ENHANCING DISASTER EARLY WARNING CAPABILITIES VIA EDGE-BASED MULTIMODAL VISION SYSTEMS: A CASE STUDY IN TUYEN QUANG

PHAN VAN NAM[1*], DO HUU SON[2]
[1]*Post and Telecommunications Institute of Technology, Ha Noi, Vietnam*
[2] *Ha Noi University of Science and Technology, Ha Noi, Vietnam*
Email address: nampv0903@gmail.com
**Corresponding author: Phan Van Nam*
http://doi.org/10.51453/3093-3706/2026/1475

ARTICLE INFO

Received: 15/02/2026
Revised: 22/3/2026
Published: 28/4/2026

KEYWORDS

Local Vision-Language; Model;
Edge Computing;
Disaster Early Warning;
Structured Data Extraction;
Visual Evidence.

ABSTRACT

Frequent extreme disasters in Tuyen Quang (e.g., 2025 Yen Son fires and Lo River floods) highlight the limitations of current systems: meteorological forecasts lack visual verification, and traditional cameras fail during bandwidth disruptions. To address this, we propose an independent edge-based early warning solution using small Vision-Language Models (sVLM) and grammar-based constrained decoding to analyze on-site imagery. Instead of free-form text, the model directly infers hazard levels into strict JSON schemas. Experiments on local hardware achieved 100% structural message integrity and 100% anomaly detection accuracy. The optimal model operates at 59.50 TPS with an end-to-end latency of just 0.1487 seconds, capping peak VRAM at 1.66 GB. By replacing continuous video streaming with verified JSON alerts, this method eliminates false alarms and optimizes bandwidth by 99.9880%, proving the viability of standalone smart monitoring stations for rapid evacuation during infrastructure blackouts.

1. INTRODUCTION

1.1. Background and Urgency

Under the impacts of global climate change, the frequency and intensity of extreme weather patterns in Southeast Asia are significantly increasing, posing unprecedented challenges for natural disaster early warning and response systems [1]. In Vietnam, Tuyen Quang is one of the northern mountainous provinces directly and severely affected by these changes due to its steep and highly fragmented terrain. In 2025 alone, the province had to cope with disasters of large scale and devastating magnitude. Specifically, a severe forest fire in Yen Son district in March 2025 [2] destroyed vast areas of vegetation, while a historic widespread flood in the Lo River basin in October 2025 submerged many residential areas, causing traffic congestion and paralyzing essential infrastructure [3].

Furthermore, modern hydrological models have indicated that flood risks in the Lo - Gam river basin are occurring with more unpredictable cycles; the rapid water concentration due to localized heavy rainfall significantly narrows the golden time for emergency response preparation [4]. This harsh reality demonstrates that while macro-forecasting systems are necessary, they are insufficient to meet on-site emergency evacuation demands. The delay in verifying on-site information or the paralysis of traditional surveillance camera systems when telecommunications networks are disrupted demands a new technological solution. Therefore, researching and applying an early warning system capable of automatic monitoring, on-site image analysis, and independent operation at the edge has become an urgent requirement to

protect the lives and properties of local residents.

Figure 1: Wildfires and Floods in Tuyen Quang in 2025

1.2. Limitations of Current Monitoring Systems

While traditional CCTV networks serve as the first line of defense in disaster management, extreme weather reveals their fatal architectural flaw: an absolute dependence on broadband infrastructure. Current field cameras act as "dumb" peripherals, continuously streaming massive video volumes to centralized data centers for processing [5]. This bandwidth reliance becomes a critical vulnerability during disasters, as floods, landslides, and power outages frequently sever fiber optics and paralyze cellular stations, instantly bottlenecking surveillance data [6].

This physical fragility exposes the inherent risk of "cascading failures" within centralized Cloud-based IoT architectures [7]. Lacking on-site analytical capabilities, these cameras become completely useless when internet connectivity drops. Consequently, the telecommunications collapse fragments the warning system into massive information "blind spots" [8], leaving command centers oblivious to rapid on-site developments and fatally delaying emergency evacuation decisions.

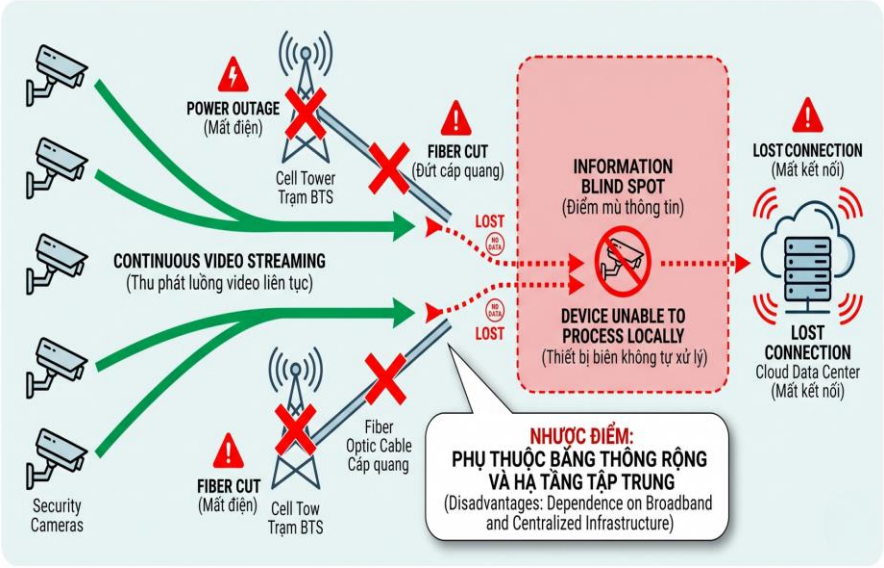

Figure 2: Diagram illustrating the cascading failure of the cloud-based camera system

When high-tech systems fail, emergency response is forced to rely on manual patrols. However, in steep and fragmented mountainous regions, this approach faces severe practical limitations. Disasters like flash floods frequently cut off access routes, causing massive information latency and preventing broad-scale site coverage [9]. Furthermore, unpredictable scenarios like rapidly spreading forest fires place human patrols at direct life-threatening risk due to limited communication [10]. Human physical endurance simply cannot sustain continuous 24/7 monitoring under extreme environmental conditions such as toxic smoke, low light, or severe storms [11]. This lack of independent on-site visual analysis creates a critical gap, highlighting the urgent need for Edge Computing technologies to replace both fragile Cloud IoT infrastructure and limited human labor.

1.3. Research Objectives and Proposed Methodlogy

To resolve dependencies on cloud infrastructure and overcome telecommunications "blind spots," this study proposes a paradigm shift from centralized cloud to edge computing. The core objective is developing a multimodal early warning system that grants monitoring devices full autonomy. By integrating sVLM, edge devices process image data directly on-site. This enables offline inference to automatically detect anomalies, such as forest fires or rising water levels, without requiring an Internet connection during analysis.

The methodological breakthrough lies in utilizing Constrained Decoding algorithms. Instead of outputting free-form text or transmitting heavy video streams, the system forces the sVLM to generate structured text messages. Converting pixel data into semantic data radically optimizes the payload, shrinking Megabyte-sized videos into micro-text files of just a few hundred Bytes. This ultra-lightweight footprint allows seamless alert broadcasting via Low-Power Wide-Area Networks (LPWAN) like LoRaWAN or SMS, completely bypassing bandwidth congestion and severed fiber optics.

Deploying this solution in Tuyen Quang targets three practical objectives: ensuring 24/7 alert continuity regardless of network collapse; providing an AI-driven "visual evidence" mechanism to verify forecasts and eliminate false alarms; and optimizing hardware costs by leveraging sVLMs compatible with small Edge GPUs. Ultimately, this research establishes a robust foundation for smart IoT monitoring networks, empowering local authorities to make immediate, highly accurate evacuation decisions amid increasingly severe natural disasters.

2. RELATED WORKS

2.1. Small Vision-Language Models (sVLM)

The modern artificial intelligence era is witnessing the rise of large-scale VLMs with superior spatial awareness and multimodal reasoning capabilities [12]. However, the barrier of tens to hundreds of billions of parameters strictly limits their deployment to Cloud Data Centers. This dependence generates significant transmission latency and the risk of system paralysis during telecommunication infrastructure failures, especially in disaster response scenarios [13]. To thoroughly overcome this limitation, the research community has shaped a new direction: designing compact sVLMs, enabling the shift of the entire visual reasoning process from the cloud directly to edge devices. Below are the typical sVLM architectures serving as the foundation for the independent disaster monitoring system.

2.1.1. LlaVA model family and visual instruction tuning priciple

LLaVA (Large Language-and-Vision Assistant) is one of the pioneering architectures demonstrating that small-scale language models can effectively integrate with image encoders via a simple projection network. By applying the "visual instruction tuning" methodology, LLaVA-1.5 has unlocked the capability for computers not only to detect objects but also to understand complex contexts through text-based instructions [14]. LLaVA's methodology laid the groundwork for training compact VLMs with logical reasoning capabilities closely matching those of colossal models, serving as a prerequisite for building environmental context-aware AI stations.

2.1.2. Qwen-VL model and detailed spatial awareness capability

Unlike the first generation of VLMs, which primarily focused on describing the overall picture, Qwen-VL is designed to pass detailed-level visual tests. Qwen-VL's architecture enables the model to accurately localize object coordinates and excels particularly in Optical Character Recognition (OCR) tasks even in complex contexts [15]. In an early disaster warning system, this capability is of vital importance, allowing the device not only to recognize the presence of flooding but also to accurately "read" the numbers on partially mud-obscured water level markers.

2.1.3. Ultra-lightweight SmolVLM2 and MobileVLM architectures for edge computing

To optimize hardware to the highest degree, model families like SmolVLM2 or MobileVLM are architected with extremely low parameter counts ([16], [17]). By applying neural network compression and knowledge distillation algorithms, these architectures thoroughly solve the memory bottleneck problem. These ultra-lightweight models can operate smoothly with a high token generation rate (TPS) on small workstations, which are only equipped with consumer-grade graphics cards (such as NVIDIA GeForce RTX 3060 with 12GB VRAM). This brings enormous economic benefits when mass-deploying monitoring Gateway stations in telecommunication blind spots.

2.1.4. Vintern-1B: Localized VLM solution for the Vietnamese context

Although international sVLMs possess high performance, they often reveal limitations in accuracy and processing speed when reasoning and outputting complex text formats in Vietnamese, as well as being less sensitive to the specific vegetation of Vietnam's mountainous terrain. To fill this gap, Vintern-1B emerged as a perfect localized solution. This is a VLM architecture with approximately 1 billion parameters, enhanced through training on a Vietnamese multimodal corpus. Evaluations at the RIVF conference system have proven Vintern-1B's superior capability in structured text extraction and anomaly detection in Vietnam [18]. Utilizing this localized sVLM at edge devices not only eliminates language barrier errors in warning JSON files but also optimizes accuracy when recognizing field images in the Lo River basin or the forests of Yen Son.

2.2. Grammar Sampling and Constrained Decoding

Despite the immensely powerful context awareness of VLMs, their nature remains that of probability-based autoregressive sequence generation models. Therefore, when required to output analysis results in strict data structure formats for downstream automation systems, these models frequently encounter "hallucination" phenomena [19]. Typical common errors include generating redundant explanatory text, arbitrarily modifying data field names, or violating structural syntax (missing quotation marks, incorrect data types). In the IoT architecture of an early disaster warning system, even the slightest syntax deviation will cause the downstream software system to fail in parsing (parsing error), leading to the risk of paralyzing the entire evacuation command flow. Thus, controlling and guaranteeing the integrity of the output is not just an optimization problem but a mandatory requirement [20].

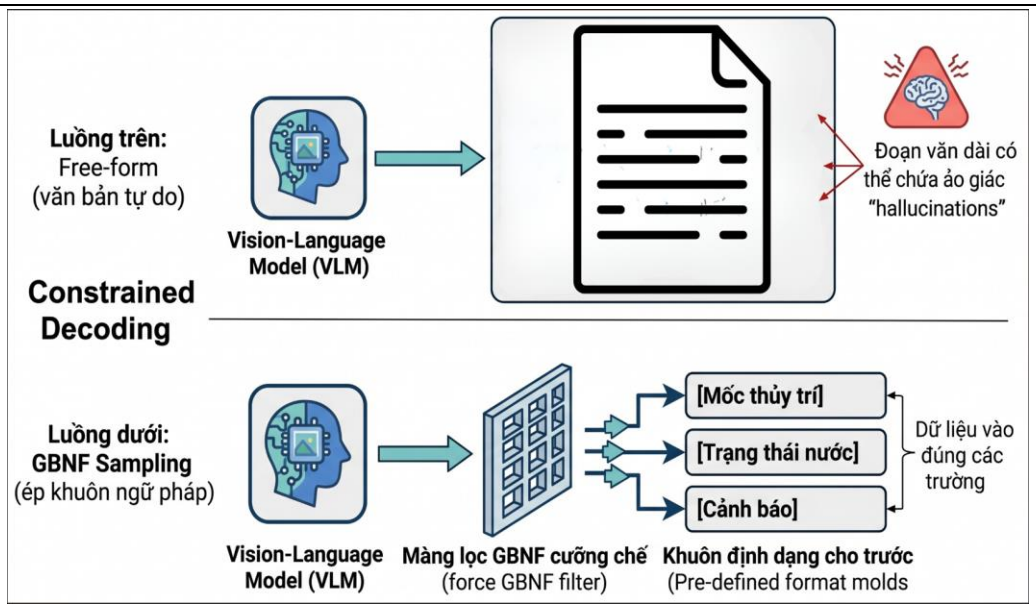

Figure 3: Comparison of conventional text generation mechanism and GBNF

To thoroughly address formatting vulnerabilities, the constrained decoding technique is applied directly at the inference layer using GBNF (GGML Backus-Naur Form) grammar sampling [21]. Instead of altering model weights, GBNF acts as a strict token-generation filter. The system pre-compiles the required JSON schema into a finite state tree grammar rule set. During inference, the GBNF filter evaluates the vocabulary's probability distribution against this grammar state at each time step. If a predicted token threatens to violate the predefined schema for instance, attempting to generate a text string when an integer is required for the MocThuyTri field, the algorithm immediately intervenes by forcing that token's probability to zero [22]. This mathematical coercion transforms the VLM from a free-text generator into a highly deterministic data extraction tool. Consequently, it completely eliminates formatting hallucinations, guaranteeing that 100% of edge-generated warning messages achieve absolute structural integrity, ready for immediate broadcast over ultra-narrowband networks without requiring manual verification.

2.3. Edge Computing in Emergency Warning

In disaster management requiring rapid, second-level responses, centralized cloud architectures reveal severe bottlenecks in latency and stability. To overcome this, edge computing has emerged as a vital paradigm shift, relocating processing and data analysis from central servers directly to the network boundary alongside IoT cameras [23]. Fusing deep learning with edge computing Edge AI, enables the on-site analysis of massive data volumes without cloud uploads, establishing new approaches for autonomous emergency response systems [24].

The first core advantage of this architecture is minimizing end-to-end latency. During rapidly developing disasters like flash floods or forest fires, transmitting high-resolution video to a central server creates a massive round-trip delay, squandering golden evacuation time. Processing emergency data directly at the edge node completely eliminates network transmission delays, allowing devices to interpolate visual context and detect anomalies almost instantaneously [25].

Email: sjttu@tqu.edu.vn

Beyond speed, the independence and resilience of edge systems are crucial. While centralized architectures risk total collapse when storms or landslides sever backbone telecommunication lines, decentralized Edge AI grants full autonomy to field devices. On-site workstations can automatically perform offline inference via sVLMs without relying on broadband Internet [26]. Consequently, even if an area's infrastructure is completely isolated, the edge monitoring system continues analyzing images. It then utilizes minuscule bandwidth, such as LoRaWAN, to transmit structured text warning packets, successfully maintaining the survival of the emergency information flow.

2.4. Related Works

To enhance disaster early warning efficiency, early systems relied on Wireless Sensor Networks (WSN) to collect physical time-series data like temperature or humidity [27]. Despite low power consumption, WSNs inherently lack "visual context." Relying solely on scalar thresholds, they frequently generate false alarms from normal environmental disturbances, such as direct sunlight causing a temperature spike, which ultimately requires manual cross-verification [28].

To overcome this "blindness", research shifted toward traditional computer vision techniques. Convolutional Neural Networks and object detection architectures like YOLO enabled high-speed detection of smoke, fire, or flooded areas from real-time video streams [29]. However, these architectures remain rigid, stopping at drawing bounding boxes with static labels. They completely lack semantic reasoning capabilities and cannot interpret contextual severity. Furthermore, outputting results as meaningless coordinate matrices forces the system to rely on an additional complex, hard-coded middleware layer to convert those physical coordinates into actionable alarm messages [30].

Overcoming the rigidity of previous technological generations, our research proposes a groundbreaking approach using VLMs. The core novelty lies in fusing visual perception and linguistic reasoning into a single end-to-end cycle [31]. Edge devices now not only "see" objects but "understand" the complex spatial context like human experts. Crucially, by utilizing the Constrained Decoding algorithm, the VLM directly infers threat levels and generates warning messages perfectly formatted as JSON schemas. This innovation eliminates the need for complex intermediate middleware, transforming surveillance cameras into independent intelligent entities capable of providing structured data directly to downstream IoT infrastructures, even when completely network-isolated.

3. PROPOSED METHODOLOGY

3.1. Overall Data Flow Architecture

To realize autonomous emergency warnings in fragmented terrain, this study designs a completely decentralized, closed-loop data flow architecture. The process begins with local data collection, where IP cameras stationed at critical points such as Lo River water markers or Yen Son fire watchtowers periodically extract static frames and transmit them directly to an on-site edge device via local networks. This strategy inherits IoT models from large-scale environmental monitoring, entirely eliminating the need to maintain bandwidth-intensive continuous cloud video streams [32].

As soon as the frame enters the edge workstation, the core analysis phase begins using the sVLM. The architecture's key differentiator is the parallel intervention of the Grammar-Based Constrained Decoding (GBNF) algorithm directly during the text generation reasoning process. Shifting deep learning inference down to edge devices effectively prevents transmission bottlenecks [33]. Instead of generating free-form descriptions, the system constrains the model to output hazard assessments strictly complying with a predefined JSON schema. This instantly transforms raw images into semantic data fields in a single computational step, bypassing any latency caused by centralized server APIs.

Finally, the generated JSON alert is extremely compact only a few hundred Bytes making it perfectly compatible for broadcasting via Low-Power Wide-Area Networks (LPWAN) like LoRaWAN. Practical evaluations prove LoRaWAN signals offer excellent obstacle penetration and stable long-distance connectivity in dense, mountainous rural environments [34], significantly outperforming traditional cellular networks in remote coverage [35]. Ultimately, this optimized architecture transforms the monitoring station into a fully independent intelligent entity capable of self-acquiring images, analyzing context, and broadcasting emergency signals without relying on vulnerable backbone fiber optic cables.

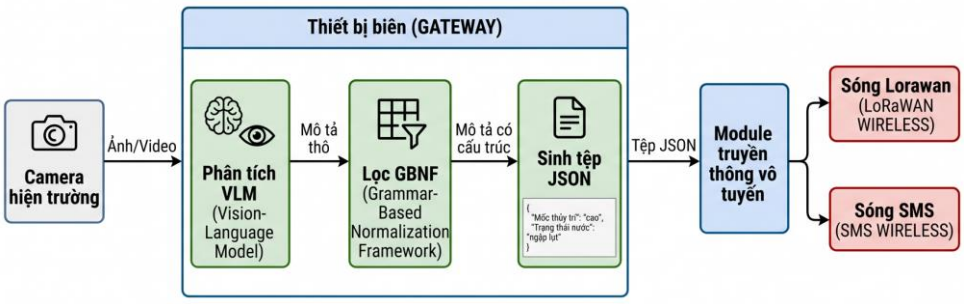

Figure 4: Overall data flow architecture of the system.

3.2. Scenario-based Message Schema Stucture

To ensure the command center's software system can automatically receive and accurately process massive alerts from monitoring stations, all edge VLM output must be uniformly standardized. The study selects JavaScript Object Notation (JSON) as the data structure framework. According to the IETF's RFC 8259 standard, JSON is an ultra-lightweight, language-independent text data interchange format optimized for machine parsing [36]. Using statically structured JSON strings instead of free text radically optimizes payload size, making it an ideal standard for message transmission protocols in low-infrastructure IoT networks [37].

The JSON Schema structure is divided into general metadata and scenario-specific data. General metadata is mandatory for every packet, including MaThietBi (Device ID), ThoiGian (Timestamp), LoaiThamHoa (Event Type), and CoCanhBao (Alert Status). The MaThietBi field provides the Gateway station's unique identifier, helping the command center accurately locate the critical area to promptly direct rescue forces. The ThoiGian field records the exact disaster detection moment, serving as a legal reference and enabling end-to-end latency measurements. The LoaiThamHoa field acts as a routing flag, allowing the central server's JSON parser to identify the situation and activate the corresponding processing flow.

Scenario-specific data is flexibly defined based on the practical situations overseen by the

monitoring station. For forest fire scenarios (e.g., the Yen Son area), the Schema requires the VLM to infer static fields like MucDoKhoi (Smoke Level: Nhe, DayDac) and ToaDoUocLuong (Estimated Coordinates based on camera perspective). For flood scenarios (e.g., the Lo River basin), the model must extract quantitative information including MocThuyTri (Water Level Marker in meters) and TrangThaiNuoc (Water Status: DangNhanh, ChayXiet, BinhThuong). Below is a complete illustrative JSON structure for a flood warning scenario generated from an edge device:

Figure 5: Illustrative JSON structure, flood warning (left), fire warning (right)

Table 1: JSON Schema message structure for the multimodal warning system

Name	Data type	Classification	Detailed Description
MaThietBi	(String)	General Management	Unique identifier of the edge Camera/Gateway device issuing the alert.
ThoiGian	(Integer) / (String)	General Management	Timestamp of image acquisition (in Epoch time or ISO 8601 format), used for latency measurement and real-time synchronization.
LoaiThamHoa	(Enum)	General Management	Disaster scenario classification (“ChayRung” or “NgapLut”).
CoCanhBao	(Boolean)	General Management	Boolean status indicating whether the system has generated an actual alert (true or false).
MucDoKhoi	(Enum)	Wildfire	Smoke density level detected through image analysis (e.g., KhongCo, Nhe, DayDac).
ToaDoUocLuong	Float	Wildfire	Estimated geographic coordinates (latitude and longitude) of the fire based on the edge device location.
MocThuyTri	(Float)	Flood	Water level extracted from visual observation of hydrological markers/gauges.
TrangThaiNuoc	(Enum)	Flood	Water flow status (e.g., BinhThuong, DangNhanh, ChayXiet) used to trigger evacuation commands.

Strictly binding these data fields not only eliminates redundant words in the VLM's analysis but also ensures the central monitoring system can redraw the disaster progression map in real-time with absolute accuracy without any manual human intervention.

3.3. Optimization of Inference Configuration Paramenters

Email: sjttu@tqu.edu.vn

To practically deploy sVLMs on resource-constrained edge devices with consumer GPUs, applying parameter quantization techniques is mandatory. Using advanced compression algorithms like GPTQ and AWQ, the system maps model weights from a 16-bit continuous space (FP16) down to a 4-bit integer format (INT4) [38]. Specifically, AWQ maintains model sensitivity by preserving the original precision for the critical 1% of weights that decisively impact the activation layer [39]. Combined with the GGUF allocation format, the VLM size shrinks from 14-16 GB to approximately 4 GB, completely solving edge storage bottlenecks.

Alongside parameter compression, the system must optimize internal cache memory by controlling context length. KV Cache VRAM consumption scales linearly with processed tokens, risking Out-Of-Memory (OOM) errors if uncontrolled [40]. Because disaster warning is a single-turn reasoning task (inputting a static frame and a JSON schema prompt), the system hard-limits the context window at 1024 tokens. This dual optimization caps peak VRAM consumption at a safe threshold, maintains high token generation rates, and reserves essential memory for smoothly operating LoRaWAN broadcasting protocols during active duty.

3.4. Evaluation Metrics

To comprehensively and objectively evaluate the performance of the multimodal disaster early warning system at the edge device, the study establishes a standardized metric framework. This framework is divided into four core indicator groups, covering everything from data structural integrity, visual cognitive capability, hardware operational performance, to transmission infrastructure optimization efficiency.

3.4.1. Structural Message Integrity

Structural Message Integrity For the task of extracting structured data for downstream automation systems, the validity of the output format is of decisive significance. The system uses the JSON Parsing Success Rate (JPSR) as the primary metric (Scholak et al., 2021). This indicator is defined as the percentage of output messages that pass the parsing library's RFC 8259 standard syntax check without throwing an exception, out of the total inference requests:

$$JPSR = \frac{N_{success}}{N_{total}} \times 100\% \quad (1)$$

Where $N_{success}$ is the number of structured JSON files successfully decoded by the central server, and N_{total} is the total number of frames subjected to testing.

3.4.2. Disaster Detection Capability

To align the measurement method with renowned international multimodal evaluation benchmarks like MMMU or Video-MME, which specialize in assessing AI contextual understanding capabilities ([41], [42]), the study adapts traditional binary classification metrics to the task of detecting field environmental anomalies. The two core indicators applied include:

Sensitivity (True Positive Rate - TPR): Measures the model's ability to not miss a disaster when an actual natural disaster occurs (forest fire or flooding).

Email: sjttu@tqu.edu.vn

False Positive Rate (FPR): Measures the frequency at which the model mistakenly identifies normal natural agents (such as fog, clouds, reflections on water) as an emergency disaster.

3.4.3. Real-time Perfomance and Resource Consumption

The study applies internal LLM/VLM inference performance measurement standards agreed upon by modern model serving architectures (Pope et al., 2023). The nature of the autoregressive model inference process is divided into two distinct mathematical phases: the parallel processing phase of the input prompt/image and the sequential token generation phase. Real-time and resource consumption metrics are defined by the following specific mathematical formulas:

Time to First Token (TTFT): The time elapsed from when the system loads the system prompt and image into the model until the first token of the JSON structure is generated. This metric reflects the processing speed of image context computation in the *prefill stage* (this is a compute-bound phase):

$$TTFT = T_{first\_token} - T_{request} \quad (2)$$

Where $T_{request}$ is the timestamp the system receives the frame and starts preprocessing, and T_{first_token} is the timestamp the first token appears at the decoding layer.

Tokens Per Second (TPS): The average number of text tokens the Edge-VLM model generates per second during the *decode stage*. This stage processes each token sequentially, thus depending on the edge hardware's memory bandwidth (memory-bound). TPS is calculated by the formula:

$$TPS = \frac{N_{token}}{T_{end} - T_{first\_token}} \quad (3)$$

Where N_{token} is the total number of output tokens, and T_{end} is the timestamp marking the completion of the entire result sequence.

End-to-End Latency (E2E): This is the most practical metric directly reflecting the end-user experience. E2E Latency encompasses the entire lifecycle of a processing flow, starting from when the edge device receives the text command, through the LLM's JSON generation inference process, until the Gateway successfully parses the syntax and is ready to trigger physical API calls. Calculation method: Calculated as the sum of the latency of the model's text generation process plus the Gateway system's parsing latency.

$$E2E = T_{api\_ready} - T_{prompt_{received}} \quad (4)$$

Where T_{api_ready} is the timestamp when the Gateway confirms the JSON file is completely valid and pushes it into the command execution queue.

Peak VRAM Consumption: The maximum graphic memory footprint of the system during the closed-loop computation cycle. This quantity is determined by the formula summing static and

Email: sjttu@tqu.edu.vn

dynamic memory energy:

$$M_{peak} = \max_{t}(M_{weights} + M_{context}(t) + M_{activation}(t)) \quad (5)$$

Where $M_{weights}$ is the fixed capacity of the model weights after quantization (Q4 or Q8), $M_{context}(t)$ is the dynamic KV Cache memory capacity that gradually increases with the context window length at time *t*, and $M_{activation}$ is the temporary memory reserved for activation layers during forward propagation. This metric is tightly pinned to ensure Out-Of-Memory (OOM) errors do not occur.

4. RESEARCH RESULTS

4.1. Environment and Test Dataset

Importantly, this experimental architecture does not aim to optimize bounding box mean Average Precision (mAP) like traditional models (e.g., YOLO), whose physical coordinate outputs require complex middleware to generate alerts. Instead, it evaluates the sVLM's ability to directly translate visual context into logically structured text (JSON). Due to this fundamental difference (physical coordinates vs. logical semantics), we omit quantitative comparisons with YOLO. Evaluation focuses entirely on structural message integrity, Token Per Second (TPS) performance, and VRAM consumption across different sVLMs on constrained edge hardware.

To assess practical feasibility, the system was simulated as an Edge Gateway typical of small-scale monitoring stations. The workstation features an Intel Core i5, 16GB RAM, and an NVIDIA GeForce RTX 3060 GPU (12GB VRAM). This consumer-grade GPU accurately reflects the strict resource limitations inherent to mass remote deployments.

Algorithmically, the experiment compares three leading sVLM using quantization to meet memory limits. The primary model, Vintern-1B (optimized for Vietnamese contexts), utilizes 8-bit (Q8) compression to retain weight precision. For an objective baseline, Qwen3-VL 4B (Q4) and SmolVLM2 2.2B (Q4) are included. These compressions shrink model weights to just 1.5–3 GB VRAM, leaving an ample safety margin on the 12GB RTX 3060 for the context cache and OS. All models operate on a shared inference framework integrated with the GBNF filter.

Due to the extreme scarcity of real flood footage in Vietnam, the test dataset (3 videos, ~1500 frames) was synthesized from two main sources:

- **Forest fire scenario:** Utilizes a subset of the standardized FLAME dataset's Test set, featuring UAV aerial imagery with accurate fire boundary labels [43].

- **Flood scenario:** Extracts frames from actual flood footage posted by Khanh Hoa Radio and Television on TikTok [44]. These close-up shots of rising water during severe weather provide high experimental value for testing the model's visual interpolation capacity.

4.2. Structural Message Integrity

In edge-based disaster warning systems, structural message integrity requires generating

Email: sjttu@tqu.edu.vn

machine-readable messages for automatic downstream processing, rather than merely describing visual content. Instead of outputting free-form text, the pipeline utilizes a vision-first approach: the VLM classifies the observed state, which is then strictly standardized into a fixed JSON schema. For forest cameras, the JSON includes MaThietBi, ThoiGian, LoaiThamHoa, MucDoKhoi, CoCanhBao, and ToaDoUocLuong. Crucially, in non-fire scenarios, LoaiThamHoa remains "ChayRung" (identifying the camera's monitored task), but MucDoKhoi is set to "KhongCo" and CoCanhBao to false. This strict logic prevents the system from misinterpreting routine transmissions as active fire alerts. For river cameras, the JSON comprises MaThietBi, ThoiGian, LoaiThamHoa, MocThuyTri, TrangThaiNuoc, and CoCanhBao. In this context, CoCanhBao evaluates to true only when the water status (TrangThaiNuoc) is "DangNhanh" or "ChayXiet", and evaluates to false when the status is "BinhThuong".

The benchmark demonstrated absolute structural integrity, with Qwen3VL-4B, Vintern-1B, and SmolVLM2 achieving a 100,0% JSON parse rate across all 189 test records (63 per model: 34 flood, 19 wildfire, and 10 non-wildfire). All generated messages successfully included the mandatory fields. Timestamps were standardized to local ISO format (e.g., 2026-05-22T02:16:09+07:00), and camera coordinates defaulted to 0,0, 0,0 to reflect practical post-installation configuration. Similarly, the river camera's MocThuyTri (water level) field defaulted to 0,0 because the current dataset lacked a readable visual gauge. Consequently, flood warnings in this experiment rely on the qualitative TrangThaiNuoc and logical CoCanhBao fields, while MocThuyTri remains in the schema for future physical sensor integration. Ultimately, this proves the post-processing mechanism is critical for transforming unstable VLM outputs into reliable business messages, effectively eliminating schema errors, missing keys, and deserialization failures at the gateway or IoT server.

Figure 6: Model testing results on a single sample, left (flood), right (forest fire)

4.3. Sensitivity and False Positives

The sensitivity of the system is evaluated based on its capability to correctly identify samples with anomalies, which means frames showing signs of a forest fire or flooding/inundation. For the forest fire task, the labels Nhe and DayDac are considered positive because both indicate the detection of smoke/fire at some level, whereas KhongCo is considered negative. For the flood task, DangNhanh and ChayXiet are viewed as warning states, while BinhThuong is non-warning. This conversion aligns with practical deployment requirements: the business layer does not necessarily

only care if the model calls the exact detailed level correctly, but first needs to know whether to generate an alert or not. Therefore, the CoCanhBao field was added to turn the classification result into a clear binary signal for the IoT system, while still retaining detailed labels like MucDoKhoi or TrangThaiNuoc to serve deeper analysis.

Table 2: Flood detection sensitivity

Model	Samples	Dang Nhanh	Chay Xiet	Binh Thuong	True	False Negative	Sensitivity
Qwen3VL-4B	34	2	32	0	34/34	0	100,0%
Vintern-1B	34	0	34	0	34/34	0	100,0%
SmolVLM2-2.2B	34	34	0	0	34/34	0	100,0%

Table 3: Forest fire detection sensitivity

Model	Samples	DayDac	Nhe	KhongCo	True Positive	False Negative	Sensitivity
Qwen3VL-4B	19	14	0	5	14/19	5	73,7%
Vintern-1B	19	7	7	5	14/19	5	73,7%
SmolVLM2	19	3	8	8	11/19	8	57,9%

Table 4: False Positive Rate on non-fire video

Model	Samples	Khong Co	Nhe	DayDac	True Negative	False Positive	False Positive Rate
Qwen3VL-4B	10	10	0	0	10/10	0	0,0%
Vintern-1B	10	10	0	0	10/10	0	0,0%
SmolVLM2	10	10	0	0	10/10	0	0,0%

Based on Table 2 data, flood detection yielded the most stable results. Exploiting the Khanh Hoa Television footage, all three models achieved absolute 100,0% sensitivity, correctly identifying 34/34 samples requiring alerts with zero False Negatives (FN). However, kinetic classification lacked consistency: Vintern-1B labeled all 34 samples as ChayXiet, SmolVLM2 chose DangNhanh, while Qwen3-VL 4B outputted 32 ChayXiet and 2 DangNhanh. This discrepancy does not affect business decisions, as both labels logically trigger CoCanhBao = true. Lacking negative river videos, this proves positive detection capabilities but cannot establish the False Positive Rate (FPR) for normal water conditions.

According to Table 3, the forest fire task exposed clear inferential divergences. Qwen3-VL 4B and Vintern-1B achieved 73,7% sensitivity (14/19 positive samples). Vintern-1B provided a realistic, softer grading (7 DayDac, 7 Nhe), whereas Qwen3-VL exhibited extreme classification (all 14 DayDac). SmolVLM2 yielded the lowest sensitivity at 57,9% (8 false negatives), struggling when thin smoke blended into the forest background or when low-resolution compression (384x216) dissipated distant visual details.

Table 4 demonstrates a highly positive false alarm scenario: all models achieved a 0,0% FPR, correctly assigning the KhongCo label to 10/10 negative frames. Completely eliminating false alarms

minimizes misdirected operational costs and builds trust. However, given the limited 10-sample negative set, these results require cautious interpretation. The system remains untested against strong visual noise like thick fog, road dust, residential smoke, or sunset background discoloration.

Overall, while the experimental scale is smaller than benchmarks like MMMU or Video-MME, this process strictly quantified VLM text outputs into statistical metrics. The pipeline demonstrates strong binary alert stability, but expanding edge disaster datasets remains critical to optimize sensitivity and accurately estimate FPR prior to real-world deployment.

4.4. Resource Consumption and Real-time Performance

The system's real-time performance is evaluated using three main metric groups: inference latency per frame, token generation speed (TPS), and peak graphic memory consumption (Peak VRAM). Experimental latency is measured end-to-end at the frame level, comprehensively encompassing image preparation, VLM inference, text generation, and JSON schema standardization. TPS reflects the output tokens generated per second, while Peak VRAM tracks the maximum GPU memory usage. These metrics align closely with standard LLM/VLM community evaluations, such as those used by NVIDIA NIM and vLLM. Although standard frameworks often track Time-To-First-Token (TTFT) and inter-token latency, this experiment deliberately excludes TTFT since the architecture does not operate in token streaming mode. Ultimately, tracking the end-to-end latency and TPS is entirely sufficient to validate the feasibility of edge deployment at a continuous 1 frame/second sampling frequency.

Table 5: Real-time performance on the flood scenario

Model	Samples	Avg Latency (s)	P95 Latency (s)	Max Latency (s)	Avg TPS	VRAM (GB)	JSON Bytes
Qwen3VL-4B	34	0,7469	0,7292	5,4324	59,50	8,67	188,1
Vintern-1B	34	0,2848	0,1968	3,8718	58,40	1,66	188,0
SmolVLM2	34	0,3620	0,1605	7,3734	26,56	3,06	189,0

Table 6: Real-time performance on the forest fire screnario

Model	Samples	Avg Latency (s)	P95 Latency (s)	Max Latency (s)	Avg TPS	VRAM (GB)	JSON Bytes
Qwen3VL-4B	19	0,4848	0,6121	0,6480	55,83	8,67	233,5
Vintern-1B	19	0,1911	0,2548	0,3938	56,34	1,66	232,4
SmolVLM2	19	0,1487	0,1747	0,4100	13,27	3,06	232,6

Table 7: Realtime performance on the non-fire screnario

Model	Samples	Avg Latency (s)	P95 Latency (s)	Max Latency (s)	Avg TPS	VRAM (GB)	JSON Bytes
Qwen3VL-4B	10	0,3906	0,5124	0,6321	50,02	8,67	235,0
Vintern-1B	10	0,1700	0,2660	0,3557	25,23	1,66	235,0
SmolVLM2	10	0,1612	0,3015	0,4386	7,88	3,06	235,0

Based on Tables 5, 6, and 7, all three models successfully meet the 1-second/frame sampling cycle throughout most of the processing time. In the flood scenario, Qwen3VL-4B recorded an average latency of 0,7469 seconds, Vintern-1B achieved 0,2848 seconds, and SmolVLM2 reached

0,3620 seconds. While the P95 latency for all models remained strictly under 1 second, the Max Latency column revealed significant initial spikes (5,4324s, 3,8718s, and 7,3734s, respectively). This reflects a common cold-start phenomenon when the runtime initializes context, caches, or loads the multimodal projector, indicating that startup latencies and stabilized latencies should be evaluated separately for continuous deployments.

The forest fire scenario demonstrated much greater latency stability. Qwen3VL-4B averaged 0,4848 seconds (P95: 0,6121s), Vintern-1B averaged 0,1911 seconds (P95: 0,2548s), and SmolVLM2 achieved the lowest average latency at 0,1487 seconds (P95: 0,1747s). However, despite its speed, SmolVLM2's average TPS was only 13,27, which is significantly lower than the other two models. This highlights that end-to-end latency and TPS are not strictly proportional, as factors like output length, chat handlers, and projector runtime process the image prompt differently.

In the non-fire scenario, Vintern-1B (0,1700s) and SmolVLM2 (0,1612s) continued to maintain lower latencies than Qwen3VL-4B (0,3906s). Crucially, when evaluating GPU resources, Vintern-1B proves to be the most economical model, maintaining a Peak VRAM of just 1,66 GB across all scenarios. SmolVLM2 follows at 3,06 GB, while Qwen3VL-4B demands the highest at 8,67 GB. Consequently, Vintern-1B is the most balanced choice for lightweight deployment on VRAM-constrained edge devices. If hardware capacity allows, Qwen3VL-4B remains the stronger candidate for maximum fire detection sensitivity. Ultimately, while all models meet the 1-second real-time requirement, the final deployment decision must balance anomaly detection sensitivity, GPU costs, parallel camera capacity, and gateway scalability.

4.5. Network Traffic Comparison

A major benefit of the edge processing architecture is the drastic reduction in network traffic achieved by transmitting lightweight JSON messages rather than full video streams to the center. After frame analysis, the system generates a tiny alert containing the device ID, timestamp, task type, status, the CoCanhBao flag, and contextual fields like coordinates or water markers. Across all 189 test messages, the average JSON size is approximately 209,2 bytes. Flood scenario messages are the most compact, averaging 188 bytes for Qwen3VL-4B and Vintern-1B, and 189 bytes for SmolVLM2. Conversely, fire and non-fire messages fluctuate between 232 and 235 bytes, reaching the maximum size due to the addition of the ToaDoUocLuong (Estimated Coordinates) structure.

Table 8: JSON size and raw image frame transmission baseline

Model	Video	Avg JSON Bytes	Baseline Frame Bytes	Bandwidth Saving
Qwen3VL-4B	lu_lut.mp4	188,1	1,572,864	99,9880%
Qwen3VL-4B	fire.mp4	233,5	1,572,864	99,9852%
Qwen3VL-4B	no_fire.mp4	235,0	1,572,864	99,9851%
Vintern-1B	lu_lut.mp4	188,0	1,572,864	99,9880%
Vintern-1B	fire.mp4	232,4	1,572,864	99,9852%
Vintern-1B	no_fire.mp4	235,0	1,572,864	99,9851%
SmolVLM2-2.2B	lu_lut.mp4	189,0	1,572,864	99,9880%
SmolVLM2-2.2B	fire.mp4	232,6	1,572,864	99,9852%

SmolVLM2-2.2B	no_fire.mp4	235,0	1,572,864	99,9851%

In the benchmark, the traditional transmission baseline is set to 1,572,864 bytes (1,5 MiB) per raw image frame. Comparing the average 209,2-byte JSON message to this baseline, the transmitted data proportion is merely 0,0133%, yielding a bandwidth saving of approximately 99,9867% per sampling cycle. At a 1 frame/second sampling frequency, the system transmits only about 0,2 KB/s per camera, whereas streaming raw frames demands 1,5 MiB/s. This massive difference is critical for mountainous IoT scenarios characterized by weak transmission lines, high data costs, and frequent connectivity interruptions. Furthermore, transmitting tiny JSONs significantly reduces storage costs, accelerates packet delivery, minimizes communication energy, and ensures the central server exclusively receives actionable, business-summarized data.

Table 9: Capacity characteristics of compressed video files in the test set

Video	Duration (s)	FPS	Frames	File Size (MB)	Compressed Bandwidth (MB/s)
lu_lut.mp4	33,10	30,0	993	16,89	0,510
fire.mp4	18,81	16,0	301	1,17	0,062
no_fire.mp4	9,25	16,0	148	0,69	0,075

Even when compared to compressed video files within the current dataset, the bandwidth savings remain significant. Transmitting the compressed lu_lut.mp4 video requires approximately 0,51 MB/s, fire.mp4 requires 0,062 MB/s, and no_fire.mp4 requires 0,075 MB/s. In stark contrast, the average JSON message utilizes merely 209,2 bytes/second at a 1-second sampling configuration. Thus, transmitting warning JSONs reduces network traffic by hundreds to thousands of times compared to compressed codecs, and the reduction is even more extreme against raw uncompressed frames. This firmly validates the edge inference architecture for disaster warning: massive visual data is processed entirely on-site, while the fragile network is only required to transport concise, structured, and verifiable conclusions.

However, transmitting exclusively JSON creates a trade-off that requires management. Without receiving the original image or video, the central server's capability for manual post-event verification is significantly reduced, complicating the review of false alarms or missed events. To resolve this, a tiered, adaptive data transmission policy is highly practical. The CoCanhBao field can act as a logical trigger: if no alert exists, only the lightweight JSON is sent. If CoCanhBao = true, or if model reliability drops, the system can automatically attach a compressed snapshot, a short video clip, or save the footage locally for later retrieval. This tiered approach retains edge processing's massive bandwidth benefits for routine monitoring while fully preserving crucial investigative capabilities when critical events actually occur.

5. CONCLUSION AND FUTURE DEVELOPMENT

5.1. Discussion Efficiency in Telecommunications Infrastructure Optimization

Deploying broadband networks in mountainous regions like Tuyen Quang is costly, and these infrastructures frequently fail during disasters. The Edge-VLM architecture resolves this vulnerability by locally processing images into ~250-byte JSON files. This micro-payload perfectly

suits Low-Power Wide-Area Networks (LPWAN) like LoRaWAN or SMS, penetrating rugged terrain over 5-15 km radiuses and ensuring uninterrupted emergency communication entirely independent of commercial grids.

Crucially, Edge AI complements, rather than replaces, meteorological macro-forecasting. While macro-models provide probabilistic early warnings, they often suffer from micro-climate errors that trigger false alarms. Edge devices fill this gap by providing on-site visual verification, digitizing physical anomalies into structured data. This establishes a foolproof decision chain: macro-models trigger readiness, but evacuation orders require VLM JSON confirmation, anchoring rescue operations directly to physical realities.

However, relying exclusively on optical sensors imposes inherent limitations. Extreme weather reduces contrast, obscuring details like thin smoke, while mud splattering creates irreversible physical blind spots, increasing False Negatives. Furthermore, standard VLMs trained on daylight data fail during nighttime operations. Future iterations must incorporate self-cleaning modules and penetrative cross-sensors like LiDAR or thermal cameras. To overcome the high costs of physical maintenance in rugged terrain, the architecture deeply integrates Over-The-Air (OTA) remote updates. This allows command centers to remotely push bug fixes, adjust schemas, or deploy new quantized weights, enabling autonomous adaptation and a highly scalable “plug and play” model.

Due to scarce real-world disaster data, this study currently operates as a Proof of Concept. Lacking close-up flood footage, the quantitative MocThuyTri field was hardcoded to 0,0 as a backward-compatible placeholder for future sensor integration; currently, evacuation relies on qualitative kinematic states (DangNhanh, ChayXiet). Additionally, the small negative dataset (10 forest frames, zero river scenarios) means the achieved 0,0% False Positive Rate remains restricted and untested against complex field noise like residential smoke or sunset reflections, making expanded localized datasets a priority.

5.2. Conclusion and Future Work

This study confirms the superior effectiveness of integrating small Vision-Language Models (sVLM) with GBNF grammar sampling directly on edge devices. Forcing VLMs to output structured JSON instead of free text eliminates formatting deviations, achieving an absolute 100% parse success rate and preventing central server crashes. This synergy optimizes computational loads via quantization and shrinks alert payloads to micro-sizes, successfully establishing an independent end-to-end architecture capable of bypassing fiber optic disruptions during disaster outbreaks.

Based on these simulated results, future development will focus on piloting Edge Gateways in disaster-prone hotspots, such as the Lo River basin and high-risk forests in Yen Son. On-site deployment will verify hardware endurance and radio stability under harsh micro-climates, while generating vital empirical data. This real-world visual data will be crucial for fine-tuning VLM weights, overcoming current visual noise limitations, and fully neutralizing false alarms before mass-scaling the early warning network.

REFERENCES

1 IPCC, “Climate Change 2023: Impacts, Adaptation and Vulnerability in Southeast Asia.

Intergovernmental Panel on Climate Change.," *IPCC,* 2023.

2. Nhan Dan Newspaper, "nhandan.vn," 2025. [Online]. Available: https://nhandan.vn/tuyen-quang-da-dap-tat-hoan-toan-dam-chay-rung-tai-nui-nghiem-post866914.html.
3. Tuyen Quang Provincial People's Committee, "Summary report on natural disaster prevention and search and rescue in Tuyen Quang province in 2025", Commanding Committee for Natural Disaster Prevention and Search and Rescue of Tuyen Quang province, 2025.
4. Thuy Duong, "Environment and Life e-Magazine," 2025. [Online]. Available: https://moitruong.net.vn/tuyen-quang-muc-nuoc-song-lo-song-gam-dang-cao-anh-huong-den-nguoi-dan-86276.html.
5. Yuyi Mao, Changsheng You, Jun Zhang, Kaibin Huang, Khaled B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials,* 2017.
6. Partha Pratim Ray, Mithun Mukherjee, Lei Shu, "Internet of Things for Disaster Management: State-of-the-Art and Prospects," *IEEE Access,* 2017.
7. Dinh C. Nguyen, Ming Ding, Pubudu N. Pathirana, Aruna Seneviratne, Jun Li, Dusit Niyato, "6G Internet of Things: A Comprehensive Survey," *IEEE Internet of Things Journal,* 2021.
8. Junaid Qadir, Anwaar Ali, Raihan ur Rasool, Andrej Zwitter, Arjuna Sathiaseelan & Jon Crowcroft, "Crisis analytics: big data-driven crisis response," *SPRINGER NATURE: Journal of International Humanitarian Action,* 2016.
9. Elizabeth A. Basha, Sai Ravela, Daniela Rus, "Model-based monitoring for early warning flood detection," *Proceedings of the 6th ACM conference on Embedded network sensor systems,* 2008.
10. Alkhatib, Ahmad, "A Review on Forest Fire Detection Techniques," *International Journal of Distributed Sensor Networks,* 2014.
11. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks,* 2002.
12. Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei et al, "Flamingo: a Visual Language Model for Few-Shot Learning," *Proceedings of Neural Information Processing Systems (NeurIPS),* 2022.
13. Yuqing Tian, Zhaoyang Zhang,, "An Edge-Cloud Collaboration Framework for Generative AI Service Provision with Synergetic Big Cloud Model and Small Edge Models," *ARXIV,* 2024.
14. Haotian Liu, Chunyuan Li, Qingyang Wu, Yong Jae Lee, "Visual Instruction Tuning," *NeurIPS,* 2023.
15. Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, Jingren Zhou, "Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond," *Computer Vision and Pattern Recognition,* 2023.
16. Xiangxiang Chu, Limeng Qiao, Xinyang Lin, Shuang Xu, Yang Yang, Yiming Hu, Fei Wei, Xinyu Zhang, Bo Zhang, Xiaolin Wei, Chunhua Shen, "MobileVLM : A Fast, Strong and Open Vision Language Assistant for Mobile Devices," *Arxiv,* 2023.
17. Cuenca, P., Wolf, T., & Tunstall, L, "SmolVLM: Small yet powerful vision-language model for edge devices. Hugging Face Technical Report.," Hugging Face Technical Report, 2024.
18. Khang T. Doan, Bao G. Huynh, Dung T. Hoang, Thuc D. Pham, Nhat H. Pham, Quan T.M. Nguyen, Bang Q. Vo, Suong N. Hoang, "Vintern-1B: An Efficient Multimodal Large Language Model for Vietnamese," Proceedings of the 18th IEEE International Conference on Computing

and Communication Technologies (RIVF), 2024.

19. Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Delong Chen, Wenliang Dai, Ho Shu Chan, Andrea Madotto, Pascale Fung, "Survey of Hallucination in Natural Language Generation," *ACM Computing Surveys,* 2022.
20. Torsten Scholak, Nathan Schucher, Dzmitry Bahdanau, "PICARD: Parsing Incrementally for Constrained Auto-Regressive Decoding from Language Models," *EMNLP,* 2021.
21. Brandon T. Willard, Rémi Louf, "Efficient Guided Generation for Large Language Models," *Arxiv: Computation and Language,* 2023.
22. Luca Beurer-Kellner, Marc Fischer, Martin Vechev, "Prompting Is Programming: A Query Language for Large Language Models," *PLDI'23: 44th ACM SIGPLAN International Conference on Programming Language Design and Implementation,* p. 2022, 2022.
23. Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, Lanyu Xu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal,* 2016.
24. Jiasi Chen, Xukan Ran, "Deep Learning With Edge Computing: A Review," *Proceedings of the IEEE,* 2019.
25. Munish Bhatia, Tariq Ahamed Ahanger, "Internet of things-fog computing-based framework for smart disaster management," *Transactions on Emerging Telecommunications Technologies,* 2020.
26. Shri Janani Muruganantham, N. Ramasubramanian, B. Shameedha Begum, "Resilient Edge-IoT Architectures by Optimizing Energy, Security and Adaptability : A Review," *2025 IEEE 9th International Conference on Information and Communication Technology (CICT),* 2025.
27. Kechar Bouabdellah, Houache Noureddine, Sekhri Larbi, "Using Wireless Sensor Networks for Reliable Forest Fires Detection☆," *Procedia Computer Science,* 2013.
28. Ramesh, Maneesha Vinodini, "Design, development, and deployment of a wireless sensor network for detection of landslides," *Ad Hoc Networks,* 2014.
29. Khan Muhammad, Jamil Ahmad, Sung Wook Baik, "Early fire detection using convolutional neural networks during surveillance for effective disaster management," *Neurocomputing,* 2018.
30. Saima Majid, Fayadh Alenezi, Sarfaraz Masood, Musheer Ahmad, Emine Selda Gündüz, Kemal Polat, "Attention based CNN model for fire detection and localization in real-world images," *Applied Soft Computing,* 2022.
31. Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, Enhong Chen, "A Survey on Multimodal Large Language Models," *Arxiv,* 2024.
32. Olakunle Elijah, Tharek Abdul Rahman, Igbafe Orikumhi, Chee Yen Leow, MHD Nour Hindia, "An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges," *IEEE Internet of Things Journal,* 2018.
33. Vestias, Mario P., "A Survey of Convolutional Neural Networks on Edge with Reconfigurable Computing," *MDPI,* 2019.
34. Ramon Sanchez-Iborra, Jesus Sanchez-Gomez, Juan Ballesta-Vinas, Maria-Dolores Cano, Antonio F. Skarmeta, "Performance Evaluation of LoRa Considering Scenario Conditions," *MDPI,* 2018.
35. Kais Mekki, Eddy Bajic, Frederic Chaxel, Fernand Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," *ICT Express,* 2019.
36. Datatracker, "https://datatracker.ietf.org/," 2017. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8259.
37. Naik, Nitin, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," *2017 IEEE International Systems Engineering Symposium (ISSE),* 2017.
38. Elias Frantar, Saleh Ashkboos, Torsten Hoefler, Dan Alistarh, "GPTQ: Accurate Post-Training

Quantization for Generative Pre-trained Transformers," *Arxiv: Machine Learning,* 2023.

39. Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, Song Han, "AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration," *Proceedings of Machine Learning and Systems,* 2023.
40. Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Anselm Levskaya, Jonathan Heek, Kefan Xiao, Shivani Agrawal, Jeff Dean, "Efficiently Scaling Transformer Inference," *Proceedings of Machine Learning and Systems,* 2023.
41. Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, Xing Sun, "Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis," *Computer Vision and Pattern Recognition,* 2024.
42. Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun et al, "MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI.," *CVPR 2024 Oral,* 2024.
43. Alireza Shamsoshoara, Fatemeh Afghah, Abolfazl Razi, Liming Zheng, Peter Fule, Erik Blasch, "The FLAME dataset: Aerial Imagery Pile burn detection using drones (UAVs)," *IEEE,* 2020.
44. Khanh Hoa Television, "ktvkhanhhoa," 2025. [Online]. Available: https://www.tiktok.com/@ktvkhanhhoa/video/7579831016852884737?_r=1&_t=ZS-96WwvpbwjSr.