buildings

MDPI

Article

Ranking-Supervised Adaptation of Vision-Language Models for Temporal Progress Ordering Toward Automated Construction Monitoring

Gilsu Jeong [1], Kyung-In Kang [2] and Hyunsoo Kim [2,*]

[1] Department of Architectural Engineering, Gachon University, 1342 Seongnam-daero, Sujeong-gu, Seongnam-si 13120, Gyeonggi-do, Republic of Korea; gsjeong@gachon.ac.kr
[2] Department of Architectural Engineering, Dankook University, 152 Jukjeon-ro, Suji-gu, Yongin-si 16890, Gyeonggi-do, Republic of Korea; kikang@dankook.ac.kr
* Correspondence: hkim13@dankook.ac.kr

Abstract

Automated interpretation of construction images and videos is important for scalable site monitoring. However, construction-oriented vision-language model (VLM) studies have largely focused on frame-level interpretation, leaving temporal reasoning over evolving construction states insufficiently established. To address this gap, this study investigated temporal progress ordering as an explicit temporal-ordinal capability and proposed ConRank, a ranking-supervised VLM framework. ConRank learned pairwise earlier–later relationships and global chronological ordering from construction videos and evaluated whether the acquired ordinal knowledge transferred to progress description generation. Results showed that ranking supervision improved both local and sequence-level ordering, while the interaction between pairwise and global supervision revealed both transferable and competing learning signals. The ranking-adapted model also produced more temporally grounded progress descriptions. By formulating construction progress as a rankable visual process and linking local ordering, global sequence consistency, and open-ended generation, this study established temporal progress ordering as an intermediate capability connecting visual recognition with higher-level automated construction monitoring.

Keywords: construction monitoring; vision-language model; construction progress; ranking supervision; video understanding; progress description generation

Check for updates

Academic Editor: Pramen P. Shrestha

Received: 21 September 2026
Revised: 15 October 2026
Accepted: 24 October 2026
Published: 27 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

As construction projects continue to increase in scale and complexity, effective project management increasingly depends on continuous and reliable monitoring of on-site conditions, activities, and progress [1–3]. Progress monitoring provides the information needed to understand actual site conditions, identify deviations from planned work, support schedule control, and make timely managerial decisions [4,5]. In parallel, visual sensing technologies such as fixed surveillance cameras, unmanned aerial vehicles (UAVs), mobile cameras, and wearable imaging platforms have enabled construction sites to be recorded at increasing spatial and temporal coverage [6,7]. However, as the volume of visual data grows, the principal bottleneck shifts from data collection to interpretation; manual review of construction images and video is labor-intensive, time-consuming, and difficult to scale [8,9]. Automated methods that can transform accumulated visual data into

actionable site information are therefore becoming an important component of construction monitoring and management [10–12].

Motivated by this need, substantial research has applied computer vision and deep learning to the automated interpretation of construction images and videos [12,13]. Existing methods can detect workers, equipment, materials, and safety-relevant objects; track assets under occlusion; recognize construction activities and work states; and support downstream applications such as safety monitoring, productivity analysis, quality assessment, and progress evaluation [14–20]. Recent studies have further addressed practical construction-site challenges such as occlusion, far-field surveillance, low-quality fixed-camera footage, limited training data, and the need to infer spatial or operational states rather than simple object presence [17,20]. Collectively, these efforts have improved the ability of automated systems to identify what is present and what is happening in construction scenes. However, construction video contains more than a collection of isolated visual states. Activities unfold, structural conditions change, equipment moves, and work sequences develop over time. As illustrated in Figure 1, construction video contains a sequence of visually evolving states rather than a collection of independent observations. Effective construction video understanding therefore requires models not only to recognize scene content at individual moments but also to capture how those observations are related across time. Such temporal and ordinal understanding provides a common basis for downstream tasks including monitoring, reporting, safety assessment, productivity analysis, and progress evaluation.



Figure 1. Examples of visual state progression across construction projects, showing sequential images from earlier to later construction states.

Recent advances in vision-language models (VLMs) and multimodal large language models (MLLMs) offer a promising means of extending construction video analysis beyond fixed-label recognition [10,21,22]. Unlike conventional task-specific vision models, these models can jointly interpret visual inputs, reason over observed content using natural language, and generate free-form descriptions, reports, or responses. These capabilities are particularly attractive for construction monitoring and reporting because visual observations can be translated directly into human-readable site information for managerial review and decision support [21,23,24]. However, useful descriptions require more than the recognition of visible objects and activities. When interpreting multiple observations from a construction process, a model must also understand how the states evolve and in which order they occur. Failure to capture this structure can lead to temporally inconsistent descriptions, incorrect

interpretations of work status, or narratives that compress distinct construction stages into an ambiguous sequence. The reliability of VLM-based construction monitoring therefore depends not only on what a model recognizes in individual frames, but also on whether it can preserve and reason about the temporal relationships among them.

Temporal ordering has long been recognized as an important source of information for video understanding. Prior studies have used chronological order verification and forward–backward temporal direction as supervisory signals for learning video representations [25,26]. More recently, video-language models have been shown to struggle with even relatively simple before–after temporal relations, highlighting persistent limitations in temporal reasoning [27]. In parallel, research on visual rankability has examined whether visual representations preserve ordinal relationships that allow observations to be arranged according to an underlying progression [28–30]. This perspective is particularly relevant to construction video, where sampled images represent successive states of an evolving construction process. In this study, construction progress ordering refers to the visual chronology of observed construction states rather than construction-specific semantic progress assessment, such as identifying WBS-defined work stages, estimating physical completion percentages, or evaluating schedule adherence. It is operationalized at two complementary levels. At the pairwise level, a model determines which of two construction states represents an earlier or later stage. At the global level, it reconstructs a chronologically consistent sequence from multiple observations. This formulation enables temporal understanding to be examined through both local state discrimination and sequence-level consistency.

Despite its relevance, progress ordering has received limited attention in construction vision research. Existing studies predominantly evaluate whether a model can recognize an object, activity, work state, or predefined progress stage [4,14,16,17,20], whereas the relative ordering among visually observed construction states has not been systematically investigated. Consequently, although reliable construction video understanding requires models to capture temporal and ordinal relationships among evolving site states, this capability has not been sufficiently established for construction video. It therefore remains unclear how effectively current VLMs can distinguish pairwise progress differences and reconstruct globally consistent construction sequences, whether ranking-specific supervision can improve this capability, and whether the resulting temporal knowledge can support more reliable downstream progress descriptions.

To address these challenges, this study investigates construction progress ordering as a temporal-ordinal learning problem and proposes ConRank (Construction Progress Ranking), a ranking-supervised vision-language framework for construction video understanding. The framework is designed around three components. First, construction progress ordering is represented through two complementary tasks: pairwise progress ordering, which learns earlier–later discrimination between two construction states, and global progress ordering, which reconstructs a chronologically consistent sequence from multiple shuffled frames. Second, the two tasks are jointly used to adapt a VLM through a Margin-Temporal Ranking Loss (MTRL), which supplements standard language-generation supervision with margin-based separation and temporal representation constraints to regularize joint pairwise and global adaptation. Third, the ranking-adapted model is extended to construction progress description through Rank Self-Verification (RSV), an inference-time procedure that explicitly elicits the learned temporal order before generating the final description. The framework is evaluated on construction-site time-lapse videos through zero-shot VLM comparisons, ranking-adaptation experiments, component ablations, and downstream description generation.

The contribution of this study is therefore not limited to improving chronological prediction accuracy. It establishes construction progress ordering as a distinct capability for evaluating temporal understanding in VLM-based monitoring, examines the relationship between local pairwise discrimination and global sequence consistency, and investigates whether ranking supervision can serve as transferable knowledge for downstream construction interpretation. In doing so, this study moves construction VLM research beyond frame-level recognition toward models that reason about how observed site states evolve over time. The resulting capability provides a foundation for temporally grounded progress descriptions and, more broadly, for future monitoring systems that connect visual observations with higher-level project information such as work stages, schedules, and planned progress.

2. Research Background and Related Work

2.1. Vision-Based Construction Image and Video Monitoring

Vision-based construction monitoring has been developed for a broad range of management objectives, including safety, productivity, quality, resource tracking, and progress assessment [31,32]. Image-based approaches have been used to detect workers, heavy equipment, personal protective equipment, structural components, and other site objects; analyze worker–equipment proximity and near-miss conditions; assess construction quality and progress through BIM or geometric information; and recover spatial or state information for safety-related reasoning [12,33–35]. These applications demonstrate that construction vision has progressively moved beyond simple object presence toward the interpretation of relationships, spatial conditions, and operational states that are directly relevant to site management. Video-based approaches further extend this capability by exploiting motion and temporal context. Construction videos have been used to track equipment, recognize worker and equipment activities, analyze work cycles and productivity, identify abnormal operations, and interpret concurrent site activities using sequential models, recurrent architectures, three-dimensional convolutional networks, self-supervised learning, and multimodal sensing [36–41]. Accordingly, construction video monitoring has evolved from frame-level detection toward dynamic interpretation of what is occurring on site over time. Despite this progress, temporal information in conventional construction vision is generally introduced to improve a predefined downstream task, such as tracking an object, classifying an activity, detecting a hazard, or estimating progress. The outputs therefore remain largely task-specific labels, trajectories, states, or numerical indicators. Even when multiple frames are analyzed, temporal information is typically treated as contextual evidence for recognition rather than as an explicit learning target that represents the relative progression among construction states [36,38,40,42]. This distinction becomes important when the objective shifts from recognizing a predefined event to understanding how a construction scene evolves as a process, which requires both the interpretation of individual observations and the relationships among them across time.

2.2. Vision-Language Models for Construction Understanding and Monitoring

Vision-language models and multimodal large language models have broadened construction visual analysis from predefined recognition outputs toward language-based scene interpretation. Construction-oriented studies have used image captioning, visual question answering, and contrastive image-language models to describe site activities, assess safety compliance, and generate textual observations from construction images [43–46]. More recent VLM and MLLM applications combine visual inputs with domain knowledge, ontologies, prompt engineering, retrieval-augmented generation, and task-specific adaptation to support open-ended monitoring and safety reasoning [21,47–50]. In parallel,

language models have also been applied to accident-case analysis and safety-guideline generation, reflecting a broader shift toward natural-language reasoning in construction management [51,52]. A closely related direction uses construction images and videos for automated documentation and reporting. Vision-language and multimodal language models have been applied to daily progress and activity captioning, automated inspection reporting, hazard reporting, drone-based progress reporting, and safety documentation from site photographs [24,53–56]. Construction video has also been combined with computer-vision-based activity and productivity analysis and language generation to produce automated daily reports [9]. These studies demonstrate a transition from extracting isolated visual indicators toward converting visual site data into human-readable records that can directly support project personnel. However, reliable descriptions and reports require more than plausible language generation; they also depend on correct interpretation of how visual states evolve over time. Existing reporting-oriented approaches generally obtain temporal structure from predefined vision modules, aggregated indicators, or independently analyzed images rather than explicitly learning the relative order among multiple construction observations [9,23,24,54,55]. Consequently, construction-oriented VLMs have not been sufficiently examined in terms of whether they preserve temporal and ordinal relationships across multiple observations and use that knowledge consistently in process-level descriptions.

2.3. Temporal Ordering and Rankability in Video Understanding

Temporal ordering has been widely used as a supervisory signal in general-domain video representation learning. Prior studies learned representations by verifying frame order, identifying incorrectly ordered subsequences, or distinguishing forward from backward temporal direction [25,26,57]. Temporal Cycle-Consistency Learning further aligned corresponding phases across videos to learn temporally structured representations [58]. In multimodal pretraining, HERO introduced Frame Order Modeling to recover the original order of shuffled video frames, and the learned representations were evaluated across several video-language tasks, including video captioning [59]. These studies show that temporal relationships can provide supervision beyond categorical action labels, although recent evaluations indicate that video-language models still struggle with basic before–after relations [27]. A related line of research examines ordinal structure and rankability in visual representations. Relative-attribute learning showed that images can be compared according to semantic attributes rather than only assigned independent labels [28]. More recent studies have adapted CLIP through explicit learning-to-rank supervision [29] and examined whether modern visual embeddings preserve globally recoverable ordinal axes [30]. Temporal or ranking-aware mechanisms have also been incorporated into video captioning: ranked temporal attention has been used to capture action dynamics [60], while temporal ordering and coherence have been introduced to improve dense video captioning from unlabeled videos [61]. A more recent study directly targeted temporally fine-grained frame captions that describe subtle visual progression throughout action sequences, demonstrating the relevance of progression-sensitive visual understanding to language generation [62]. Although previous studies have demonstrated the value of temporal ordering and visual rankability for video understanding, the relationship between pairwise discrimination and global sequence ordering remains insufficiently explored. Furthermore, the transfer of independently learned ranking knowledge to downstream construction progress descriptions has received limited attention.

2.4. Research Gap and Study Positioning

Based on the preceding literature review, three research gaps can be identified in construction video understanding. First, existing construction monitoring and reporting studies have primarily focused on recognizing objects, activities, and predefined progress states or generating descriptions from visual observations, while explicit evaluation of temporal ordering among construction states remains limited. Second, although temporal order learning and visual ranking have been investigated in general-domain research, the interaction between pairwise and global ordering supervision during joint VLM adaptation has not been sufficiently examined in construction videos. Third, existing temporal captioning approaches have primarily incorporated temporal information into language generation rather than investigating whether independently learned ranking knowledge can transfer to construction progress descriptions.

To address these gaps, this study proposes ConRank, which formulates construction progress ordering through pairwise and global ranking tasks, integrates both forms of supervision using MTRL, and evaluates the transfer of learned temporal knowledge to progress description generation through RSV. By connecting explicit temporal ordering with downstream language generation, this study investigates temporal-ordinal understanding as an intermediate capability for construction progress monitoring.

3. Methodology

3.1. Overall Framework

Construction progress is represented by successive visual states whose relationships are inherently ordinal: one state precedes another, and multiple states together form a temporal sequence. ConRank is designed around this structure by treating construction progress ordering as an intermediate capability for VLM-based construction video understanding. Rather than relying directly on free-form description generation, the framework first adapts a VLM to distinguish earlier and later construction states and to reconstruct globally consistent progress sequences. The resulting temporal-ordinal knowledge is then reused to support downstream construction progress description.

As shown in Figure 2, ConRank consists of two stages: (1) ranking-supervised adaptation and (2) rank self-verification with progress description generation. In Stage 1, pairwise and global progress ordering are formulated as complementary training tasks for a single VLM backbone. Pairwise ordering emphasizes local earlier–later discrimination between two construction states, whereas global ordering requires sequence-level consistency across multiple states. The two task formats are mixed during training and update the same trainable adapter parameters, allowing both forms of supervision to shape a shared temporal representation. Their joint optimization is further coordinated through the Margin-Temporal Ranking Loss (MTRL), which is detailed in Section 3.2. The resulting ranking-adapted backbone is transferred to Stage 2 without additional parameter updates. Given uniformly sampled chronological frames, the ranking-adapted VLM first performs Rank Self-Verification (RSV) to explicitly verify their temporal order and produce a textual rank context. The same visual observations, together with the verified rank context, are then used for rank-conditioned construction progress description generation. This two-stage design separates the acquisition of temporal-ordering capability from its downstream use, enabling the effects of ranking adaptation and inference-time rank verification to be examined independently.

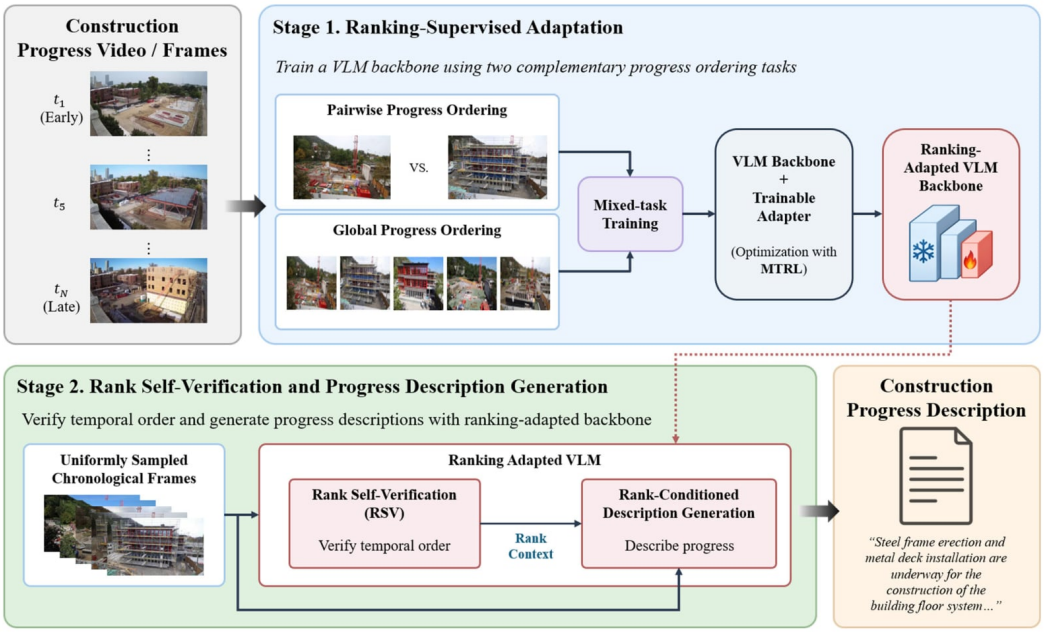


Figure 2. Overall framework of ConRank, consisting of ranking-supervised adaptation and rank-conditioned construction progress description generation.

3.2. Stage 1: Ranking-Supervised Adaptation

Ranking-supervised adaptation stage is designed to equip a single VLM backbone with temporal-ordinal understanding before the model is used for downstream description generation. Its input is a mixed set of ranking instances constructed from frames sampled within the same construction video, where each instance follows either a pairwise or global ordering format and is paired with the corresponding instruction and target response. The output is a ranking-adapted VLM backbone whose shared trainable adapter has learned both local earlier–later discrimination and sequence-level chronological consistency. As illustrated in Figure 3, this stage comprises three components: pairwise progress ordering, global progress ordering, and joint ranking adaptation with the proposed Margin-Temporal Ranking Loss (MTRL).

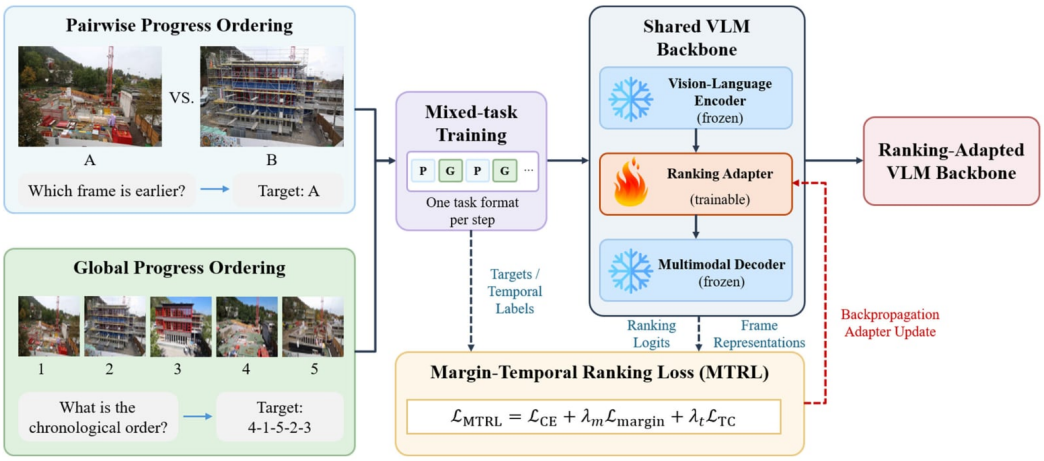


Figure 3. Ranking-supervised adaptation in ConRank, where pairwise and global progress-ordering tasks jointly train a shared VLM backbone using MTRL.

3.2.1. Pairwise Progress Ordering

Pairwise progress ordering provides the most local form of ranking supervision by teaching the model to distinguish the relative progress relationship between two construc-

tion states. This formulation is particularly important for visually similar states, where progress differences may be expressed through limited structural change while background conditions, equipment, temporary objects, or other incidental visual factors remain similar. For each training instance, two frames from the same construction video are assigned randomly to display positions A and B, and the model is instructed to identify which displayed frame represents the earlier construction state. Random assignment prevents the presentation order itself from revealing the correct answer. Let t_A and t_B denote the original temporal positions of the frames displayed as A and B, respectively. The ground-truth pairwise label is defined as follows.

$$y_{AB} = \begin{cases} A, & t_A < t_B \\ B, & t_B < t_A \end{cases} \quad (1)$$

The input therefore consists of two displayed frames and a pairwise-ordering instruction, while the target is the display label of the earlier frame. Because A and B are presentation labels rather than temporal indices, the task requires the model to infer relative construction progress from visual content rather than input position. Pairwise ordering consequently converts temporal progression into a binary earlier–later decision and provides direct supervision for local state discrimination.

3.2.2. Global Progress Ordering

Pairwise discrimination alone does not guarantee that multiple construction states can be organized into a globally consistent progression. Global progress ordering therefore extends the ranking task from two-frame comparison to sequence-level chronological reconstruction. A set of K frames sampled from the same construction video is randomly shuffled and presented simultaneously, and the model is instructed to recover the correct order from the earliest to the latest construction state. The input is thus a shuffled multi-frame set with a global-ordering instruction, and the target is a permutation of the displayed frame positions. Let σ denote the random permutation used to shuffle the sampled frames and let $t_{\sigma(k)}$ denote the original temporal position of the frame shown at display position k. The ground-truth chronological permutation is defined as

$$\pi = \text{argsort}\left(t_{\sigma(1)}, t_{\sigma(2)}, \ldots, t_{\sigma(k)}\right) \quad (2)$$

Here, π returns the display-position indices in ascending temporal order. For example, $\pi = (3, 1, 5, 2, 4)$ indicates that the frame displayed in position 3 is the earliest, followed by positions 1, 5, 2, and 4. In contrast to the local binary decision of pairwise ordering, the global task requires one consistent progression across all presented observations and therefore supplies sequence-level ranking supervision.

3.2.3. Joint Ranking Adaptation with MTRL

The two ordering tasks capture different temporal granularities and are therefore combined within a single ranking-adaptation process rather than assigned to separate model branches. Pairwise and global instances are merged into one mixed training set. At each optimization step, the VLM receives one task format with its corresponding instruction and target, while the same trainable adapter parameters are updated regardless of whether the instance is pairwise or global. This interleaved training scheme allows local discrimination and global sequence ordering to shape a shared ranking-adapted backbone.

Standard sequence-level cross-entropy supervises generation of the correct ranking response, but output-token supervision alone does not explicitly enforce separation between competing pairwise decisions or constrain the frame-level representations used for temporal

reasoning. MTRL therefore augments the cross-entropy objective with a margin-ranking term and a temporary contrastive term:

$$\mathcal{L}_{\text{MTRL}} = \mathcal{L}_{\text{CE}} + \lambda_m \mathcal{L}_{\text{margin}} + \lambda_t \mathcal{L}_{\text{TC}} \tag{3}$$

where $\mathcal{L}_{\text{CE}}$ is the sequence-level cross-entropy loss, $\mathcal{L}_{\text{margin}}$ strengthens separation between the correct and incorrect pairwise answer logits, and $\mathcal{L}_{\text{TC}}$ encourages discriminative separation between temporally distinct frame-level hidden representations. The weighting coefficients are set to $\lambda_m = 0.5$ and $\lambda_t = 0.1$. For pairwise instances, the margin-ranking term is defined as:

$$\mathcal{L}_{\text{margin}} = \max\left(0, m - \left(z_{\text{correct}} - z_{\text{wrong}}\right)\right) \tag{4}$$

where z_{correct} and z_{wrong} are the logits of the correct and incorrect answer tokens, respectively, and the margin is set to $m = 1.0$. The term penalizes predictions when the correct-answer logit does not exceed the incorrect-answer logit by at least the specified margin. Because the global task produces a multi-position permutation rather than a binary answer, $\mathcal{L}_{\text{margin}} = 0$ for global-ordering instances. The frame-level representations used in the temporal-contrastive loss are obtained from the final hidden layer of the VLM. For each input frame, the hidden states corresponding to its visual tokens are mean-pooled and subsequently L2-normalized to obtain a single frame-level representation. The temporal-contrastive loss is then computed using these representations for the two presented frames in pairwise instances or chronologically adjacent frame pairs in global instances. The temporal-contrastive term discourages excessive similarity between temporally distinct frame representations and is defined as:

$$\mathcal{L}_{\text{TC}}(i, j) = \max\left(0, \text{sim}\left(\mathbf{h}_i, \mathbf{h}_j\right) - \tau\right) \tag{5}$$

where $\mathbf{h}_i$ and $\mathbf{h}_j$ are L2-normalized frame-level hidden representations and $\tau = 0.5$ is the similarity threshold. For a pairwise instance, the loss is computed directly from the two presented frames. For a global instance, it is computed over ground-truth chronologically adjacent frame pairs and averaged across the sequence. The same threshold is applied regardless of temporal gap. This term is intended to improve state separability rather than explicitly encode temporal distance. During training, the pretrained VLM parameters remain frozen, and only the trainable LoRA adapter parameters are updated. The cross-entropy loss propagates gradients through the output logits for both task formats, while the margin-ranking loss additionally contributes gradients through the correct and incorrect answer logits for pairwise instances. The temporal-contrastive loss propagates gradients through the frame-level hidden representations. All applicable loss terms contribute to optimizing the same LoRA adapter parameters rather than separate model components. Through this joint objective, Stage 1 produces a single ranking-adapted VLM backbone that encodes both pairwise and global ordering knowledge and is subsequently used in Stage 2 without additional training.

3.3. Stage 2: Rank Self-Verification and Progress Description Generation

Stage 2 is designed to reuse the temporal-ordering capability acquired during ranking-supervised adaptation when generating a free-form construction progress description. The input is a sequence of N frames uniformly sampled from a construction video and arranged in chronological order, and the output is a natural-language description of the observed construction progression. The ranking-adapted VLM obtained from Stage 1 is used throughout this stage without any additional fine-tuning or parameter update. As shown in Figure 2, Stage 2 consists of two consecutive inference steps: Rank Self-Verification (RSV), which makes the temporal order explicit as textual rank context, and rank-conditioned progress description generation, which combines that context with the same visual observations to produce the final description.

3.3.1. Rank Self-Verification

Rank self-verification (RSV) is an inference-time prompting step intended to elicit an explicit temporal-ordering response before progress description generation. Direct description generation can rely primarily on salient objects or activities in individual frames, whereas RSV requires the ranking-adapted VLM to first attend to the relative progression among the presented construction states. The procedure therefore introduces an intermediate reasoning step that externalizes temporal-order information before it is used for description generation. For each video, N frames are uniformly sampled and supplied to the ranking-adapted VLM in their original chronological order together with an order-verification instruction. The frames are not shuffled or reordered during RSV. Thus, RSV does not independently verify or correct the chronological arrangement of the input frames. Rather, it provides a model-generated ordering statement as additional context for subsequent description generation. Instead, the model is asked to explicitly state the inferred temporal order in a global-ordering style response, such as 1, 2, 3, …, N. The generated order is retained as textual rank context for the next step. Thus, RSV takes the chronologically ordered frame sequence and an order-verification instruction as input and produces an explicit model-generated ordering statement as output, while leaving both the visual input and model parameters unchanged.

3.3.2. Rank-Conditioned Progress Description Generation

The second step uses the temporal-order information made explicit by RSV to guide construction progress description generation. The same (N) chronologically ordered frames used during RSV are presented again to the same ranking-adapted VLM, together with a description-generation instruction and the RSV-derived rank context. Keeping the visual input unchanged isolates the contribution of the explicit rank context from changes in frame selection or ordering. The model then generates a natural-language description of the construction progression conditioned jointly on the visual observations and the RSV-generated rank context. No separate pairwise or global ranking task is executed at this stage because the effects of those tasks are already embedded in the ranking-adapted backbone from Stage 1. No additional optimization is performed. The input to this step therefore comprises the chronological frame sequence, the description instruction, and the RSV-generated rank context, while the output is the final construction progress description. This design provides a direct mechanism for examining whether temporal-order knowledge learned through ranking supervision can be reused during downstream language generation.

4. Experiment Settings

4.1. Dataset

The dataset was constructed to support three complementary evaluations: pairwise progress ordering, global progress ordering, and downstream construction progress description generation. Construction-site time-lapse videos were collected as long-term visual records of project evolution, representative progress states were extracted for ranking supervision, and a separate chronologically sampled frame set was prepared for description evaluation.

4.1.1. Video Collection and Progress-Sensitive Frame Sampling

This study collected a total of 50 construction-site time-lapse videos from YouTube. Each video records the progression of a construction project over an extended period, and the collection covers major stages ranging from early earthwork to project completion. These videos provide chronologically ordered visual records in which structural components,

temporary facilities, equipment, materials, and overall site configuration change as construction advances. As the videos were obtained from publicly available sources, detailed information on the original recording equipment, camera installation, and acquisition settings could not be consistently verified. Although the collected videos represent different construction projects with varying viewpoints and recording conditions, it was not possible to confirm whether cameras remained permanently installed throughout the recording periods. Therefore, this study focused on a standardized frame-sampling and selection procedure applied to the collected videos, rather than controlling their original recording conditions. Although the collected videos exhibit variations in lighting and weather conditions, detailed environmental metadata, including sunlight, cloud cover, precipitation, and recording time, were not consistently available. As these conditions were not separately annotated, their individual effects on model performance could not be quantitatively evaluated. To construct a compact but progress-sensitive frame pool, 200 frames were first sampled uniformly from each video. Changes between consecutive sampled frames were measured using SSIM on the HSV hue and saturation channels, SSIM on Sobel edge maps, and perceptual hash (pHash) distance after CLAHE normalization. Frame transitions with an HSV-based SSIM below 0.6 or a pHash distance of at least 15 were selected as change points, and the candidate regions were expanded by ± 2 sampled frames. When more than 30 candidate frames were obtained, they were uniformly downsampled; when fewer than 30 were available, additional frames were uniformly sampled from the video to obtain exactly 30 representative frames per video.

4.1.2. Ranking Dataset Construction

Before ranking-instance generation, the 50 videos were randomly shuffled using a fixed seed of 42 and divided at the video level into 35 training, 5 validation, and 10 test videos, with no overlap between the subsets. This prevents frames from the same construction project from appearing across multiple subsets and reduces the risk of visual leakage between training and evaluation data. Pairwise ranking instances were generated from the 30 representative frames of each video. A total of 60 pairs were constructed per video and divided into Hard, Medium, and Easy groups according to the temporal index gap between the two selected frames. Let $\Delta = |i - j|$ denote the absolute difference between the indices of the two frames in the 30-frame representative sequence. Hard, Medium, and Easy pairs were defined as $\Delta \leq 5$, $6 \leq \Delta \leq 15$, and $\Delta > 15$, respectively. Smaller index gaps were expected to correspond to more visually similar construction states and were therefore treated as harder ordering cases, whereas larger index gaps were expected to provide clearer progress differences. Each difficulty group contained 20 pairs. Within every pair, the earlier and later frames were randomly assigned to display positions A and B with equal probability to eliminate positional cues. Figure 4 shows representative Easy and Hard examples. Global ranking instances were constructed from the same 30-frame pool. For each video, 30 groups were generated, each containing 5 frames. The frames in each group were randomly shuffled, and the target sequence was defined by their original chronological order. Duplicate frame combinations were excluded. Figure 5 illustrates the resulting global ordering task. Across the 50 videos, the final split contained 2100 pairwise and 1050 global training instances, 300 pairwise and 150 global validation instances, and 600 pairwise and 300 global test instances. Final ranking performance was reported only on the test split.

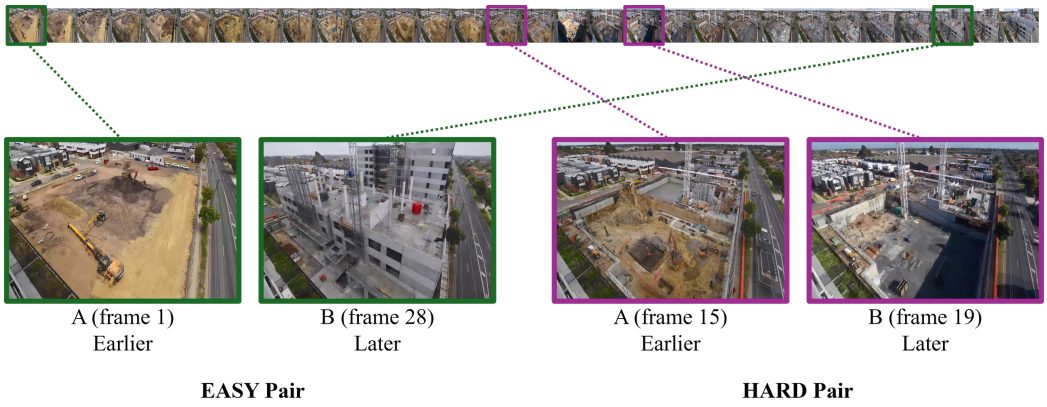


Figure 4. Example of the pairwise progress ordering task, where frame pairs are compared according to their temporal construction progress, with Hard and Easy pairs representing smaller and larger ordinal gaps within the representative-frame sequence, respectively.

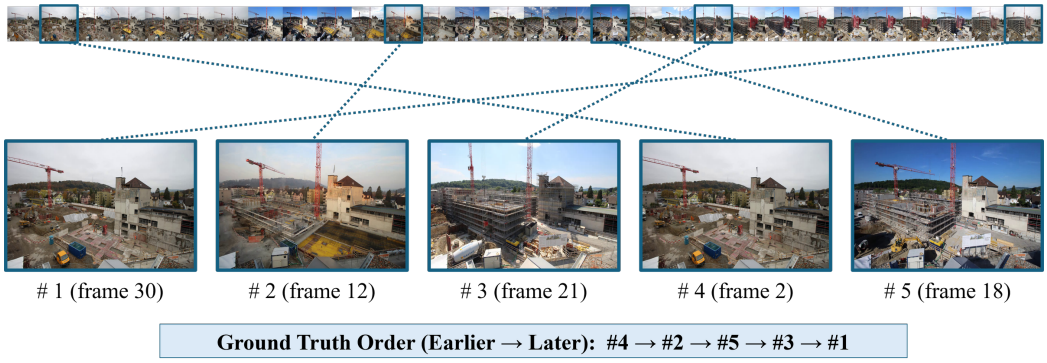


Figure 5. Example of the global progress ordering task, where shuffled construction frames are reordered according to their chronological progress.

4.1.3. Progress-Description Reference Data

In addition to the ranking datasets used in Stage 1, a separate frame set was prepared from the 10 test videos for evaluating progress-description generation in Stage 2. From each video, 100 frames were sampled uniformly at evenly spaced temporal positions from the beginning to the end of the video and retained in chronological order. These sequences were used as the common visual input for evaluating progress-description generation. To construct consistent ground-truth reference annotations without relying entirely on unconstrained manual writing, GPT-4o, a commercial multimodal API model, was used as an annotation tool to generate initial frame-level descriptions and one overall temporal progress description for each test video. The generated outputs were not adopted directly as ground truth. The reference descriptions were manually reviewed by three researchers and, where necessary, revised to remove unsupported statements, correct construction terminology, and ensure consistency with the observed temporal progression. The resulting human-validated descriptions were used as ground-truth reference annotations for evaluation. GPT-4o was used only to assist in constructing the reference descriptions and was not used to generate the descriptions evaluated in this experiment. All evaluated descriptions were generated using Qwen2.5-VL-7B-based variants to examine the effects of ranking-supervised adaptation and RSV under a consistent model backbone. The same 100 chronologically ordered frames were provided to every description-generation variant so that differences in performance reflected model adaptation and the use of rank context rather than differences in visual input.

4.2. Experimental Design

The experiments were organized to evaluate ConRank at 3 levels: local pairwise ordering, global multi-frame ordering, and transfer of the learned ordering capability to construction progress description generation.

Experiment 1 evaluates whether a model can identify the earlier construction state from 2 frames. Zero-shot baselines and Qwen2.5-VL-7B-based ranking variants are compared on the 600 test pairs, with performance additionally examined across the Hard, Medium, and Easy groups to assess sensitivity to ordinal separation within the representative-frame sequence.

Experiment 2 extends the evaluation from binary comparison to sequence-level ordering. Each model receives 5 shuffled construction frames and is required to recover their correct temporal order. The same zero-shot and ranking-adapted model configurations used in Experiment 1 are compared on the 300 test groups. This experiment examines whether local earlier–later discrimination transfers to a globally consistent sequence and whether joint ranking adaptation improves multi-frame ordering beyond task-specific training.

Experiment 3 evaluates whether the temporal knowledge learned from ranking supervision transfers to construction progress description generation. All variants receive the same 100 chronologically ordered frames from each of the 10 test videos. The Base Qwen2.5-VL-7B model generates descriptions without ranking adaptation, ConRank w/o RSV uses the ranking-adapted backbone but generates descriptions directly without the RSV step, and the full ConRank additionally incorporates RSV-derived rank context. This comparison separates the effect of ranking adaptation from the additional effect of inference-time rank self-verification.

4.3. Competing Models

The ranking experiments include both non-adapted baselines and Qwen2.5-VL-7B-based fine-tuned variants. Random prediction was included as a chance-level baseline, and CLIP was included as a conventional vision-language representation baseline. Qwen2.5-VL-7B served as the representative open-source VLM, while GPT-4o and Gemini-3.1-Pro were included as representative proprietary multimodal models. The generative VLMs were evaluated under zero-shot settings using the same test data and task instructions, without task-specific adaptation. Qwen2.5-VL-7B was also used as the backbone for all fine-tuned ranking variants because its open weights allow task-specific adaptation. Pairwise Only was trained only with pairwise supervision, whereas Global Only was trained only with global-ordering supervision. Pairwise + Global used the mixed-task training scheme without MTRL, providing a direct comparison for evaluating whether simply combining the 2 task formats is sufficient. ConRank used the same mixed pairwise and global supervision together with MTRL. For the description experiment, the comparison was restricted to Base Qwen2.5-VL-7B, ConRank w/o RSV, and the full ConRank so that the effects of ranking adaptation and RSV could be isolated sequentially.

4.4. Evaluation Metrics and Protocol

Pairwise progress ordering was evaluated on 600 frame pairs from the 10 test videos. Performance was measured using overall accuracy and separate accuracies for the Hard, Medium, and Easy groups defined in Section 4.1.2. Reporting performance by difficulty level allows the evaluation to distinguish ordering of visually similar states from cases with larger ordinal separation within the representative-frame sequence. Global progress ordering was evaluated on 300 frame groups from the same 10 test videos. Kendall's τ was used to measure agreement between the predicted and ground-truth orderings at the pairwise relation level, while Exact Match measured the proportion of sequences

for which the complete predicted permutation exactly matched the ground-truth order. Together, these metrics distinguish partially correct ordering from fully correct sequence reconstruction. Construction progress description was evaluated on the 10 test videos using the same 100-frame chronological sequence for every model variant. Description quality was assessed using BERTScore F1 for semantic similarity, CLIPScore for visual-text alignment, GPT-4-based ratings of Process Accuracy, Temporal Flow, and Fluency, and an NLI-based contradiction score. To mitigate potential evaluation bias arising from GPT-4o-assisted reference construction and GPT-based ratings, the evaluation incorporated complementary metrics, including BERTScore F1, CLIPScore, and NLI-based contradiction scores. In particular, CLIPScore assesses visual-text alignment without relying on the reference descriptions. Nevertheless, potential stylistic and evaluator biases cannot be entirely excluded, and the results were therefore interpreted across multiple evaluation dimensions rather than relying solely on GPT-based ratings.

4.5. Implementation Details

ConRank and the Qwen2.5-VL-7B-based fine-tuned variants were implemented from the Qwen/Qwen2.5-VL-7B-Instruct checkpoint. LoRA was applied to the query and value projection layers of the attention modules, with a rank of 16 and a scaling factor of 32, while the pretrained model parameters remained frozen. Training used bf16 precision without quantization, and the maximum visual input size was set to 512 × 512 pixels per image. AdamW with a cosine learning-rate schedule was used with a peak learning rate of 0.0001 and 100 warm-up steps. Training was conducted for 5 epochs with a batch size of 1, gradient accumulation of 8, and gradient clipping with a maximum norm of 1.0. The training configurations differed in the number of examples processed per epoch. Pairwise Only used 2100 pairwise instances, while Global Only used 1050 global instances. Both Pairwise + Global and ConRank used the same 3150 training instances, consisting of 2100 pairwise (66.7%) and 1050 global (33.3%) instances. Although all configurations were trained for five epochs with the same optimization settings, the different dataset sizes resulted in different numbers of optimization steps. Specifically, Pairwise Only, Global Only, Pairwise + Global, and ConRank completed 1315, 660, 1970, and 1970 optimizer steps, respectively. All experiments were conducted on a single NVIDIA H100 NVL GPU with 96 GB memory using Python 3.10.20, PyTorch 2.10.0 with CUDA 12.8, Transformers 5.5.4, and PEFT 0.18.1. GPT-4o and Gemini-3.1-Pro were accessed through their respective APIs.

For description evaluation, BERTScore F1 was computed using RoBERTa-large (17 layers, without IDF weighting or baseline rescaling). Reference-free CLIPScore used CLIP ViT-B/32, averaging cosine similarities across 10 frames with a scaling factor of 2.5 and a 77-token text limit. NLI contradiction was evaluated using facebook/bart-large-mnli by averaging sentence-level contradiction probabilities against the reference descriptions, with a 512-token input limit. GPT-4o evaluated Process Accuracy, Temporal Flow, and Fluency on a five-point scale using reference and generated descriptions without images, with one evaluation per sample and the default temperature of 1.0.

5. Experiment Results

This chapter presents the results of the three experiments defined in Section 4.2. The first experiment evaluated local progress-ordering capability through pairwise earlier–later discrimination, the second examined sequence-level temporal consistency through global multi-frame ordering, and the third investigated whether the learned ordering capability transferred to construction progress description generation. Together, these experiments assessed the effectiveness of ranking-supervised adaptation and its transfer to a downstream language-generation task.

5.1. Pairwise Progress Ordering Results

The pairwise experiment examined how reliably each model distinguished the relative progress of two construction states and how this capability changed with ordinal separation within the representative-frame sequence. Table 1 summarizes the results across the Hard, Medium, and Easy groups. Among the zero-shot VLMs, Gemini-3.1-Pro achieved the highest overall accuracy of 82.00%, followed by GPT-4o at 69.00%, whereas the base Qwen2.5-VL-7B achieved 57.83%. Performance generally increased from Hard to Easy cases, supporting the intended difficulty stratification based on ordinal separation: frame pairs with larger ordinal gaps were more readily distinguished than pairs with smaller gaps. Fine-tuning substantially improved the ordering performance of Qwen2.5-VL-7B. Pairwise Only, Global Only, and Pairwise + Global achieved overall accuracies of 78.33%, 80.67%, and 81.33%, respectively. ConRank achieved the highest overall accuracy of 84.33%, exceeding Pairwise + Global by 3.00 percentage points and Gemini-3.1-Pro zero-shot by 2.33 percentage points. The improvement produced by MTRL was most pronounced for the Hard group, where accuracy increased from 68.00% for Pairwise + Global to 72.50% for ConRank.

Table 1. Pairwise progress-ordering performance of zero-shot and fine-tuned models.

Model	Input Configuration	Setting	Hard	Medium	Easy	Overall
Random	-	Baseline	50.00	50.00	50.00	50.00
CLIP	-	Zero-shot	47.50	49.00	49.00	48.50
Qwen2.5-VL-7B	-	Zero-shot	54.50	55.50	63.50	57.83
GPT-4o	-	Zero-shot	48.00	73.00	86.00	69.00
Gemini-3.1-Pro	-	Zero-shot	**75.00**	**80.00**	**91.00**	**82.00**
Qwen2.5-VL-7B	Pairwise Only	Fine-tuned	64.50	82.00	88.50	78.33
Qwen2.5-VL-7B	Global Only	Fine-tuned	69.50	84.50	88.00	80.67
Qwen2.5-VL-7B	Pairwise + Global	Fine-tuned	68.00	85.00	91.00	81.33
ConRank (Ours)	Pairwise + Global + MTRL	Fine-tuned	**72.50**	**87.50**	**93.00**	**84.33**

Note: Bold values indicate the best results within each setting.

5.2. Global Progress Ordering Results

The global ordering experiment evaluated whether the models could extend local progress cues to a consistent sequence across multiple construction states. Table 2 reports the results for the 5-frame ordering task. Zero-shot Qwen2.5-VL-7B showed limited sequence-level ordering capability, with a Kendall's τ of 0.089 and an Exact Match of 1.67%. The proprietary models performed substantially better under zero-shot evaluation, with GPT-4o and Gemini-3.1-Pro reaching Kendall's τ values of 0.380 and 0.390, respectively. The fine-tuned variants further showed that supervision from one ordering format could affect the other. Pairwise Only, despite receiving no global-ordering supervision, improved to $\tau = 0.174$ and 9.00% Exact Match. Global Only reached $\tau = 0.373$ and 23.67% Exact Match. In contrast, directly combining pairwise and global training without MTRL resulted in $\tau = 0.304$ and 19.00% Exact Match. ConRank increased these values to $\tau = 0.412$ and 35.33%, achieving the best global-ordering performance among all evaluated configurations. Compared with Global Only, ConRank improved Kendall's τ by 0.039 and Exact Match by 11.66 percentage points, and it also exceeded the strongest zero-shot model, Gemini-3.1-Pro. Figure 6 jointly summarizes the results of Experiments 1 and 2. Figure 6a shows the pairwise accuracy trend across the Hard, Medium, and Easy groups, while Figure 6b compares pairwise overall accuracy with global Kendall's τ. The combined view shows that the task-specific variants emphasize different aspects of ordering performance, whereas ConRank attains the strongest overall balance between pairwise and global ordering.

Table 2. Global progress-ordering performance of zero-shot and fine-tuned models.

Model	Input Configuration	Setting	Kendall's τ	Exact Match
Random	-	Baseline	0.000	0.83
CLIP	-	Zero-shot	0.010	1.33
Qwen2.5-VL-7B	-	Zero-shot	0.089	1.67
GPT-4o	-	Zero-shot	0.380	26.67
Gemini-3.1-Pro	-	Zero-shot	**0.390**	**31.00**
Qwen2.5-VL-7B	Pairwise Only	Fine-tuned	0.174	9.00
Qwen2.5-VL-7B	Global Only	Fine-tuned	0.373	23.67
Qwen2.5-VL-7B	Pairwise + Global	Fine-tuned	0.304	19.00
ConRank (Ours)	Pairwise + Global + MTRL	Fine-tuned	**0.412**	**35.33**

Note: Bold values indicate the best results within each setting.

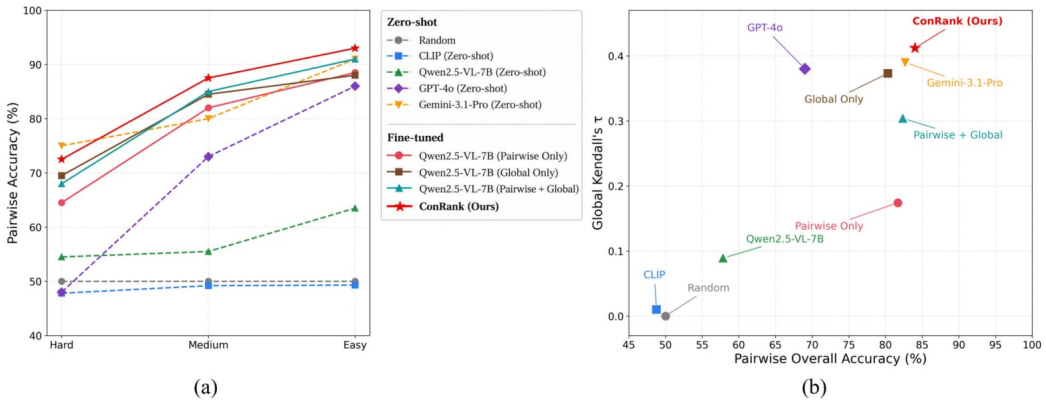


Figure 6. Ranking performance of zero-shot and fine-tuned models: (**a**) pairwise ordering accuracy across Hard, Medium, and Easy difficulty levels; (**b**) relationship between pairwise overall accuracy and global ordering performance.

5.3. Construction Progress Description Results

The description experiment examined whether the temporal-ordering capability acquired through ranking supervision contributed to a downstream generation task and whether RSV provided an additional benefit at inference time. Table 3 compares the Base model, ConRank w/o RSV, and the full ConRank. ConRank w/o RSV improved all reported metrics relative to the Base model. BERTScore F1 increased from 0.879 to 0.880 and CLIPScore from 0.608 to 0.620. The MLLM-based Process Accuracy, Temporal Flow, and Fluency scores increased from 2.80, 3.70, and 4.40 to 3.50, 4.20, and 4.70, respectively, while NLI Contradiction decreased from 0.053 to 0.031. Adding RSV further increased BERTScore F1 to 0.894, CLIPScore to 0.695, Process Accuracy to 3.90, and Fluency to 4.80, while reducing NLI Contradiction to 0.004. Temporal Flow was the only metric that did not increase further with RSV, decreasing slightly from 4.20 to 4.10. Accordingly, the full ConRank achieved the best performance on 5 of the 6 reported description metrics, whereas ConRank w/o RSV obtained the highest Temporal Flow score. The qualitative example in Figure 7 further illustrates the difference in generated descriptions. Compared with the Base model, ConRank produced a more stage-specific account of the observed construction sequence, with clearer separation of successive construction states rather than compressing multiple stages into a generic summary. This qualitative pattern is consistent with the improvements in Process Accuracy and the reduction in NLI Contradiction reported in Table 3.

Table 3. Comparison of construction progress description performance.

Model	BERT Score F1	CLIP Score	MLLM Judge			NLI Contradiction
			Process Accuracy (1–5)	Temporal Flow (1–5)	Fluency (1–5)	
Base (Qwen2.5 7B)	0.879	0.608	2.80	3.70	4.40	0.053
ConRank w/o RSV	0.880	0.620	3.50	**4.20**	4.70	0.031
ConRank (Ours)	**0.894**	**0.695**	**3.90**	4.10	**4.80**	**0.004**

Note: Bold values indicate the best performance for each metric.

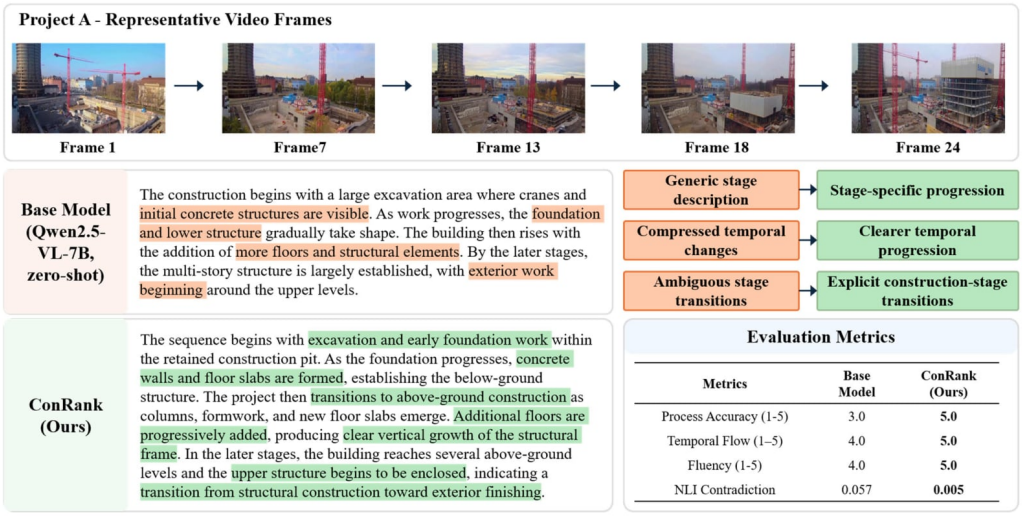


Figure 7. Representative qualitative comparison of construction progress descriptions, where highlighted phrases contrast generic or compressed Base descriptions with ConRank's clearer stage-specific temporal progression.

6. Discussion

The experimental results revealed three connected aspects of construction progress understanding with VLMs: the relationship between local and global ordering, the effect of coordinating heterogeneous ranking supervision, and the transfer of learned ordering knowledge to progress description generation. The discussion therefore focuses on how pairwise and global tasks interact, why MTRL improves joint adaptation, how ranking supervision influences downstream generation, and what these findings imply for construction monitoring.

6.1. Temporal Understanding and Cross-Task Transfer

Pairwise and global ordering capture different but related aspects of temporal construction understanding. Pairwise ordering requires the model to distinguish the relative progress of two states, whereas global ordering requires the same notion of progression to remain consistent across an entire sequence. The lower accuracy observed for Hard pairwise cases indicated that fine-grained progress differences were more difficult to distinguish when structural change was limited and incidental visual factors such as illumination, viewpoint, temporary objects, or local site activity remained prominent. In contrast, larger ordinal gaps within the representative-frame sequence provide stronger structural and process cues, which explains the general increase in performance from Hard to Easy cases. Further examination of the error distribution shows that 55 of the 94 incorrect pairwise predictions (58.5%) occurred in the Hard group, followed by 25 (26.6%) in the Medium group and 14 (14.9%) in the Easy group. However, attributing these errors to specific construction stages or scene conditions is challenging because the dataset does not include correspond-

ing annotations. Moreover, pairwise and global ordering instances may span multiple construction stages, making it difficult to associate an ordering error with a single stage.

The global task imposes an additional consistency requirement because several local relationships must be resolved simultaneously. A model may correctly judge many individuals earlier–later relations without producing a fully consistent sequence when those judgments are combined. This distinction was reflected in the relatively low Exact Match values compared with the stronger pairwise results, particularly under zero-shot evaluation. At the same time, the task-specific fine-tuning results indicate that the two formulations share transferable temporal information. Pairwise Only improved global ordering despite receiving no global-ordering supervision, while Global Only substantially improved pairwise accuracy relative to the base Qwen2.5-VL-7B model. These results suggest that local earlier–later discrimination and global sequence reconstruction are not independent capabilities. Instead, both appear to rely on a shared ordinal representation of construction progression, with each task emphasizing a different temporal granularity.

6.2. Task Interference and the Role of MTRL

Cross-task transfer does not imply that simply mixing pairwise and global supervision will improve both tasks simultaneously. Pairwise + Global training improved pairwise accuracy relative to the single-task variants but reduced global performance compared with Global Only. This pattern suggests that the two tasks provide related but non-identical optimization signals. Pairwise supervision is based on a binary decision between two states, whereas global supervision requires generation of a multi-position permutation. Their different output structures and temporal scopes can therefore introduce competing learning pressures when they are optimized only through token-level cross-entropy.

MTRL supplements the common sequence-generation objective with ranking-specific constraints during joint pairwise and global adaptation. The margin term directly strengthens separation between competing pairwise answers, while the temporal-contrastive term discourages overly similar representations for temporally distinct frame pairs in both task formats. The resulting improvement from Pairwise + Global to ConRank was particularly evident in global ordering, where Kendall's τ increased from 0.304 to 0.412 and Exact Match from 19.00% to 35.33%. Pairwise performance also improved, with the largest gain occurring in the Hard group. This pattern suggests that the additional ranking-aware constraints are most useful when visual differences are subtle or when multiple local relationships must be coordinated into one consistent ordering. These results support the role of MTRL as a ranking-aware regularizer for joint adaptation rather than simply adding another source of supervision. The experiments did not directly analyze gradient conflict or representation geometry, so the precise internal mechanism remains to be established. However, the contrast between naive mixed-task training and MTRL-guided adaptation indicates that additional ranking-aware constraints are beneficial when local and global ordering are learned within a shared backbone. Additional ablation and sensitivity analyses were conducted to examine the individual contributions of the MTRL components and the effects of their hyperparameter settings (Appendix A). The results indicate that the margin-ranking term contributes more to pairwise ordering, whereas the temporal-contrastive term provides greater improvements in global ordering. Their combination achieves the best overall performance, while the sensitivity analysis reveals relatively stable pairwise performance but greater sensitivity in global ordering, particularly to the similarity threshold. The performance differences between single-task and joint configurations should also be interpreted considering their unequal training-set sizes and optimization steps. Nevertheless, the comparison between Pairwise + Global and ConRank provides a more controlled evaluation of MTRL, as both configurations use identical training data and optimization settings.

6.3. Transfer to Construction Progress Description Generation

The description results show that the effect of ranking supervision extends beyond the explicit ordering tasks used during adaptation. ConRank w/o RSV improved the description metrics relative to the Base model even though no additional ordering task was executed during generation. Previous studies have primarily explored temporal ordering as a supervisory signal for video representation learning or focused on generating construction reports directly from visual observations. In contrast, the present results suggest that explicitly learned ranking knowledge can be transferred to construction progress description generation, even without an additional ordering step during inference. The qualitative example in Figure 7 supports the same interpretation: compared with the Base model, the ranking-adapted model separates successive construction stages more explicitly rather than compressing multiple states into a generic summary. RSV introduces an additional inference-time prompting step that makes the temporal ordering explicit before description generation. Since the input frames are already arranged chronologically, RSV does not independently validate or correct their temporal order. Therefore, the observed improvements should be interpreted as the effects of incorporating model-generated rank context rather than as evidence of independent temporal verification. The improvements in BERTScore F1, CLIPScore, Process Accuracy, Fluency, and NLI Contradiction after adding RSV suggest that this explicit rank context can help convert latent ordering knowledge into a more faithful description of observed progress. Temporal Flow decreased slightly from 4.20 to 4.10. This pattern indicates that stronger temporal conditioning does not uniformly improve every aspect of language generation. Making the sequence structure more explicit may improve stage-specific accuracy and reduce contradiction while providing little additional benefit to narrative smoothness. This trade-off is important because progress reporting requires both factual temporal grounding and readable language. The results therefore support RSV as a useful inference-time bridge between ranking and generation, while also showing that its effect should be interpreted across multiple evaluation dimensions rather than through a single metric.

6.4. Implications for Construction Monitoring

The results suggest that temporal ordering can serve as an intermediate capability between frame-level visual recognition and higher-level construction monitoring. Pairwise ordering is directly relevant to local comparison between two site observations, such as determining whether a later image reflects a more advanced visual state. Global ordering extends this capability to multiple observations, supporting sequence organization and consistency checking across accumulated site images or video frames. These functions are especially relevant when monitoring systems must interpret how site conditions evolve over time rather than only identify what is visible in an individual frame. The transfer to description generation further suggests practical applications in construction progress monitoring and reporting. For example, periodically collected site images could be processed to generate descriptions of how visible construction states evolve over time, providing preliminary content for daily or weekly progress reports. These descriptions could also support routine progress monitoring by helping site managers review changes in construction activities and identify transitions between successive work stages. In such applications, the generated descriptions would serve as supporting information for construction personnel rather than replacing their assessments. The scope of these applications should be distinguished from construction-specific semantic progress assessment. ConRank learns the chronological relationships among observed visual states but does not identify WBS-defined work packages, estimate physical completion percentages, or evaluate schedule adherence. Such assessments would require additional project information, including WBS definitions, planned

schedules, BIM models, or quantity records, and their alignment with visual observations. Within this scope, the present results position temporal ordering as a foundational VLM capability for organizing and interpreting longitudinal construction imagery. Although the YouTube time-lapse videos used in this study cover different construction projects with varying viewpoints and environmental conditions, the original recording configurations could not be fully verified. Nevertheless, these videos were considered suitable for investigating the temporal ordering capabilities of VLMs across diverse construction scenes. However, the dataset does not fully represent real-world construction monitoring conditions, particularly continuous CCTV footage with occlusions, motion blur, and irregular observations. Further validation under practical monitoring conditions is therefore needed to establish the generalizability of the proposed framework.

7. Conclusions

This study investigated whether vision-language models can understand construction progress as an ordered sequence of evolving visual states rather than as a collection of independent frames. Construction progress ordering was formulated through pairwise earlier–later discrimination and global chronological ordering, and ConRank was proposed to adapt a shared VLM using both forms of ranking supervision. The study further examined how local and global ordering supervision interacted during adaptation and whether the resulting temporal knowledge could be transferred from explicit ranking tasks to construction progress description generation. The framework was evaluated through zero-shot comparisons, task-specific and joint ranking adaptation, and progress-description experiments using construction time-lapse videos.

The contributions of this study are threefold. First, it establishes construction progress ordering as a distinct temporal-ordinal capability for VLM-based construction monitoring and provides a structured formulation that evaluates both local state discrimination and sequence-level chronological consistency. Second, it clarifies the relationship between pairwise and global ordering by showing that the two tasks share transferable ordinal information while also introducing different temporal granularity and supervision requirements, thereby motivating coordinated rather than naive joint adaptation. Third, it provides evidence that ranking-supervised temporal knowledge can transfer beyond explicit ordering to process-oriented description generation, positioning progress ordering as an intermediate capability between visual recognition and higher-level construction monitoring and reporting.

Despite the significant findings, this paper has several limitations. Although the dataset includes construction time-lapse videos with diverse visual and environmental conditions, it does not fully represent the variability of real-world construction monitoring footage. In particular, the original recording configurations could not be comprehensively verified, and the proposed framework has not been evaluated on continuously recorded CCTV footage. Therefore, the generalizability of the findings to practical monitoring environments remains to be established. Furthermore, the ranking evaluation was based on 10 held-out construction videos, with multiple frame pairs and groups derived from each video. Therefore, the observed performance differences should be interpreted within the evaluated dataset rather than as evidence of general superiority across construction projects. In addition, the supervision represents visual chronology rather than physical percent-complete, work-package completion, or deviation from a planned schedule. The current ranking objective also uses simplified temporal constraints, including a fixed similarity threshold, and the effect of ranking supervision was evaluated primarily through task performance rather than direct analysis of the learned multimodal representation. The downstream description evaluation was limited to the 10 held-out test videos, and broader validation is needed to establish the

generality of the observed transfer. Furthermore, although the reference descriptions were manually validated and multiple evaluation metrics were employed, potential stylistic and evaluator biases associated with GPT-4o-assisted reference construction and GPT-based evaluation cannot be entirely excluded.

To address these issues, future research should therefore validate progress ordering on broader and more naturally recorded construction videos, including fixed surveillance cameras, UAV, and mobile-site footage, and extend ranking supervision to account for temporal or progress distance. Representation-level analysis could further clarify how ordinal supervision reshapes multimodal features and where temporal ambiguity remains. Future studies could further incorporate construction stage- and environmental condition-specific annotations to systematically investigate ordering failures across different construction processes and visual conditions, including transitions between multiple construction stages. Future studies should also incorporate independently prepared human-authored reference descriptions and blinded expert evaluations to further assess factual accuracy and temporal consistency while minimizing potential reference and evaluator biases. More importantly, visual chronology should be linked with schedule, WBS, BIM, quantity, and activity information so that relative visual progression can be connected to measurable project progress. Such integration would extend the current framework from temporal interpretation of construction imagery toward plan-versus-actual comparison, work-stage assessment, and automated progress monitoring.

Author Contributions: Conceptualization, G.J. and H.K.; methodology, G.J.; software, G.J.; validation, G.J.; formal analysis, G.J.; investigation, G.J.; data curation, G.J.; visualization, G.J.; writing—original draft preparation, G.J.; writing—review and editing, G.J., K.-I.K. and H.K.; supervision, K.-I.K. and H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was by a grant (RS-2022-00143493) from the Digital-Based Building Construction and Safety Supervision Technology Research Program funded by the Ministry of Land, Infrastructure, and Transport of the Korean Government, and the Gachon University research fund of 2026 (GCU-202601430001).

Data Availability Statement: Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Ablation and Sensitivity Analysis of MTRL

To examine the individual contributions of the Margin-Temporal Ranking Loss (MTRL) components, an ablation study was conducted by separately applying the margin-ranking and temporal-contrastive terms under the same joint pairwise and global ranking setting. Table A1 presents the results for four configurations.

Table A1. Contribution of individual MTRL components to pairwise and global progress ordering.

Configuration	λ_m	λ_t	Hard	Medium	Easy	Overall	Kendall's τ	Exact Match
Pairwise + Global	0	0	68.00	85.00	91.00	81.33	0.304	19.00
+ Margin	0.5	0	71.00	86.00	92.00	83.00	0.327	21.67
+ Temporal-Contrastive	0	0.1	69.50	86.00	91.50	82.33	0.371	28.33
+ MTRL (ConRank)	0.5	0.1	**72.50**	**87.50**	**93.00**	**84.33**	**0.412**	**35.33**

Note: Hard, Medium, Easy, and Overall represent pairwise ordering accuracy (%), while Exact Match is reported as a percentage (%). Note: Bold values indicate the best performance for each metric.

The results show that the margin-ranking term provides a larger improvement in pairwise ordering accuracy, whereas the temporal-contrastive term contributes more substantially to global ordering performance. Combining both terms achieves the highest performance across all evaluated metrics, indicating their complementary contributions to joint ranking adaptation.

To evaluate the sensitivity of ConRank to the MTRL hyperparameters, each parameter was varied individually while the others were fixed at their default values ($\lambda_m = 0.5$, $\lambda_t = 0.1$, $m = 1.0$, and $\delta = 0.5$). Table A2 summarizes the results.

Table A2. Sensitivity of ConRank to MTRL hyperparameters.

Hyperparameter	Value	Hard	Medium	Easy	Overall	Kendall's τ	Exact Match
Default	–	**72.50**	**87.50**	**93.00**	**84.33**	**0.412**	**35.33**
λ_m (margin weight)	0.25	71.00	87.00	92.50	83.50	0.398	33.00
	1.0	73.00	87.50	92.50	84.33	0.389	31.67
λ_t (contrastive weight)	0.05	71.50	87.00	92.50	83.67	0.376	29.67
	0.2	71.50	87.00	93.00	83.83	0.405	34.33
m (margin)	0.5	71.00	87.00	92.50	83.50	0.401	33.67
	2.0	72.00	87.50	93.00	84.17	0.396	32.33
δ (similarity threshold)	0.3	71.00	86.50	92.50	83.33	0.394	32.67
	0.7	70.00	86.50	92.00	82.83	0.352	26.33

Note: One hyperparameter was varied at a time, with the remaining parameters fixed at their default values. Pairwise ordering accuracy and Exact Match are reported as percentages (%). Note: Bold values indicate the best performance for each metric.

The sensitivity analysis shows relatively limited variations in pairwise ordering accuracy across the tested parameter settings. Global ordering performance exhibits greater sensitivity, particularly to the temporal-contrastive weight and similarity threshold. Increasing the similarity threshold to 0.7 reduces Kendall's τ from 0.412 to 0.352 and Exact Match from 35.33% to 26.33%. Overall, the default configuration provides the best global ordering performance among the evaluated settings, although these results do not establish the global optimality of the selected hyperparameters.

References

1. Li, L.; Huang, Z.; Wang, J.; Du, B.; Dai, L. Automated Construction Monitoring Based on Computer Vision: A Comprehensive Review. *Dev. Built Environ.* **2026**, *25*, 100832. [CrossRef]
2. Kim, J. Visual Analytics for Operation-Level Construction Monitoring and Documentation: State-of-the-Art Technologies, Research Challenges, and Future Directions. *Front. Built Environ.* **2020**, *6*, 575738. [CrossRef]
3. Xiong, R.; Wang, Y.; Cai, J.; Liu, K.; Zhu, Y.; Tang, P.; El-Gohary, N. OpenConstruction: A Systematic Synthesis of Open Visual Data Sets for Data-Centric Intelligence in Construction Monitoring. *J. Comput. Civ. Eng.* **2026**, *40*, 03126002. [CrossRef]
4. Reja, V.K.; Varghese, K.; Ha, Q.P. Computer Vision-Based Construction Progress Monitoring. *Autom. Constr.* **2022**, *138*, 104245. [CrossRef]
5. Sami Ur Rehman, M.; Shafiq, M.T.; Ullah, F. Automated Computer Vision-Based Construction Progress Monitoring: A Systematic Review. *Buildings* **2022**, *12*, 1037. [CrossRef]
6. Han, M.; Zhang, J.; Huang, Y.; Xu, J.; Chen, X.; Chen, B.M. Enhancing Worker Monitoring and Management on Large-Scale Construction Sites with UAVs and Digital Twin Modeling. *Autom. Constr.* **2025**, *174*, 106108. [CrossRef]
7. Houng, S.C.; Pal, A.; Lin, J.J. 4D BIM and Reality Model–Driven Camera Placement Optimization for Construction Monitoring. *J. Constr. Eng. Manag.* **2024**, *150*, 04024045. [CrossRef]
8. Yang, J.; Park, M.-W.; Vela, P.A.; Golparvar-Fard, M. Construction Performance Monitoring via Still Images, Time-Lapse Photos, and Video Streams: Now, Tomorrow, and the Future. *Adv. Eng. Inform.* **2015**, *29*, 211–224. [CrossRef]
9. Xiao, B.; Wang, Y.; Zhang, Y.; Chen, C.; Darko, A. Automated Daily Report Generation from Construction Videos Using ChatGPT and Computer Vision. *Autom. Constr.* **2024**, *168*, 105874. [CrossRef]

10. Maaz, M.; Rasheed, H.; Khan, S.; Khan, F.S. Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models. *arXiv* **2023**. [CrossRef]
11. Shin, Y.; Seo, S.; Koo, C. Synthetic Video Generation Process Model for Enhancing the Activity Recognition Performance of Heavy Construction Equipment—Utilizing 3D Simulations in Unreal Engine Environment. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 74–82. [CrossRef]
12. Paneru, S.; Jeelani, I. Computer Vision Applications in Construction: Current State, Opportunities & Challenges. *Autom. Constr.* **2021**, *132*, 103940. [CrossRef]
13. Li, H.; Deng, H.; Deng, Y. Towards Worker-Centric Construction Scene Understanding: Status Quo and Future Directions. *Autom. Constr.* **2025**, *171*, 106005. [CrossRef]
14. Choi, H.; Ko, Y.; Kim, G.; Han, S. Comparative Analysis of Action Recognition Methodologies Using Construction Data in CCTV for Construction Sites. *Korean J. Constr. Eng. Manag.* **2026**, *27*, 79–88. [CrossRef]
15. Ahn, S.; Seo, S.; Koo, C. Tracking Reliability Estimation Model for Construction Equipment under Occlusion—Unreal Engine Simulation and Deep Learning Approach. *Korean J. Constr. Eng. Manag.* **2026**, *27*, 60–67. [CrossRef]
16. Jeong, G.; Jung, M.; Park, S.; Park, M.; Ahn, C.R. Contextual Multimodal Approach for Recognizing Concurrent Activities of Equipment in Tunnel Construction Projects. *Autom. Constr.* **2024**, *158*, 105195. [CrossRef]
17. Jeong, G.; Lee, J.; Park, M.; Ahn, C.R. Depth-Aware Detection of Hanging Objects for State Reasoning in Construction Sites. *Autom. Constr.* **2026**, *182*, 106757. [CrossRef]
18. Jeong, G.; Park, S.; Lee, J.; Park, M.; Ahn, C.R. HangCon: Benchmark Data Set for Enhanced Detection of Hanging Objects in Construction Sites. *J. Comput. Civ. Eng.* **2025**, *39*, 04025002. [CrossRef]
19. Kim, H.; Kim, J.; Kim, M. A Study on Construction Material Recognition Using YOLO and Virtual Images. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 90–101. [CrossRef]
20. Shin, Y.; Seo, S.; Koo, C. Deep Learning-Based Automated Method for Enhancing Excavator Activity Recognition in Far-Field Construction Site Surveillance Videos. *Autom. Constr.* **2025**, *173*, 106099. [CrossRef]
21. Chan, C.-F.; Wong, P.K.-Y.; Guo, X.; Cheng, J.C.P.; Chan, J.P.-C.; Leung, P.-H.; Tao, X. Context-Aware Vision-Language Model Agent Enriched with Domain-Specific Ontology for Construction Site Safety Monitoring. *Autom. Constr.* **2025**, *177*, 106305. [CrossRef]
22. Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. Qwen2.5-VL Technical Report. *arXiv* **2025**. [CrossRef]
23. Pu, H.; Yang, X.; Li, J.; Guo, R. AutoRepo: A General Framework for Multimodal LLM-Based Automated Construction Reporting. *Expert Syst. Appl.* **2024**, *255*, 124601. [CrossRef]
24. Jung, Y.; Cho, I.; Hsu, S.-H.; Golparvar-Fard, M. VisualSiteDiary: A Detector-Free Vision-Language Transformer Model for Captioning Photologs for Daily Construction Reporting and Image Retrievals. *Autom. Constr.* **2024**, *165*, 105483. [CrossRef]
25. Misra, I.; Zitnick, C.L.; Hebert, M. Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2016; Volume 9905, pp. 527–544, ISBN 978-3-319-46447-3.
26. Wei, D.; Lim, J.; Zisserman, A.; Freeman, W.T. Learning and Using the Arrow of Time. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8052–8060.
27. Bagad, P.; Tapaswi, M.; Snoek, C.G.M. Test of Time: Instilling Video-Language Models with a Sense of Time. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 2503–2516.
28. Parikh, D.; Grauman, K. Relative Attributes. In *Proceedings of the 2011 International Conference on Computer Vision*; IEEE: Piscataway, NJ, USA, 2011; pp. 503–510.
29. Yu, W.-H.; Lin, Y.-Y.; Yang, M.-H.; Tsai, Y.-H. Ranking-Aware Adapter for Text-Driven Image Ordering with CLIP. In *Proceedings of the International Conference on Learning Representations*; Yue, Y., Garg, A., Peng, N., Sha, F., Yu, R., Eds.; ICLR: Singapore, 2025; Volume 2025, pp. 80614–80633.
30. Sonthalia, A.; Uselis, A.; Oh, S.J. On the Rankability of Visual Embeddings. *arXiv* **2025**. [CrossRef]
31. Kim, J.-Y.; Kim, S.; Kim, G.-H. Progress Measurement of Structural Frame Construction Using Point Cloud Data. *Korean J. Constr. Eng. Manag.* **2024**, *25*, 37–46. [CrossRef]
32. Park, M.-G.; Kim, K.-H. Development of an Automatic Classification Model for Construction Site Photos with Semantic Analysis Based on Korean Construction Specification. *Korean J. Constr. Eng. Manag.* **2024**, *25*, 58–67. [CrossRef]
33. Sherafat, B.; Ahn, C.R.; Akhavian, R.; Behzadan, A.H.; Golparvar-Fard, M.; Kim, H.; Lee, Y.-C.; Rashidi, A.; Azar, E.R. Automated Methods for Activity Recognition of Construction Workers and Equipment: State-of-the-Art Review. *J. Constr. Eng. Manag.* **2020**, *146*, 03120002. [CrossRef]
34. Kim, K.-H.; Kim, T.-J.; Park, Y.-H.; Yoon, S.-H.; Jeon, J.-H.; Kim, J.-W. Barriers to the Adoption of Computer Vision-Based Object Recognition in Construction Sites—Analyzing Barriers to Safety Technology Adoption Using the Technology Acceptance Model. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 87–96. [CrossRef]

35. Han, S.; Won, J.; Koo, C. A Strategic Approach to Enhancing the Practical Applicability of Vision-Based Detection and Classification Models for Construction Tools—Sensitivity Analysis of Model Performance Depending on Confidence Threshold. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 102–109. [CrossRef]
36. Ghelmani, A.; Hammad, A. Self-Supervised Contrastive Video Representation Learning for Construction Equipment Activity Recognition on Limited Dataset. *Autom. Constr.* **2023**, *154*, 105001. [CrossRef]
37. Lin, Z.-H.; Chen, A.Y.; Hsieh, S.-H. Temporal Image Analytics for Abnormal Construction Activity Identification. *Autom. Constr.* **2021**, *124*, 103572. [CrossRef]
38. Roberts, D.; Golparvar-Fard, M. End-to-End Vision-Based Detection, Tracking and Activity Analysis of Earthmoving Equipment Filmed at Ground Level. *Autom. Constr.* **2019**, *105*, 102811. [CrossRef]
39. Chen, C.; Zhu, Z.; Hammad, A. Automated Excavators Activity Recognition and Productivity Analysis from Construction Site Surveillance Videos. *Autom. Constr.* **2020**, *110*, 103045. [CrossRef]
40. Kim, J.; Chi, S. Action Recognition of Earthmoving Excavators Based on Sequential Pattern Analysis of Visual Features and Operation Cycles. *Autom. Constr.* **2019**, *104*, 255–264. [CrossRef]
41. Choi, H.; Ko, Y.; Kim, G.; Han, S. Development of Deep Learning Methodology of Object and Action Recognition for Discrete Event Simulation in Construction. *Korean J. Constr. Eng. Manag.* **2026**, *27*, 61–70. [CrossRef]
42. Jung, S.; Jeoung, J.; Kang, H.; Hong, T. 3D Convolutional Neural Network-based One-stage Model for Real-time Action Detection in Video of Construction Equipment. *Comput.-Aided Civ. Infrastruct. Eng.* **2022**, *37*, 126–142. [CrossRef]
43. Erfani, A.; Mansouri, A. Applications of Multimodal Large Language Models in Construction Industry. *Adv. Eng. Inform.* **2026**, *69*, 103909. [CrossRef]
44. Liu, H.; Wang, G.; Huang, T.; He, P.; Skitmore, M.; Luo, X. Manifesting Construction Activity Scenes via Image Captioning. *Autom. Constr.* **2020**, *119*, 103334. [CrossRef]
45. Ding, Y.; Liu, M.; Luo, X. Safety Compliance Checking of Construction Behaviors Using Visual Question Answering. *Autom. Constr.* **2022**, *144*, 104580. [CrossRef]
46. Tsai, W.-L.; Le, P.-L.; Ho, W.-F.; Chi, N.-W.; Lin, J.J.; Tang, S.; Hsieh, S.-H. Construction Safety Inspection with Contrastive Language-Image Pre-Training (CLIP) Image Captioning and Attention. *Autom. Constr.* **2025**, *169*, 105863. [CrossRef]
47. Wang, Y.; Luo, H.; Fang, W. An Integrated Approach for Automatic Safety Inspection in Construction: Domain Knowledge with Multimodal Large Language Model. *Adv. Eng. Inform.* **2025**, *65*, 103246. [CrossRef]
48. Guo, K.X.; Wong, P.K.-Y.; Cheng, J.C.P.; Chan, C.-F.; Leung, P.-H.; Tao, X. Enhancing Visual-LLM for Construction Site Safety Compliance via Prompt Engineering and Bi-Stage Retrieval-Augmented Generation. *Autom. Constr.* **2025**, *179*, 106490. [CrossRef]
49. Liu, Y.; Li, L.; Su, X.; Lin, H.; Zhang, L.; Guan, D.; Cai, H. Retrieval-Augmented Multimodal Large Language Models for Visual Question Answering of Construction Occupational Health and Safety Hazards. *J. Manag. Eng.* **2026**, *42*, 04026044. [CrossRef]
50. Li, Y.; Xu, F.; Zhang, Z.; Mei, X.; Huang, H. Construction Site Fall Hazard Identification and Automated Captioning Using Adapted Vision-Language Models. *Autom. Constr.* **2026**, *183*, 106790. [CrossRef]
51. Choe, S.H.; Kim, H.S. Development of Generative AI-Based System for Extracting and Analyzing Construction Accident Cases. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 3–14. [CrossRef]
52. Lee, J.; Ahn, S. Development of AI Prototype for Generating Construction Safety Guidelines Through Fine-Tuning of Large-Scale Language Model. *Korean J. Constr. Eng. Manag.* **2025**, *26*, 20–31. [CrossRef]
53. Pu, H.; Yang, X.; Shi, Z.; Jin, N. Automated Inspection Report Generation Using Multimodal Large Language Models and Set-of-Mark Prompting. In Proceedings of the 41st International Symposium on Automation and Robotics in Construction (ISARC 2024), Lille, France, 3 June 2024.
54. Chen, Q.; Yin, X. Tailored Vision-Language Framework for Automated Hazard Identification and Report Generation in Construction Sites. *Adv. Eng. Inform.* **2025**, *66*, 103478. [CrossRef]
55. Liu, Y.; Noghabaei, M.; Bonilla, M. Automated Construction Progress Reporting from Drone Images via Vision-Language Models. In *Proceedings of the Construction Research Congress 2026*; American Society of Civil Engineers: San Antonio, TX, USA, 2026; pp. 518–528.
56. Chang, C.-Y.; Khanum, A.; Su, S.-G.; Tsai, M.-H.; Hsu, H.-H.; Chen, W.-T.; Lu, H.-W. Optimizing Construction Safety: Multimodal Prompts For Automated Image Descriptions In Daily Construction Reports. In *Proceedings of the 2025 IEEE International Conference on Image Processing Workshops (ICIPW)*; IEEE: Anchorage, AK, USA, 2025; pp. 263–268.
57. Fernando, B.; Bilen, H.; Gavves, E.; Gould, S. Self-Supervised Video Representation Learning with Odd-One-Out Networks. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Honolulu, HI, USA, 2017; pp. 5729–5738.
58. Dwibedi, D.; Aytar, Y.; Tompson, J.; Sermanet, P.; Zisserman, A. Temporal Cycle-Consistency Learning. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Long Beach, CA, USA, 2019; pp. 1801–1810.

59. Li, L.; Chen, Y.-C.; Cheng, Y.; Gan, Z.; Yu, L.; Liu, J. HERO: Hierarchical Encoder for Video+Language Omni-Representation Pre-Training. In *Proceedings of the Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 2046–2065.
60. Cherian, A.; Wang, J.; Hori, C.; Marks, T.K. Spatio-Temporal Ranked-Attention Networks for Video Captioning. In *Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*; IEEE: Snowmass Village, CO, USA, 2020; pp. 1606–1615.
61. Wu, H.; Liu, H.; Qiao, Y.; Sun, X. DIBS: Enhancing Dense Video Captioning with Unlabeled Videos via Pseudo Boundary Enrichment and Online Refinement. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA, 17–21 June 2024; pp. 18699–18708.
62. Xue, Z.; An, J.; Yang, X.; Grauman, K. Progress-Aware Video Frame Captioning. In Proceedings of the 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 11–15 June 2025; pp. 13639–13650.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.