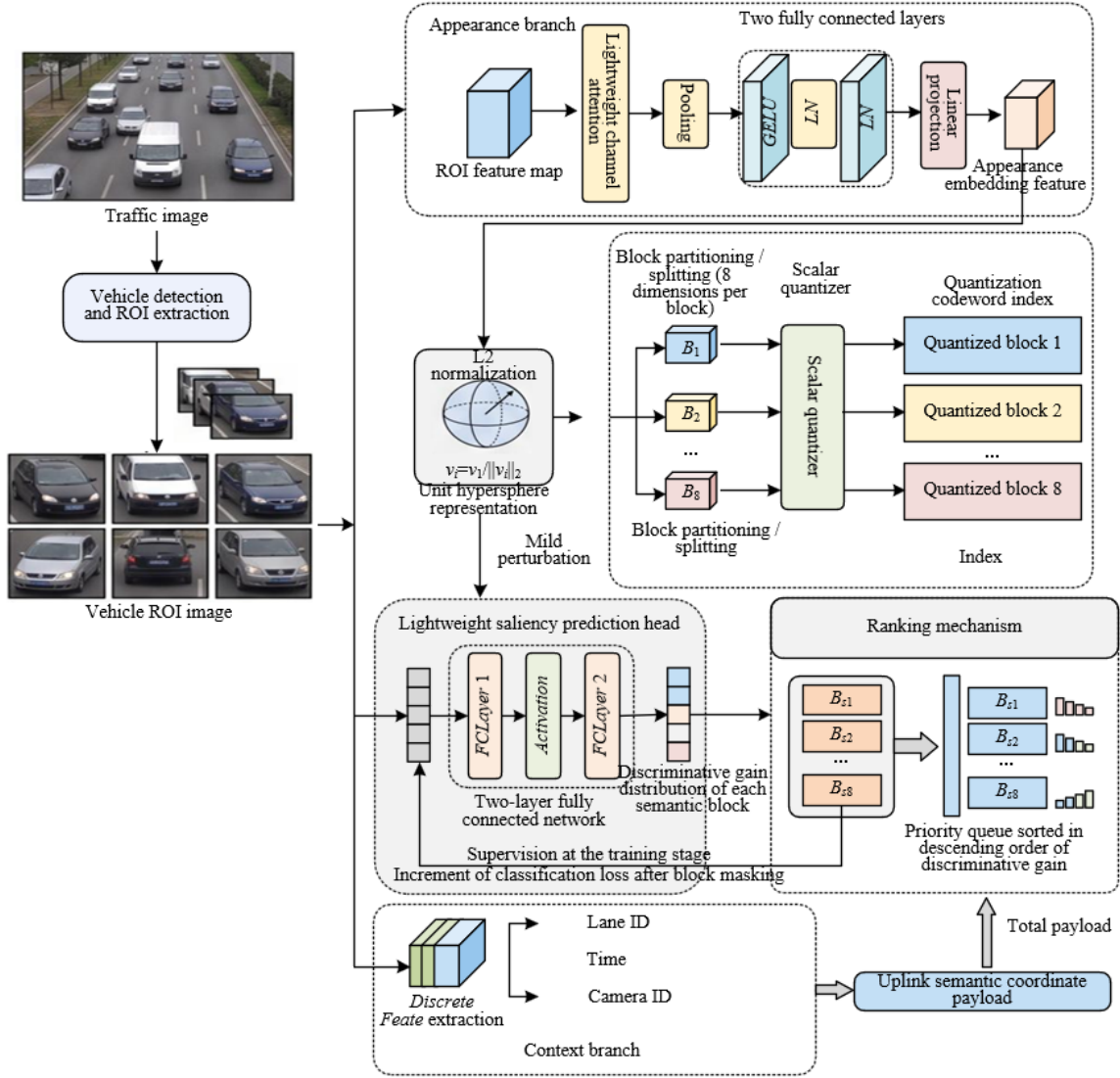


Figure 2. Structure of semantic coordinate encoding of vehicle identity and semantic block priority organization

To match the NB-IoT narrowband transmission budget, z_i is quantized into semantic coordinates q_i . Let the semantic dimension be $d = 64$, which is divided into $K = 8$ semantic blocks, each with 8 dimensions. Each semantic block is independently scalar-quantized, and the per-dimension quantization bit width w_k can be adapted between $w_{\min} = 4$ and $w_{\max} = 16$ bit, corresponding to a per-block payload $b_k = 8w_k$ bit. The quantization process is written as:

$$q_i = Q(z_i; B_i) = \sum_{k=1}^K q_{i,k} e_k$$

where, $q_{i,k}$ is the quantization codeword of the k -th semantic block, e_k is the corresponding basis vector, and when all blocks are transmitted, the total semantic payload satisfies

$B_i = \sum_{k=1}^K b_k$. When 16-bit quantization is adopted per dimension, the payload of each block is 128 bit, i.e., 16 bytes, and 8 blocks amount to 128 bytes. Block-wise quantization supports progressive transmission and differentiated bit widths; blocks with high discriminative gain can be allocated more bits, while blocks with low gain can be coarsely quantized or delayed for transmission. In actual deployment, each semantic block carries the block index, the quantization codeword, and necessary check information, and the protocol overhead is counted separately in the communication statistics.

Training the semantic coordinates with only the classification loss tends to retain background and illumination information that is irrelevant to vehicle identity. To this end, an information bottleneck constraint is introduced, so that q_i

maximizes the predictive capability for the vehicle label y_i while compressing the original region information. Its variational upper bound is:

$$L_{IB} \leq E_{q_i \sim Q} [-\log p(y_i | q_i)] + \beta D_{KL}(Q(q_i | x_i) \| p(q_i))$$

where, the first term is the classification likelihood, the second term constrains the quantized posterior $Q(q_i | x_i)$ to be close to the prior $p(q_i)$, and β controls the compression strength. This constraint drives the encoder to discard variation factors irrelevant to identity, such as illumination, weather, and background texture. At the same time, triplet metric learning is adopted to enhance intra-class compactness and inter-class separability:

$$L_{reid} = \sum_i \max(0, \|q_i^a - q_i^p\|_2^2 - \|q_i^a - q_i^n\|_2^2 + m)$$

where, q_i^a , q_i^p , and q_i^n are the anchor, positive sample, and negative sample semantic coordinates, respectively, and m is the margin. During training, online hard example mining is adopted to preferentially select negative samples that are similar to the anchor but have different vehicle identities, and positive samples that differ greatly from the anchor. This strategy enables the semantic coordinates to retain vehicle identity discriminative capability after NB-IoT low-bit quantization.

The cloud classifier takes the semantic coordinates and the context embedding as input:

$$\hat{p}_i = \text{softmax}(W_c[q_i; s_i] + b_c)$$

The classification loss is:

$$L_{cls} = - \sum_i y_i \log \hat{p}_i$$

The quantization loss consists of the reconstruction error and an entropy regularization:

$$L_{quant} = \|z_i - q_i\|_2^2 - \eta H(q_i)$$

where, W_c and b_c are the classifier weight and bias, y_i is the vehicle label, and $H(q_i)$ is the codeword distribution entropy; a negative entropy term is adopted in the minimization objective to suppress excessive concentration of codewords, and η controls the regularization strength. The classification loss guarantees the separability of vehicle categories, the re-identification loss guarantees cross-frame identity consistency, the information bottleneck loss suppresses redundancy, and the quantization loss controls the bit cost. Through joint training, q_i output by the encoder is no longer a generic visual feature, but a compact semantic coordinate oriented to vehicle identity discrimination.

2.4 Discriminative gain estimation and semantic block organization

To determine which semantic blocks should preferentially occupy the limited NB-IoT uplink bits, this paper uses the discriminative gain for vehicle identity as the basis for

semantic block ranking. For the k -th semantic block $q_{i,k}$ of the i -th vehicle, the increment of the classification loss after masking this block is defined as the discriminative gain:

$$\alpha_{i,k} = \max(0, L_{cls}(q_i^{(-k)}) - L_{cls}(q_i))$$

where, $q_i^{(-k)}$ denotes the semantic coordinates after setting the k -th semantic block to the missing state, and L_{cls} is the vehicle classification loss. If masking a semantic block causes the classification loss to increase significantly, it indicates that this block contributes highly to the discrimination of the current vehicle identity, and it should be transmitted with priority when the transmission budget is limited. This definition can be directly computed by one block masking experiment, and is kept consistent with the block priority in the subsequent progressive transmission.

To make the discriminative gain quickly available at the inference stage, a lightweight saliency prediction head h_ϕ is designed to perform feed-forward estimation of the semantic block importance. This prediction head consists of two fully connected layers, takes the L_2 -normalized appearance embedding z_i as input, and outputs a K -dimensional importance vector:

$$\hat{\alpha}_i = \text{softmax}(h_\phi(z_i)) \in \Delta^{K-1}$$

where, ϕ is the prediction head parameter, Δ^{K-1} denotes the K -dimensional probability simplex, and the components of $\hat{\alpha}_i$ correspond to the predicted discriminative gain of each semantic block. In the training stage, the increment of the classification loss after block masking is used as the supervision signal, so that $\hat{\alpha}_i$ approaches α_i . The number of parameters of this prediction head is less than 0.1 M, and it is used only for training and inference at the edge, without increasing the burden on the cloud. At inference, the edge sorts the semantic blocks in descending order according to $\hat{\alpha}_i$ and generates a transmission priority queue, so that the limited uplink budget preferentially carries the semantic components that contribute most to vehicle identity discrimination.

Semantic block organization follows three principles: independently decodable, progressively fusible, and feedback queryable. Being independently decodable requires each semantic block to contain a complete quantization codeword and block index, so that the cloud can decode and perform preliminary classification without waiting for all the blocks. Progressive fusion is achieved by introducing masked reconstruction and a progressive classification loss in training, so that the combination of the first several blocks still maintains semantic consistency and avoids feature space misalignment caused by partial reception. Feedback queryability associates the block index with the spatial region, so that the cloud can request, through a downlink query vector, that the edge supplement a specific block or increase the bit width of a specific block. Finally, each vehicle generates a priority queue $\{\pi_i(1), \pi_i(2), \dots, \pi_i(K)\}$ sorted in descending order of discriminative gain, where $\pi_i(1)$ is the semantic block with the highest discriminative gain. This queue is truncated for transmission when the link quality is poor, retaining only the first several high-gain blocks; it is transmitted in full when the link quality is good, so as to improve the vehicle re-identification accuracy. As a result, semantic block organization is no longer a static partitioning, but a

transmission unit that is dynamically coupled with the link state and the cloud recognition requirement.

2.5 Link-state-aware progressive semantic transmission

NB-IoT link quality exhibits significant time variability and spatial non-uniformity, and a fixed compression ratio or a fixed feature dimension can hardly guarantee the availability of key vehicle semantics when the link degrades. To this end, the edge estimates the link state $Q_t = \{\text{SNR}_t, \text{RSRP}_t, \text{retrans}_t, P_t\}$ in each transmission time slot, where SNR_t is the signal-to-noise ratio, RSRP_t is the reference signal received power, retrans_t is the number of retransmissions, and P_t is the available application-layer payload budget of the current time slot. The first two determine the bit error rate and the available

modulation and coding scheme, the number of retransmissions reflects the historical congestion level, and the application-layer payload budget constrains the amount of data that the current time slot can carry. Based on this, the edge decides which semantic blocks to send and how many bits to use for each block. When the link quality is poor, only the first m semantic blocks with the highest discriminative gain are sent, ensuring that key attributes such as vehicle category and color remain recognizable; when the link quality is good, the number of semantic blocks is increased or the quantization bit width is raised, so as to improve the vehicle re-identification accuracy. This strategy avoids the loss of key semantics caused by a fixed compression ratio when the link degrades, and at the same time prevents redundant bits from being wasted when the link is good.

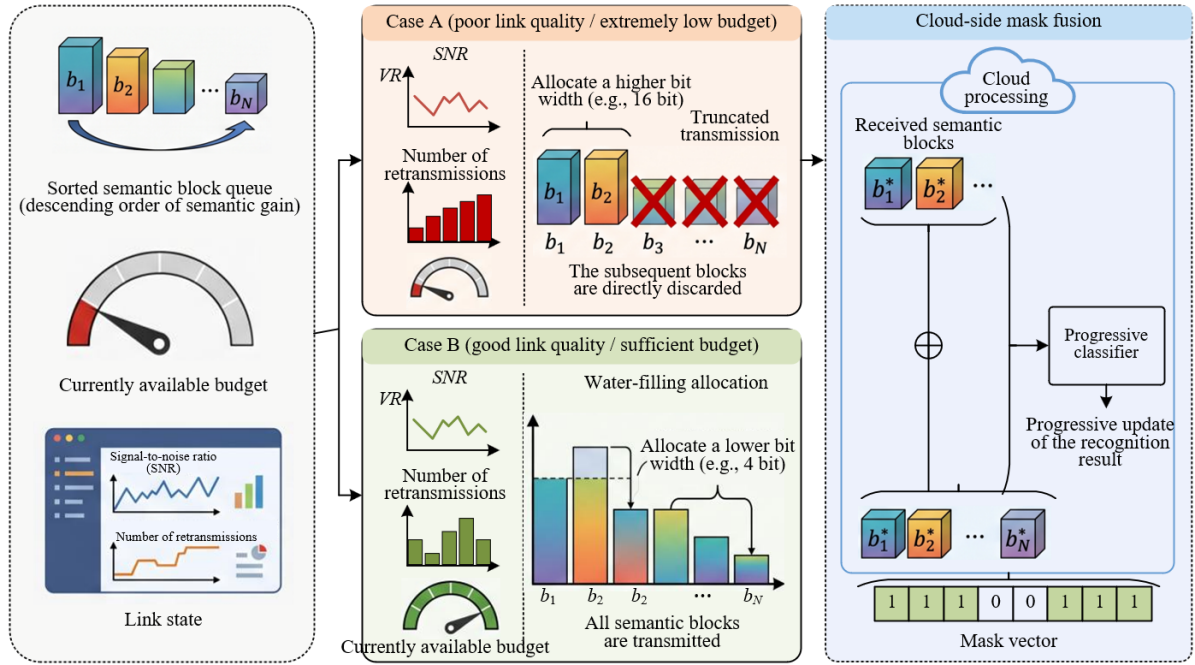


Figure 3. Schematic diagram of link-state-aware water-filling bit allocation and progressive transmission mechanism

As shown in Figure 3, the water-filling bit allocation mechanism can dynamically switch strategies according to the real-time available budget: it performs truncated transmission to guarantee key features when the link quality is poor, and performs full transmission to improve the re-identification accuracy when the budget is sufficient. Given the available uplink budget B_t of the current time slot and the set of blocks to be transmitted $\pi_t \subseteq \{1, \dots, K\}$, progressive block selection can be formalized as an expected utility maximization problem:

$$\pi_t^* = \underset{\pi_t}{\operatorname{argmax}} \mathbb{E}_y \left[U \left(y_i, \hat{y}_i(q_{i,\pi_t}) \right) \right], \text{ s.t. } \sum_{k \in \pi_t} b_k(Q_t) \leq B_t$$

where, K is the total number of semantic blocks, q_{i,π_t} is the combination of semantic blocks corresponding to the set π_t , $\hat{y}_i$ is the vehicle recognition result obtained by the cloud based on the received semantic blocks, the utility U can be the negative cross-entropy or the mutual information $I(y_i; q_{i,\pi_t})$, and $b_k(Q_t)$ is the number of bits occupied by the k -th semantic block under the current link state. Since $K = 8$ and the number of blocks is small, a greedy strategy is adopted to select semantic blocks in descending order of $\alpha_{i,k}/b_k$ until the budget

is exhausted, with a complexity of $O(K \log K)$, which is suitable for real-time operation at the edge. When multiple vehicles share the uplink budget, the number of blocks is further allocated according to the vehicle recognition uncertainty and the detection confidence, giving priority to high-confidence vehicles and high-risk vehicles, so that the limited bits are also allocated on demand among vehicle instances.

After the set of blocks to be transmitted is determined, the quantization bit width is adaptively adjusted with the link quality and the semantic block importance. Let b_k denote the number of payload bits of the k -th semantic block, whose value ranges from $b_{\min} = 32$ to $b_{\max} = 128$ bit, corresponding to 4 to 16 bit quantization per dimension. Given the discriminative gain $\alpha_{i,k}$, water-filling bit allocation is adopted:

$$b_k = \operatorname{clip} \left(\operatorname{round} \left(b_{\min} + (B_t - |\pi_t| b_{\min}) \frac{\alpha_{i,k}}{\sum_{j \in \pi_t} \alpha_{i,j}} \right), b_{\min}, b_{\max} \right)$$

where, $b_{\min}$ and $b_{\max}$ are the minimum and maximum quantization bit widths of a semantic block, respectively, $\operatorname{round}()$ is the rounding operation, $\operatorname{clip}()$ limits the result to the range $[b_{\min}, b_{\max}]$, $\alpha_{i,k}$ is the discriminative gain of the k -th

semantic block, and $\sum_{j \in \pi_t} a_{i,j}$ is the total discriminative gain of the current set of blocks to be transmitted. Blocks with high discriminative gain obtain more bits, while blocks with low gain use the lowest bit width. This allocation can be regarded as an approximate solution to maximizing the semantic discriminative gain under the total bit constraint. When the link is poor, B_i is small, and the bit width is concentrated on the first m high-gain blocks; when the link is good, B_i is large, and more semantic blocks obtain a medium bit width. Compared with a uniform bit width, water-filling allocation improves the recognition accuracy of the first few semantic blocks under the same budget, making the early results of progressive transmission more usable.

After receiving part of the semantic blocks, the cloud adopts a masked progressive classifier to update the vehicle recognition posterior:

$$\hat{p}_i^{(m)} = \text{softmax} \left(W_c [\tilde{q}_{i,\pi_t}; M_{\pi_t}; s_i] + b_c \right)$$

where, $\tilde{q}_{i,\pi_t}$ is the concatenation of the received semantic blocks, M_{π_t} is the block mask used to indicate the missing semantic blocks, s_i is the scene context embedding, and W_c and b_c are the classifier weight and bias. The mask vector and the indices of the received blocks are jointly input, so that the classifier can distinguish a zero value from an unreceived state, and avoid missing blocks being misjudged as valid zero values. As more semantic blocks arrive, the cloud posterior is progressively refined, and the current recognition uncertainty is computed:

$$H_i^{(m)} = - \sum_{c=1}^C \hat{p}_{i,c}^{(m)} \log \hat{p}_{i,c}^{(m)}$$

where, C is the number of vehicle categories, $\hat{p}_{i,c}^{(m)}$ is the predicted probability of the c -th category, and $H_i^{(m)}$ is the current posterior entropy. If $H_i^{(m)}$ is lower than the preset threshold, the transmission of this vehicle is terminated early to save the uplink budget; if $H_i^{(m)}$ remains high, the subsequent feedback refinement is triggered. This mechanism makes the transmission policy itself a controllable variable of the vehicle recognition accuracy rather than a fixed preprocessing step, and provides an explicit triggering basis for edge-cloud feedback refinement.

2.6 Dynamic refinement driven by edge-cloud semantic feedback

After the cloud completes progressive classification, in addition to outputting the vehicle category and re-identification result, it also needs to evaluate whether the current recognition state is reliable, so as to decide whether to request the edge to supplement semantics. The classification uncertainty is measured by the current posterior entropy $H_i^{(m)}$, whose computation is consistent with that in Section 2.5. The gallery-matching ambiguity is defined as:

$$A_i = 1 - \max_j \text{sim}(q_i^{(m)}, r_j)$$

where, $q_i^{(m)}$ is the currently received semantic coordinates, r_j is the vehicle gallery prototype, and sim is the cosine

similarity. If A_i is high, it indicates that the current semantic coordinates are similar to multiple known vehicles and it is difficult to uniquely determine the identity. Feedback is triggered when $H_i^{(m)} > \tau_H$, or $A_i > \tau_A$, or $\max_c \hat{p}_{i,c}^{(m)} < \tau_p$, where τ_H , τ_A , and τ_p control the category uncertainty, the gallery ambiguity, and the maximum category confidence, respectively. A lower threshold triggers more refinement, which improves the recognition accuracy but increases the uplink bytes; a higher threshold saves the transmission budget but reduces the recall rate of difficult samples.

If feedback is triggered, the cloud sends a compact query vector through the NB-IoT downlink:

$$u_i = \text{MLP}([H_i^{(m)}, A_i, \hat{p}_i^{(m)}, \tilde{b}_i]) \in \mathbb{R}^r, r \ll d$$

where, $\hat{p}_i^{(m)}$ is the current category posterior, $\tilde{b}_i$ is the normalized position and scale, r is the dimension of the query vector, which is usually taken as 16 or 32, and d is the dimension of the semantic coordinates. This query vector occupies only a small number of downlink bits, yet contains four types of information: the current uncertainty, the gallery ambiguity, the category posterior, and the spatial position. Based on this, the edge determines which region of the vehicle should be focused on, which semantic blocks need a higher bit width, and whether the region of interest needs to be re-cropped. Compared with the scheme without feedback, this downlink query enables the cloud recognition state to reversely affect the edge encoding behavior, so that the uplink semantic supplement has an explicit direction.

The specific secondary query and feature enhancement mechanism is shown in Figure 4. With the Feature-wise Linear Modulation (FiLM) module, the edge uses the downlink query vector to dynamically enhance specific channels in the original high-resolution image, thereby extracting the most complementary local semantics. After receiving u_i , the edge performs high-resolution cropping of the corresponding region of interest in the original frame, and modulates the appearance encoder in a FiLM manner:

$$\gamma_i, \beta_i = \text{MLP}(u_i), h_i^+ = \gamma_i \odot h_i + \beta_i \\ z_i^+ = E_{\text{app}}^+(\text{ROIAlign}(F_{\text{hr}}, b_i); u_i), \Delta q_i = Q(z_i^+)$$

where, F_{hr} is the high-resolution feature, E_{app}^+ is the refinement encoder with query modulation, γ_i and β_i are the FiLM modulation parameters, $\odot$ denotes element-wise multiplication, h_i and h_i^+ are the features before and after modulation, respectively, and Δq_i is the supplementary semantics. FiLM modulation enables the edge to dynamically enhance specific channels according to the cloud query. When the ambiguity mainly comes from color difference, color-related channels are enhanced; when the ambiguity mainly comes from vehicle type difference, contour- and grille-related channels are enhanced. The supplementary semantics usually contain only 2 to 4 high-priority blocks, avoiding occupying too much uplink budget again.

The cloud fuses the supplementary semantics Δq_i with the existing semantics $q_i^{(m)}$ to update the vehicle recognition posterior:

$$\hat{p}_i^{\text{new}} = \text{softmax}(W_c [q_i^{(m)}; \Delta q_i; s_i] + b_c),$$

where, s_i is the scene context embedding, and W_c and b_c are the classifier weight and bias. Under the constraint of the total uplink budget B_{total} , the refinement resources are allocated according to the marginal utility:

$$\max_{S, \{b_i\}} \sum_{i \in S} \Delta U_i(b_i), \text{ s.t. } \sum_{i \in S} b_i \leq B_{\text{total}}$$

where, S is the set of vehicles to be refined, b_i is the supplementary bits allocated to the i -th vehicle, and $\Delta U_i(b_i) \approx H_i^{(m)} - H_i^{(m)}(b_i)$ denotes the uncertainty reduction brought by the additional bits. Lagrangian optimality gives:

$$b_i^* = \operatorname{argmax}_b [\Delta U_i(b) - \lambda b]$$

where, λ is the Lagrange multiplier used to balance the recognition gain and the bit cost. This allocation gives priority to refining vehicle instances that are highly uncertain, visually ambiguous, and contribute the most to the global recognition gain, so that the limited uplink budget is concentrated on the difficult samples that truly need supplementary semantics. Through this feedback loop, the cloud guides the quality of the uplink semantics at the edge through an extremely narrowband downlink control channel, achieving on-demand refinement and the dynamic closure of semantic transmission.

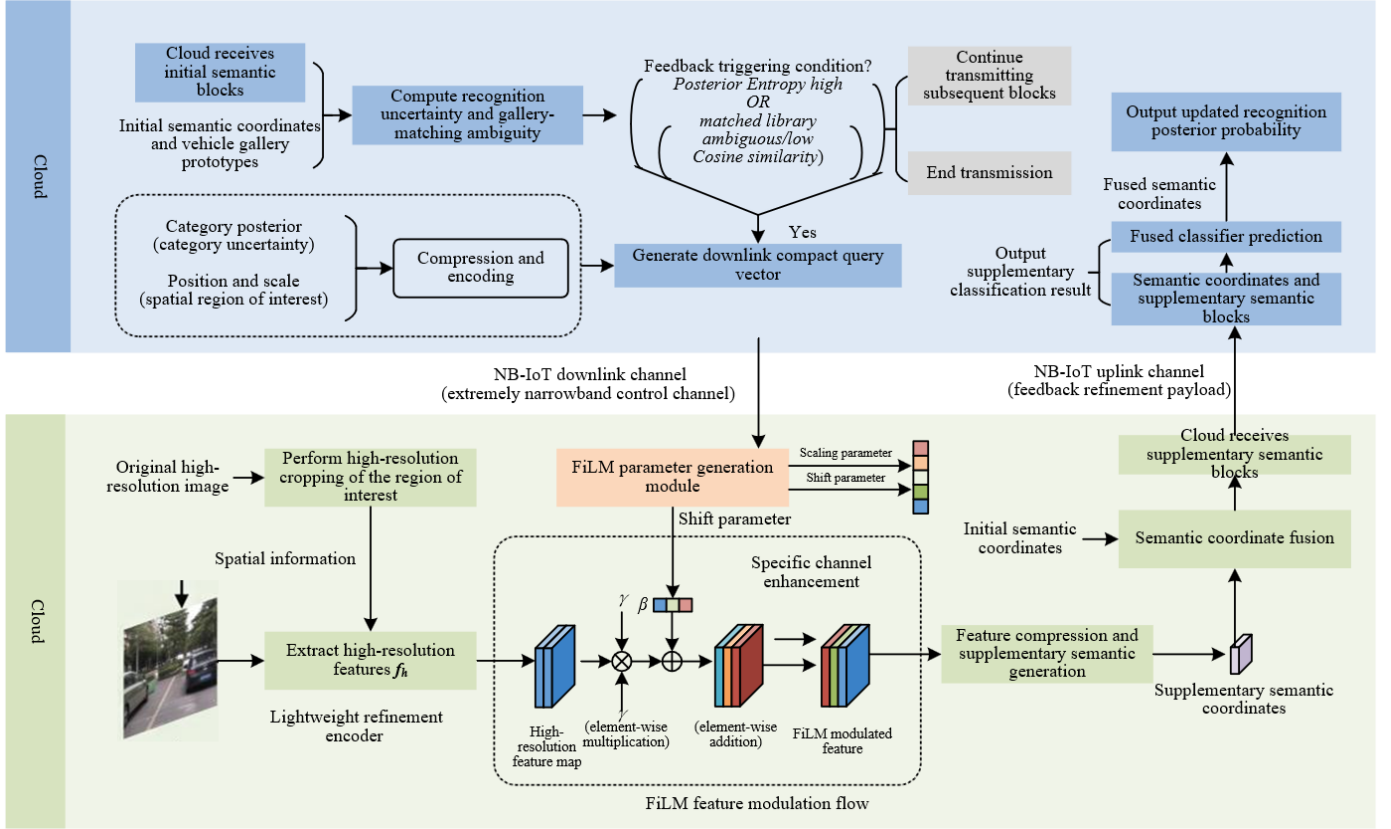


Figure 4. Flowchart of edge dynamic refinement based on downlink compact query and Feature-wise Linear Modulation (FiLM)

2.7 Unified training and inference

Unified training incorporates the dual-branch semantic encoder, the quantizer, the saliency prediction head, the cloud-side masked progressive classifier, and the feedback refinement module into the same optimization process. The training is divided into two stages. In the first stage, the lightweight detector is fixed, and the encoder, the quantizer, and the classifier are optimized. The total loss is written as:

$$L = L_{\text{cls}} + \lambda_1 L_{\text{reid}} + \lambda_2 L_{\text{quant}} + \lambda_3 L_{\text{ctx}} + \lambda_4 R(B) + \lambda_5 L_{\text{fb}}$$

where, L_{cls} is the vehicle classification loss, L_{reid} is the vehicle re-identification loss, L_{quant} is the quantization reconstruction and entropy regularization loss, L_{ctx} is the context consistency loss, $R(B)$ is the transmission bit cost, L_{fb} is the feedback refinement loss, and λ_1 to λ_5 are the weights of the respective losses. In the second stage, the saliency prediction head and the feedback refinement module are added, and joint fine-tuning is performed with the block masking, progressive

classification, and feedback fusion losses, so that the classifier adapts to partial semantic input. During training, part of the semantic blocks are randomly dropped and the quantization bit width is randomly set, so as to simulate NB-IoT link fluctuation. The optimizer adopts Adam with Decoupled Weight Decay (AdamW), with an initial learning rate of 1×10^{-3} , a batch size of 64, a semantic dimension $d = 64$, and the number of semantic blocks $K = 8$. The loss weights λ_1 to λ_5 are determined by grid search on the validation set.

The inference stage is executed in the order of edge encoding, link-adaptive transmission, cloud-side progressive recognition, and feedback refinement. The edge detects vehicles and extracts ROIs; the dual-branch encoder outputs the semantic coordinates q_i and the context embedding s_i ; the saliency prediction head outputs $\hat{a}_i$ and generates the semantic block priority queue. According to the link state Q_t and the available budget B_t , the edge selects the set of blocks to be transmitted, allocates the quantization bit width, and sends the semantic coordinates according to the priority. After receiving part of the semantic blocks, the cloud performs masked

progressive classification and computes the classification uncertainty $H_i^{(m)}$ and the gallery-matching ambiguity A_i . If the triggering condition is satisfied, the cloud sends the compact query vector u_i through the downlink; based on this, the edge performs high-resolution re-encoding of the specified ROI and sends back the supplementary semantics Δq_i , and the cloud updates the recognition result after fusion. The whole pipeline can be terminated early when the budget is exhausted or the uncertainty is lower than the threshold. The computation of a single frame at the edge is controlled within 1.5 Giga Floating-Point Operations (GFLOPs), the number of parameters is less than 5M, the complexity of semantic block ranking and bit width allocation is $O(K\log K)$, and the cloud only processes the 64-dimensional semantic coordinates and a lightweight classifier. The semantic coordinates of a single vehicle are about 128 bytes, which satisfies NB-IoT single-packet or few-packet transmission, and progressive transmission and early termination keep the satisfaction rate of the prescribed transmission budget at 100%.

3. EXPERIMENTAL DESIGN AND RESULT ANALYSIS

3.1 Experimental setup

3.1.1 Datasets and evaluation metrics

The experiments adopt UA-DETRAC, VeRi-776, VehicleID, CityFlow, and a self-collected NB-IoT traffic image dataset. UA-DETRAC is used for vehicle detection and classification, VeRi-776 and VehicleID are used for vehicle re-identification, CityFlow is used for cross-camera scenarios, and the self-collected dataset is used for NB-IoT link simulation and transmission budget testing. The evaluation metrics include vehicle detection mean Average Precision

(mAP), classification Top-1 and Top-5, re-identification Rank-1 and mAP, the number of transmitted bytes per vehicle, the processing latency of the edge and cloud algorithms, edge energy consumption, the transmission budget satisfaction rate, the number of retransmissions, and the progressive accuracy–bitrate curve.

3.1.2 Narrowband Internet of Things simulation and implementation details

The edge input resolution is set to 256×256 , the detector adopts a lightweight YOLO variant, and the backbone is MobileNetV3-Small. The semantic dimension is $d = 64$, the number of semantic blocks is $K = 8$, and the per-dimension quantization bit width is $w_{\min} = 4$, $w_{\max} = 16$. The NB-IoT simulation sets the SNR to -5 to 15 dB, the RSRP to -120 to -80 dBm, the application-layer effective payload budget to 128, 256, and 512 bytes, the transmission time ratio budget to 1%, and the number of retransmissions to 0 to 4. The compared methods include Tiny-You Only Look Once (YOLO) with metadata, Joint Photographic Experts Group, Quality factor 30 (JPEG-Q30) with cloud recognition, High Efficiency Video Coding (HEVC)-Low, SplitComputing, Deep Joint Source-Channel Coding (DeepJSCC), generic semantic communication, and the proposed method. All comparisons are tested under the same link conditions, and the uplink load actually produced by each method is recorded.

Table 1 shows that the self-collected NB-IoT dataset covers typical urban traffic scenarios; the semantic coordinates are constrained within the range of 64 dimensions, 8 blocks, and 4–16 bit per-dimension quantization, and the typical transmission volume of a single vehicle is 96–160 bytes, which can match the 128/256-byte application-layer payload budget. This configuration enables the proposed method to complete one or a few packet transmissions within the prescribed application-layer payload budget, providing a fair budget baseline for the subsequent main experiments.

Table 1. Narrowband Internet of Things (NB-IoT) simulation configuration

Item	Setting
Datasets	UA-DETRAC, VeRi-776, VehicleID, CityFlow, self-collected NB-IoT dataset
Data split	Public datasets follow their standard training/testing splits; self-collected dataset 8k/2k
Input resolution	256×256 or 384×384
Semantic dimension d	64
Number of semantic blocks K	8
Per-dimension quantization bit width	4–16 bit
Simulated effective uplink rate	40–200 kbps
Application-layer effective payload budget	128/256/512 B
SNR	-5–15 dB
RSRP	-120–80 dBm
Transmission time ratio budget	1%
Retransmission	0–4 times

Note: SNR = Signal-to-Noise Ratio; PSRP = Reference Signal Received Power; NB-IoT = Datasets and Narrowband Internet of Things.

Table 2. Comparison of main experiments

Method	Bytes per Vehicle	UA-DETRAC Top-1 / %	VeRi-776 Rank-1 / %	VeRi-776 mAP / %	Transmission Budget Satisfaction Rate / %
Tiny-YOLO + metadata	16	89.2	61.5	42.3	100
JPEG-Q30 + cloud	4096	82.6	58.2	40.1	12
HEVC-Low	2048	84.1	60.4	43.6	25
SplitComputing	512	88.5	68.7	51.2	68
DeepJSCC	128	90.3	72.6	55.8	100
Generic semantic communication	128	91.1	74.0	57.4	100
The proposed method	128	93.8	82.4	67.1	100

Note: YOLO = You Only Look Once; JPEG-Q30 = Joint Photographic Experts Group-Quality factor 30; HEVC = High Efficiency Video Coding; DeepJSCC = Deep Joint Source-Channel Coding; mAP = mean Average Precision.

3.2 Vehicle recognition and re-identification experiments

Experiment 1 compares the classification and re-identification performance of each method under the same or a controlled transmission budget. Table 2 gives the UA-DETRAC classification Top-1, the VeRi-776 Rank-1 and mAP, the average number of bytes per vehicle, and the transmission budget satisfaction rate.

In Table 2, the proposed method achieves 93.8% Top-1, 82.4% Rank-1, and 67.1% mAP under the budget of 128 bytes per vehicle. Compared with DeepJSCC, the proposed method improves Top-1 by 3.5 percentage points, Rank-1 by 9.8 percentage points, and mAP by 11.3 percentage points. Compared with generic semantic communication, Rank-1 and mAP are improved by 8.4 and 9.7 percentage points, respectively. This shows that although generic semantic communication can compress visual information, it is not optimized for vehicle identity discrimination, whereas the semantic coordinates of vehicle identity in this paper preserve cross-frame identity consistency. Tiny-YOLO with metadata transmits only 16 bytes and its classification Top-1 is acceptable, but its Rank-1 is only 61.5% and its mAP is only

42.3%, indicating that metadata loses appearance semantics and cannot support re-identification. Although JPEG-Q30 and HEVC-Low have a large number of bytes, their transmission budget satisfaction rates are only 12% and 25%, which can hardly meet the requirements under the currently prescribed low-power transmission budget. SplitComputing transmits 512 bytes with a transmission budget satisfaction rate of 68%, but its Rank-1 is still 13.7 percentage points lower than that of the proposed method. The proposed method achieves the best trade-off among recognition accuracy, re-identification capability, and transmission budget satisfaction.

3.3 Link adaptation and progressive transmission experiments

Experiment 2 verifies the second innovation. Figure 5 compares fixed compression and the proposed progressive transmission in terms of the first-block accuracy, the full accuracy, the number of retransmissions, and the budget satisfaction rate under different SNRs. Table 3 gives the classification and re-identification performance when semantic blocks arrive block by block.



Figure 5. Progressive transmission performance under different link qualities

Table 3. Progressive recognition performance when semantic blocks arrive block by block

Number of Arrived Blocks m	Top-1 / %	Rank-1 / %	mAP / %	Cumulative Bytes per Vehicle
1	76.4	52.1	39.8	16
2	83.2	61.5	48.6	32
3	88.6	72.1	56.3	48
4	91.0	76.8	61.2	64
5	92.4	79.5	64.0	80
6	93.2	81.2	66.0	96
8	93.8	82.4	67.1	128

Note: mAP = mean Average Precision.

Figure 5 shows that at a low SNR of -5 dB, the Top-1 of the first block of the proposed method has reached 84.5%, higher than the 78.2% of the full transmission of fixed compression, indicating that discriminative gain ranking concentrates the

most critical vehicle identity information in the highest-priority block. As the SNR increases to 15 dB, the full Top-1 of the proposed method reaches 94.1% and Rank-1 reaches 83.0%, while the number of retransmissions decreases from

2.8 to 0.7. Fixed compression retransmits 3.6 times when the link is poor, which tends to increase the pressure on the transmission time budget. Through prioritizing the transmission of high-gain blocks and adaptive bit width, the proposed progressive transmission significantly reduces retransmission under the same budget satisfaction rate. This result verifies the effectiveness of link-state awareness and progressive transmission.

Table 3 shows that the first 3 semantic blocks account for only 37.5% of the full 128-byte payload, yet achieve 88.6% Top-1, 72.1% Rank-1, and 56.3% mAP. The first 4 blocks reach 91.0% Top-1 and 61.2% mAP. The mAP gains brought by the 5th to the 8th block are 2.8, 2.0, and 1.1, respectively, showing an obvious marginal decrease. This phenomenon indicates that the saliency prediction head effectively identifies the blocks with high discriminative gain, so that the cloud can still quickly obtain usable recognition results when the link is poor or the budget is tight. If the link quality is good, continuing to send the remaining blocks can further improve the re-identification accuracy.

3.4 Ablation experiments on semantic encoding and quantization

Experiment 3 verifies the first innovation. Table 4 performs ablation on the dual branch, the information bottleneck, metric learning, discriminative gain ranking, and adaptive bit width.

In Table 4, the baseline uses only single-branch encoding

and uniform quantization, and its Rank-1 is 68.3%. After adding the dual branch, Top-1 is improved by 3.5 percentage points, indicating that the context embedding helps category determination. After adding the information bottleneck, Rank-1 is improved by 3.6 percentage points and mAP is improved by 3.4 percentage points, showing that suppressing background and illumination redundancy is especially important for re-identification. After adding metric learning, Rank-1 and mAP are improved by 3.3 and 3.3 percentage points, respectively. After adding discriminative gain ranking, the full accuracy improvement is limited, but the Top-1 of the first 3 blocks is increased from 82.6% to 88.6%, indicating that ranking mainly improves the early usability of progressive transmission. Finally, with adaptive bit width added, the full Top-1 and Rank-1 reach 93.8% and 82.4%, respectively, and mAP reaches 67.1%. This ablation verifies the component complementarity of the three innovations: the information bottleneck and metric learning improve identity discrimination, while discriminative gain ranking and adaptive bit width improve transmission efficiency.

3.5 Feedback refinement and resource allocation experiments

Experiment 4 verifies the third innovation. Table 5 compares no feedback, uniform refinement, and on-demand refinement, and analyzes the sensitivity of the triggering threshold.

Table 4. Ablation on semantic encoding and quantization

Configuration	Top-1 / %	Rank-1 / %	mAP / %	Top-1 of the First 3 Blocks / %	Bytes per Vehicle
Baseline	82.1	68.3	52.4	72.1	128
+ dual branch	85.6	72.5	56.8	75.4	128
+ dual branch + IB	88.2	76.1	60.2	79.8	128
+ dual branch + IB + metric	90.1	79.4	63.5	82.6	128
+ dual branch + IB + metric + α ranking	92.3	81.0	65.8	88.6	128
+ all (adaptive bit width)	93.8	82.4	67.1	88.6	128

Note: IB: Information Bottleneck; mAP = mean Average Precision.

Table 5. Feedback refinement strategies and threshold analysis

Strategy	Top-1/%	Rank-1 / %	mAP / %	Bytes per Vehicle	Difficult Sample Recall / %
No feedback	90.2	75.6	60.3	96	62.4
Uniform refinement	92.1	79.8	64.7	160	71.2
On-demand refinement (low threshold)	94.1	82.9	67.5	152	80.1
On-demand refinement (medium threshold)	93.8	82.4	67.1	128	78.5
On-demand refinement (high threshold)	92.6	80.5	64.9	104	70.3

Note: mAP: mean Average Precision.

In Table 5, the no-feedback strategy uses only 96 bytes, with a Rank-1 of 75.6% and a difficult sample recall of 62.4%. Uniform refinement increases the bytes to 160 and improves Rank-1 to 79.8%, but the gain per bit is limited. On-demand refinement with the medium threshold achieves 93.8% Top-1, 82.4% Rank-1, and 67.1% mAP under 128 bytes, saving 32 bytes compared with uniform refinement, improving mAP by 2.4 percentage points, and improving difficult sample recall by 7.3 percentage points. The low threshold triggers more refinement, with slightly higher accuracy, but the bytes increase to 152. The high threshold saves bytes, but the difficult sample recall drops to 70.3%. Therefore, the medium threshold can be selected in actual deployment as a trade-off between accuracy and budget. This result shows that feedback refinement concentrates the limited uplink budget on vehicles

with high uncertainty, avoiding the waste caused by uniform allocation.

3.6 Cross-domain robustness and system overhead experiments

Experiment 5 verifies the generalization capability and the system overhead. Figure 6 compares the performance under daytime, nighttime, rain and fog, occlusion, and motion blur conditions. Table 6 compares the system overhead and the transmission budget satisfaction.

Figure 6 shows that the proposed method achieves a Rank-1 of 85.1% and an mAP of 70.2% under daytime conditions, which are 13.1 and 15.1 percentage points higher than those of DeepJSCC, respectively. Under nighttime conditions, the

Rank-1 of the proposed method is 80.4%, 15.1 percentage points higher than that of DeepJSCC. Under the occlusion condition, the improvement is 16.5 percentage points. Under the motion blur condition, the improvement is 15.8 percentage points. The reason is that the information bottleneck and metric learning make the semantic coordinates focus on vehicle identity-related structures rather than global textures that are susceptible to illumination, rain and fog, and blur. Feedback refinement further performs high-resolution re-encoding of difficult samples, alleviating the loss of discriminative information caused by occlusion and blur. This result shows that the proposed method has strong robustness under cross-domain and degraded visual conditions.

In Table 6, the proposed method has an edge latency of 42

ms, a cloud latency of 35 ms, and an edge energy consumption of 18.5 mJ, all better than those of DeepJSCC and SplitComputing. The proposed method uses 128 bytes per vehicle, with a transmission budget satisfaction rate of 100% and 0.9 retransmissions. Although SplitComputing has a low cloud latency, its 512 bytes per vehicle lead to a budget satisfaction rate of only 68% and 2.2 retransmissions. Through progressive transmission and early termination, the proposed method reduces invalid transmission and retransmission, providing a basis for further deployment verification of NB-IoT low-power and low-rate scenarios. Combining Figure 6 and Table 6, the proposed method achieves a good balance between cross-domain robustness and system overhead.

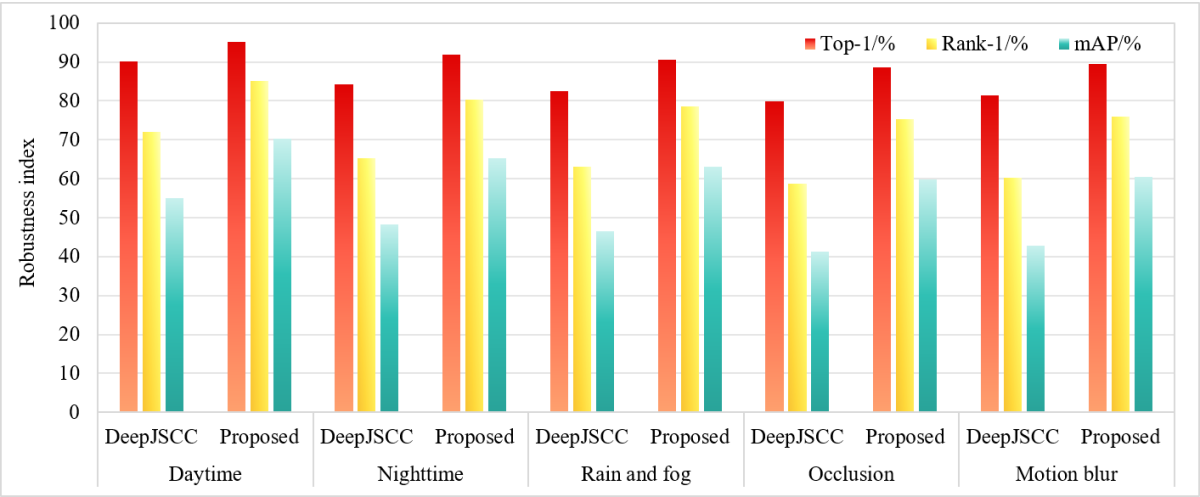


Figure 6. Robustness under cross-domain and degraded conditions

Table 6. System overhead and transmission budget satisfaction

Method	Edge Latency / ms	Cloud Latency / ms	Edge Energy Consumption / mJ	Bytes per Vehicle	Transmission Budget Satisfaction Rate / %	Number of Retransmissions
The proposed method	42	35	18.5	128	100	0.9
DeepJSCC	68	52	31.2	128	100	1.8
SplitComputing	85	40	40.1	512	68	2.2

Note: DeepJSCC = Deep Joint Source-Channel Coding.

3.7 Visual analysis of vehicle identity semantic extraction and feedback refinement

To intuitively examine the capability of the proposed method to extract vehicle identity-related visual information in complex traffic scenes, traffic surveillance images containing multiple vehicle categories and complex backgrounds are selected for visual analysis, as shown in Figure 7. Through object detection and ROI cropping at the edge, the vehicle to be recognized is separated from the original scene, and the local information with potential identity discriminative value, such as the body contour, the top additional structure, the side logo, and the rear functional structure, is further presented. This result shows that the vehicle ROI can provide a relatively independent visual input for identity semantic encoding, so that the subsequent feature extraction focuses on the target vehicle rather than the whole traffic scene. Combined with the progressive recognition results in Table 3, the first three semantic blocks achieve 88.6% classification Top-1 accuracy and 72.1% re-identification Rank-1 accuracy under a transmission volume

of 48 bytes, indicating that the semantic block organization driven by discriminative gain can preserve effective vehicle recognition information under a limited transmission budget. It should be pointed out that the annotation of the local elements in the figure is used to explain the potential discriminative cues in the vehicle appearance, and does not directly represent the spatial attention distribution learned by the model.

To examine the supporting role of the proposed feedback mechanism for vehicle identity matching under the condition of insufficient initial semantics, a cross-camera retrieval scenario of visually similar vehicles is constructed to show the process of cloud feedback, edge local refinement, and matching result update, as shown in Figure 8. In the initial stage, the cloud performs vehicle retrieval based on the received semantic blocks, and judges whether supplementary semantics are needed by using the classification uncertainty and the gallery-matching ambiguity. After the feedback is triggered, the edge performs high-resolution re-encoding of the specified vehicle region according to the downlink query, extracts supplementary visual information such as headlights,

grille, and local structures, and sends it back to the cloud through quantized semantic blocks. Combined with the overall experimental results in Table 5, the on-demand refinement strategy achieves a Rank-1 accuracy of 82.4% and an mAP of 67.1% under an average transmission volume of 128 bytes, both 6.8 percentage points higher than those of the no-feedback strategy, and the difficult sample recall rate is

increased from 62.4% to 78.5%. The results show that on-demand feedback oriented to recognition uncertainty can reduce unnecessary semantic transmission, improve the utilization efficiency of the limited uplink budget, and improve the vehicle identity discrimination performance in complex traffic scenes.

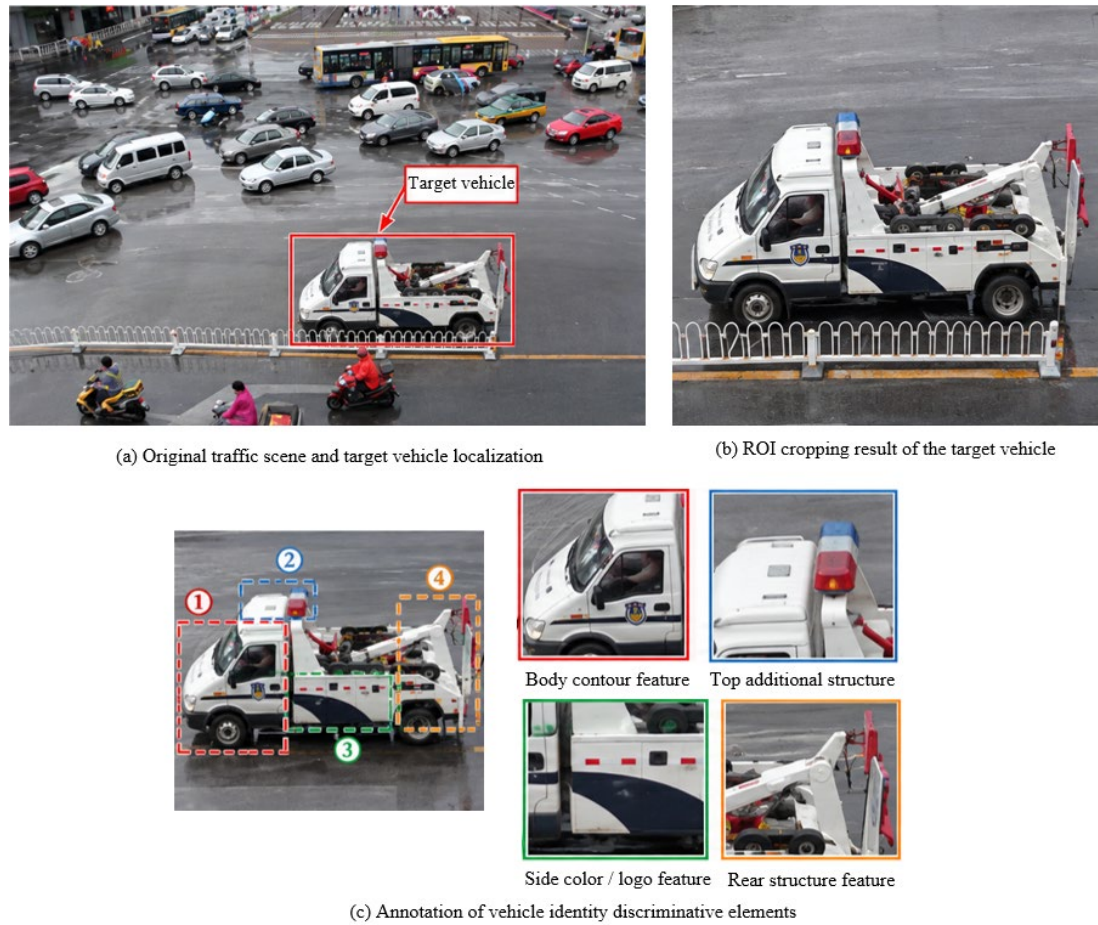


Figure 7. Schematic diagram of target vehicle extraction and identity discriminative elements

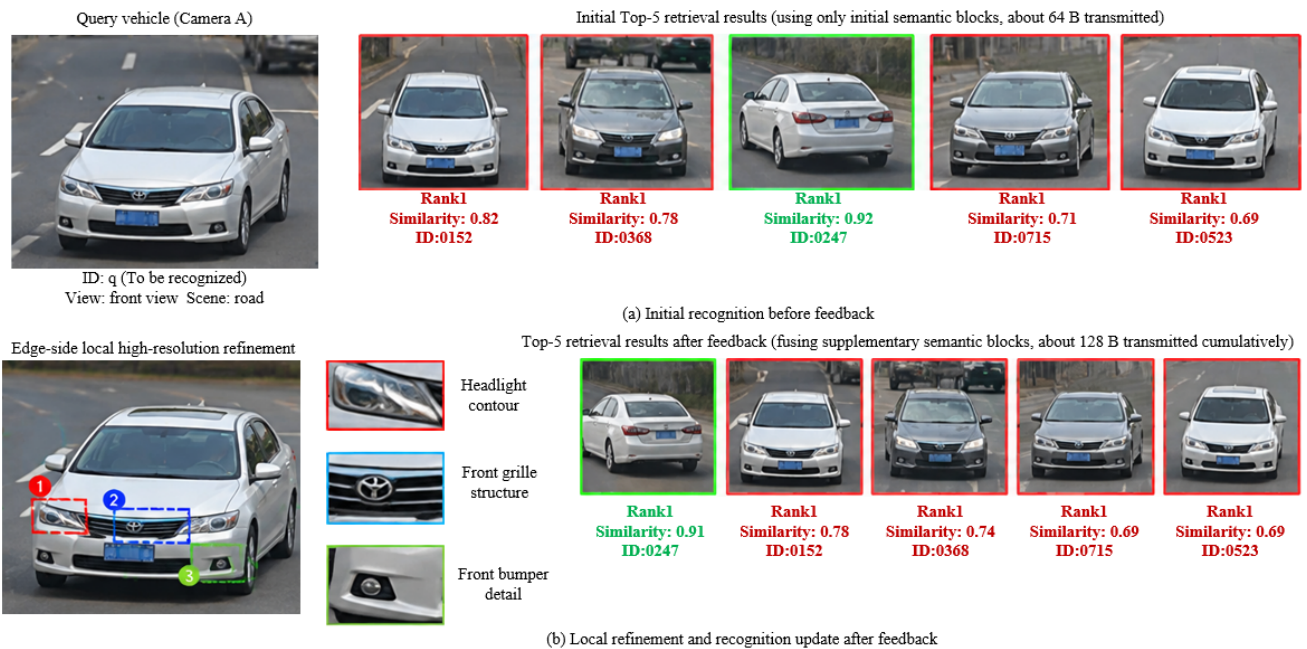


Figure 8. Schematic diagram of cloud feedback-driven local refinement and matching update for difficult vehicles

4. CONCLUSION

This paper addresses the contradiction between the high-dimensional redundancy of traffic images and the vehicle recognition requirement in the NB-IoT narrowband uplink scenario, and proposes a framework of semantically compact encoding, link-adaptive progressive transmission, and edge-cloud feedback refinement, which uses the discriminative gain for vehicle identity as a unified variable. The edge encodes the vehicle region into quantizable, rankable, and queryable semantic coordinates of vehicle identity, and suppresses background and illumination redundancy through the information bottleneck and metric learning. The transmission side allocates semantic blocks and quantization bit widths according to the link state and the discriminative gain, so that the limited uplink bits preferentially carry the semantic components with the highest identity discriminative capability. The cloud triggers compact downlink queries based on the progressive classification result, guiding the edge to perform high-resolution re-encoding of difficult vehicles and to send back supplementary semantics. Experiments on UA-DETRAC, VeRi-776, VehicleID, and the self-collected NB-IoT traffic image dataset show that, under a per-vehicle transmission budget of tens to a few hundred bytes, the proposed method outperforms existing compressive transmission and edge metadata schemes on vehicle classification and re-identification tasks, and satisfies the prescribed low-power transmission budget constraint. The results show that shifting the transmitted object from visual data to semantic coordinates of vehicle identity can effectively accomplish semantic compression of traffic images and vehicle recognition under narrowband constraints.

Although the proposed framework performs stably in most scenarios, extreme occlusion, long-distance nighttime capture, and severe multi-vehicle overlap may still lead to incomplete region-of-interest detection at the edge, so that the semantic blocks cannot cover the key identity regions and the gain of the supplementary semantics from feedback refinement is limited. Future work will introduce multi-frame spatio-temporal semantic aggregation, using vehicle motion continuity to compensate for the semantic missing in a single frame. An incremental update mechanism for the cross-camera vehicle gallery will be constructed to improve the re-identification generalization capability in open scenarios. End-to-end deployment tests will also be carried out in a real NB-IoT network to further verify the system feasibility under real link latency, transmission budget, and energy consumption constraints. In addition, the joint optimization of detection and semantic encoding is expected to reduce front-end redundant computation and provide a more compact solution for low-power traffic visual perception.

REFERENCES

- [1] Zhu, W., Wang, Z., Wang, X., et al. (2023). A dual self-attention mechanism for vehicle re-identification. *Pattern Recognition*, 137: 109258. <https://doi.org/10.1016/j.patcog.2022.109258>
- [2] Nadif, S., Sabir, E., Elbiaze, H., Haqiq, A. (2022). Traffic-aware mean-field power allocation for ultradense NB-IoT networks. *IEEE Internet of Things Journal*, 9(21): 21811-21824. <https://doi.org/10.1109/JIOT.2022.3182854>
- [3] Abbas, M.T., Grinnemo, K.J., Brunstrom, A., et al. (2024). Evaluating the impact of pre-configured uplink resources in narrowband IoT. *Sensors*, 24(17): 5706. <https://doi.org/10.3390/s24175706>
- [4] Michelinakis, F., Al-Selwi, A.S., Capuzzo, M., Zanella, A., Mahmood, K., Elmokashfi, A. (2020). Dissecting energy consumption of NB-IoT devices empirically. *IEEE Internet of Things Journal*, 8(2): 1224-1242. <https://doi.org/10.1109/JIOT.2020.3013949>
- [5] Zhang, H., Kuang, Z., Cheng, L., Liu, Y., Ding, X., Huang, Y. (2024). AIVR-Net: Attribute-based invariant visual representation learning for vehicle re-identification. *Knowledge-Based Systems*, 289: 111455. <https://doi.org/10.1016/j.knsys.2024.111455>
- [6] Gandor, T., Nalepa, J. (2022). First gradually, then suddenly: Understanding the impact of image compression on object detection using deep learning. *Sensors*, 22(3): 1104. <https://doi.org/10.3390/s22031104>
- [7] Wang, Z., Li, F., Zhang, Y., Zhang, Y. (2023). Low-rate feature compression for collaborative intelligence: Reducing redundancy in spatial and statistical levels. *IEEE Transactions on Multimedia*, 26: 2756-2771. <https://doi.org/10.1109/TMM.2023.3303716>
- [8] Ma, S., Qiao, W., Wu, Y., et al. (2023). Task-oriented explainable semantic communications. *IEEE Transactions on Wireless Communications*, 22(12): 9248-9262. <https://doi.org/10.1109/TWC.2023.3269444>
- [9] Duan, L., Liu, J., Yang, W., Huang, T., Gao, W. (2020). Video coding for machines: A paradigm of collaborative compression and intelligent analytics. *IEEE Transactions on Image Processing*, 29: 8680-8695. <https://doi.org/10.1109/TIP.2020.3016485>
- [10] Khan, A., Khattak, K.S., Khan, Z.H., Gulliver, T.A., Abdullah. (2023). Edge computing for effective and efficient traffic characterization. *Sensors*, 23(23): 9385. <https://doi.org/10.3390/s23239385>
- [11] Hu, W., Zhan, H., Shivakumara, P., Pal, U., Lu, Y. (2024). TANet: Text region attention learning for vehicle re-identification. *Engineering Applications of Artificial Intelligence*, 133: 108448. <https://doi.org/10.1016/j.engappai.2024.108448>
- [12] Tu, J., Chen, C., Huang, X., He, J., Guan, X. (2022). DFR-ST: Discriminative feature representation with spatio-temporal cues for vehicle re-identification. *Pattern Recognition*, 131: 108887. <https://doi.org/10.1016/j.patcog.2022.108887>
- [13] Wang, S., Wang, Z., Wang, S., Ye, Y. (2022). Deep image compression toward machine vision: A unified optimization framework. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(6): 2979-2989. <https://doi.org/10.1109/TCSVT.2022.3230843>
- [14] Hawlader, F., Robinet, F., Frank, R. (2024). Leveraging the edge and cloud for V2X-based real-time object detection in autonomous driving. *Computer Communications*, 213: 372-381. <https://doi.org/10.1016/j.comcom.2023.11.025>
- [15] Wang, D., Wang, Q., Tu, Z., et al. (2024). Vision-language constraint graph representation learning for unsupervised vehicle re-identification. *Expert Systems with Applications*, 255: 124495. <https://doi.org/10.1016/j.eswa.2024.124495>
- [16] Kim, Y., Jeong, H., Yu, J., et al. (2023). End-to-end learnable multi-scale feature compression for VCM. *IEEE Transactions on Circuits and Systems for Video*

- Technology, 34(5): 3156-3167. <https://doi.org/10.1109/TCSVT.2023.3302858>
- [17] Jiang, W., Shen, H., Xu, Z., Yang, C., Yang, J. (2024). A feature compression method based on similarity matching. *Displays*, 83: 102728. <https://doi.org/10.1016/j.displa.2024.102728>
- [18] He, Y., Yu, G., Cai, Y. (2023). Rate-adaptive coding mechanism for semantic communications with multi-modal data. *IEEE Transactions on Communications*, 72(3): 1385-1400. <https://doi.org/10.1109/TCOMM.2023.3335977>
- [19] Lyu, Z., Zhu, G., Xu, J., Ai, B., Cui, S. (2024). Semantic communications for image recovery and classification via deep joint source and channel coding. *IEEE Transactions on Wireless Communications*, 23(8): 8388-8404. <https://doi.org/10.1109/TWC.2023.3349330>
- [20] Sun, S., Qin, Z., Xie, H., Tao, X. (2024). Task-oriented scene graph-based semantic communications with adaptive channel coding. *IEEE Transactions on Wireless Communications*, 23(11): 17070-17083. <https://doi.org/10.1109/TWC.2024.3450697>
- [21] Pang, X., Zheng, Y., Nie, X., Yin, Y., Li, X. (2024). Multi-axis interactive multidimensional attention network for vehicle re-identification. *Image and Vision Computing*, 144: 104972. <https://doi.org/10.1016/j.imavis.2024.104972>
- [22] Wang, W., An, P., Huang, X., Huang, K., Yang, C. (2023). Intermediate deep feature coding for human-machine vision collaboration. *Journal of Visual Communication and Image Representation*, 95: 103859. <https://doi.org/10.1016/j.jvcir.2023.103859>
- [23] Nagamatsu, N., Ise, K., Hara, Y. (2024). Mixed-precision neural architecture search and dynamic split point selection for split computing. *IEEE Access*, 12: 137439-137454. <https://doi.org/10.1109/ACCESS.2024.3455251>
- [24] Cao, Z., Cheng, Y., Zhou, Z., et al. (2024). Edge-cloud collaborated object detection via bandwidth adaptive difficult-case discriminator. *IEEE Transactions on Mobile Computing*, 24(2): 1181-1196. <https://doi.org/10.1109/TMC.2024.3474743>
- [25] Zhang, H., Wang, H., Li, Y., Long, K., Nallanathan, A. (2023). DRL-driven dynamic resource allocation for task-oriented semantic communication. *IEEE Transactions on Communications*, 71(7): 3992-4004. <https://doi.org/10.1109/TCOMM.2023.3274145>