*applied sciences*

MDPI

*Article*

# Constraint-Oriented YOLOv7 Optimization for Panoramic Object Detection: Representation, Crowded-Scene Detection, and Deployment-Oriented Model Compression

**Qiang Zhang [1,2], Jianjing Shen [1,2,*], Chenyang He [3], Zeqi Xie [1,2] and Yun Tie [4]**

[1] Faculty of Engineering, Zhengzhou SIAS University, Zhengzhou 451150, China; 22367@sias.edu.cn (Q.Z.); 11616@sias.edu.cn (Z.X.)
[2] Henan Provincial Engineering Research Center on Digital Twin Technology for Intelligent Manufacturing, Zhengzhou SIAS University, Zhengzhou 451150, China
[3] School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou 450001, China; mo2n0221@gs.zzu.edu.cn
[4] School of Cyber Science and Engineering, Zhengzhou University, Zhengzhou 450001, China; ieytie@zzu.edu.cn
* Correspondence: shenjj@sias.edu.cn

**Abstract**

Panoramic object detection is constrained at three levels of the detection pipeline: equirectangular projection deforms the input representation, crowded scenes weaken feature discrimination and complicate overlapping predictions, and high-resolution imagery increases the computational cost. We organized these challenges in a constraint-oriented framework comprising representation-, detection-, and computation-level branches around a common YOLOv7 reference architecture. The branches were independently configured and evaluated rather than cumulatively optimized as a single detector. At the representation level, spherical rotation augmentation, adaptive sampling, routed attention, and detail-preserving transformation yielded an 86.9% $mAP_{50}$ and a 62.7% $mAP_{50:95}$ on full PanoWe, increases of 1.8 and 0.9 percentage points over the branch-specific YOLOv7 reference, while the recall increased by 3.0 points and the precision decreased by 1.1 points. At the detection level, strengthened cross-scale interaction, overlap-aware post-processing, and a normalized Wasserstein distance (NWD) localization term yielded an 86.2% $mAP_{50}$, a 64.1% $mAP_{50:95}$, and a 91.1% recall on a manually curated 1264-image crowded-scene subset, with a 1.1-point precision decrease relative to its reference. At the computation level, partial convolution and layer-adaptive magnitude pruning retained an 83.8% $mAP_{50}$ with 19.4 million parameters and 43.0 theoretical GFLOPs, reductions of 47.4% and 58.9% at a cost of 1.3 $mAP_{50}$ points. These preserved single-run results support a stage-aligned analysis of the accuracy, recall, and model-complexity trade-offs; they do not establish additive gains across branches or statistically significant superiority. The throughput measured on the recorded hardware platform and an external evaluation on the public 360-Indoor benchmark are additionally reported to characterize the deployment behavior and generalization beyond the in-house collection.

**Keywords:** panoramic image; object detection; geometric distortion; crowded-scene detection; model compression; network pruning

Check for updates

Academic Editors: Thomas Lindner and João M. F. Rodrigues

Received: 31 July 2026
Revised: 28 October 2026
Accepted: 29 October 2026
Published: 30 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

Panoramic images capture an approximately 360° horizontal and 180° vertical field of view and therefore provide substantially more scene context than perspective images.

This coverage is useful in autonomous driving, aerial monitoring, and panoramic scene understanding [1–4]. Applying a conventional detector directly to panoramic imagery, however, exposes constraints at three levels of its pipeline: nonuniform geometric distortion affects representation, small and overlapping objects challenge feature interaction and prediction, and high-resolution inputs increase the computational cost. These levels provide the organizing structure of this study. Figure 1 illustrates the representation-level mismatch between a fixed convolutional grid and a learned deformable pattern on the same ERP scene.



(**a**) (**b**)

**Figure 1.** Sampling locations on the same ERP panoramic scene. The regular grid remains spatially uniform, whereas the deformable pattern moves sampling points toward the distorted object support. (**a**) Regular convolution with a fixed $3 \times 3$ sampling grid. (**b**) Deformable convolution with learned sampling offsets.

The first problem arises from equirectangular projection (ERP), the most common format for storing and processing panoramic images. Mapping a sphere onto a plane causes latitude-dependent stretching, especially near the poles [3,5,6]. This spatial deformation conflicts with the translation-invariance assumption of standard convolutional neural networks (CNNs), so kernels with fixed sampling positions may not follow the contours of distorted objects.

The second problem is the loss of information about small objects. Because panoramic images cover a large field of view at a high resolution, many objects occupy only a small fraction of the image and frequently overlap. Geometric distortion and repeated network downsampling can suppress their fine textures and high-frequency boundary cues [7]. Once these signals have decayed during cross-level propagation, conventional feature fusion may not recover enough information for reliable classification and localization [8,9].

The third problem concerns the computational complexity. High-resolution panoramic inputs require considerable computational and memory resources. Increasing the network depth or adding attention modules may improve the accuracy, but they also increase the model size and arithmetic cost. Reducing parameters and theoretical floating-point operations (FLOPs) without a disproportionate loss of detection accuracy is therefore an important model-compression objective. The latency obtained on a deployment platform additionally depends on the execution library and on whether the induced sparsity is supported by the target hardware, so model-complexity reduction and throughput measured on the recorded platform are reported separately in this study.

Previous studies usually concentrated on one constraint level. Spherical CNNs and projection-specific operators address representation through geometry-specific sampling, conversion, or reprojection [5,10,11]; crowded-object methods strengthen multiscale interactions or prediction processing; and efficient detectors reduce operator or parameter redundancy. Treating these directions only as a list of modules obscures where each intervention acts and which trade-off it is intended to control. We therefore ask: *How should a detector be adapted at each of the representation, detection, and computation levels to address the principal constraints of panoramic object detection, and what level-specific benefits and costs are supported by*

*the available experiments?* The corresponding questions test (i) whether projection-oriented transformations improve aggregate detection, (ii) whether stronger interaction and overlap-aware post-processing favor recall in crowded scenes, and (iii) whether partial convolution and pruning reduce the complexity with limited accuracy loss. The three levels are organized by a common constraint-to-stage design logic; the corresponding configurations are trained and evaluated independently, and their gains are not assumed to be additive.

As illustrated in Figure 2, the study organizes distortion-oriented representation, crowded-scene detection, and model compression as three levels organized by one constraint-oriented design logic. The main contributions are as follows:

- We formulated a stage-aligned mapping from panoramic-detection constraints to representation-, detection-, and computation-level interventions. This mapping supplies a common rationale and comparison vocabulary; the three configurations are evaluated independently and their gains are not assumed to be additive.
- At the representation level, we adapted published spherical augmentation, deformable sampling, routed attention, and detail-preserving downsampling to an ERP-preserving YOLOv7 configuration and analyzed its aggregate accuracy, recall, precision, and feature-computation trade-off on full PanoWe.
- At the detection level, we combined Star-based interactions, gather-and-distribute fusion, and soft-NMS and quantified the resulting recall–precision trade-off on a manually curated 1264-image crowded-scene subset. The evidence is explicitly limited to aggregate subset metrics rather than size-specific AP.
- At the computation level, we combined partial convolution and layer-adaptive magnitude pruning (LAMP), recovering both cumulative and pruning-sensitivity evidence to characterize the accuracy–parameter–FLOP trade-off on full PanoWe rather than inferring the unmeasured device latency.

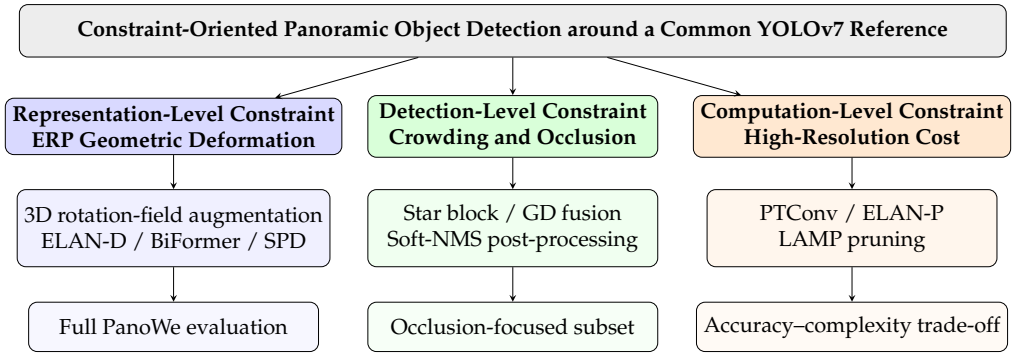



**Figure 2.** Constraint-oriented framework for panoramic object detection. The common research logic maps representation-, detection-, and computation-level constraints to stage-aligned interventions and outcomes. The three configurations were evaluated independently; arrows do not denote cumulative training across branches.

## 2. Related Work

### 2.1. Panoramic Object Detection and Geometry-Aware Representation

ERP is widely used for panoramic imagery, but its nonlinear spherical-to-planar mapping produces nonuniform geometric distortion. Geometry-aware approaches perform feature learning with spherical sampling or alternative spherical and polyhedral representations [3,5,10]. ERP-based architectures instead retain the native projection and explicitly account for its boundary discontinuity and latitude-dependent deformation [4,6].

Data-driven geometry modeling provides another approach. Deformable convolution adjusts sampling locations to match local shape changes [12,13], while distortion-aware filters modify receptive fields according to the panoramic projection [6]. Transformer-based

models use long-range self-attention to connect spatially separated regions [4,14,15]. These studies provide complementary solutions, but optimize different parts of the panoramic-detection pipeline. Geometry-specific sampling offers a projection prior, but can require specialized operators; alternative spherical or polyhedral representations reduce ERP deformation at the cost of representation conversion; ERP-native transformers model the long-range structure, but add computational cost. The present study does not claim a new projection operator. Instead, it examines a task-specific integration of local adaptive sampling, routed contextual interaction, and detail-preserving downsampling while retaining the ERP representation.

### 2.2. Crowded-Scene and Occlusion-Aware Detection

Small objects occupy few pixels in panoramic images and are often densely distributed or occluded. Feature pyramid networks combine shallow spatial detail with deeper semantic features and have become a standard component of object detectors [9]. Across conventional small-object detection and infrared small-target detection, later work has used feature enhancement, contextual reasoning, and attention to amplify weak target responses [8,16,17].

Prediction-stage optimization is also relevant to crowded scenes. Conventional non-maximum suppression (NMS) can remove valid detections when objects overlap. Soft-NMS decays scores instead of discarding boxes at a fixed overlap threshold [18]. The normalized Wasserstein distance (NWD) models bounding boxes as Gaussian distributions and is less sensitive than the intersection over union (IoU) to small coordinate shifts [19]. These mechanisms complement multiscale feature fusion by acting on localization and overlapping predictions. NWD was adopted in the detection-level branch described in Section 3.4, where it complemented soft-NMS by making the localization term less sensitive to small box displacements.

### 2.3. Efficient Object Detection and Model Compression

Efficient detectors are needed for mobile robots, aerial inspection, and other resource-constrained systems. One line of work designs lightweight operators, including depthwise separable and partial convolution, to reduce the cost of feature extraction [20,21]. Another compresses trained networks by pruning weights or allocating sparsity across layers [22]. These approaches address different forms of redundancy, although aggressive compression can reduce the accuracy. Panoramic detection places additional pressure on memory and computation because both the input resolution and the scene complexity are high.

Table 1 summarizes the distinct scopes of representative approaches. The literature provides strong objective-specific mechanisms, but their positions in the panoramic-detection pipeline and their different accuracy, recall, and efficiency criteria are not naturally expressed by a single module-level comparison. This motivates the constraint-oriented view used here: geometry handling is assigned to representation, crowded-scene interaction and post-processing to detection, and redundancy reduction to computation. The view is an organizing and design framework, not a claim that all mechanisms have been jointly optimized.

**Table 1.** Scope of representative method families relevant to this study. This is a methodological comparison rather than a common-protocol accuracy ranking.

| Method Family | Representative Studies | Primary Mechanism | Limitation Motivating the Present Study |
| --- | --- | --- | --- |
| Spherical or alternative representations | SphereNet, SpherePHD, and spherical/polyhedral learning [3,5,10] | Projection-specific sampling or a non-ERP feature representation | Specialized geometry or representation conversion can complicate the reuse of a conventional ERP detector. |

**Table 1.** *Cont.*

| Method Family | Representative Studies | Primary Mechanism | Limitation Motivating the Present Study |
| --- | --- | --- | --- |
| ERP-native distortion modeling | Distortion-aware filters and PanoSwin [4,6] | Latitude-aware filtering, boundary modeling, or panoramic attention | Primarily addresses geometric representation; computational and crowded-scene objectives remain separate. |
| Small- or crowded-object detection | Feature pyramids, contextual enhancement, soft-NMS, and NWD [9,16,18,19] | Multiscale fusion, contextual interaction, and overlap-aware prediction processing | Evidence on conventional or specialized small-target data does not establish size-specific gains on panoramic data. |
| Efficient detection and compression | Mobile operators, partial convolution, and LAMP [20–22] | Operator-level redundancy reduction and weight pruning | Parameter and FLOP reductions do not necessarily translate into device-level acceleration. |

## 3. Proposed Method

This section presents a constraint-oriented framework for adapting a conventional detector to panoramic imagery. The framework maps each identified constraint to the stage at which it enters the pipeline, the intervention used to address it, and the outcome used to evaluate it. YOLOv7 provides the common reference architecture. The three branches instantiate the framework independently because the preserved experiments do not contain a jointly trained configuration.

### *3.1. Overall Architecture*

Figure 2 summarizes the framework. Its unifying principle is stage alignment: a representation constraint is addressed before and within feature extraction, a detection constraint is addressed through feature interaction and prediction processing, and a computation constraint is addressed through operator redesign and weight pruning. Each branch starts from a YOLOv7 reference and produces a constraint-specific trade-off. The branches are conceptually connected by this mapping, but empirically parallel rather than a serial network.

### *3.2. Framework Integration and Experimental Scope*

The framework is integrated at the level of problem formulation rather than model weights. Spherical rotation, ELAN-D, BiFormer, and SPD address how ERP content is represented; Star blocks, GD fusion, and soft-NMS address how weak and overlapping evidence is propagated and retained; and ELAN-P and LAMP address how computations and parameters are allocated. This stage–constraint–outcome mapping gives the three branches a common design language and enables their trade-offs to be interpreted together.

Their experimental evidence nevertheless remains independent. A future combined implementation is technically possible because the interventions act at partly separable stages, but additivity cannot be assumed. DCNv4, routed attention, GD fusion, and SPD can increase memory access or the arithmetic cost and therefore counteract the compression objective. Conversely, unstructured pruning may remove weights important to routing or adaptive sampling unless sparsity and fine-tuning are re-optimized. Soft-NMS can be attached without retraining, but changes the precision–recall operating point. Accordingly, the framework guides configuration selection rather than representing a validated all-in-one detector: representation-level adaptation is the relevant branch when ERP deformation is primary, the detection-level branch when crowded-scene recall is primary, and the computation-level branch when the model size and theoretical arithmetic cost dominate.

### *3.3. Representation-Level Branch: Distortion-Oriented Learning*

This strategy combines projection-aware data augmentation, deformable sampling, sparse global interaction, and detail-preserving downsampling.

**Three-dimensional rotation-field augmentation.** An ERP image is first mapped from pixel coordinates to the unit sphere. The augmentation formulation permits Euler rotations about the longitude and latitude axes, followed by reprojection and bilinear resampling. The preserved implementation record states that reflection padding was used when resampling outside the finite image array; this choice concerns the resampling step of the rotation augmentation. Continuous periodic modeling of the ERP seam—circular padding along the longitude axis and seam-consistent handling of boxes that cross the boundary—constitutes a distinct line of spherical-representation research [3,4,10], which this study does not duplicate; seam-crossing objects were consequently not evaluated separately here. The example panels preserved for Figure 3 show longitude-rotated projections, but their exact angle metadata were not retained.




**Figure 3.** Representative three-dimensional rotation-field augmentation. (**a**) Original ERP panorama; (**b**–**d**) preserved examples obtained after longitude rotation and reprojection. The numerical angles used to create these legacy examples were not retained in the available experimental record.

**ELAN-D with a dynamic receptive field.** A standard convolution samples a fixed regular grid and cannot adapt to the latitude-dependent deformation of ERP images. We replaced selected standard convolutions with DCNv4 [23] to construct the ELAN-D module shown in Figure 4. A simplified form of deformable aggregation at output position $p_0$ is

$$y(p_0) = \sum_{p_n \in R} w(p_n) x(p_0 + p_n + \Delta p_n) \Delta m_n, \tag{1}$$

where $R$ is the regular sampling grid, $w(p_n)$ denotes the kernel weight, and $\Delta p_n$ and $\Delta m_n$ are the learned sampling offset and modulation factor, respectively. The learned offsets allow the receptive field to follow distorted object contours instead of remaining fixed on the ERP grid.

**Bi-level routing attention.** The long-range context can reconnect features separated by local deformation, but global attention has a quadratic complexity. Let a feature map be partitioned into $S$ non-overlapping regions, with region descriptors $\mathbf{Q}^r, \mathbf{K}^r \in \mathbb{R}^{S \times d_r}$. BiFormer computes the region-level affinity matrix $\mathbf{A}^r = \mathbf{Q}^r (\mathbf{K}^r)^T \in \mathbb{R}^{S \times S}$ and retains $k$ routed region indices per query, $\mathbf{I}^r = \text{topkIndex}(\mathbf{A}^r) \in \mathbb{N}^{S \times k}$ [24]. For token tensors

$\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N \times d}$, gather$(\cdot, \mathbf{I}^r)$ selects tokens belonging to the routed regions, and fine-grained interaction is restricted to those tokens:

$$\mathbf{O} = \text{Attention}(\mathbf{Q}, \text{gather}(\mathbf{K}, \mathbf{I}^r), \text{gather}(\mathbf{V}, \mathbf{I}^r)) + \text{LCE}(\mathbf{V}). \quad (2)$$

Here, Attention denotes the scaled dot-product attention and LCE denotes the local context enhancement term implemented with a depthwise convolution on $\mathbf{V}$. The output $\mathbf{O} \in \mathbb{R}^{N \times d}$ has the same token and channel dimensions as $\mathbf{Q}$. This sparse routing establishes long-range dependencies without applying dense global attention to every token pair.

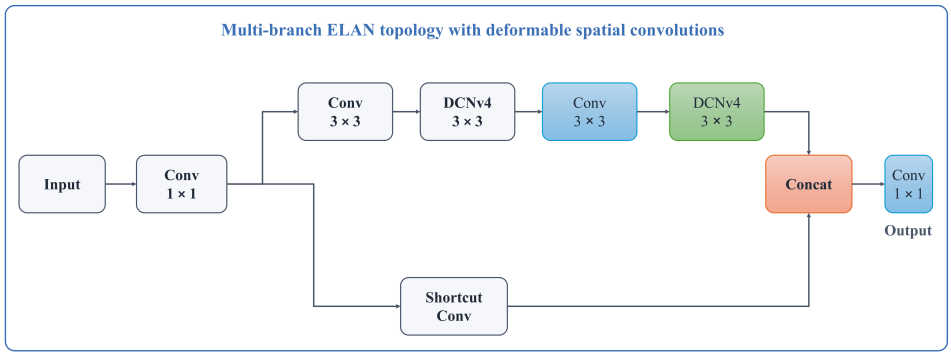



**Figure 4.** ELAN-D module. Blue blocks denote standard feature transformations, and the highlighted blocks replace selected spatial convolutions with deformable convolution. Multi-branch features are concatenated before the output projection.

**Space-to-depth transformation.** Repeated strided operations can erase the high-frequency boundaries of small objects. We therefore used a space-to-depth (SPD) transformation [7]. For $X \in \mathbb{R}^{H \times W \times C}$ and downsampling factor $s$, submaps were extracted as $f_{x,y} = X[x : H : s, y : W : s]$ and concatenated along the channel dimension. The output has a spatial size $(H/s) \times (W/s)$ and $Cs^2$ channels. The transformation rearranges local values rather than discarding them, preserving information that would otherwise be lost during downsampling. Figure 5 shows the BiFormer hierarchy and the spatial-to-channel rearrangement used in this stage.

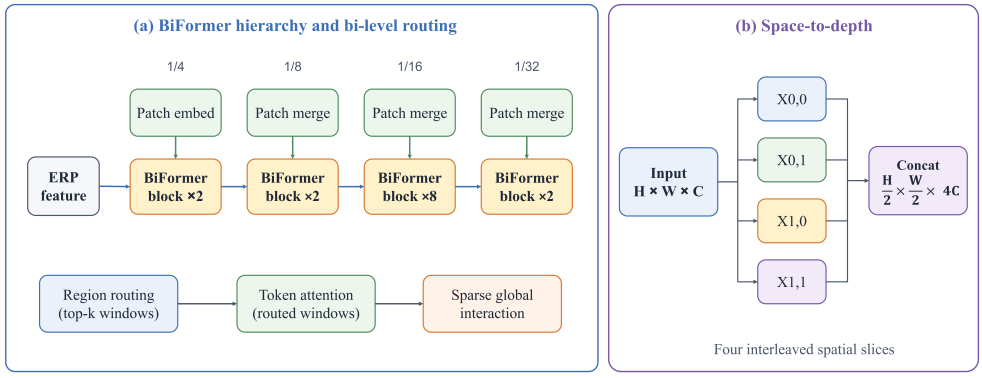



**Figure 5.** Two components of the distortion-oriented detector: (**a**) sparse region-to-region and token-to-token routing in BiFormer and (**b**) rearrangement of spatial samples into the channel dimension in SPD.

### *3.4. Detection-Level Branch: Occlusion-Focused Interaction*

This strategy targets crowded scenes containing overlapping and partially occluded objects through nonlinear feature interaction and direct cross-scale information transfer. It was evaluated independently from the distortion-oriented strategy, and its evidence is limited to aggregate metrics on the condition-selected subset described in Section 4.1.

**Feature interaction with the Star Operation.** The backbone uses the element-wise Star Operation shown in Figure 6a [25]. Given a feature tensor $\mathbf{F}$, two parallel mappings are multiplied as

$$\Phi(\mathbf{F}) = (\mathbf{W}_1\mathbf{F} + \mathbf{B}_1) \odot (\mathbf{W}_2\mathbf{F} + \mathbf{B}_2). \tag{3}$$

The multiplication introduces second-order cross terms between feature channels. Stacking these blocks increases the order of implicit interactions without explicitly constructing the corresponding high-dimensional tensor.

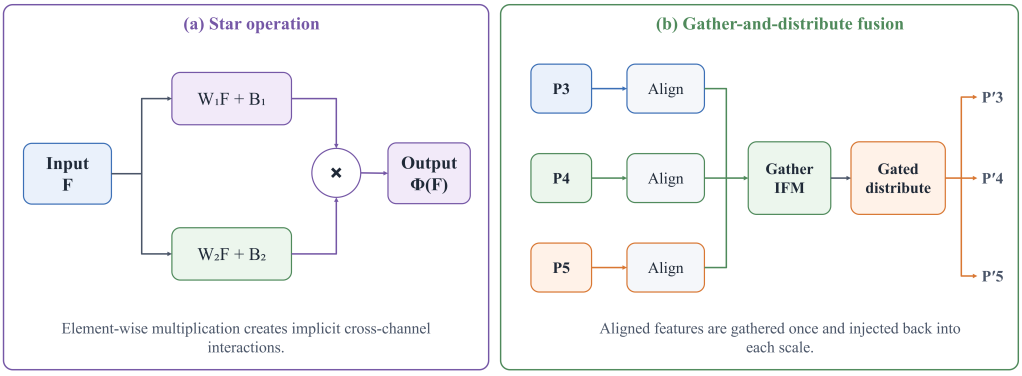



**Figure 6.** Occlusion-focused feature interaction. (**a**) The Star Operation multiplies two learned mappings to introduce implicit cross-channel interactions. (**b**) The GD mechanism aligns pyramid features, gathers a global representation, and distributes gated information back to each scale.

**Gather-and-distribute aggregation.** Let $\mathbf{F}_i \in \mathbb{R}^{H_i \times W_i \times C_i}$ denote the $i$th pyramid feature. To avoid attenuation caused by sequential top-down or bottom-up propagation, the gather-and-distribute (GD) mechanism shown in Figure 6b uses $\mathcal{T}_i$ to resize each input by average pooling or bilinear interpolation and project it to a common spatial size $H_a \times W_a$ and channel width $C_a$. The aligned tensors are concatenated and processed by the information fusion mapping $\mathcal{H}_{\text{IFM}}$ [26]:

$$\mathbf{F}_{\text{global}} = \mathcal{H}_{\text{IFM}}(\text{Concat}(\mathcal{T}_1(\mathbf{F}_1), \ldots, \mathcal{T}_n(\mathbf{F}_n))). \tag{4}$$

The fused tensor $\mathbf{F}_{\text{global}} \in \mathbb{R}^{H_a \times W_a \times C_g}$ is split into scale-specific global features. In the distribute phase, $\mathcal{H}^i_{\text{inject}}$ resizes and projects the corresponding global feature to $H_i \times W_i \times C_i$, $\sigma$ is a sigmoid gate, and $\odot$ is element-wise multiplication:

$$\mathbf{F}'_i = \mathbf{F}_i + \mathbf{F}_i \odot \sigma\Big(\mathcal{H}^i_{\text{inject}}(\mathbf{F}_{\text{global}})\Big). \tag{5}$$

This direct path allows a high-level context to modulate fine-resolution features without passing through every intermediate pyramid level.

**Overlap-aware post-processing.** We replaced hard non-maximum suppression (NMS) with soft-NMS to reduce the immediate removal of valid overlapping boxes [18]. Soft-NMS decays a candidate score as a function of its overlap with a higher-scoring box instead of discarding it at a single overlap threshold. The corresponding preserved ablation is therefore interpreted as a post-processing change that shifts the precision–recall operating point, rather than as evidence of improved learned feature representations. A normalized Wasserstein distance (NWD) term [19] is additionally applied to the localization loss. When modeling two boxes $A = (c^a_x, c^a_y, w_a, h_a)$ and $B = (c^b_x, c^b_y, w_b, h_b)$ as two-dimensional Gaussians, their similarity is

$$\text{NWD}(A, B) = \exp\left(-\frac{\sqrt{W_2^2(A, B)}}{C}\right), \tag{6}$$

where $W_2^2$ is the second-order Wasserstein distance between the two Gaussians and $C$ is a dataset-level scale constant. The localization loss becomes $\mathcal{L} = \lambda\,\mathcal{L}_{\mathrm{NWD}} + (1-\lambda)\,\mathcal{L}_{\mathrm{IoU}}$. Because NWD varies smoothly with small displacements, it stabilizes assignment for small objects, for which a displacement of a few pixels can drive IoU to zero. $C$ and $\lambda$ are constants fixed for a given dataset and training configuration; the exact numerical values used in the original run are not retained in the archive, so the NWD term is reported as part of the preserved ablation sequence rather than as a separately tuned setting.

### *3.5. Computation-Level Branch: Model-Compression Design*

The model-compression strategy reduces spatial and parameter redundancy in the YOLOv7 baseline. It is not obtained by pruning either of the two preceding detector variants.

**ELAN-P with partial convolution.** High-resolution panoramic feature maps contain substantial channel redundancy. As illustrated in Figure 7, PTConv combines partial convolution with pointwise convolution and applies spatial convolution only to a channel subset $C_p$ [21]. Its computational cost is

$$\mathrm{FLOPs}_{\mathrm{PTConv}} = HW\left(k^2 C_p^2 + C^2\right). \tag{7}$$

We set $C_p = C/4$ and used PTConv in the ELAN-P module. The remaining channels bypass the spatial convolution and are mixed by the subsequent pointwise operation.

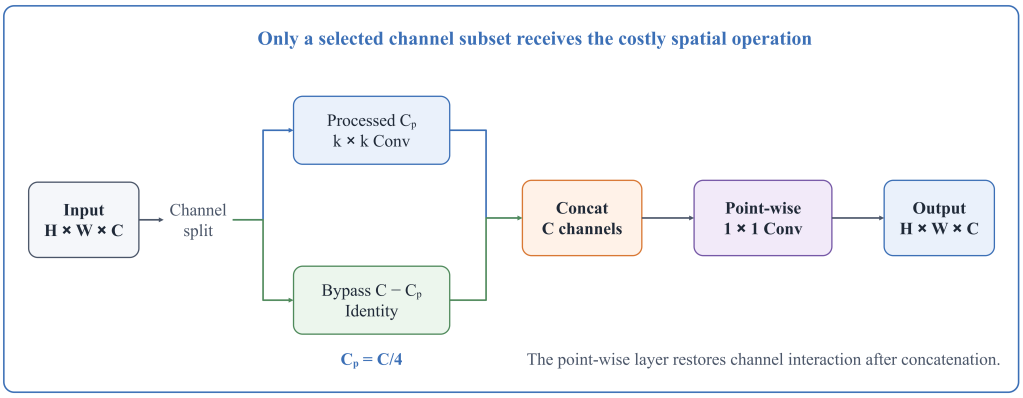



**Figure 7.** PTConv structure. Spatial convolution was applied only to a selected channel subset, while the remaining channels bypassed the spatial operator; a pointwise convolution then restored channel interaction.

**Layer-adaptive magnitude pruning.** After training, LAMP assigns sparsity according to the relative importance of the weights in each layer [22]. For weights sorted by ascending absolute magnitude and indexed by $u$, the score is

$$\mathrm{score}(u; W) = \frac{(W[u])^2}{\sum_{v \geq u} (W[v])^2}. \tag{8}$$

Global ranking by this normalized score reduces bias caused by different weight scales across layers. The connections that contribute most to the learned representation are retained under the target parameter budget.

## 4. Experiments

The distortion-oriented, occlusion-focused, and model-compression configurations were evaluated separately; values from one branch were therefore not used as the baseline of another branch.

### *4.1. Dataset, Experimental Provenance, and Verification Scope*

**Dataset provenance.** PanoWe is an in-house collection of 1971 high-resolution indoor and outdoor panoramas assembled from three sources: frames extracted from videos acquired with laboratory equipment, images captured in physical scenes using KanDao four-lens and Teche six-lens panoramic cameras, and high-resolution panoramas collected from public image or video platforms. Bounding boxes were manually drawn with LabelImg for ten categories: person, chair, car, window, table, monitor, sofa, door, picture, and bed. The full collection was randomly divided at the image level in a 7:2:1 ratio, producing 1380 training, 394 validation, and 197 test images, and was used for the representation- and computation-level experiments.

For the detection-level experiment, 1264 images containing crowded scenes with small or partially occluded objects were selected manually from the panoramic collection before the image-level 7:2:1 split. This chronology rules out selection on the basis of an already defined test partition, but it does not remove the limitation associated with qualitative condition-based sampling. Quantitative selection thresholds and persistent scene identifiers were not retained in the record. We therefore treated this set as a manually curated exploratory crowded-scene subset rather than a standardized small-object or occlusion benchmark. Its aggregate results neither represent full-distribution PanoWe performance nor establish the performance by object size or occlusion severity.

Figure 8 illustrates the indoor and outdoor environments represented in PanoWe.



**Figure 8.** Representative indoor and outdoor ERP scenes from PanoWe.

**Experimental provenance.** The recorded experiments used Ubuntu 20.04, Python 3.8, PyTorch 1.13, an NVIDIA RTX 4090 GPU, and a 16-core Intel Xeon Gold 6430 CPU. The retained thesis and manuscript materials also identified the branch architectures, module locations, dataset counts, image-level split ratio, evaluation metrics, and the single-run tables analyzed below. Table 2 summarizes the verification status and the interpretation permitted for each category of information. In this paper, "main descriptive evidence" denotes the role of a preserved result in the analysis; it does not imply complete historical training reproducibility.

**Table 2.** Archival provenance and verification scope of the preserved study materials.

| Category | Available Record | Verification Status | Permitted Interpretation |
|---|---|---|---|
| Dataset and annotation | PanoWe size, classes, acquisition sources, LabelImg annotation, and image-level split ratio | Traceable | Dataset provenance and sampling context |
| Architecture and placement | Branch structures, module locations, and method equations | Traceable | Architectural and methodological description |
| Results and ablations | Branch-specific tables, cumulative ablations, and pruning-sensitivity records | Traceable archival results | Descriptive within-branch analysis |
| Training configuration | Key hyperparameters and the exact optimization schedule are not traceable from the available archive | Partial | Exact historical retraining is not claimed |
| Reproducibility artifacts | Fixed split files, training code, model weights, and random-seed records are not present in the available archive | Incomplete | No exact split replication or statistical inference |

The reported measurements were consequently treated as preserved single-run results. No standard deviations, confidence intervals, or statistical-significance claims were made, and present-day YOLOv7 defaults were not substituted for unrecorded historical settings. The image-level split also did not establish scene-disjoint generalization because scene identifiers and a duplicate-screening record were absent. Parameters and theoretical FLOPs are reported as model-complexity measures; the throughput in frames per second (FPS) is additionally reported as measured on the hardware and software platform recorded above, and is therefore a platform-specific observation rather than a hardware-independent quantity.

### *4.2. Representation-Level Branch Evaluation*

Table 3 reports the preserved cumulative comparison against the branch-specific YOLOv7 reference on full PanoWe. The final configuration increased the $mAP_{50}$ from 85.1% to 86.9%, the $mAP_{50:95}$ from 61.8% to 62.7%, and the recall from 83.2% to 86.2%, while the precision decreased from 86.7% to 85.6% and the FPS from 120 to 96. Within the recorded sequence, DCNv4, BiFormer, and SPD added 0.6, 0.9, and 0.3 $mAP_{50}$ points at their respective insertion steps. These single-run differences describe an aggregate accuracy–recall–computation trade-off on PanoWe; they were not interpreted as a common-protocol detector ranking or as latitude- or seam-specific evidence.

**Table 3.** Cumulative ablation of the distortion-oriented model on full PanoWe.

| Configuration | Precision (%) | Recall (%) | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) | FPS |
|---|---|---|---|---|---|
| YOLOv7 | 86.7 | 83.2 | 85.1 | 61.8 | 120 |
| + DCNv4 | 86.5 | 82.6 | 85.7 | 61.1 | 108 |
| + DCNv4 + BiFormer | 85.3 | 85.9 | 86.6 | 62.3 | 100 |
| + DCNv4 + BiFormer + SPD | 85.6 | 86.2 | 86.9 | 62.7 | 96 |

Table 4 compares alternative placements of conventional (C) and deformable (D) convolutions within ELAN-D. This controlled structure comparison supports the selection of the C–D–C–D–C arrangement, but remains a single-run result.

Table 5 reports the preserved attention-only comparison. These runs compare alternative attention choices attached to YOLOv7 and are distinct from the cumulative configuration in Table 3.

**Table 4.** Preserved comparison of ELAN-D convolution arrangements on PanoWe. C and D denote conventional and DCNv4-based spatial convolutions, respectively.

| ELAN-D Arrangement | Precision (%) | Recall (%) | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) |
|---|---|---|---|---|
| C–C–D–D–D | 63.4 | 68.9 | 64.1 | 39.4 |
| C–C–C–D–D | 82.3 | 78.2 | 84.8 | 59.9 |
| D–C–D–C–D | 79.8 | 81.1 | 83.2 | 58.6 |
| C–D–C–D–C | 86.5 | 82.6 | 85.7 | 61.1 |

**Table 5.** Preserved comparison of attention mechanisms on PanoWe.

| Configuration | Precision (%) | Recall (%) | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) |
|---|---|---|---|---|
| YOLOv7 | 86.7 | 83.2 | 85.1 | 61.8 |
| YOLOv7 + SE [27] | 88.1 | 84.3 | 85.7 | 62.3 |
| YOLOv7 + CBAM [28] | 87.2 | 86.4 | 85.9 | 62.7 |
| YOLOv7 + BiFormer | 88.7 | 85.6 | 86.3 | 63.7 |

The augmentation ablation in Table 6 reports higher aggregate metrics with spherical rotation and reprojection for both YOLOv7 and the distortion-oriented model. For the latter, augmentation increased the $mAP_{50}$ from 85.3% to 86.9% and the $mAP_{50:95}$ from 61.9% to 62.7%. This association supports retaining the augmentation in the recorded configuration, but it does not isolate the performance by latitude, seam proximity, or distortion severity.

**Table 6.** Effect of three-dimensional rotation-field augmentation on PanoWe.

| Model | Aug. | Precision (%) | Recall (%) | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) |
|---|---|---|---|---|---|
| YOLOv7 | No | 84.2 | 81.1 | 83.7 | 60.6 |
| YOLOv7 | Yes | 86.7 | 83.2 | 85.1 | 61.8 |
| Distortion-oriented model | No | 85.1 | 84.7 | 85.3 | 61.9 |
| Distortion-oriented model | Yes | 85.6 | 86.2 | 86.9 | 62.7 |

### *4.3. Detection-Level Branch Evaluation*

Table 7 reports the preserved cumulative comparison on the manually curated 1264-image crowded-scene subset. The evaluated branch ended with soft-NMS followed by the NWD localization term. Relative to the branch-specific YOLOv7 reference, the final configuration increased the recall from 78.3% to 91.1%, the $mAP_{50}$ from 83.6% to 86.2%, and the $mAP_{50:95}$ from 61.1% to 64.1%, while the precision decreased from 86.5% to 85.4%. The net precision cost of the branch was therefore 1.1 percentage points. These were aggregate changes on the condition-selected subset and must not be interpreted as the full-PanoWe, object-size-specific, or occlusion-severity-specific performance.

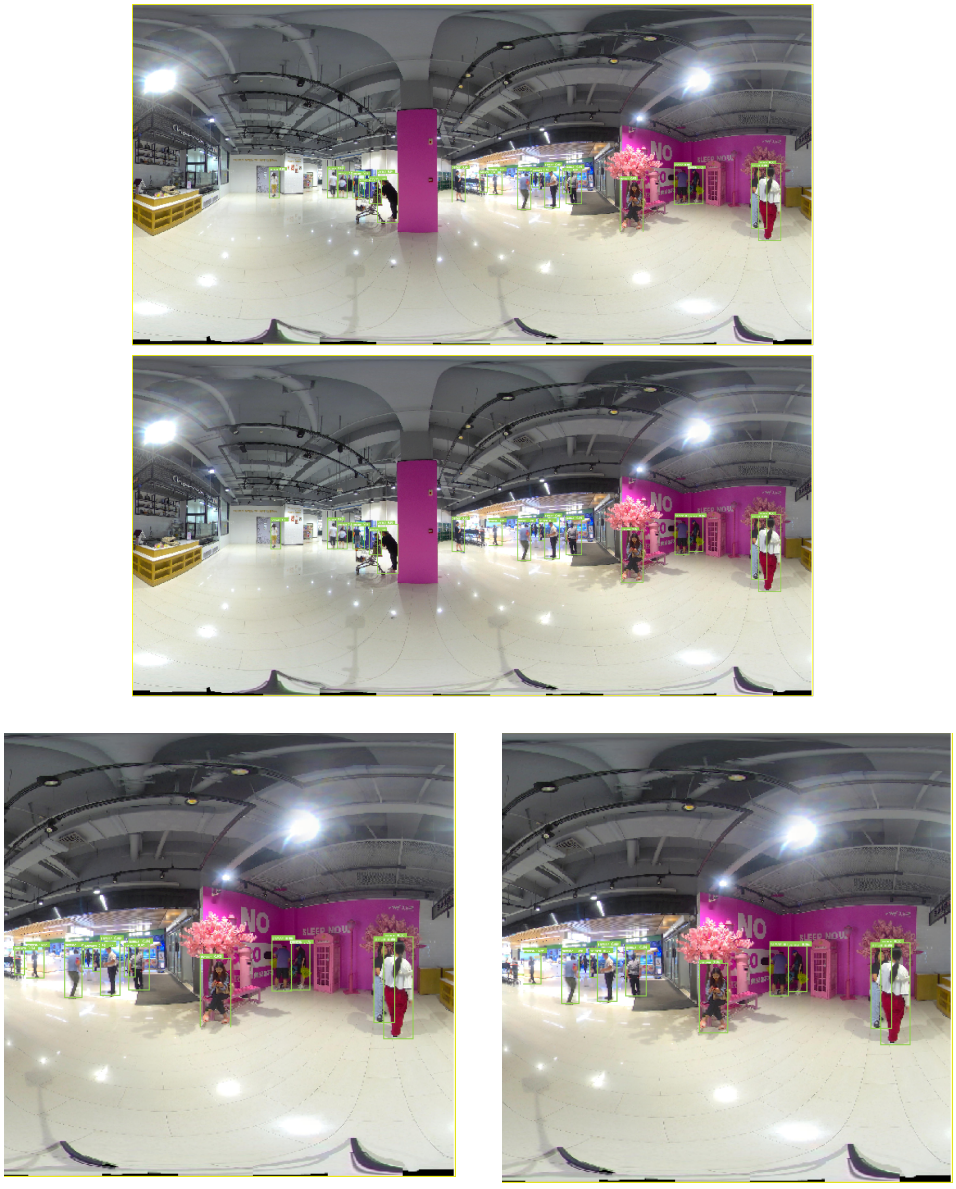
The cumulative sequence also separates learned architectural changes from post-processing. From YOLOv7 to Star block + GD, the precision, recall, $mAP_{50}$, and $mAP_{50:95}$ changed by +1.4, +7.3, +1.3, and +1.7 percentage points, respectively. Adding soft-NMS to that learned configuration changed the same metrics by −4.7, +5.1, +0.5, and +0.5 points. The first stage therefore recorded a joint feature-interaction and aggregation change, whereas the second was a post-processing-induced operating-point shift toward recall. Adding NWD changed the same metrics by +2.2, +0.4, +0.8, and +0.8 points, recovering most of the precision given up at the soft-NMS step while further increasing the recall; this is consistent with a scale-aware localization term producing fewer spurious assignments on small boxes, although the single-run record did not separate this effect from the accompanying threshold shift. Because these rows are cumulative single-run observations, they do not establish statistically independent causal effects or superiority at every confidence threshold.

**Table 7.** Ablation on the 1264-image occlusion-focused subset.

| Configuration | Precision (%) | Recall (%) | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) |
|---|---|---|---|---|
| YOLOv7 | 86.5 | 78.3 | 83.6 | 61.1 |
| + Star block | 86.1 | 82.4 | 84.2 | 61.7 |
| + Star block + GD | 87.9 | 85.6 | 84.9 | 62.8 |
| + Star block + GD + soft-NMS | 83.2 | 90.7 | 85.4 | 63.3 |
| + Star block + GD + soft-NMS + NWD | 85.4 | 91.1 | 86.2 | 64.1 |

### *4.4. Qualitative Visualization*

Figure 9 compares the occlusion-focused configuration with its YOLOv7 baseline on one crowded panoramic scene containing closely spaced and partially occluded people.



(**a**) (**b**)

**Figure 9.** Qualitative comparison using a preserved crowded scene. The full image contains the occlusion-focused result in the upper half and the YOLOv7 baseline in the lower half; enlarged crops make the crowded right-side detections easier to inspect. This single selected example illustrates the aggregate recall trend, but does not demonstrate the robustness across scenes, object sizes, occlusion levels, or ERP seams. (**a**) Enlarged right-side crop: occlusion-focused configuration. (**b**) Enlarged right-side crop: YOLOv7 baseline.

In this selected example, the upper result contains detections near the freezer and behind the shopping cart that are absent from the lower baseline output. The enlarged crops improve the visual inspection of those regions. No additional preserved matched examples or failure cases were available, so the figure is treated only as an illustration and not as an independent performance estimate.

### 4.5. Computation-Level Branch Evaluation

Table 8 reports the preserved within-branch comparison on full PanoWe. ELAN-P reduces the parameter count from 36.9 to 32.9 million and theoretical FLOPs from 104.7 to 86.1 G, with a 0.4-point decrease in the $mAP_{50}$. After LAMP pruning, the model retained 19.4 million parameters and 43.0 theoretical GFLOPs. Relative to the branch-specific YOLOv7 reference, the reductions were 47.4% and 58.9%, while the $mAP_{50}$ and $mAP_{50:95}$ decreased by 1.3 and 1.4 points. LAMP removed individual weights and produced unstructured sparsity. Realizing the corresponding reduction in the wall-clock time requires a sparse execution kernel or supporting hardware, so the pruning results below are reported as model-complexity reductions and are not presented as measured speed-ups.

**Table 8.** Ablation of the model-compression configuration on full PanoWe.

| Configuration | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) | FLOPs (G) | Params (M) |
|---|---|---|---|---|
| YOLOv7 | 85.1 | 61.8 | 104.7 | 36.9 |
| + ELAN-P | 84.7 | 61.2 | 86.1 | 32.9 |
| + ELAN-P + LAMP | 83.8 | 60.4 | 43.0 | 19.4 |

The archived experimental record also includes the separate pruning-sensitivity series in Table 9. ELAN-P was trained before LAMP scoring, pruning, and fine-tuning. We renamed the original "speed-up factor" field as the nominal computation-reduction factor to prevent it from being read as measured hardware acceleration, while retaining the historical label in the caption for traceability. A nominal factor of 2 was selected in the recorded analysis as the accuracy–complexity compromise. The accuracy values in this series differed from the separately trained branch ablation in Table 8; the two tables were therefore interpreted separately rather than pooled.

**Table 9.** Preserved LAMP pruning-sensitivity series for ELAN-P. The nominal computation-reduction factor was originally recorded as the "speed-up factor"; it denotes a theoretical-computation target rather than measured hardware acceleration. Values are not pooled with Table 8.

| Nominal Computation-Reduction Factor | $mAP_{50}$ (%) | $mAP_{50:95}$ (%) | FLOPs (G) | Params (M) |
|---|---|---|---|---|
| 1.0 | 86.3 | 61.2 | 86.1 | 32.9 |
| 1.5 | 85.7 | 60.6 | 57.4 | 23.3 |
| 2.0 | 85.4 | 60.4 | 43.0 | 19.4 |
| 2.5 | 84.0 | 59.2 | 34.4 | 14.8 |
| 3.0 | 81.1 | 55.6 | 28.7 | 12.4 |

Figure 10 visualizes the same five archived pruning targets without introducing additional measurements. Moving from 86.1 to 43.0 theoretical GFLOPs retains an 85.4% $mAP_{50}$ in this sensitivity series; more aggressive targets showed a progressively larger accuracy cost. The curve characterizes a theoretical accuracy–complexity frontier, not a latency or energy curve.

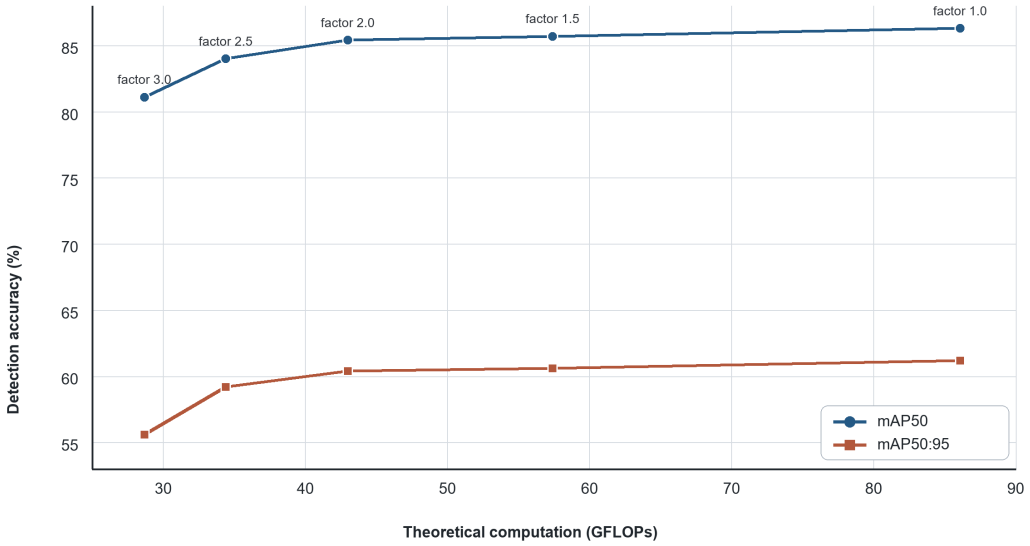



**Figure 10.** Accuracy–theoretical-computation trade-off for the five preserved LAMP pruning targets. Point labels show the nominal computation-reduction factor, originally recorded as the "speed-up factor." No hardware latency is inferred.

### *4.6. External-Dataset Evaluation on 360-Indoor*

To test whether the behavior observed on the in-house collection was specific to it, the representation-level configuration was additionally trained and evaluated on 360-Indoor [29], a public panoramic benchmark that differs from PanoWe in the acquisition device, scene type, and category set.

Table 10 shows that the ordering obtained on PanoWe was broadly preserved on this independent collection: the representation-level configuration attained the highest $mAP_{50}$ at a parameter count comparable to the YOLOv7 baseline, exceeding the strongest baseline entry by 1.1 percentage points. Because the baseline entries were reproduced from the source publication and were not re-trained under a common protocol, this comparison indicates that the improvement was not confined to PanoWe, but it did not establish cross-domain transfer: no model trained on PanoWe was evaluated on 360-Indoor without re-training, and such a zero-shot protocol is not part of the preserved record.

**Table 10.** Comparison on the 360-Indoor benchmark. Baseline entries are reproduced from the comparison table reported in [4]; because the benchmark predates several of these methods, the baseline entries followed the protocol of that source rather than a common re-run under identical training settings.

| Method | $mAP_{50}$ (%) | Parameters (M) |
|---|---|---|
| ResNet-50 + SphereConv | 21.1 | 72.0 |
| YOLOv3 | 24.5 | 61.5 |
| PanoSwin-T+ | 29.4 | 47.0 |
| Faster R-CNN + ResNet-101 | 30.2 | 60.0 |
| FPN + ResNet-101 | 33.6 | 63.0 |
| YOLOv7 | 35.8 | 36.9 |
| Ours (representation-level) | 36.9 | 36.7 |

### *4.7. Constraint-Oriented Synthesis*

The three evaluations used different evidence scopes and therefore cannot be pooled into a single model score or ranked against one another. They can, however, be synthesized through the common stage–constraint–outcome logic introduced above. Table 11 reports changes relative to the corresponding YOLOv7 reference within each branch.

The representation-level branch yielded a moderate aggregate-accuracy gain while shifting the operating balance toward recall and increasing the feature-computation cost. The detection-level branch produced the largest recall shift, accompanied by a lower precision, on the curated crowded-scene subset. The computation-level branch substantially reduced the parameters and theoretical FLOPs at a measurable accuracy cost. These are different responses to different constraints rather than competing estimates of one final detector.

**Table 11.** Branch-specific intervention–outcome–cost summary based on the preserved single-run record. Values were not compared across evidence scopes or interpreted as a jointly trained result.

| Branch | Intervention | Observed Benefit | Observed Cost or Trade-Off | Permitted Inference |
|---|---|---|---|---|
| Representation | Rotation augmentation, DCNv4, BiFormer, and SPD | $mAP_{50}$ +1.8 pp; $mAP_{50:95}$ +0.9 pp; Recall +3.0 pp | Precision −1.1 pp; FPS 120 to 96 | Aggregate accuracy–recall–feature-computation trade-off on full PanoWe |
| Crowded scene | Star block, GD fusion, soft-NMS, and NWD | Recall +12.8 pp; $mAP_{50}$ +2.6 pp; $mAP_{50:95}$ +3.0 pp | Precision −1.1 pp | Recall-oriented aggregate change on the manually curated 1264-image subset |
| Compression | ELAN-P and LAMP | Parameters −47.4%; Theoretical FLOPs −58.9% | $mAP_{50}$ −1.3 pp; $mAP_{50:95}$ −1.4 pp | Accuracy–model-complexity trade-off on full PanoWe; throughput not measured for this branch |

The representation branch targets latitude-dependent deformation methodologically, but the archived evaluation is aggregate rather than stratified by the latitude or distortion severity. ERP is horizontally periodic, and circular padding along the longitude axis, $u_{\mathrm{wrap}} = u \bmod W$, is the topology-consistent boundary treatment; the seam-consistent handling of boxes that cross the boundary belongs to the spherical-representation line of research [3,4,10] that this study did not duplicate. The preserved augmentation instead used reflection padding in its resampling step, so the reported results are aggregate and were not stratified by the distance to the seam.

The synthesis provides a selection logic keyed to the dominant constraint, and is not a claim that the three configurations compose into a single jointly trained detector. When projection deformation is the dominant concern, the representation branch supplies the relevant adaptation and its aggregate accuracy–recall–feature-computation evidence. When missed overlapping targets are more costly than additional false positives, the detection branch offers a recall-oriented operating point for the curated subset. When the parameter count or theoretical arithmetic budget is dominant, the computation branch exposes a tunable compression curve. Selecting more than one branch would require joint training and evaluation because interactions among branches are not identified by the preserved results.

This constraint-oriented selection logic also distinguishes the present study from recent YOLO developments in its research objective and adaptation strategy. YOLO26 couples architectural and training changes for general-purpose real-time vision, including a dual-head design supporting NMS-free inference and a detection head without distribution focal loss [30]. The present study retained YOLOv7 as its reference architecture and evaluated three independent configurations targeting panoramic representation, crowded-scene detection, and model compression. The detection configuration preserves an explicit soft-NMS stage, while the compression configuration combines partial convolution with LAMP pruning. Its contribution therefore lies in quantifying the effects and trade-offs of these adaptations in the evaluated panoramic settings.

YOLO-World addresses a different task setting: it extends YOLO to open-vocabulary detection through vision–language pre-training, a re-parameterizable vision–language path aggregation network, and region–text contrastive learning [31]. In contrast, the present study considered supervised detection over fixed, dataset-specific category inventories, without text-conditioned classification or open-vocabulary recognition. The distinctive strength of the present approach is its alignment of task-specific adaptations with the constraints of panoramic imagery: geometry-oriented representation learning, recall-oriented detection in crowded scenes, and explicit accuracy–complexity trade-offs for model compression.

## 5. Conclusions

This study organizes panoramic object detection as a constraint-oriented problem spanning representation, detection, and computation. Its contribution is the stage-aligned mapping from each constraint to an intervention and an observable outcome, rather than a claim that the adopted operators form a new primitive or that the three configurations compose into one jointly trained detector. Three independently evaluated YOLOv7 configurations instantiate this mapping and support a common within-branch trade-off analysis.

On full PanoWe, the representation configuration reached an 86.9% $mAP_{50}$ and 62.7% $mAP_{50:95}$ while shifting the balance toward recall and reducing the FPS from 120 to 96. On the manually curated 1264-image crowded-scene subset, Star-based interaction and GD fusion produced positive changes in all four aggregate metrics before soft-NMS shifted the operating point further toward recall; the final configuration reached a 91.1% recall with an 85.4% precision, with the NWD term recovering 2.2 of the 4.7 precision percentage points given up at the soft-NMS step. On full PanoWe, ELAN-P and LAMP retained an 83.8% $mAP_{50}$ with 19.4 million parameters and 43.0 theoretical GFLOPs. On the public 360-Indoor benchmark, the representation-level configuration reached a 36.9% $mAP_{50}$, exceeding the strongest reproduced baseline by 1.1 percentage points. Together, these results expose the accuracy–feature-computation, recall–precision, and accuracy–model-complexity trade-offs under three distinct evidence scopes.

The evidence remains descriptive: repeated or stratified validation is not represented in the retained record, and the throughput was not measured for the compression branch. Accordingly, the study does not claim exact historical retraining reproducibility, statistical superiority, size- or seam-specific robustness, additive gains across branches, or hardware-independent acceleration. The FPS values reported for the representation branch are measurements on one recorded platform, and the 360-Indoor baselines were reproduced from a published comparison rather than re-trained under a common protocol.

**Author Contributions:** Conceptualization, Q.Z. and J.S.; methodology, Q.Z., J.S. and C.H.; software, Q.Z., C.H. and Z.X.; validation, Q.Z., J.S., C.H. and Z.X.; formal analysis, Q.Z. and C.H.; investigation, Q.Z., C.H. and Z.X.; resources, J.S., Z.X. and Y.T.; data curation, Q.Z., C.H. and Z.X.; writing—original draft preparation, Q.Z.; writing—review and editing, J.S., C.H. and Y.T.; visualization, Q.Z., C.H. and Z.X.; supervision, J.S. and Y.T.; project administration, J.S. and Y.T.; funding acquisition, not applicable. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The PanoWe collection includes laboratory-acquired imagery, camera-acquired scenes, and material obtained from third-party public platforms. A complete redistribution package, fixed split files, the training code, and model weights are not present in the available study archive. The corresponding author may provide the documented architecture descriptions and tabulated records used in this analysis, subject to the acquisition and third-party-source conditions applicable to the underlying imagery.

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

## References


1. Kinzig, C.; Cortés, I.; Fernández, C.; Lauer, M. Real-time seamless image stitching in autonomous driving. In *Proceedings of the 2022 25th International Conference on Information Fusion (FUSION)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–8. https://doi.org/10.23919/FUSION49751.2022.9841307.
2. Gao, J.; Hu, Z.; Bian, K.; Mao, X.; Song, L. AQ360: UAV-aided air quality monitoring by 360-degree aerial panoramic images in urban areas. *IEEE Internet Things J.* **2021**, *8*, 428–442. https://doi.org/10.1109/JIOT.2020.3004582.
3. Coors, B.; Condurache, A.P.; Geiger, A. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer International Publishing: Cham, Switzerland, 2018; pp. 518–533.
4. Ling, Z.; Xing, Z.; Zhou, X.; Cao, M.; Zhou, G. Panoswin: A pano-style swin transformer for panorama understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2023; pp. 17755–17764.
5. Su, Y.C.; Grauman, K. Learning spherical convolution for fast features from 360 imagery. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2017; Volume 30.
6. Tateno, K.; Navab, N.; Tombari, F. Distortion-aware convolutional filters for dense prediction in panoramic images. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer International Publishing: Cham, Switzerland, 2018; pp. 707–722.
7. Sunkara, R.; Luo, T. No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*; Springer: Cham, Switzerland, 2022; pp. 443–459. https://doi.org/10.1007/978-3-031-26409-2_27.
8. Zhang, H.; Li, M.; Miao, D.; Pedrycz, W.; Wang, Z.; Jiang, M. Construction of a feature enhancement network for small object detection. *Pattern Recognit.* **2023**, *143*, 109801. https://doi.org/10.1016/j.patcog.2023.109801.
9. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2017; pp. 2117–2125.
10. Lee, Y.; Jeong, J.; Yun, J.; Cho, W.; Yoon, K.J. Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2019; pp. 9181–9189.
11. Li, M.; Meng, M.; Zhou, Z. RepF-Net: Distortion-aware Re-projection Fusion Network for Object Detection in Panorama Image. In *Proceedings of the Asian Conference on Computer Vision*; Springer Nature: Cham, Switzerland, 2022; pp. 74–89.
12. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*; IEEE: Piscataway, NJ, USA, 2017; pp. 764–773.
13. Zhu, X.; Hu, H.; Lin, S.; Dai, J. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2019; pp. 9308–9316.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2017; Volume 30.
15. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 213–229.
16. Yang, B.; Zhang, X.; Zhang, J.; Luo, J.; Zhou, M.; Pi, Y. EFLNet: Enhancing feature learning network for infrared small target detection. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5906511. https://doi.org/10.1109/TGRS.2024.3365677.
17. Shen, Y.; Li, Q.; Xu, C.; Chang, C.; Yin, Q. Graph-based context learning network for infrared small target detection. *Neurocomputing* **2025**, *616*, 128949. https://doi.org/10.1016/j.neucom.2024.128949.
18. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS–improving object detection with one line of code. In *Proceedings of the IEEE International Conference on Computer Vision*; IEEE: Piscataway, NJ, USA, 2017; pp. 5561–5569.
19. Xu, C.; Wang, J.; Yang, W.; Yu, H.; Yu, L.; Xia, G.S. Detecting tiny objects in aerial images: A normalized Wasserstein distance and a new benchmark. *ISPRS J. Photogramm. Remote Sens.* **2022**, *190*, 79–93. https://doi.org/10.1016/j.isprsjprs.2022.06.002.
20. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2018; pp. 4510–4520.

21. Chen, J.; Kao, S.h.; He, H.; Zhuo, W.; Wen, S.; Lee, C.H.; Chan, S.H.G. Run, don't walk: Chasing higher FLOPS for faster neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2023; pp. 12021–12031.
22. Lee, J.; Park, S.; Mo, S.; Ahn, S.; Shin, J. Layer-adaptive sparsity for the magnitude-based pruning. In Proceedings of the International Conference on Learning Representations, Virtual, 3–7 May 2021.
23. Xiong, Y.; Li, Z.; Chen, Y.; Wang, F.; Zhu, X.; Luo, J.; Wang, W.; Lu, T.; Li, H.; Qiao, Y.; et al. Efficient deformable convnets: Rethinking dynamic and sparse operator for vision applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2024; pp. 5652–5661.
24. Zhu, L.; Wang, X.; Ke, Z.; Zhang, W.; Lau, R.W. Biformer: Vision transformer with bi-level routing attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2023; pp. 10323–10333.
25. Ma, X.; Dai, X.; Bai, Y.; Wang, Y.; Fu, Y. Rewrite the stars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2024; pp. 5694–5703.
26. Wang, C.; He, W.; Nie, Y.; Guo, J.; Liu, C.; Wang, Y.; Han, K. Gold-YOLO: Efficient object detector via gather-and-distribute mechanism. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2023; Volume 36, pp. 51094–51112.
27. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2018; pp. 7132–7141.
28. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer International Publishing: Cham, Switzerland, 2018; pp. 3–19.
29. Chou, S.H.; Sun, C.; Chang, W.Y.; Hsu, W.T.; Sun, M.; Fu, J. 360-indoor: Towards learning real-world objects in 360deg indoor equirectangular images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; IEEE: Piscataway, NJ, USA, 2020; pp. 834–842. https://doi.org/10.1109/WACV45572.2020.9093262.
30. Jocher, G.; Qiu, J.; Liu, M.; Lyu, S.; Akyon, F.C.; Kalfaoglu, M.E. Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models. *arXiv* **2026**, arXiv:2606.03748 . https://doi.org/10.48550/arXiv.2606.03748.
31. Cheng, T.; Song, L.; Ge, Y.; Liu, W.; Wang, X.; Shan, Y. YOLO-World: Real-Time Open-Vocabulary Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2024; pp. 16901–16911.


**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.