

RESEARCH ARTICLE



Knowledge-Guided Multimodal Deep Learning for BIM-Based Construction Progress and Quality Verification from Site Images and Video

Mohammed Hijazi^{1*}

¹*Freyssinet Saudi Arabia, Saudi Arabia*

Abstract: Construction progress and quality verification require visual evidence to be interpreted against object identity, schedule state, and installation dependencies. This study proposes a knowledge-guided multimodal method that combines 32×32 synthetic image crops, a 15-variable building information modeling (BIM) vector, five schedule variables, neural late fusion, explicit construction-rule factors, and confidence-based abstention. A reproducible benchmark of 4,800 observations across eight synthetic project domains was evaluated using a complete seven-cell factorial analysis of image, BIM, and schedule modalities, together with recent compact visual backbones. On held-out Project 7, BIM-only achieved 43.5% accuracy and 0.309 macro-F1, schedule-only achieved 87.0% and 0.825, image + schedule achieved 89.3% and 0.859, and image + BIM + schedule achieved 87.8% and 0.833. These results show that the large schedule gain primarily recovers temporal relationships deliberately encoded in the synthetic generator and should not be interpreted as independent evidence of field-level causal superiority. The knowledge-guided output achieved 90.5% accuracy and 0.866 macro-F1 and eliminated the defined predecessor violations. At a confidence threshold of 0.65, 80.3% of observations were processed automatically at 95.6% retained accuracy. A separate synthetic defect task reached 0.958 F1 using image features, while BIM fusion reduced recall. The results support an auditable simulation-based proof of concept; field validation with independently labeled, registered site imagery remains necessary before operational deployment.

Keywords: building information modeling, multimodal deep learning, computer vision, construction progress monitoring, quality verification

1. Introduction

Reliable construction control depends on knowing what has been installed, where it has been installed, whether the work corresponds to the approved model and schedule, and whether the observed condition is acceptable. Conventional progress reporting is still dominated by site walks, photographs, spreadsheets, and subjective percentage estimates. These practices are costly to repeat, difficult to audit at object level, and vulnerable to reporting lag. Building information modeling (BIM) provides a structured representation of design objects and their attributes, while site images and video provide frequent evidence of physical conditions. The unresolved problem is how to combine both forms of information so that an artificial intelligence system does more than recognize pixels: it must reason about project-specific meaning.

Foundational work demonstrated the value of comparing sensed geometry and visual evidence with as-planned models and schedules [1]. Recent research has extended and synthesized this

foundation through computer-vision progress-monitoring frameworks, digital-twin-based activity tracking, BIM-linked image segmentation, indoor component recognition, real-time object detection, and prefabricated-module monitoring [2–8]. Systematic reviews nevertheless identify persistent limitations involving data scarcity, occlusion, changing viewpoints, domain shift, weak interoperability with project information, and limited treatment of uncertainty [9, 10].

A further limitation is conceptual. Many systems use BIM only as a visualization reference or post-processing target. In contrast, BIM contains information that can constrain an artificial intelligence (AI) prediction before it becomes a project decision. A component marked as complete should not normally precede a mandatory predecessor; an object scheduled for a later phase should be treated differently from an overdue object; and visually similar components can be distinguished using class, dimensions, discipline, level, zone, and system context. These relationships correspond to knowledge representation and reasoning, an area explicitly relevant to Artificial Intelligence and Applications. Multimodal learning offers a practical mechanism for combining visual and structured project data [11, 12], while selective

*Corresponding author: Mohammed Hijazi, Freyssinet Saudi Arabia, Saudi Arabia. Email: mohammed.hijazi@fsa.com.sa

prediction provides a way to retain human oversight when confidence is inadequate [13, 14].

This paper proposes and computationally evaluates a knowledge-guided multimodal deep-learning method for BIM-based construction progress and quality verification. The work is intentionally framed as a simulation-based proof of concept rather than evidence from a live construction project. Synthetic site-like images are paired with BIM attributes, schedule variables, and precedence states. The controlled setting permits ablation of each information source, comparison with recent visual backbones, and direct measurement of logical violations and confidence-based review.

The contribution is fourfold. First, the method treats BIM and schedule data as active machine learning inputs rather than passive visualization. Second, it combines neural probabilities with explicit construction knowledge and records the rule that modifies each prediction. Third, it provides a detailed, reproducible inference procedure and compares the method with compact Swin Transformer V2, EfficientViT, and ConvNeXt V2 baselines on the same held-out project domain. Fourth, it reports a negative multimodal result for quality verification, showing that BIM fusion is not universally beneficial when the target is dominated by visible defect cues.

2. Related Work and Research Gap

2.1. BIM-linked progress monitoring

Automated progress monitoring has evolved from manual image comparison toward geometric registration, visual recognition, and model-based status inference. Early work associated unordered daily images with Industry Foundation Classes (IFC)-based objects [1]. Recent reviews and digital-twin research have synthesized the progression from visual capture and geometric registration to BIM-linked status inference and closed-loop control [2, 3]. More recent vision-based systems have used Mask Region-Based Convolutional Neural Networks (CNNs) and BIM to quantify indoor work completion, while multi-technology architectures merge complementary site data streams to improve project-status acquisition [4, 15].

Recent studies have strengthened geometric and robotic monitoring through IFC integration, uncertainty-aware change detection, mobile mapping, and online as-built model updating. Kavaliauskas et al. [16] integrated point clouds with IFC-based BIM and highlighted the persistent challenges of as-built object identification and coordinate alignment. Meyer et al. [17] incorporated uncertainties when comparing indoor point clouds with BIM, while Vassena et al. [18] connected four-dimensional BIM with repeated mobile-mapping surveys. Spinner and Degani [19] injected schedule and spatial prior knowledge into robotic BIM updating, and Choi et al. [20] integrated drone imagery and AI with BIM-based site management. These contributions demonstrate strong geometric reasoning, but acquisition, registration, and processing can remain resource intensive for frequent visual reporting.

2.2. Computer vision and quality inspection

Computer vision provides a lower-cost path when cameras are available on mobile devices, fixed viewpoints, robots, or unmanned platforms. Reviews show that convolutional networks, object detectors, and segmentation models have become central to automated construction monitoring, while also emphasizing

dataset and generalization limitations [9, 10]. Recent applications include real-time object detection [5], indoor progress recognition and work-completion estimation [6], deep-learning monitoring of prefabricated volumetric construction [7], time-series supervision of temporary facilities [21], and BIM-linked bricklaying progress tracking [8]. For quality inspection, performance still depends on defect visibility, class balance, background variation, and transfer to unseen projects.

Visual systems face several ambiguities. Absence in an image can mean that an object is not installed, is outside the camera view, or is occluded. Partial installation may resemble a defect. Different mechanical, electrical, and plumbing components may have similar local appearance. Image evidence therefore benefits from context, but the context must be represented in a form that a model can learn or a reasoning component can enforce.

2.3. Multimodal learning, reasoning, and uncertainty

Multimodal machine learning studies how information from distinct modalities can be represented, aligned, fused, and jointly learned [11, 12]. Recent visual architectures provide relevant comparison points: Swin Transformer V2 uses scaled cosine window attention and residual post-normalization [22]; ConvNeXt V2 combines modern convolutional blocks with global response normalization [23]; and EfficientViT targets memory-efficient attention for high-speed recognition [24]. In the present problem, visual appearance is one modality and structured BIM/schedule attributes form another, so backbone improvements must be distinguished from gains caused by project context.

Knowledge-based reasoning complements statistical learning. BIM can be transformed into object graphs representing containment, connectivity, support, accessibility, and sequence relationships [25], for example, derived directed representative graphs for mechanical, electrical, and plumbing systems from BIM data. Data readiness is equally important: [26] identified limitations in time-series support, geometry extraction, and IFC-to-AI conversion toolchains. For progress monitoring, a small transparent rule set can prevent impossible states without replacing the neural model. Finally, operational use requires uncertainty management. Modern neural networks may produce overconfident probabilities [14], while selective classification allows a system to abstain below a confidence threshold [13].

Table 1 summarizes the progression from visual and geometric monitoring toward the proposed knowledge-guided architecture.

The research gap is therefore not the absence of computer vision or BIM comparison in isolation. It is the lack of an experimentally transparent architecture that (i) states each technical inference step, (ii) quantifies all non-empty combinations of image, BIM, and schedule information rather than only a sequential addition path, (iii) separates learned context from explicit symbolic constraints, and (iv) compares those effects with recent visual backbone families within one reproducible held-out-project benchmark.

2.4. Design requirements and testable propositions

The literature synthesis was translated into six design requirements. R1 is object traceability: every inference must refer to a BIM object or an explicitly unresolved candidate. R2 is modality separation: visual, BIM, schedule, and rule contributions must be removable so that their value can be measured. R3 is

Table 1
Positioning of representative monitoring approaches

Approach	Primary evidence	Remaining limitation
Photograph/4D BIM	Images or photogrammetry	Registration and limited project semantics
Scan-vs-BIM	Point clouds + model geometry	Acquisition cost and tolerance sensitivity
Deep visual inspection	Images/video	Weak schedule and object context
Multimodal BIM-AI	Images + structured attributes	Needs modality ablation and domain testing
Knowledge-guided AI	Neural outputs + explicit rules	Rules require governance and validation

constraint transparency: a prediction altered by project knowledge must retain the original probability and identify the applied rule. R4 is domain separation: evaluation should hold out an entire project-like visual domain rather than distribute nearly identical project images across training and test sets. R5 is selective automation: the system must be able to abstain and route an observation to a reviewer. R6 is evidence-bound reporting: progress and quality claims must be limited to what the supplied modality can establish.

These requirements support four testable propositions. P1 tests whether BIM features add information beyond image-only learning when object appearance is ambiguous. P2 tests the incremental and stand-alone predictive value of schedule features; because the synthetic generator uses temporal variables to construct progress states, schedule-related gains are interpreted as recovery of programmed benchmark relationships rather than causal or field-performance evidence. P3 states that precedence rules should reduce sequence-inconsistent installed predictions beyond the unconstrained schedule-aware model. P4 states that the value of BIM fusion is task dependent and may be small or negative when labels are determined mainly by visible anomalies. A complete 2^3-1 factorial analysis of image, BIM, and schedule modalities is used to expose marginal contributions and interaction effects instead of presenting only the strongest final model.

3. Proposed Knowledge-Guided Method

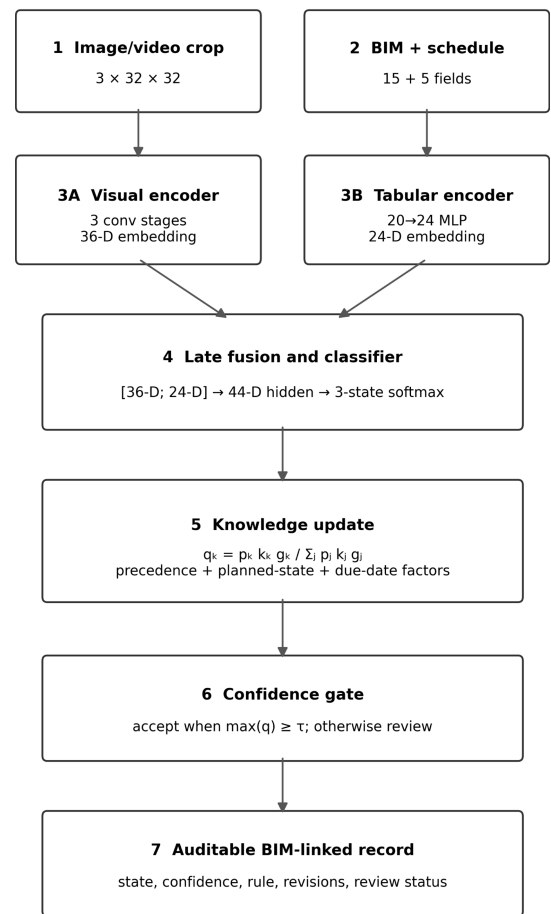
Figure 1 summarizes the seven technical stages of the proposed method. This section defines only the contributed inference method; benchmark construction and training procedures are reported separately in Section 4.

3.1. Problem definition and inputs

For observation i , the method receives an RGB object crop $x_i \in \mathbb{R}^{3 \times 32 \times 32}$, a BIM feature vector $b_i \in \mathbb{R}^{15}$, a schedule vector $s_i \in \mathbb{R}^5$, and a rule record r_i . The BIM vector contains six one-hot object classes, normalized width and height, normalized floor, four one-hot zones, and two one-hot disciplines. The schedule vector contains three planned-state indicators, normalized inspection time, and normalized due-date delta. The rule record contains predecessor completion and the unnormalized planned-state fields used for transparent post-model reasoning.

The target progress state y_i has three classes: not installed, partially installed, and installed. The benchmark supplies an object-centered observation and its BIM link; therefore, the proposed method evaluates state inference rather than camera localization or image-to-BIM registration. In a field system, association confidence would be checked before Step 1, and unresolved candidates would be routed to review.

Figure 1
Seven-stage knowledge-guided multimodal inference method, from BIM-linked observation to auditable decision
Knowledge-guided BIM-AI pipeline



3.2. Visual and structured encoders

The visual encoder applies three 3×3 convolutional stages with 10, 20, and 36 channels. Rectified linear units follow each convolution; the first two stages use 2×2 max pooling, and adaptive average pooling converts the final activation map into a 36-dimensional vector v_i . The compact design limits capacity so that improvements from structured project data remain observable.

The structured encoder concatenates b_i and s_i into a 20-variable vector and maps it through a 20→24 linear layer with a rectified linear unit, producing embedding z_i . Late fusion concatenates $[v_i; z_i]$ into 60 variables and applies a 60→44 hidden

layer, rectified linear activation, and dropout of 0.12, followed by a $44 \rightarrow 3$ output. The neural probability vector is $p_i = \text{softmax}(W_0 \text{ReLU}(W_f [v_i; z_i] + c_f) + c_0)$. The BIM-only configuration uses the same path with the 15-variable b_i vector.

3.3. Knowledge-guided probability update

The knowledge layer converts project rules into a nonnegative factor vector $g_i = [g_0, g_1, g_2]$. Four transparent factors were used in the proof of concept. If a mandatory predecessor is incomplete, [1.50, 0.90, 0.06] suppresses the installed state. If the planned state is not installed, [1.18, 0.95, 0.42] discourages premature completion. If the planned state is installed and predecessors are complete, [0.65, 0.97, 1.13] modestly favors completion. If due-date delta exceeds 0.20 with completed predecessors, [0.82, 0.97, 1.07] provides a smaller overdue adjustment. When several conditions apply, their factors are multiplied elementwise.

The adjusted probability is $q_{i,k} = p_{i,k} g_{i,k} / \sum_j p_{i,j} g_{i,j}$. This soft-constraint formulation preserves visual evidence while making technically implausible states less likely. The system retains p_i , g_i , q_i , the triggered rule identifiers, and the rule-set version so that every change from neural output to final output can be reproduced.

3.4. Confidence and policy gate

Let $c_i = \max(q_i)$. The automatic state is $\hat{y}_i = \text{argmax}(q_i)$ only when $c_i \geq \tau$ and no mandatory-review policy applies. Otherwise, the method returns abstain and routes the observation to human review. The threshold τ controls the measurable trade-off between automated coverage and retained accuracy; it does not replace project-specific review requirements for contractual, safety-critical, or ambiguous observations.

3.5. Step-by-step inference procedure

Step 1—Validate the input link. Confirm that the crop is associated with one BIM object candidate and that the object, model revision, schedule revision, capture time, and evidence hash are available. If association is unresolved, stop and request review.

Step 2—Encode the evidence. Normalize x_i using the fixed channel statistics used by the submitted training pipeline, one-hot encode the categorical BIM and planned-state fields, and normalize geometry, floor, inspection time, and due-date delta.

Step 3—Compute v_i and z_i with the visual and structured encoders.

Step 4—Fuse both embeddings and calculate the three-state neural distribution p_i .

Step 5—Evaluate the governed construction conditions in r_i , multiply the applicable factor vectors, and calculate q_i .

Step 6—Apply the confidence and mandatory-review gate.

Step 7—Write a BIM-linked record containing the selected state or abstention, p_i , q_i , confidence, rule identifiers, model and rule versions, and reviewer status.

The neural path requires approximately 11,939 trainable parameters in the BIM-plus-schedule configuration. Rule evaluation is constant-time for the four conditions used here, so the contributed reasoning layer adds negligible inference cost relative to visual encoding while providing an explicit logical check.

3.6. Quality-verification branch

The quality branch uses the same visual principle but replaces the three-state softmax with a binary sigmoid for visible anomaly presence. A BIM-fusion variant concatenates the visual embedding with object attributes. No progress rule factors are applied because the synthetic anomaly labels are visual and do not encode specification tolerances. Keeping this branch separate tests whether multimodal context is relevant to the target rather than assuming that fusion must always improve performance.

4. Experimental Design

4.1. Synthetic benchmark

A synthetic benchmark was created to evaluate the method without claiming access to confidential live-project data. It contains 4,800 raster images at 64×64 pixels, generated across eight project domains with 600 observations per project. Six classes were represented: column, beam, wall, pipe, duct, and cable tray. Each domain used a distinct background, object palette, clutter level, brightness, and noise profile. Project 7 deliberately used darker lighting, greater noise, additional clutter, and a higher low-light probability to create a held-out domain shift.

Figure 2 reports six exact observations from held-out Project 7 rather than illustrative construction photographs. The samples are P7-0000, P7-0030, P7-0050, P7-0012, P7-0047, and P7-0029. For each sample, the figure shows the raw 64×64 generator output, the exact 32×32 array supplied to the progress classifier, the ground-truth state, and the actual prediction and confidence from the evaluated schedule-aware checkpoint (progress_schedule.pt). These are crop-level three-state classification inputs; no external site photographs, detection boxes, or segmentation masks are used.

The benchmark generator is supplied as generate_v2.py and uses fixed seed 20260625. Geometry is parameterized by class-specific normalized width and height distributions. Progress state is generated from inspection time, planned start and finish, project delay, class delay, floor effect, stochastic deviation, and predecessor completion. Visible objects may contain synthetic anomalies such as cracks, deformation, misalignment, or connection irregularities. Occlusion bars, perspective lines, Gaussian blur, brightness variation, and pixel noise are added. These patterns are deliberately non-photorealistic and are used only for controlled proof-of-concept comparisons. The exact sample identifiers and checkpoint probabilities shown in Figure 2 are also provided in the supplementary sample-audit file.

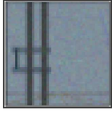
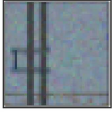
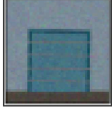
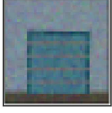
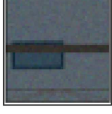
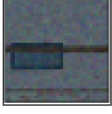
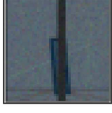
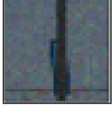
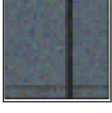
Table 2 lists the benchmark scale, labels, BIM fields, schedule variables, and generated visual disturbances.

4.2. Data partition and preprocessing

Projects 0–6 formed the development set. A stratified 15% subset was used for validation, while project 7 was held out entirely for testing, yielding 600 progress observations. This project-level split prevents visually related observations from the same synthetic domain appearing in both development and test sets. For quality verification, only partial or installed objects were used, producing 384 held-out observations with 25% defect prevalence.

Images were downsampled from the stored 64×64 generator outputs to exact 32×32 model inputs using every second pixel and then normalized with fixed channel statistics used by the submitted training pipeline. Categorical BIM fields were one-hot encoded, and dimensions, floor, inspection time, and

Figure 2
Held-out synthetic benchmark samples and schedule-aware predictions

ID / class	64×64	32×32	GT → prediction (confidence)
P7-0000 Cable Tray			P → P (0.788) plan P; pred Y; occ M; NL
P7-0030 Wall			I → P (0.473) plan I; pred Y; occ L; NL
P7-0050 Duct			I → I (0.546) plan I; pred Y; occ L; LL
P7-0012 Column			I → I (0.792) plan I; pred Y; occ M; LL
P7-0047 Beam			NI → NI (0.892) plan P; pred N; occ M; NL
P7-0029 Pipe			NI → NI (0.982) plan NI; pred N; occ M; LL

Note: NI = not installed; P = partially installed; I = installed; pred = predecessor complete; occ = L/M/H; NL = normal light; LL = low light.

Table 2
Synthetic benchmark composition and variables

Category	Specification
Scale	4,800 images; eight project domains; 600 observations each
Object classes	Column, beam, wall, pipe, duct, cable tray
Progress labels	Not installed, partially installed, installed
BIM features	Class, width, height, floor, zone, discipline
Schedule features	Inspection time, planned state, due-date delta
Knowledge input	Predecessor completion and schedule-state rules
Visual variation	Clutter, occlusion, low light, blur, noise, domain style
Quality label	Synthetic visible anomaly for installed/partial objects

due-date delta were normalized. The BIM vector contained 15 variables, and the schedule vector contained five variables. Predecessor completion was excluded from learned inputs and supplied only to the knowledge layer so that neural and rule contributions remained separable.

4.3. Proposed configurations and training

Table 3 defines the complete seven-cell factorial analysis of the three learned modalities (image, BIM, and schedule), the knowledge-guided post-model configuration, and the three compact recent image-only comparison families.

The proposed progress networks used AdamW, learning rate 0.003, weight decay 10^{-4} , batch size 256, cross-entropy loss, seven epochs, and validation checkpoint selection by macro-F1. Image-containing multimodal models used the same three-stage CNN and a 24-unit structured embedding followed by a 44-unit fused hidden layer. Structured-only factorial cells used the corresponding tabular path without the CNN: BIM-only used $15 \rightarrow 24 \rightarrow 44 \rightarrow 3$, schedule-only used $5 \rightarrow 24 \rightarrow 44 \rightarrow 3$, and BIM + schedule used $20 \rightarrow 24 \rightarrow 44 \rightarrow 3$. The four factorial cells added in the revision were trained with the same split, preprocessing, optimization settings, epoch count, and fixed seed 20260625 as the original configurations. The knowledge layer uses the image + BIM + schedule checkpoint without adding trainable parameters.

Quality networks used convolutional stages with 12, 24, and 40 channels. A weighted random sampler balanced training batches. Nine epochs, AdamW with learning rate 0.0028, binary cross-entropy with logits, and validation average precision were used. The validation threshold maximizing F1 was frozen before test evaluation.

Table 3
Complete modality factorial configurations, knowledge-guided configuration, and recent visual comparison models

Configuration	Learned inputs	Defining component/reasoning
Image-only CNN	Image	Three-stage compact CNN
BIM-only multilayer perceptron (MLP)	15 BIM fields	15→24→44→3 tabular network
Schedule-only MLP	5 schedule fields	5→24→44→3 tabular network
BIM + schedule MLP	15 BIM + 5 schedule fields	20→24→44→3 tabular network
Image + BIM	Image + 15 BIM fields	Late feature fusion
Image + schedule	Image + 5 schedule fields	Late feature fusion
Image + BIM + schedule	Image + BIM + schedule fields	Late feature fusion
Knowledge-guided	Image + BIM + schedule neural probabilities	Rule factors + confidence gate
Compact Swin V2	Image	Shifted-window cosine attention
Compact EfficientViT	Image	Local convolution + efficient attention
Compact ConvNeXt V2	Image	Depthwise convolution + GRN

4.4. Comparative study with recent visual models

Three recent architecture families were implemented in compact form to fit the low-resolution benchmark and to avoid introducing external pretrained data. Compact Swin Transformer V2 used 4×4 patches, stage depths [1,1,2], heads [2,4,8], a 4×4 window, scaled cosine attention, and 258,753 parameters [22]. Compact EfficientViT used local depthwise convolution and lightweight grouped/linear attention in 24- and 48-channel stages, totalling 48,075 parameters [24]. Compact ConvNeXt V2 used 7×7 depthwise convolution, stage widths of 16/32/64, and global response normalization, totalling 61,859 parameters [23].

All comparison models used only the image crop, the same development/validation/project-7 split, development-only normalization, AdamW, validation-selected checkpoints, and the same fixed seed. No ImageNet or other pretraining was used. Training was limited to 10–12 epochs with cosine learning-rate decay. The comparison therefore evaluates architecture families on the same synthetic domain rather than reproducing their full published-scale configurations.

4.5. Evaluation measures and comparison protocol

Progress classification was assessed using accuracy, balanced accuracy, macro-F1, and confusion matrices. Macro-F1 gives equal weight to the three progress states. The complete factorial design evaluated all seven non-empty subsets of image, BIM, and schedule modalities: image; BIM; schedule; image + BIM; image + schedule; BIM + schedule; and image + BIM + schedule. A sequence-violation rate measured the proportion of held-out observations predicted as installed while the mandatory predecessor was incomplete. Recent backbones were compared separately on the same held-out project, and parameter counts were reported to distinguish visual capacity from project-context contributions.

Rule ablation isolated precedence, schedule-state, and overdue effects. Robustness was summarized for occlusion and lighting subsets. Selective prediction was evaluated through coverage, review rate, retained accuracy, and retained macro-F1. Quality verification used precision, recall, F1, receiver operating characteristic area under the curve (ROC-AUC), and average precision [27].

4.6. Reproducibility and experimental boundaries

The supplementary package includes the benchmark generator, 4,800 generated images, metadata, proposed-model scripts,

factorial-analysis script and results, checkpoints, prediction arrays, confusion matrices, and result tables. Deterministic seeds were set for Python, NumPy, and PyTorch; the project-level development/test split was fixed before training; and test labels were evaluated only after validation checkpoint selection. Figure 2 is traceable to explicit sample identifiers and the submitted schedule-aware checkpoint.

The purpose is controlled architecture and information ablation, not maximum synthetic accuracy. Results are fixed-seed benchmark runs and should not be interpreted as estimates of field performance. The complete factorial analysis is especially important because schedule variables participate in the synthetic target-generation process. Therefore, a strong schedule-only or schedule-fusion result demonstrates that the architecture can recover information intentionally embedded in the benchmark; it does not demonstrate an unknown causal relationship or real-site generalization. Compact recent models preserve defining block types but are smaller than their published ImageNet configurations.

The proposed schedule-aware neural path is substantially smaller than the recent visual comparators: 11,939 trainable parameters versus 48,075–258,753. End-to-end deployment cost would nevertheless be dominated by capture, detection, registration, and object association, which are outside the present classifier-level experiment.

All claims are therefore limited to the supplied synthetic benchmark, the project-level domain holdout, the defined symbolic rules, and the reported metrics. Generated observations must not be represented as photographs from a real construction project. The factorial schedule results are treated as a benchmark-consistency test, not as evidence that schedule features would yield the same magnitude of improvement on independently labeled field data.

5. Results

5.1. Progress-state classification

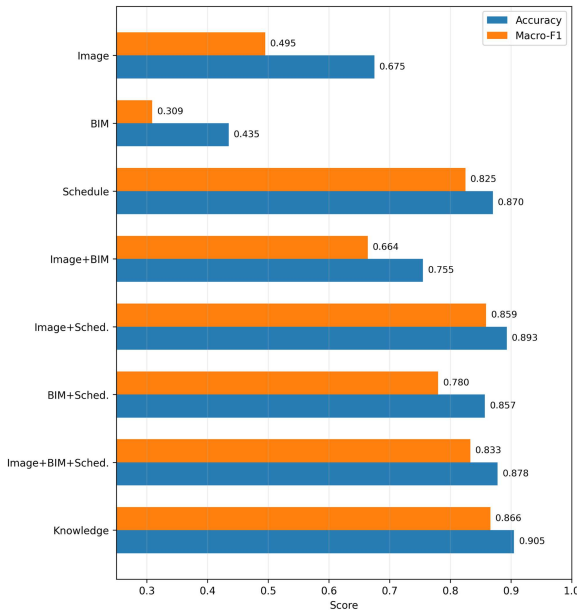
Table 4 reports the complete image/BIM/schedule factorial analysis, the knowledge-guided output, and the three compact recent visual backbones on held-out Project 7. Figure 3 visualizes the seven non-empty modality combinations plus the knowledge-guided result.

The complete factorial analysis changes the interpretation of the sequential ablation. BIM-only was weak (43.5% accuracy;

Table 4
Held-out-project complete modality factorial analysis, knowledge-guided result, and recent visual comparators

Model (trainable parameters)	Accuracy	Balanced acc.	Macro-F1	Violations
Image-only CNN (10.4 k)	0.675	0.556	0.495	3.67%
BIM-only MLP (1.6 k)	0.435	0.345	0.309	17.00%
Schedule-only MLP (1.4 k)	0.870	0.815	0.825	3.17%
Image + BIM (11.8 k)	0.755	0.658	0.664	4.67%
Image + schedule (11.6 k)	0.893	0.844	0.859	3.17%
BIM + schedule (1.7 k)	0.857	0.769	0.780	3.17%
Image + BIM + schedule (11.9 k)	0.878	0.827	0.833	1.67%
Knowledge-guided (11.9 k + rules)	0.905	0.857	0.866	0.00%
Compact Swin V2 (258.8 k)	0.505	0.417	0.370	8.67%
Compact EfficientViT (48.1 k)	0.722	0.684	0.688	3.17%
Compact ConvNeXt V2 (61.9 k)	0.900	0.874	0.876	1.00%

Figure 3
Accuracy and macro-F1 for the complete seven-cell image/BIM/schedule factorial analysis plus the knowledge-guided output on held-out Project 7



0.309 macro-F1), whereas schedule-only reached 87.0% accuracy and 0.825 macro-F1. Image + BIM improved the compact image-only CNN from 67.5%/0.495 to 75.5%/0.664, but image + schedule was the strongest unconstrained factorial cell at 89.3% accuracy and 0.859 macro-F1. BIM + schedule reached 85.7%/0.780, and image + BIM + schedule reached 87.8%/0.833. Thus, BIM adds useful context to the small image-only CNN, but it does not provide an additional gain once the schedule signal is present in this synthetic generator. This negative interaction is reported explicitly rather than hidden by the original sequential presentation.

The schedule-dominant result directly reflects the benchmark construction: planned state, inspection time, and due-date relationships are closely related to variables used to generate progress state. The factorial analysis therefore confirms that the large schedule gain is primarily recovery of programmed temporal structure. It should be read as an architecture sanity check,

not as evidence that schedule variables independently predict real construction progress with comparable accuracy. The knowledge-guided output still reached 90.5% accuracy, 0.857 balanced accuracy, and 0.866 macro-F1 with zero defined predecessor violations. Among recent image-only comparators, compact ConvNeXt V2 remained strongest in macro-F1 at 0.876 and 90.0% accuracy, reinforcing that the proposed contribution is not a superior visual representation but an auditable integration of project context, explicit constraints, and review.

5.2. Knowledge-rule ablation

Table 5 and Figure 4 isolate the effects of precedence and schedule-state reasoning.

Table 5
Ablation of symbolic rules

Rule setting	Accuracy	Macro-F1	Violations
No rules	0.878	0.833	1.67%
Precedence only	0.893	0.854	0.00%
Schedule state only	0.888	0.843	1.67%
All rules	0.905	0.866	0.00%

The precedence rule provided the clearest safety-related benefit. It eliminated installed predictions when the predecessor was incomplete and raised macro-F1 from 0.833 to 0.854. The schedule-state rule alone produced a smaller improvement and did not remove predecessor violations because it had no access to that relationship. Combining all rules yielded the strongest overall result. This pattern illustrates why schedule features and knowledge rules should not be treated as interchangeable: schedule variables support statistical prediction, whereas explicit dependencies can enforce a project constraint.

5.3. Synthetic quality verification

Table 6 and Figure 5 compare the two quality-classification configurations.

The image-only quality classifier achieved precision, recall, and F1 of 0.958. The BIM-fusion model increased precision to 0.976 but reduced recall to 0.854, resulting in a lower F1 of 0.911.

Figure 4
Macro-F1 and sequence-violation rate under four symbolic-rule configurations

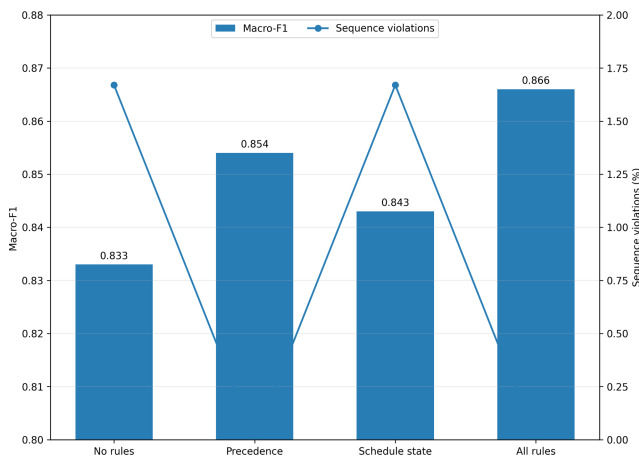
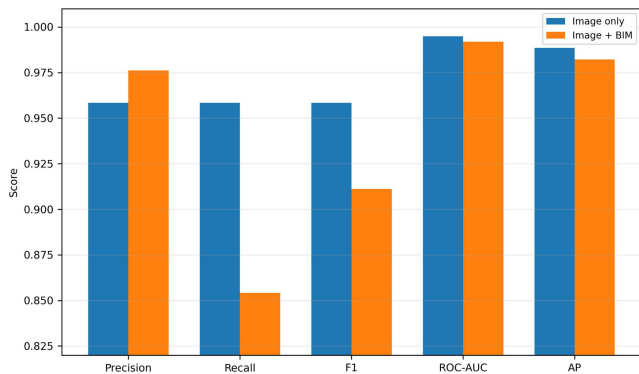


Table 6
Visible-object quality results

Model	Precision	Recall	F1	ROC-AUC	AP
Image-only quality CNN	0.958	0.958	0.958	0.995	0.988
Image + BIM fusion	0.976	0.854	0.911	0.992	0.982

Figure 5
Performance of the image-only and BIM-fusion quality classifiers on synthetic visual anomalies



ROC-AUC and average precision remained high for both models. The appropriate interpretation is not that BIM context is unhelpful to quality management. Rather, the generated anomalies were predominantly visual, and the structured BIM variables did not encode the relevant acceptance criteria. A real quality system would require tolerances, specifications, approved samples, inspection test plans, and object-specific defect definitions in addition to geometry and class.

5.4. Robustness and selective review

Figure 6 stratifies progress performance by generated occlusion and lighting conditions.

Accuracy remained between 89.4% and 91.1% for low-occlusion, moderate-occlusion, normal-lighting, and low-light

Figure 6
Knowledge-guided progress performance across occlusion and lighting subsets. The high-occlusion group contains only 22 observations

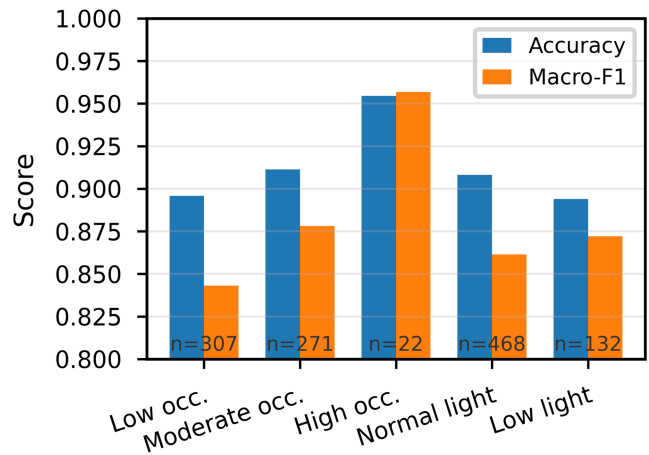
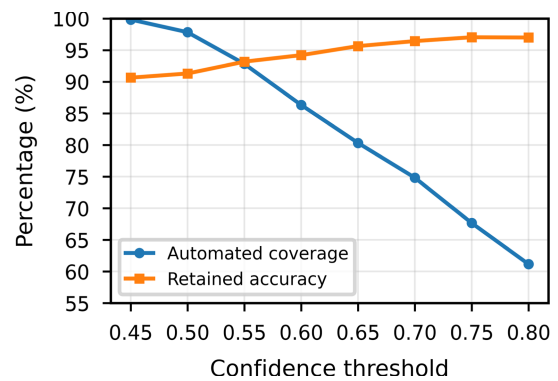


Figure 7
Trade-off between automated coverage and retained accuracy as the confidence threshold increases



subsets. The high-occlusion subset produced 95.5% accuracy, but its sample size was only 22, and the apparent increase should not be generalized. Because the synthetic occluders were simple bars rather than complex real-site obstructions, the robustness experiment establishes consistency within the benchmark, not field robustness.

Figure 7 shows how increasing the confidence threshold trades automated coverage for higher retained accuracy.

Selective review created a practical accuracy-coverage trade-off. At threshold 0.50, the system retained 97.8% of observations at 91.3% accuracy. At threshold 0.65, automated coverage was 80.3%, the review rate was 19.7%, and retained accuracy reached 95.6% with retained macro-F1 of 0.915. At threshold 0.70, retained accuracy rose to 96.4% while coverage fell to 74.8%. These values show how a project can choose an operating point based on the cost of review and the consequences of an incorrect status.

5.5. Error and comparison analysis

The knowledge-guided confusion matrix contained 215 correct not-installed predictions, 69 correct partial predictions, and 259 correct installed predictions. The main error concentration involved the partial state: 26 partial observations were classified as not installed and 14 as installed; 14 installed observations were

classified as partial and two as not installed. Figure 2 intentionally includes an actual checkpoint error (P7-0030: installed ground truth predicted as partial, confidence 0.473) so that the visual evidence panel reflects the evaluated model rather than only successful examples. The partial state remains the principal ambiguity because its generated visible extent overlaps both endpoint states.

The complete factorial comparison provides a stronger modality interpretation than the original sequential path. Schedule-only nearly matched the full schedule-aware neural model, while image + schedule exceeded image + BIM + schedule on both accuracy and macro-F1. BIM-only was weak, and adding BIM to the schedule signal did not improve the synthetic target. This pattern is consistent with the generator logic rather than evidence of a generally dominant schedule modality. The recent-backbone comparison adds a separate domain-generalization observation: compact ConvNeXt V2 extracted strong appearance features and produced the highest macro-F1, whereas compact Swin V2 showed a large validation-to-held-out decline. These results separate visual representation quality from project-information governance.

The quality-model comparison also suggests a modality-optimization issue. The BIM branch increased precision but reduced recall, indicating that the fused classifier became more conservative. Because class and dimensions were not direct causes of the synthetic anomaly, the tabular branch may have introduced class-conditional priors that discouraged some positive predictions. In a real dataset, this risk can be tested by permutation importance, class-stratified metrics, and comparison with specification-aware attributes. A multimodal model should be rejected when it reduces the metric aligned with the inspection objective.

Selective review concentrates human effort on the most uncertain cases, but confidence ranking itself must be monitored. Review outcomes should be analyzed by trade, state, camera, and project to detect systematic under-referral. The retained accuracy reported here is conditional on generated test data; operational thresholds should be recalibrated whenever the model, project domain, capture device, or class distribution changes.

6. Discussion

6.1. Interpretation of the multimodal gains

The complete factorial analysis reveals that the apparent multimodal gain is not uniformly additive. BIM attributes improve the small image-only CNN, but schedule-only already reaches 87.0% accuracy and 0.825 macro-F1, and image + schedule is the strongest unconstrained factorial cell at 89.3%/0.859. Adding BIM to image + schedule does not improve the synthetic benchmark. Because inspection time, planned state, and due-date relationships are deliberately connected to target generation, the schedule result is best interpreted as recovery of available programmed context. The recent backbone study adds another qualification: compact ConvNeXt V2 slightly exceeds the knowledge-guided method in macro-F1 using images alone. The defensible contribution is therefore the auditable information architecture—BIM linkage, explicit rule effects, probability lineage, zero defined predecessor violations, and confidence-based review—rather than a claim of superior visual modeling or independent schedule causality.

The quality experiment provides an equally important boundary condition. BIM attributes did not improve detection of

visually salient synthetic anomalies. The multimodal model sacrificed recall even though its precision increased. This outcome suggests that additional modalities should be selected from the decision requirements rather than added because they are available. For quality verification, useful nonvisual inputs may include specification clauses, permissible tolerances, method statements, approved material data, object history, and sensor measurements. Generic BIM geometry alone may be insufficient.

6.2. Implications for BIM and common data environment practice

An operational implementation would require more than a trained classifier. BIM objects must have stable identifiers, consistent classifications, verified location data, and links to schedule activities. Model revisions and approved status must be controlled in the common data environment. Visual observations require timestamps, camera metadata, spatial registration, and a record of visibility. Predecessor relationships must be maintained as governed project knowledge rather than informal assumptions. These conditions align with continuing BIM–AI integration challenges and with recent evidence that BIM/IFC data preparation, multi-technology status acquisition, and image-to-model integration are central to dependable AI deployment [15, 20, 26, 28].

The proposed pipeline can support progress dashboards, discrepancy registers, and targeted site review. It should not independently certify payment, safety, or contractual completion. A suitable workflow would capture visual evidence, associate observations with candidate BIM objects, infer progress or anomaly probabilities, apply approved rules, refer uncertain or high-risk cases, record the reviewer decision, and feed confirmed labels back into model improvement. The process therefore forms a controlled human–AI loop rather than an autonomous replacement for construction professionals.

6.3. Reliability, governance, and explainability

The symbolic layer improves a form of explainability because its effect can be expressed in project language: for example, “installed probability reduced because the mandatory predecessor is incomplete.” This is more actionable than a saliency map alone. Nevertheless, the neural component remains statistical and may respond to spurious visual cues. Confidence also requires calibration; high softmax probability is not proof of correctness [14]. Before deployment, calibration should be evaluated on project-specific validation data, and abstention thresholds should be differentiated by object criticality and decision type.

Governance should cover data ownership, consent for site imagery, cybersecurity, model versioning, rule approval, bias across trades and environments, and retention of original evidence. The pipeline should log inputs, model version, rule version, original and adjusted probabilities, confidence threshold, reviewer identity, and final disposition. These records enable challenge and correction, which are essential when AI contributes to project controls.

6.4. Limitations

The primary limitation is the synthetic nature of the benchmark. The images use simplified shapes, generated clutter, and programmed defect cues; therefore, the absolute performance values cannot be transferred to a construction site. More importantly, temporal variables used by the learned schedule branch

are closely related to the variables used to generate progress labels. The complete factorial analysis exposes this dependence: schedule-only performs strongly, and image + schedule exceeds the full image + BIM + schedule model. These values demonstrate benchmark recoverability, not independent causal evidence or expected field performance. The experiment also assumes an object-centered observation and does not solve camera localization, image-to-BIM registration, multi-object association, or temporal tracking.

The rule weights were selected as transparent proof-of-concept factors rather than calibrated from project data. Predecessor completion also participates in target generation and is supplied only to the post-model knowledge layer; consequently, elimination of the defined predecessor inconsistency demonstrates enforcement of a programmed constraint, not proof that all real sequence errors can be eliminated. The predecessor flag represents one simplified dependency, while real schedules contain multiple predecessors, leads, lags, work packages, approved exceptions, and out-of-sequence progress. The high-occlusion subgroup was small, quality labels were binary and visually dominant, and the recent architecture comparison used compact non-pretrained configurations and one fixed seed. These limitations define the next experimental stages rather than validate operational performance.

6.5. Future validation

Future work should begin with a consented, anonymized dataset from multiple projects and trades. Site images should be registered to federated BIM, and object identities should be verified independently. The test design should separate projects, devices, seasons, and contractors to quantify domain generalization. Progress labels should be produced through documented inspection criteria and inter-rater agreement. Quality experiments should use object-specific tolerances and compare visual, geometric, and textual information. Additional studies should evaluate temporal video models, vision transformers, graph neural networks over BIM relationships, calibrated uncertainty, and active learning from reviewer feedback.

6.6. Proposed field-validation protocol

A staged field-validation protocol is recommended. Stage 1 should use retrospective images from one completed zone to test registration, object association, and label definitions without influencing project decisions. Stage 2 should prospectively capture a small set of object classes with controlled viewpoints and independent inspector labels. Stage 3 should introduce multiple devices and uncontrolled capture while maintaining shadow-mode operation: the AI produces recommendations, but official records continue through the existing process. Stage 4 may permit limited automated updates for low-risk, high-confidence observations, subject to audit sampling. Contractual certification should remain outside the automated scope until legal, governance, and reliability requirements are established.

Ground truth should be object based and time stamped. At least two qualified reviewers should independently classify ambiguous states, with disagreement resolved through documented criteria. Registration accuracy and visibility should be labeled separately from progress state. Dataset partitions should be by project or physical zone, not by random frames from the same video. Reported results should include all classes, confusion matrices, calibration, abstention coverage, processing time, and failure examples. A field benchmark should also record how often

an observation cannot be associated with a unique object, since association failure is a major source of end-to-end error.

6.7. Deployment architecture and lifecycle management

The deployment architecture can be organized into capture, integration, inference, governance, and presentation services. Capture services ingest mobile, fixed-camera, drone, or robotic imagery. Integration services resolve project coordinates, candidate BIM objects, model revision, and schedule status. Inference services execute visual and tabular encoders. Governance services apply rules, confidence thresholds, mandatory-review policies, and audit logging. Presentation services display the image, projected model object, original and adjusted probabilities, rule explanation, and links to the issue-management workflow.

Model lifecycle management should define retraining triggers and rollback procedures. Drift may be caused by a new project type, different temporary works, camera upgrades, changing lighting, or a revised classification system. Monitoring should compare confidence, referral rate, class distribution, reviewer disagreement, and error sampling with the validated baseline. A new model should not replace the previous version solely because aggregate accuracy improves; it must also satisfy critical-class, calibration, and sequence-consistency criteria. Historical decisions should remain linked to the model and rule versions that produced them.

Interoperability is also central. The system should exchange object identifiers and status through open or documented interfaces rather than embedding decisions in proprietary image files. IFC can support object semantics, but project-specific property sets and schedule links require consistent mapping. Issue records may use BIM Collaboration Format or an equivalent governed schema. The goal is not to force all processing into the BIM authoring tool but to maintain an auditable link between physical evidence, AI inference, review, and authoritative project information.

6.8. Threats to validity

Internal validity is affected by programmed data-generation relationships. Progress labels are constructed from inspection time, planned start and finish, project delay, class delay, floor effect, stochastic deviation, and predecessor completion, while learned schedule inputs include planned-state indicators, inspection time, and due-date delta. The complete factorial analysis quantifies the resulting dependence: schedule-only reaches 87.0% accuracy and 0.825 macro-F1, and image + schedule reaches 89.3%/0.859. These results demonstrate architectural recovery of encoded context rather than discovery of an unknown causal mechanism. Construct validity is limited because simple raster shapes are proxies for construction components and binary anomaly labels are proxies for quality. External validity is limited to generated domains, and conclusion validity is affected by fixed-seed experiments and compact architecture choices.

The study mitigates these threats through explicit disclosure of the generator, exact sample identifiers and model inputs, held-out-domain evaluation, complete modality factorial analysis, rule ablation, multiple metrics, an intentionally displayed checkpoint error, and disclosure of negative findings. It does not claim that the reported performance will persist on site. The defensible contribution is the evaluated, auditable information architecture and the direction of effects within a controlled benchmark.

Table 7
Recommended stages for field validation

Stage	Primary objective	Permitted decision use
1. Retrospective pilot	Test registration, association, and labels	Research analysis only
2. Controlled prospective	Validate selected classes and viewpoints	Inspector comparison
3. Shadow operation	Test uncontrolled capture and drift	No automatic official update
4. Limited automation	Automate low-risk, high-confidence cases	Audited project records

Independently labeled field evidence is required to estimate operational accuracy, causal value of schedule context, and economic benefit.

Table 7 converts the proposed field-validation process into four progressively more consequential stages.

6.9. Operational utility and economic evaluation

Classification accuracy alone does not establish project value. Operational evaluation should measure the time required to capture, associate, review, and close each observation, the number of site visits avoided, the delay between physical work and verified status, and the rate at which automated findings lead to corrective action. The correct comparator is the existing progress or inspection process, including its labor, travel, rework, and reporting delay. A system with lower classification accuracy may still be useful if it rapidly prioritizes the right areas for human inspection, while a highly accurate classifier may offer little value if object association and review consume more effort than the manual process.

Economic analysis should distinguish one-time setup from recurring operation. Setup includes BIM data preparation, schedule linking, camera planning, integration, validation, and training. Recurring cost includes capture, computation, storage, reviewer time, maintenance, and retraining. Benefits may include reduced inspection effort, earlier discrepancy detection, improved evidence for coordination, more consistent progress records, and reduced rework. These benefits should be quantified cautiously and should not be inferred from the synthetic accuracy values in this study.

A useful field trial can compare matched zones or reporting periods. One group follows the conventional process, while another receives AI-assisted prioritization without removing required inspections. Outcomes should include reporting cycle time, reviewer workload, disagreement rate, unresolved observations, correction lead time, and user acceptance. Qualitative feedback is important because reviewers may reject a technically accurate system if its explanations, images, or model links are difficult to verify. The most defensible early use case is decision support and evidence organization rather than autonomous certification.

6.10. Responsible AI and minimum audit record

Construction images may contain workers, visitors, vehicle registrations, security arrangements, and proprietary methods. Responsible deployment therefore requires a documented capture purpose, access control, retention period, and masking process where personal or security-sensitive content is not needed. Dataset reuse for model training should be authorized separately from routine project documentation. Cloud processing and remote access should follow project cybersecurity requirements, especially when model geometry and schedule status are commercially sensitive.

Human oversight must be substantive rather than ceremonial. Reviewers need access to the original image, the associated BIM object, relevant schedule information, rule explanation, and a simple mechanism to correct the output. Corrections should not be treated automatically as ground truth until their quality is checked; otherwise, rushed review can create noisy feedback loops. Appeals and exception handling are also needed for legitimate out-of-sequence work, temporary installations, and approved design changes.

Table 8 specifies the minimum evidence, version, probability, threshold, and review fields needed for auditability.

Table 8
Minimum fields in an auditable AI verification record

Record field	Purpose
Evidence identifier and hash	Preserve the original source and detect alteration
Capture time, device, and view	Establish temporal and visual context
BIM object and model revision	Link the decision to authoritative geometry and data
Schedule activity and revision	Identify the planned state used by the system
Model and rule-set versions	Reproduce neural and symbolic processing
Original/adjusted probabilities	Show the effect of knowledge guidance
Threshold and review status	Explain automation or abstention
Reviewer decision and reason	Maintain accountability and feedback

The audit record should be immutable or revision controlled and should survive changes to dashboards or model versions. It enables error investigation, supports governance reviews, and separates the evidential history from later reinterpretation. Responsible AI in this setting is therefore not limited to model explainability; it also depends on information management, human authority, security, and the ability to reconstruct how a project status was produced.

7. Conclusions

This study presented a seven-stage knowledge-guided multimodal method combining visual evidence, BIM attributes, schedule state, explicit construction rules, and confidence-based human review. The revision adds a complete 2^3-1 factorial analysis. BIM-only achieved 43.5% accuracy and 0.309 macro-F1; schedule-only achieved 87.0% and 0.825; image + BIM achieved 75.5% and 0.664; image + schedule achieved 89.3% and 0.859;

BIM + schedule achieved 85.7% and 0.780; and image + BIM + schedule achieved 87.8% and 0.833. The knowledge-guided output achieved 90.5% accuracy, 0.866 macro-F1, and zero defined predecessor violations.

The factorial results materially narrow the claim. Schedule information is highly predictive in this benchmark because temporal relationships are deliberately embedded in the target generator; therefore, the schedule gain is evidence that the architecture recovers available benchmark context, not evidence of an independently discovered causal mechanism or expected field-level improvement. BIM helps the compact image-only CNN but does not improve the image + schedule model in this synthetic setting. Compact ConvNeXt V2 remains a strong image-only comparator at 90.0% accuracy and 0.876 macro-F1. At a confidence threshold of 0.65, the knowledge-guided method processes 80.3% of observations automatically at 95.6% retained accuracy, while the separate quality task again shows that BIM fusion is beneficial only when structured variables are relevant to the target.

The evidence remains a simulation-based proof of concept whose principal contribution is an auditable information and governance architecture rather than demonstrated field performance or a universally superior visual model. Field deployment requires independently labeled and registered site imagery, verified BIM-object association, project-calibrated rules, development-only preprocessing, multi-seed and multi-project evaluation, uncertainty calibration, and professional oversight. Within those limits, the method provides a transparent and reproducible basis for studying how visual evidence, project information, symbolic constraints, and governed human review can be combined.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by the author.

Conflicts of Interest

The author declares that he has no conflicts of interest related to this work.

Data Availability Statement

The data that support the findings of this study are openly available in Zenodo at <https://doi.org/10.5281/zenodo.22122960>.

Author Contribution Statement

Mohammed Hijazi: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration.

References

- [1] Golparvar-Fard, M., Peña-Mora, F., & Savarese, S. (2015). Automated progress monitoring using unordered daily construction photographs and IFC-based building information models. *Journal of Computing in Civil Engineering*, 29(1), 04014025. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000205](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000205)
- [2] Reja, V. K., Varghese, K., & Ha, Q. P. (2022). Computer vision-based construction progress monitoring. *Automation in Construction*, 138, 104245. <https://doi.org/10.1016/j.autcon.2022.104245>
- [3] Pal, A., Lin, J. J., Hsieh, S.-H., & Golparvar-Fard, M. (2023). Automated vision-based construction progress monitoring in built environment through digital twin. *Developments in the Built Environment*, 16, 100247. <https://doi.org/10.1016/j.dibe.2023.100247>
- [4] Wei, W., Lu, Y., Zhong, T., Li, P., & Liu, B. (2022). Integrated vision-based automated progress monitoring of indoor construction using mask region-based convolutional neural networks and BIM. *Automation in Construction*, 140, 104327. <https://doi.org/10.1016/j.autcon.2022.104327>
- [5] Yang, J., Wilde, A., Menzel, K., Sheikh, M. Z., & Kuznetsov, B. (2023). Computer vision for construction progress monitoring: A real-time object detection approach. In L. M. Camarinha-Matos, X. Boucher, & A. Ortiz (Eds.), *Collaborative networks in digitalization and society 5.0* (pp. 660–672). Springer. https://doi.org/10.1007/978-3-031-42622-3_47
- [6] Ekanayake, B., Wong, J. K. W., Fini, A. A. F., Smith, P., & Thengane, V. (2024). Deep learning-based computer vision in project management: Automating indoor construction progress monitoring. *Project Leadership and Society*, 5, 100149. <https://doi.org/10.1016/j.plas.2024.100149>
- [7] Chua, W. P., & Cheah, C. C. (2024). Deep-learning-based automated building construction progress monitoring for prefabricated prefinished volumetric construction. *Sensors*, 24(21), 7074. <https://doi.org/10.3390/s24217074>
- [8] Magdy, R., Hamdy, K. A., & Essawy, Y. A. (2025). Real-time progress monitoring of bricklaying. *Buildings*, 15(14), 2456. <https://doi.org/10.3390/buildings15142456>
- [9] Sami Ur Rehman, M., Shafiq, M. T., & Ullah, F. (2022). Automated computer vision-based construction progress monitoring: A systematic review. *Buildings*, 12(7), 1037. <https://doi.org/10.3390/buildings12071037>
- [10] Samsami, R. (2024). A systematic review of automated construction inspection and progress monitoring (ACIPM): Applications, challenges, and future directions. *CivilEng*, 5(1), 265–287. <https://doi.org/10.3390/civileng5010014>
- [11] Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), 423–443. <https://doi.org/10.1109/TPAMI.2018.2798607>
- [12] Xu, P., Zhu, X., & Clifton, D. A. (2023). Multimodal learning with transformers: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10), 12113–12132. <https://doi.org/10.1109/TPAMI.2023.3275156>
- [13] Geifman, Y., & El-Yaniv, R. (2017). Selective classification for deep neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 4885–4894.
- [14] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International Conference On Machine Learning*, 1321–1330.
- [15] Hasan, S., & Sacks, R. (2023). Integrating BIM and multiple construction monitoring technologies for acquisition of project status information. *Journal of Construction Engineering and Management*, 149(7), 04023051. <https://doi.org/10.1061/JCEMD4.COENG-12826>
- [16] Kavaliauskas, P., Fernandez, J. B., McGuinness, K., & Jurelionis, A. (2022). Automation of construction progress

- monitoring by integrating 3D point cloud data with an IFC-based BIM model. *Buildings*, 12(10), 1754. <https://doi.org/10.3390/buildings12101754>
- [17] Meyer, T., Brunn, A., & Stilla, U. (2022). Change detection for indoor construction progress monitoring based on BIM, point clouds and uncertainties. *Automation in Construction*, 141, 104442. <https://doi.org/10.1016/j.autcon.2022.104442>
- [18] Vassena, G. P. M., Perfetti, L., Comai, S., Mastrolembro Ventura, S., & Ciribini, A. L. C. (2023). Construction progress monitoring through the integration of 4D BIM and SLAM-based mapping devices. *Buildings*, 13(10), 2488. <https://doi.org/10.3390/buildings13102488>
- [19] Spinner, A., & Degani, A. (2024). Online as-built building information model update for robotic monitoring in construction sites. *Journal of Intelligent & Robotic Systems*, 110(2), 50. <https://doi.org/10.1007/s10846-024-02087-2>
- [20] Choi, W., Na, S., & Heo, S. (2024). Integrating drone imagery and AI for improved construction site management through building information modeling. *Buildings*, 14(4), 1106. <https://doi.org/10.3390/buildings14041106>
- [21] Yoon, S., & Kim, H. (2025). Time-series image-based automated monitoring framework for visible facilities: Focusing on installation and retention period. *Sensors*, 25(2), 574. <https://doi.org/10.3390/s25020574>
- [22] Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., . . . , & Guo, B. (2022). Swin transformer v2: Scaling up capacity and resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11999–1200. <https://doi.org/10.1109/CVPR52688.2022.01170>
- [23] Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I. S., & Xie, S. (2023). ConvNeXt v2: Co-designing and scaling convNets with masked autoencoders. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16133–16142. <https://doi.org/10.1109/CVPR52729.2023.01548>
- [24] Liu, X., Peng, H., Zheng, N., Yang, Y., Hu, H., & Yuan, Y. (2023). EfficientViT: Memory efficient vision transformer with cascaded group attention. In *2023 IEEE/CVF Conference On Computer Vision And Pattern Recognition*, 14420–14430. <https://doi.org/10.1109/CVPR52729.2023.01386>
- [25] Han, J., Zhou, X., Zhang, W., Guo, Q., Wang, J., & Lu, Y. (2022). Directed representative graph modeling of MEP systems using BIM data. *Buildings*, 12(6), 834. <https://doi.org/10.3390/buildings12060834>
- [26] Du, S., Hou, L., Zhang, G., Tan, Y., & Mao, P. (2024). BIM and IFC data readiness for AI integration in the construction industry: A review approach. *Buildings*, 14(10), 3305. <https://doi.org/10.3390/buildings14103305>
- [27] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>
- [28] Khan, A. A., Bello, A. O., Arqam, M., & Ullah, F. (2024). Integrating building information modelling and artificial intelligence in construction projects: A review of challenges and mitigation strategies. *Technologies*, 12(10), 185. <https://doi.org/10.3390/technologies12100185>

How to Cite: Hijazi, M. (2026). Knowledge-Guided Multimodal Deep Learning for BIM-Based Construction Progress and Quality Verification from Site Images and Video. *Artificial Intelligence and Applications*. <https://doi.org/10.47852/bonviewAIA620211089>