Chain of Thoughts Based Prompting for Incident Detection Using Large Language Model

Lorraine Chambers* †, Affiong Udokisai*, Haitham Mahmoud*, Khalid ismail*, Nazim Mohammed†, Waldo Cervantes*, Mohamed Gaber*, Rehan Bhana*
*College of Computing, Birmingham City University, Birmingham, United Kingdom
†Conway Packaging Services (CPS) Ltd., Birmingham, United Kingdom
Corresponding Authors: Haitham Mahmoud and Rehan Bhana ({haitham.mahmoud, rehan.bhana}@bcu.ac.uk)

***Abstract*—Warehouse safety is crucial but challenging due to the variety of incidents that occur, such as human errors and equipment failures. Traditional incident reporting often focuses only on the event itself, lacking context on the actions leading to the incident. To address this, this work introduces a system using video analysis to track and document the sequence of events before, during, and after an incident. This system integrates You Only Look Once (YOLO) models YOLOv9, YOLOv11 and Faster R-CNN (Region-based Convolutional Neural Network) to detect and annotate real-time events, capturing the entire incident sequence. Moreover, a language model automatically generates detailed and clear health and safety reports based on the detected actions, ensuring their accuracy and relevance. Testing demonstrated the system's effectiveness in detecting key incidents like forklift mishandling and goods falling. YOLOv11 achieved a precision of 0.806 and a recall of 0.955, with a mean Average Precision (mAP) mAP50 score of 0.972. The system also showed strong sequence detection accuracy, with key events identified with a recall of 1.0 in some cases. Reports generated using Generative Pre-trained Transformer (GPT)-based models showed strong alignment with human-readable text, with a cosine similarity score of 0.874 and a Bidirectional Encoder Representations from Transformers (BERT) F1 score of 0.879. These results indicate that the system improves safety practices by providing comprehensive, actionable insights.**

***Index Terms*—Chain-of-Thought Prompting (CoT), Large Language Models (LLM), Warehouse Safety, Smart Warehousing.**

I. Introduction

Ensuring safety and operational efficiency in warehouses is crucial as e-commerce growth increases goods volume and operational complexity [1]. This expansion heightens accident risks: in the UK, the Transportation and Storage sector reported 15 fatalities and 29,000 non-fatal injuries between 2022–2023 [2], while in the US, the sector had a non-fatal injury rate of 220 per 10,000 workers [3]. In Great Britain, manufacturing data show an average of 94,000 work-related health issues annually between 2021–2024, with fatal injuries steady at 16–17 per year since 2019 [4]. Leading fatality causes in 2024 were moving objects (21%), falls from height (21%), and moving machinery (16%). These figures highlight the need for continuous research and technological innovation in workplace safety [5]. Traditional health and safety monitoring often underreports incidents, delaying safety improvements [6]. Computer vision now enables real-time visual analysis to enhance incident detection and response [7]. Warehouse safety encompasses equipment handling, protocol adherence, machinery risk mitigation, fall prevention, and correct goods placement. This paper focuses on developing computer vision systems for video analysis to automate health and safety report generation in warehouses. These systems monitor unsafe behaviour, equipment faults, hazards, and PPE compliance [7], [8]. Event reporting via video faces challenges, often capturing only the incident, not the surrounding context. Therefore, video annotation targets the full event sequence. Large Language Models (LLMs) are integrated to generate human-readable, coherent reports synthesising incident timelines [9].

While some construction sector research explores automated report generation from visual inspections [10], no prior work combines computer vision with LLMs for automated warehouse health and safety reporting. Hence, this paper aims to introduce computer vision to automatically produce comprehensive health and safety reports via LLMs of incidents that occur in warehouses. Using computer vision, we analyse video and generate health and safety reports directly from it. The contributions are two-fold:

1) Event detection computer vision model to identify health and safety incidents, along with the events that lead up to and follow the incident, thereby producing a coherent flow of events related to the incident. The system uses advanced detection techniques to identify hazards, including mishandled equipment, unstable placements, and movement around forklifts and machinery. Models such as YOLOv9 and YOLOv11 effectively pinpoint and track incidents, offering a clear understanding of causation.
2) Use of LLMs using a prompt generated from the event detection model to generate an immediate comprehensive health and safety report. Advanced language models like GPT-4o, Llama 3.1, Mistral and specialised variations process event data streamline reporting. By dynamically generating prompts informed by detected incidents, the system ensures tailored and actionable safety reports.

This paper is structured as follows: Section II discusses related work. Section III provides an overview of the system. Section IV, outlines the experimental setup. In Section V, our method is analysed with other object detection models

and LLMs. Section VI summarises our findings and identifies future work.

II. Related Work

This section discusses current warehouse health and safety reporting software and technologies, highlighting their limitations and the need for more advanced approaches. It then explores object detection and recent advances in large language models, establishing the foundation and context for the proposed method. Current hazard incident detection and reporting methods range from fully manual processes to hazard management software and computer vision-driven warehouse solutions. Hazard management software, like SafetyCulture [11] and SafetyIQ [12], relies on human-recorded incidents, with analytics but no real-time comprehensive reporting. Computer vision-based solutions like Optima [13], Protex AI [14], and KIBSI [15] offer hazard detection by monitoring existing cameras; Optima focuses on wearable and sensor-based monitoring. While these systems provide some reporting and analytics, none, to our knowledge, deliver comprehensive real-time health and safety incident reporting.

There are studies on warehouse safety and the role of technology in this area. Hofstra et al [16] investigates how warehouse safety can be assessed and facilitated, highlighting the importance of integrating technology, such as video surveillance, to monitor safety behaviors and identify potential hazards. Auyong et al [17] reviews the impact of occupational safety and health measures in logistics and discuss that integrating video technology with existing safety protocols can lead to more effective hazard identification and risk mitigation. The integration of video surveillance in warehouse safety management offers real-time incident response.

A major industry in which computer vision for health and safety has been studied is construction. They have similar monitoring needs to the logistics industry. In surveys, computer vision and IoT are the prevailing technologies for health and safety management, out of the two, computer vision is dominant because of its non-intrusive approach to data collection, supporting the scalability of it [18].

A. Object Detection

Object detection technologies are at the forefront of automated health and safety incident detection. Recent advances in the accuracy and speed of object detection have facilitated this transformational change in computer vision technologies across a multitude of industries such as industrial automation, construction and healthcare. Most notably, the YOLO [19] and Faster Region Convolutional Neural Network (Faster R-CNN) [20] algorithms [21] are leading the way. Recent studies involving object detection in health and safety include PPE monitoring in manufacturing [22], and on construction sites, deep learning and computer vision is used for health and safety management, utilising object detection for PPE via YOLO-5 [23]. However, these depict the single situation of personnel not wearing suitable PPE, but do not describe events leading up to and after an incident.

B. LLM Report Generation

LLMs like GPT-4 [24], built on transformer architecture and self-attention [25], have advanced NLP by learning complex language patterns from diverse datasets. They excel in tasks such as text generation, summarisation, translation, and question-answering [10], and are widely used in chatbots and content creation. LLMs are utilised in supply chain management for tasks such as enhancing the accuracy of demand forecasts, improving risk assessment and mitigation, and automating logistics processes. By analysing large volumes of unstructured data such as market reports, social media sentiment, and consumer reviews, LLMs help to refine demand predictions [26]. LLMs have been studied for comprehensive incident reporting in construction, where unmanned vehicles gather scene data and a multimodal LLM automatically generates inspection reports [10]. However, to the best of our knowledge, there is no study of their application in health and safety reporting. Table I presents an overview of software solutions for identifying and managing hazard incidents. These tools cater to various needs, including streamlined reporting processes, real-time hazard detection, and warehouse safety management. SafetyCulture and SafetyIQ focus on efficient reporting and proactive issue resolution. Optima combines IoT devices and drones for inventory automation and safety analytics. Protex AI and KIBSI utilise advanced computer vision techniques to monitor safety incidents, such as PPE compliance and forklift operations. The table highlights features like Inventory Tracking (IT), Hazard Detection (HD), and Reporting (R), with solutions originating from regions such as Australia, the UK, Ireland, and the USA.

TABLE I: Studies on Hazard Incident Detection and Reporting where IT=Inventory Tracking, HD=Hazard Detection, R=Reporting.

Name	Description	Features	Type	Country
Safety Culture [11]	Tool for hazard reporting, enabling quick issue resolution.	R	Hazard Software	Australia
SafetyIQ [12]	Real-time hazard ID, prioritisation, and control.	R	Hazard Software	Australia
Optima [13]	Wearables, IoT, drones for inventory and hazard reporting.	IT, R	WMS+HD	UK
Protex AI [14]	AI-based PPE and incident monitoring via CCTV.	HD, R	WMS+HD	Ireland
KIBSI [15]	CV for forklift, people, and object tracking via cameras.	IT, HD, R	WMS+HD	USA

A main consideration when using LLMs is the prompt that is supplied to the LLM. LLM prompting techniques optimise model performance across a range of tasks and applications. Techniques range from zero-shot, where no prior examples are given in the prompt, to instruction prompts that provide explicit instructions. Chain-of-Thought (CoT) prompting encourages step-by-step reasoning and enhances the abilities of LLMs significantly. It is used in medical applications which

require a high degree of accuracy [27]. Through rigorous experimentation, Wei et al demonstrated the effectiveness of CoT prompting in improving the depth and coherence of LLM responses [28]. In their studies, they applied this technique, achieving a state-of-the-art accuracy of 90.2% on mathematical and commonsense reasoning benchmarks. This underscores the potential of CoT prompting to significantly elevate the performance of LLMs in tasks that require reasoning.

Although LLMs have made significant progress in multiple areas, their application in generating health and safety reports remains underexplored. To address this gap, our research aims to investigate the potential of automated generation of health and safety reports for the warehouse by combining the use of object detection models and LLMs. Existing literature shows that object detection models are increasingly proficient in their capabilities and LLMs excel at understanding and generating text content. However, the application of these models to automatically interpret warehouse incident scenes to generate comprehensive reports represents an area that has yet to be thoroughly explored. Therefore, the contribution of this study is the development and evaluation of a framework that combines image capture of warehouse incidents with the text generation capabilities of LLMs, aiming to provide an innovative, automated health and safety report generation method aimed at the warehouse logistics industry.

III. System Description

In this section, a comprehensive system description is provided via the process flow, algorithm formalisation and description of the main constituent modules. Figure 1 shows the process flow. A detailed analysis of the video feed is employed, starting with frame extraction, event detection, date and time extraction via Optical Character Recognition (OCR), prompt engineering, and application of the prompt to an LLM to produce a comprehensive health and safety report from a video. Algorithm 1 formally describes this flow. The output of the object detection model can be expressed as:

$$\mathcal{N}(I) = \{(x_i, y_i, w_i, h_i, p_{\text{obj}}, \{p_c\})\}_{i=1}^{N}$$

Where I is the input image, (x_i, y_i) are the centre coordinates for the bounding box, (w_i, h_i) are the width and height of the bounding box. p_{obj} is the objectness score for bounding box i, $\{p_c\}$ is the predicted class probabilities for C classes associated with bounding box i and N is the total number of predicted bounding boxes. The underlying architectures of the event detection models follow the original specifications provided by their respective authors and vary accordingly. These models are adapted for the warehouse event detection task through training on the annotated event classes detailed in Table II.

Initially, a video file (V), an object detection model ($\mathcal{N}$), an OCR engine (O) and a language model (L) are required. An empty event list (A) is also initialised (Line 1), alongside a frame counter (Line 2). The video file (V) is opened. For each 100th frame (F_i), object detection is applied (Lines 4 to 7), resulting in the event detected in the frame (Line 8).

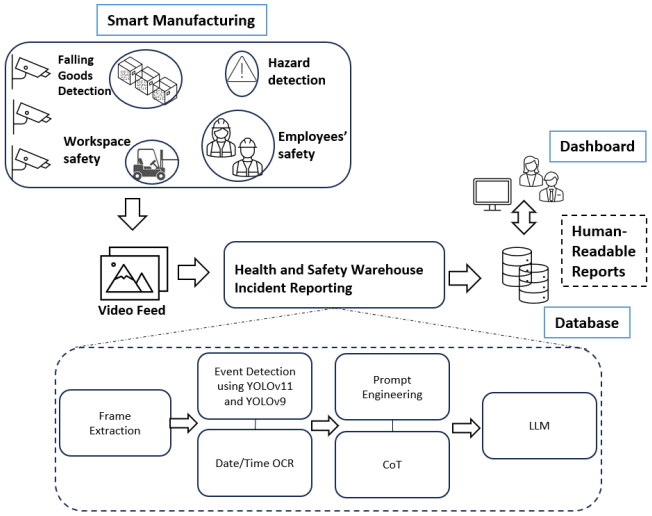

Fig. 1: This process flow aims to generate a Comprehensive Health and Safety Report using video footage and AI-driven analysis.

Algorithm 1 Incident Detection and Reporting

Require: Video file V, object detection model $\mathcal{N}$, OCR engine O, Language model L
Ensure: Incident report R
1: Initialize an empty list $A \leftarrow []$
2: Initialize frame counter $i \leftarrow 0$
3: Open video file V
4: **while** video has frames **do**
5: Read frame F_i from V
6: **if** $i \bmod 100 = 0$ **then** ▷ process every 100th frame
7: Apply object detection model $\mathcal{N}$:

$$\mathcal{N}(F_i) = \{(x, y, w, h, p_{\text{obj}}, \{p_c\})\}_{j=1}^{N}$$

8: Get A_i from $\mathcal{N}(F_i)$ ▷ get bounding boxes & probabilities
9: **if** 'Incident'in A_i **then** ▷ detect incident
10: Apply OCR engine O to F_i
11: Get T_i from OCR ▷ extract date & time information
12: $A \leftarrow (A_i, T_i)$
13: **end if**
14: **end if**
15: $i \leftarrow i + 1$ ▷ increment frame counter
16: **end while**
17: $U \leftarrow \text{unique}(A)$ ▷ get unique events U
18: $P \leftarrow \text{instruction} + U$ ▷ generate CoT prompt P
19: $R \leftarrow L(P)$ ▷ generate report from language model using prompt
return R

If an incident is detected from the events extracted from the predicted bounding boxes (Line 9), the algorithm employs an OCR engine (O) to extract the date and time (T) from the frame (Lines 10 and 11). These details are appended to the events list (A) (Line 12). Following the frame processing, duplicate events are removed from the events list (A), giving unique events of (U) (Line 17). A CoT prompt (P) is generated from a pre-defined instruction combined with the event list U (Line 18). A generic representation of the prompt is shown in Listing 1. Finally, this prompt is submitted to a language

model (L) to create a comprehensive incident report (R). The following sections provide further description of the modules shown in Figure 1.

A. Frame Extraction and Event Detection

The video input V can be represented as a sequence of frames $V = \{F_1, F_2, ..., F_n\}$, where F_i is the i-th frame in the video and n is the total number of frames. For efficiency, every 100th frame of the video is extracted (F_i). For event detection, object detection is utilised ($\mathcal{N}(F_i)$). To achieve this, the data is annotated with specific events in videos from differing camera viewpoints as shown in Figure 2.

(a) Camera angle one (b) Camera angle two

Fig. 2: A picture frame of CCTV footage.

The annotation process involves several key tasks. First, it is necessary to detect specific events within the video, such as the movement of a forklift toward the truck and the forklift carrying goods. These events are critical indicators of the operational flow within the warehouse. By labelling these events, the model can learn to recognise the sequence of events leading up to the incident, which is essential for understanding causality and for developing preventive measures.

To maximise the utility of the annotated data, it is also important to consider the quality and granularity of the labels. High-quality annotations, where all relevant details are accurately labelled, contribute to the effectiveness of the training process. The granularity of the labels and the detail and specificity of the annotations also play a significant role. For example, annotating not just the presence of a forklift but also its specific actions, such as lifting or lowering goods, provides richer information for the model to learn from. Table II shows the annotations that were applied.

TABLE II: Class name and type

Class ID	Class Name
0	Forklift driver and another personnel inspecting the incident
1	Forklift driver and another personnel trying to lift the goods
2	Forklift driver inspecting the incident
3	Forklift lifting goods from the truck
4	Forklift moving goods away from the truck
5	Forklift moving towards the truck to carry goods
6	Forklift preparing to carry goods from the truck
7	Goods fell from the forklift carrying it - Incident

B. Date and Time OCR

The CCTV footage provides a visual record of the incident and offers precise time-stamped evidence. This time-stamped data (T_i) is essential for reconstructing the incident timeline from frame F_i as in $T_i = O(F_i)$. EasyOCR [29] was employed to ensure the accuracy of the extraction of the incident date and time. The advantages of EasyOCR are high accuracy, attributed to deep learning models, and its superior reliability [30]. On the other hand, the Events detected from the object detection and OCR outputs are aggregated into an event list (A) as discussed (Line 12): $A = \{(E_i, T_i)\}_{i=1}^{m}$, where E_i represents the detected event and m is the total number of events.

C. Prompt Engineering, Chain-of-Thoughts (CoTs) and Large Language Model

LLMs can struggle with complex reasoning tasks, limiting their full potential. This can be addressed to some extent via prompt engineering. Prompt engineering involves the crafting of task-specific instructions, or prompts, to guide a model's responses without altering its internal parameters. CoT prompting was introduced by [28]. This technique aims to improve LLMs' logical and methodical information processing by encouraging them to adopt a structured, step-by-step reasoning approach.

The innovation of CoT prompting lies in its ability to transform LLMs' responses, resulting in more organised and reflective outputs compared to conventional prompts that do not provide a logical sequence. By explicitly prompting the model to consider each step of a reasoning process, CoT prompting allows LLMs to decompose complex problems into manageable parts, similar to how humans approach multi-step tasks. For instance, when solving a multi-step math problem, a CoT prompt not only seeks the final answer but also requires the model to detail the entire reasoning process leading to that answer. This approach mimics human cognitive strategies, where intermediate steps are crucial to arriving at an accurate solution. The CoT prompt is applied to LLM models that are optimised for chat-based interactions, ensuring that a complex prompt can be accurately interpreted.

The CoT prompt (P) is constructed from an instruction and the output of the object detection model (U). P = Instruction + Sequence of Events (Line 18). The possible outputs of the object detection model are shown in Table 1. The CoT prompt pre-defined instruction and the placement of the actions are shown in Listing 1, where "`Action 1`" to "`Action n`" are replaced by the output from the object detection model (U), alongside the addition of the extracted date and time of the incident (T_i).

```
"You will create an incident report based on the
following sequence of events. Please analyze each
event step by step and then compile the details
into a structured report.
Sequence of Events:

- Action 1
- Action 2
...
- Action n
Date:
Time:

Let's break this down step by step. First, indicate
the date and time of the incident, then give a brief
```

```
description of incident by explaining the sequence
of event. Also describe the root cause, any injuries/
goods damage, actions taken, and any recommendations."
```

Listing 1: CoT prompt

IV. Experimental Study

A. Video Data

In the development of this solution, the necessary data was provided by Conway Packaging Services (CPS) Ltd., a well-established warehouse known for its efficient logistics and packaging operations. The dataset comprises two important pieces of video evidence, CCTV footage that shows an incident that occurred from two distinct camera angles within the warehouse environment. These videos are used as the foundation for analysing the incident, as the video provides extensive coverage of the event from two different perspectives, enhancing the reliability of the analysis. The content of the videos centres around a certain operational scenario where a forklift is seen carrying out the routine task of moving goods from a truck stationed at the unloading bay. The forklift operator, following standard procedures, begins transporting the goods across the warehouse floor. However, during this process of transportation, an unexpected event occurs - the goods being carried by the forklift slip and fall to the ground, causing an incident. This incident is a critical point of analysis, as it involves not only a potential loss of goods but also raises questions about safety protocols, equipment handling, and operational efficiency within the warehouse environment.

B. Models

Three models, YOLOv9, YOLOv11 and Faster R-CNN, are employed for event detection due to their advanced object detection capabilities. YOLOv9, being an earlier version, and YOLOv11, the latest model, are both pivotal for accurate event detection tasks. They are both fine-tuned to an early stopping of 100, with YOLOv9 and YOLOv11 reaching 284 and 586 epochs, respectively. The YOLO models follow a one-stage object detection philosophy, where an input image is divided into a grid. Each cell directly predicts bounding box coordinates, objectness scores, and class probabilities in a single forward pass. This negates the need for region proposal stages used in two-stage detectors such as Faster R-CNN [20]. YOLOv11 builds upon YOLOv9 by introducing larger-kernel convolutions and lightweight self-attention to capture long-range dependencies between pixels, enabling it to focus on relevant features regardless of their spatial distance. This improves local and global feature extraction. A decoupled head for classification and localisation, an anchor-free prediction strategy, and dynamic label assignment enhance detection accuracy and offer greater flexibility in handling diverse object scales and shapes. In contrast, the Faster R-CNN architecture uses a region proposal network to first generate potential object regions (proposals) and then refines these regions for classification and bounding box regression. In addition, four Small Language Models (SLMs) and three LLMs are utilised for event detection, selected for their cutting-edge performance and chat functionalities. SLMs, which encompass fewer than 20 billion parameters, include Llama-2-7b-chat-hf, Meta-Llama-3.1-8b-Instruct, Mistral-7b-Instruct-v0.3, and gpt-4o-mini. The selected LLMs, known for their expansive parameter counts and advanced capabilities, are gpt-3.5-turbo, gpt-4-turbo, and gpt-4o.

General-purpose SLMs and LLMs as opposed to domain-specific ones are selected due to their strong generalisation, instruction-following capabilities, and ability to generate structured outputs across diverse domains. While domain-specific models trained on industrial safety data offer targeted performance, they typically require large, annotated datasets that are scarce in the warehouse field and lack the linguistic capabilities necessary for generating comprehensive and human-readable reports. General LLMs benefit from large-scale pre-training on diverse corpora and alignment techniques, making them effective in tasks involving structured generation with minimal domain adaptation [31], [32]. Additionally, techniques such as chain-of-thought prompting [28] further enhance their reasoning and coherence in structured report generation. In the absence of publicly available, large-scale safety-specific datasets, general LLMs offer a practical and scalable alternative. Incorporating these models ensures comprehensive coverage of event detection scenarios, leveraging state-of-the-art technology for optimal results.

C. Evaluation

Performance evaluation is conducted in two distinct phases: Event detection model evaluation and language model evaluation. For the event detection model, the mAP, precision and recall metrics are employed. Evaluating the output of the language models necessitates a combination of quantitative and qualitative methods. Quantitative methods include cosine similarity and BERT metrics of precision, recall, and F1-Score. BERT evaluates text based on contextual embeddings, allowing it to assess semantic similarity more effectively than traditional metrics, which rely on surface-level word matching and often fail to capture the deeper meaning of sentences. To compute these metrics, an incident report, reflecting a real incident documented by a manager or warehouse operative, is utilised. This incident report serves as a benchmark, ensuring that the language models' output aligns closely with human-generated reports. The incident report includes comprehensive details of the event, such as the time, location, nature of the incident, and any injuries or damages incurred.

Domain experts from CPS carried out a qualitative evaluation on the highest-scoring generated report. Their judgments were converted into quantitative scores. The generated report was expected to provide a clear and complete account of the incident, including the date, time, description, sequence of events, root cause, any injuries or damage, actions taken, and preventive recommendations. Experts rated each of these elements on a scale of 1 to 10, also identifying factual inaccuracies (hallucinations), evaluating the use of language and terminology for suitability in a professional health and

safety context, and whether the reports were appropriate for the intended audience in terms of clarity and relevance. This thorough evaluation process ensures that the LLM's outputs are reliable and comparable to human-generated documentation.

V. Experimental Results

In this section, the experimental results are analysed and discussed. Firstly, the event detection in the video frames, followed by the report generation. Table III shows the metrics for the event detection in the video frames. The values in bold signify the highest metric values between YOLOv9, YOLOv11 and Faster R-CNN for all classes combined.

TABLE III: Video frame event detection metrics.

Class	Precision	Recall	F1-Score	mAP50	mAP50-95
YOLOv9					
All	0.711	0.840	0.770	0.837	0.679
0	0.743	1.000	0.852	0.845	0.694
1	0.834	1.000	0.910	0.995	0.972
2	0.737	1.000	0.849	0.913	0.844
3	0.464	0.333	0.385	0.381	0.195
4	0.765	0.667	0.712	0.913	0.633
5	0.663	1.000	0.797	0.995	0.699
6	0.574	0.723	0.639	0.663	0.437
7	0.911	1.000	0.953	0.995	0.956
YOLOv11					
All	**0.806**	**0.955**	**0.874**	**0.972**	**0.810**
0	0.794	0.969	0.874	0.945	0.768
1	0.971	1.000	0.985	0.995	0.970
2	1.000	0.952	0.975	0.995	0.897
3	0.527	0.756	0.622	0.863	0.588
4	0.673	1.000	0.805	0.995	0.838
5	0.717	1.000	0.835	0.995	0.748
6	1.000	0.962	0.980	0.995	0.699
7	0.769	1.000	0.869	0.995	0.975
Faster R-CNN					
All	0.525	1.000	0.669	0.964	0.482
0	0.524	1.000	0.688	0.958	0.479
1	0.500	1.000	0.667	1.000	0.500
2	0.333	1.000	0.500	0.916	0.458
3	0.333	1.000	0.500	0.835	0.417
4	0.526	1.000	0.690	1.000	0.500
5	0.818	1.000	0.900	1.000	0.500
6	0.833	1.000	0.909	1.000	0.500
7	0.333	1.000	0.500	1.000	0.500

The evaluation results indicate that YOLOv11 significantly outperformed YOLOv9 across all object detection metrics. YOLOv11 achieved a precision of 0.806, compared to YOLOv9's 0.711 and Faster R-CNN's 0.525, demonstrating its superior ability to predict true positives. Additionally, YOLOv11 exhibited a recall of 0.955 versus 0.840 for YOLOv9, highlighting its strength in detecting true positives. Faster R-CNN did have an overall recall of 1.000, indicating that it did not miss any positive instances; however, with a recall of 0.525, the F1-Score of 0.669 is lower than 0.770 achieved by YOLOv9. The mAP scores further confirmed YOLOv11's dominance, with mAP50 at 0.972 compared to 0.837 for YOLOv9 and 0.964 for Faster R-CNN, and mAP50-95 at 0.810 versus 0.679 and 0.482, respectively. These results underscore YOLOv11's robustness and precision, making it the preferred model for event detection. Table IV shows the unique events that were predicted by YOLOv11, where

TABLE IV: YOLOv11 Detected Events

ID	Detection
1	1. Forklift moving towards the truck to carry goods
2	1. Forklift lifting goods from the truck 2. Forklift preparing to carry goods from the truck
3	1. Forklift lifting goods from the truck 2. Forklift moving goods away from the truck
4	1. Forklift moving goods away from the truck
5	1. Forklift moving goods away from the truck 2. Forklift moving towards the truck to carry goods
6	1. Forklift driver and another personnel inspecting the incident 2. Forklift driver inspecting the incident
7	1. Goods fell from the forklift carrying it - Incident
8	1. Forklift driver inspecting the incident 2. Goods fell from the forklift carrying it - Incident
9	1. Forklift driver and another personnel inspecting the incident 2. Forklift driver inspecting the incident 3. Goods fell from the forklift carrying it - Incident
10	1. Forklift driver and another personnel inspecting the incident 2. Goods fell from the forklift carrying it - Incident
11	1. Forklift driver and another personnel inspecting the incident
12	1. Forklift driver and another personnel trying to lift the goods
13	1. Forklift driver and another personnel inspecting the incident 2. Forklift driver and another personnel trying to lift the goods

the ID column is a number to identify the event and the Detection column lists the object detections that occurred in that frame. It can be seen that the flow of events is generally correct, showing the movement of the forklift towards the truck, carrying goods from it, the goods falling, and personnel inspecting it. However, some events are slightly out of order. For instance, after the forklift truck moves away from the truck, the event detection indicates that it then moves towards the truck again in event 5, detection 2, indicating that there is difficulty determining which direction the forklift is travelling in. The forklift driver and other personnel also inspect the incident before the incident happens in event 6, detection 2. However, the language models make logical sense of this and produce a correct flow of events over multiple runs, demonstrating that this did not impact the quality or coherence of the LLMs output.

For language model evaluation, Table V shows cosine similarity and BERT metrics, where Llama 2 7B is the Llama-2-7b-chat-hf model, Llama 3.1 8B is the Meta-Llama-3.1-8b-Instruct model and Mistral 7B is the Mistral-7b-Instruct-v0.3 model. Mistral-7B emerged as the best-performing Small Language Model (SLM), achieving the highest cosine similarity score of 0.874 and a BERT F1-score of 0.894. Among LLMs, GPT-3.5-turbo displayed superior quantitative performance with a BERT F1-score of 0.883, precision of 0.870, and recall of 0.895. Qualitative analysis revealed that GPT-4o produced the most coherent and human-readable responses, though it scored slightly lower in quantitative metrics. This suggests that GPT-3.5-turbo is optimal for accuracy-driven applications, while GPT-4o may be preferred for tasks emphasising fluency and coherence.

Table VI details the comparative timings for video analysis, recorded in seconds, with the values in brackets representing standard deviations. Measurements were conducted over five runs and encompass the full workflow, including frame ex-

TABLE V: Report similarity results

Model	Type	Cosine Similarity	BERT Precision	BERT Recall	BERT F1
Llama 2 7B	SLM	0.740	0.843	0.880	0.862
Llama 3.1 8B	SLM	0.750	0.820	0.867	0.843
Mistral 7B	SLM	**0.874**	0.865	0.893	0.879
gpt-4o-mini	SLM	0.784	0.827	0.877	0.852
gpt-3.5-turbo	LLM	0.819	**0.870**	**0.895**	**0.883**
gpt-4-turbo	LLM	0.780	0.825	0.873	0.848
gpt-4o	LLM	0.708	0.825	0.879	0.851
YOLO-NAS					
Llama 2 7B	SLM	0.76	0.88	0.90	0.89
gpt-3.5-turbo	SLM	0.81	0.88	0.91	0.89
YOLOv11					
Llama 2 7B	SLM	0.857	0.848	0.899	0.873
Llama 3.1 8B	SLM	0.867	0.853	0.893	0.873
Mistral 7B	SLM	0.860	**0.878**	**0.911**	**0.894**
gpt-4o-mini	SLM	**0.868**	0.847	0.892	0.869
gpt-3.5-turbo	LLM	0.841	0.871	0.905	0.888
gpt-4-turbo	LLM	0.777	0.836	0.896	0.865
gpt-4o	LLM	0.797	0.838	0.898	0.867
Faster R-CNN					
Llama 2 7B	SLM	0.860	0.855	0.894	0.874
Llama 3.1 8B	SLM	0.849	0.842	0.890	0.865
Mistral 7B	SLM	**0.878**	0.868	0.910	0.888
gpt-4o-mini	SLM	0.835	0.844	0.889	0.866
gpt-3.5-turbo	LLM	0.852	**0.878**	**0.911**	**0.894**
gpt-4-turbo	LLM	0.818	0.842	0.894	0.867
gpt-4o	LLM	0.797	0.836	0.890	0.862

TABLE VI: Timings for video analysis in seconds

	YOLOv9	YOLOv11	Faster R-CNN
Llama 2 7B	49.394 (0.279)	39.924 (0.094)	39.104 (0.096)
Llama 3.1 8B	25.594 (0.180)	31.487 (0.041)	30.270 (0.053)
Mistral 7B	20.606 (0.135)	24.893 (0.112)	24.486 (0.048)
gpt-4o-mini	8.220 (0.681)	6.915 (0.726)	6.759 (0.811)
gpt-3.5-turbo	**3.434 (0.991)**	**3.408 (0.498)**	**3.245 (0.236)**
gpt-4-turbo	9.983 (0.586)	9.660 (0.642)	11.300 (2.285)
gpt-4o	7.173 (1.031)	6.135 (0.612)	7.612 (0.790)

traction, object detection, date and time extraction, and report generation. The lowest timings are highlighted in bold. The findings demonstrate that the GPT models outperform the Llama and Mistral models in report generation, with GPT-3.5-turbo offering the most efficient reports, achieving a minimum time of 3.245 seconds using Faster R-CNN. Regarding object detection, YOLOv11 achieves the quickest processing times for three of the models, while YOLOv9 and Faster R-CNN each deliver the fastest results for two models. As previously discussed in this paper, GPT-3.5 is particularly well-suited to accuracy-driven tasks, showcasing exceptional performance and also real-time capabilities. GPT-4.0 is preferable for tasks prioritising fluency and coherence and ranks among the top three models in terms of speed, further affirming its efficiency and accuracy. Using object detection models and LLMs with fewer parameters generally leads to faster real-time performance. Processing location also plays a key role. If the model runs locally, sufficient GPU power is essential to maintain responsiveness. Alternatively, leveraging cloud-based providers like GPT can offer faster results due to their robust infrastructure. However, this approach involves trade-offs between resource availability, internet connectivity, and the cost of using external services.

The comparative analysis between SLMs and LLMs revealed interesting patterns. While LLMs like GPT-3.5-turbo demonstrated better contextual understanding and the ability to generate nuanced, detailed reports, SLMs such as Mistral-7B provided competitive and consistent results, making them a cost-effective alternative for simpler tasks. The use of advanced prompting techniques, such as CoT prompting, further enhanced LLM performance in complex reasoning tasks. To optimise warehouse incident reporting, YOLOv11 should be adopted as the primary model for event detection. Its superior metrics, particularly in precision and mAP scores, make it well-suited for accurately identifying incidents and associated events. For report generation, GPT-3.5-turbo is recommended due to its strong alignment with human-generated reports and excellent performance across quantitative evaluations. However, in scenarios requiring greater coherence and readability, GPT-4o may be a better choice.

In resource-constrained settings, SLMs like Mistral-7B can be leveraged as an efficient and effective alternative, particularly for less complex tasks. Incorporating prompt engineering techniques, such as CoT prompting, is also recommended to enhance the logical reasoning and contextual depth of language models. This approach ensures that the models not only generate accurate reports but also provide actionable insights for safety improvements. The primary difference between SLMs and LLMs lies in their parameter count and associated capabilities. LLMs, with billions of parameters, excel in capturing complex language patterns and producing nuanced, detailed outputs, making them ideal for dynamic and context-heavy tasks. In contrast, SLMs are computationally lighter and offer consistent performance for simpler applications, ensuring efficiency without significant compromise in accuracy. While LLMs like GPT-3.5-turbo and GPT-4o shine in tasks requiring adaptability and depth, SLMs such as Mistral-7B demonstrate their value in cost-effective and streamlined operations.

In general, the integration of YOLOv11 for object detection and GPT-3.5-turbo or GPT-4o for report generation offers a powerful framework for automating warehouse safety reporting. These technologies can significantly enhance safety protocols by providing detailed, accurate, and timely insights into incidents, ultimately improving operational efficiency and reducing risks. Llama and Mistral produce consistent results, with a standard deviation of 0.000, whereas the GPT models all vary with a maximum standard deviation of 0.023. YOLOv11 performs better than YOLOv9 and Faster R-CNN. Mistral shows superior performance for SLMs, whereas GPT 3.5 shows superior performance for LLMs. Although the overall best performance is the Mistral SLM, indicating that SLMs are sufficient for this task. Qualitative analysis of the outputs reveals that GPT-4o generates the most coherent and accurate responses. The GPT-4-turbo model, while providing additional details such as the forklift being taken out of operational duty for a mechanical check, still performed well. Although GPT-3.5-turbo showed superior performance based on the BERT score, and GPT-4-turbo excelled in cosine similarity, human evaluation indicates that GPT-4o's outputs are the most fluent

and precise.

VI. Conclusion and Future Work

This paper presents a novel system that leverages computer vision and LLM to enhance health and safety incident reporting in warehouse environments. By integrating YOLOv9, YOLOv11 and Faster R-CNN for event detection and advanced LLMs for report generation, the system provides a comprehensive and automated method for capturing incidents and their contextual details. Evaluation results highlight its effectiveness, with YOLOv11 achieving a mAP50 of 0.972 and GPT-based models demonstrating high alignment with human-readable reports. This innovation addresses gaps in traditional methods by enabling real-time, detailed, and actionable insights, contributing to improved safety practices and operational efficiency.

Future directions include expanding the dataset to encompass diverse warehouse settings and incident types, ensuring the broader applicability of the model. Additional enhancements could involve integrating multimodal data, such as audio and sensor readings, to improve incident detection accuracy. Moreover, developing real-time alert systems and incorporating predictive analytics may further assist in preventing potential hazards. Collaboration with industry stakeholders for real-world deployment and iterative feedback will also be prioritised to refine and validate the system's practical utility.

Acknowledgment

This research was supported by funding from Innovate UK BridgeAI under Grant Number [10122554]. The authors would like to thank Innovate UK for their support and the opportunity to conduct this study in collaboration with Conway Packaging Services Ltd.

References

[1] N. Boysen, R. de Koster, and F. Weidinger, "Warehousing in the e-commerce era: A survey," *European Journal of Operational Research*, vol. 277, no. 2, pp. 396–411, Sep. 2019.

[2] I. Polanowski, "Transportation and storage statistics in Great Britain, 2023," *Health and Safety Executive*, 2023.

[3] National Safety Council, "Most dangerous industries," 2022. [Online]. Available: https://injuryfacts.nsc.org/work/industry-incidence-rates/most-dangerous-industries/

[4] Z. Tan, A. Beigi, S. Wang, R. Guo, A. Bhattacharjee, B. Jiang, M. Karami, J. Li, L. Cheng, and H. Liu, "Large language models for data annotation: A survey," *arXiv preprint arXiv:2402.13446*, 2024.

[5] A. Battistoni, "4 A Green New Deal for Care: Revaluing the Work of Social and Ecological Reproduction," in *The Green New Deal and the Future of Work*, C. Calhoun and B. Fong, Eds. Columbia University Press, Aug. 2022, pp. 103–141.

[6] S. Hayward, K. van Lopik, and A. West, "A holistic approach to health and safety monitoring: Framework and technology perspective," *Internet of Things*, vol. 20, p. 100606, Nov. 2022.

[7] A. Daios, A. Xanthopoulos, D. Folinas, and I. Kostavelis, "Towards automating stocktaking in warehouses: Challenges, trends, and reliable approaches," *Procedia Computer Science*, vol. 232, pp. 1437–1445, Jan. 2024.

[8] R. Bhana, H. Mahmoud, and M. Idrissi, "Smart Industrial Safety using Computer Vision," in *2023 28th International Conference on Automation and Computing (ICAC)*, Aug. 2023, pp. 1–6.

[9] M. Sudarshan, S. Shih, E. Yee, A. Yang, J. Zou, C. Chen, Q. Zhou, L. Chen, C. Singhal, and G. Shih, "Agentic llm workflows for generating patient-friendly medical reports," *arXiv preprint arXiv:2408.01112*, 2024.

[10] H. Pu, X. Yang, J. Li, and R. Guo, "AutoRepo: A general framework for multimodal LLM-based automated construction reporting," *Expert Systems with Applications*, vol. 255, p. 124601, Dec. 2024.

[11] "SafetyCulture | Improve Workplace Operations," https://safetyculture.com/.

[12] "SafetyIQ Workforce Safety Management Platform," https://safetyiq.com/.

[13] "Optima WS," https://www.optima-ws.co.uk/.

[14] "Protex AI - Proactive Safety Powered by AI," https://www.protex.ai/.

[15] "KIBSI," https://kibsi.com/.

[16] N. Hofstra, B. Petkova, W. Dullaert, G. Reniers, and S. de Leeuw, "Assessing and facilitating warehouse safety," *Safety Science*, vol. 105, pp. 134–148, Jun. 2018.

[17] H.-N. Auyong, S. Zailani, and L. Surienty, "Perceived safety management practices in the logistics sector," *Work*, vol. 53, no. 4, pp. 729–735, Jan. 2016.

[18] S. Arshad, O. Akinade, S. Bello, and M. Bilal, "Computer vision and IoT research landscape for health and safety management on construction sites," *Journal of Building Engineering*, vol. 76, p. 107049, Oct. 2023.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[21] N. Rane, "YOLO and Faster R-CNN object detection for smart Industry 4.0 and Industry 5.0: Applications, challenges, and opportunities," Rochester, NY, Oct. 2023.

[22] M. Shahin, F. F. Chen, A. Hosseinzadeh, H. Khodadadi Koodiani, H. Bouzary, and A. Shahin, "Enhanced safety implementation in 5S + 1 via object detection algorithms," *The International Journal of Advanced Manufacturing Technology*, vol. 125, no. 7, pp. 3701–3721, Apr. 2023.

[23] M. M. Alateeq, F. R. P. P., and M. A. S. Ali, "Construction Site Hazards Identification Using Deep Learning and Computer Vision," *Sustainability*, vol. 15, no. 3, p. 2358, Jan. 2023.

[24] e. a. OpenAI, "GPT-4 Technical Report," Mar. 2024.

[25] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.

[26] G. Krishnamoorthy, M. V. Kurkute, and J. Sreeram, "Integrating LLMs into AI-Driven Supply Chains: Best Practices for Training, Development, and Deployment in the Retail and Manufacturing Industries," *Journal of Artificial Intelligence Research and Applications*, vol. 4, no. 1, pp. 592–627, Mar. 2024.

[27] J. Miao, C. Thongprayoon, S. Suppadungsuk, P. Krisanapan, Y. Radhakrishnan, and W. Cheungpasitporn, "Chain of Thought Utilization in Large Language Models and Application in Nephrology," *Medicina*, vol. 60, no. 1, p. 148, Jan. 2024.

[28] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. V. Le, and D. Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, Dec. 2022.

[29] jaidedai, "Jaided AI: EasyOCR demo," https://www.jaided.ai/easyocr/.

[30] D. Vedhaviyassh, R. Sudhan, G. Saranya, M. Safa, and D. Arun, "Comparative Analysis of EasyOCR and TesseractOCR for Automatic License Plate Recognition using Deep Learning Algorithm," in *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, Dec. 2022, pp. 966–971.

[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[32] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024.