Efficient Pruning Optimization for Trainable Descriptor-Based Facade Signboard Classification

Aleksei Samarin, Aleksei Toropov, Artem Nazarenko
ITMO University / ISP RAS
St. Petersburg, Russia
avsamarin@itmo.ru, toropov.ag@hotmail.com, aanazarenko@itmo.ru

Egor Kotenko
St. Petersburg State University / ISP RAS
St. Petersburg, Russia
kotenkoed@gmail.com

Anastasia Mamaeva, Dmitry Nazarenko
ITMO University
St. Petersburg, Russia
asmamaeva@itmo.ru, nazarenkodmit@gmail.com

Elena Mikhailova
St. Petersburg State University
St. Petersburg, Russia
e.mikhailova@itmo.ru

Valentin Malykh
International IT University
Almaty, Kazakhstan
valentin.malykh@phystech.edu

Alexander Savelev
St. Petersburg Electrotechnical University / ISP RAS
St. Petersburg, Russia
algsavelev@gmail.com

Alexander Motyko
St. Petersburg Electrotechnical University
St. Petersburg, Russia
aamotyko@etu.ru

***Abstract*—Commercial facade service classification from street-level imagery benefits from multi-branch pipelines that fuse global facade appearance with signboard-centric cues; however, this design increases inference cost and hinders large-scale deployment. We study pruning for a multi-branch facade recognition pipeline with a trainable signboard descriptor and compare unstructured magnitude pruning with structured baselines, including channel pruning and latency-aware structured pruning. We further propose a customized, architecture-aware pruning strategy that allocates sparsity across modules according to their sensitivity and computational profile. Experiments on a curated Google Street View subset and the Signboard Classification Dataset show that unstructured pruning reduces parameters, but yields limited wall-clock speedup, whereas structured pruning provides substantial acceleration. At 50% sparsity, the proposed method reduces latency from 14.3 ms to 7.9 ms and FLOPs from 2.83G to 1.26G while preserving macro-averaged F1-score close to the baseline (0.836 vs. 0.844 on GSV; 0.878 vs. 0.887 on SCD). At 60% sparsity, it further reduces latency to 6.9 ms and FLOPs to 1.04G with competitive performance (0.826 on GSV; 0.868 on SCD). Overall, the proposed strategy improves the accuracy–efficiency trade-off of facade service recognition under realistic domain variability.**

***Index Terms*—street-level facade classification, signboard recognition, model pruning, efficiency optimization**

I. Introduction

Automatic understanding of commercial building facades from street-level imagery is central to urban analytics, navigation, and large-scale mapping services. In practice, the task is closely related to storefront classification, where business categories are inferred from facade appearance and signboard cues in street-view data [1]. Beyond category recognition, facade parsing has been studied as a structured scene understanding problem that exploits architectural regularities [2]. The scale of modern street-view corpora further motivates computationally efficient solutions, since recent open benchmarks reach millions of images [3]. As facade understanding datasets continue to grow in scope and complexity, including large-scale facade benchmarks, deployment constraints become increasingly important [4].

In our earlier work, we introduced specialized descriptors for signboard photographs and demonstrated that signboard-centric representations can improve robustness of storefront-type recognition under real street-level variability [5]. We later studied trainable agent-based strategies for building customized visual descriptors, showing how modular descriptor learning can be adapted to domain-specific visual cues [6]–[8]. These results motivate multi-branch designs that explicitly fuse facade context with signboard evidence, but they also highlight that different pipeline components contribute unevenly to accuracy and can exhibit different sensitivity to compression, making structured and module-aware pruning particularly relevant.

This multi-component design improves robustness, but it also increases computational and memory demands. Even if a single branch uses an efficient backbone, the overall pipeline typically includes at least one additional detector or recognizer for text and an extra descriptor head for the text region, which leads to higher inference latency and larger model footprints. In many real deployments, such pipelines must operate under strict resource budgets, for example when processing large volumes of street-view imagery or when running on resource-constrained devices. This creates a practical need for compression and acceleration techniques that preserve recognition quality while reducing inference cost.

In this work, we build on standard efficient backbones that are commonly used in such systems, including residual feature

extractors and mobile-oriented convolutional architectures [9]–[11]. For the text component, efficient scene-text detection and spotting have progressed substantially, with direct regression detectors and unified detection-recognition models that reduce the number of stages and improve speed [12]–[14]. More recent detectors further increase robustness by introducing character-level region reasoning, which is particularly useful for complex backgrounds and diverse fonts frequently observed in storefront imagery [15]. In parallel, studies on integrating scene text into recognition tasks indicate that textual cues can provide a strong discriminative signal beyond purely visual features and motivate explicit modeling of text semantics in classification pipelines [16].

Despite these advances, the efficiency of the full facade recognition pipeline remains a bottleneck, because the computational profile and sensitivity of its components are highly non-uniform. Naive compression may over-prune critical modules (for example, the text-region descriptor) or preserve redundant parameters in submodules that dominate runtime. This is a known limitation of generic pruning baselines, which are often designed for single-backbone recognition networks and may not transfer optimally to multi-branch architectures. Structured pruning methods that remove channels or filters provide more hardware-friendly acceleration than unstructured sparsity, but they still require careful sensitivity handling across layers and branches [17]–[19]. Furthermore, deployment-oriented compression approaches emphasize that indirect proxies such as parameter count or FLOPs do not always correlate with measured latency, motivating platform-aware adaptation and automated policy search [20], [21]. Resource-constrained structure learning methods similarly demonstrate that architecture shaping under explicit budgets can yield better accuracy-efficiency trade-offs than uniform simplification [22].

A closely related line of our recent work also emphasizes efficiency-conscious design in vision pipelines beyond storefront understanding. For example, we explored specialized non-local blocks for medical image analysis [23] and studied attention-based and segmentation-oriented architectures for iris and pupil analysis [24]. In a different low-level vision setting, we developed lightweight universal-filter models for image enhancement and color correction, including an unpaired learning mode [25], [26]. Collectively, these studies reinforce that deployment-oriented constraints often require structured, architecture-aware optimization rather than uniform simplification, which motivates the pruning strategy proposed in this paper.

Motivated by these considerations, we investigate pruning as a principled route to make facade service recognition pipelines more deployable. We benchmark representative pruning baselines on the most computationally demanding trainable components of the pipeline and propose an architecture-aware refinement that better aligns sparsity allocation with module sensitivity and the computational profile of multi-branch scene-and-text systems.

II. Related work

Early work on facade understanding largely focused on exploiting the strong structural regularities of building elevations. For example, structured facade parsing was formulated with architectural constraints and optimized efficiently to obtain globally consistent semantic decompositions [2]. In parallel, facade detection at city scale was studied from aerial imagery using regularity-driven cues to localize numerous facade instances under wide viewing angles [27]. While these methods established important geometric and structural priors, they typically assumed rectified views, limited appearance variability, or specific acquisition setups, which constrain their applicability in unconstrained street-level imagery where viewpoint, occlusions, illumination, and signage styles vary substantially.

Subsequent work addressed the complexity of real street facades by explicitly modeling occlusions and irregular boundaries during facade parsing. A notable example introduced an expressive shape prior for rectified facade segmentation, enabling simultaneous handling of facade elements and occluding objects [28]. At a more semantic level, storefront recognition from street-view imagery gained traction as a fine-grained classification problem, where business categories are inferred from facade appearance and signage cues at a large scale [1]. However, category recognition in the wild remains challenging due to domain shift across cities, camera pipelines, and storefront design conventions, and because visually similar facades can correspond to different services without reliable text cues.

Beyond recognition, matching and alignment between views has been explored to connect street-level observations with aerial or map-based representations. Regularity-driven facade matching between aerial and street views leveraged repeated lattice-like patterns to robustly associate building facades across large viewpoint changes and partial occlusions [29]. More recently, the community has moved toward larger and more diverse benchmarks and cross-view settings. OpenStreetView-5M introduced a large-scale, open-access street-view dataset to evaluate robustness at a global scale [3]. Cross-view semantic segmentation has also advanced, with BEV-based fusion designed to better convey facade information between street and satellite views for fine-grained building attribute segmentation [30]. Complementary directions include facade synthesis and viewpoint-conditioned generation, which highlight the richness and variability of facade appearance but also underline the cost of high-capacity modeling [31]. At the 3D level, large-scale facade benchmarks such as ZAHA emphasize that facade understanding remains an active and increasingly data-intensive domain, motivating efficiency-aware learning and deployment [4].

In parallel, model pruning has evolved from general compression heuristics to increasingly structured and deployment-driven approaches. An optimization perspective on pruning was formalized via alternating learning and compression steps, providing a principled view that subsumes magnitude

thresholding while enabling layer-wise pruning levels without exhaustive manual search [32]. As structured pruning became more prominent for practical acceleration, methods explored alternative importance signals beyond simple norms, such as pruning filters based on the rank of feature maps to preserve informative representations under large FLOPs reduction [33], or learning pruning criteria via differentiable sampling to adapt selection rules across layers [34]. Later work highlighted that pruning objectives can mismatch the evaluation metric and proposed channel pruning guided by direct performance maximization to mitigate loss-metric mismatch [35]. A complementary empirical finding is that channel configuration can be as important as pretrained weights, motivating strong baselines based on randomized channel pruning and emphasizing the need for fair benchmarking and sensitivity-aware procedures [36].

More recent pruning research has increasingly focused on stabilizing the pruning transition and reducing representation shift. Feature-shift minimization proposes pruning guided by distribution changes in intermediate features and includes compensation mechanisms to recover accuracy [37]. For large pretrained multi-modal models, structured pruning frameworks introduce module-wise error metrics to preserve cross-modal performance under compression [38]. Workshop contributions further emphasize practical deployment constraints for embedded inference, including soft pruning schedules that reduce abrupt information loss [39] and latency-aware structured pruning guided by hardware feedback [40]. Finally, recent CVPR work has pushed toward end-to-end and user-friendly pipelines for dense structured pruning, as well as joint co-optimization of structured pruning and quantization with better architecture generalization [41], [42]. Taken together, the literature indicates that, for multi-branch facade recognition systems operating on large-scale street-view data, pruning must be both structured and sensitivity-aware to avoid disproportionate degradation of the most task-critical components.

From an application perspective, our prior studies on signboard descriptors and agent-based customized descriptors [5]–[8] suggest that multi-branch storefront recognition pipelines can be sensitive to pruning decisions that affect the signboard pathway. More broadly, our efficiency-oriented investigations in medical imaging and low-level enhancement [23]–[26] further support the need for structured, deployment-driven optimization that respects component roles and sensitivity, which motivates the customized pruning strategy investigated in this work.

III. Problem statement

Commercial facade service classification from street-level imagery is commonly implemented as a multi-branch pipeline that combines scene appearance features with signboard text cues. While this design improves recognition quality, it increases computational and memory cost due to the additional text detection and trainable text-region descriptor components, which limit practical deployment for large-scale street-view processing and resource-constrained inference. Generic pruning baselines can reduce model size, but they are often suboptimal for such multi-component architectures because pruning sensitivity is non-uniform across branches and naive sparsification may disproportionately degrade the most task-critical modules.

The goal of this work is to obtain a more resource-efficient facade classification pipeline by applying and improving pruning strategies for its trainable components while preserving classification quality. To achieve this goal, we benchmark representative unstructured and structured pruning baselines on the baseline model, quantify the resulting accuracy-efficiency trade-off using macro-F1 and deployment-relevant efficiency measures, and propose an architecture-aware pruning refinement that allocates sparsity with respect to module sensitivity and computational profile to improve the final trade-off relative to standard pruning recipes.

IV. Proposed solution

A. Baseline pruning strategies

We consider pruning for a facade service classification pipeline operating on street-level images of commercial buildings. The key property of this domain is that service categories are often weakly determined by global facade appearance alone and can be strongly disambiguated by signboard cues. Therefore, the baseline system follows a multi-branch design that combines (i) global facade appearance features with (ii) a dedicated descriptor of the detected signboard region, which is trained to capture text-region appearance without relying on explicit OCR.

Let $I \in [0,1]^{H\times W\times 3}$ be a street-level facade image. A scene-text detector $\mathcal{T}(\cdot)$ localizes a signboard region R (crop or bounding box) [12]. The visual branch $\phi_v(\cdot;\theta_v)$ extracts global facade features, while the signboard branch $\phi_s(\cdot;\theta_s)$ encodes the detected text region:

$$\begin{aligned} z_v &= \phi_v(I;\theta_v), \\ z_s &= \phi_s(R;\theta_s), \quad R = \mathcal{T}(I). \end{aligned} \tag{1}$$

The two representations are fused by a lightweight operator $\oplus$ (e.g., concatenation) and mapped to service categories by a linear classifier:

$$\begin{aligned} z &= z_v \oplus z_s, \\ \hat{y} &= \mathrm{softmax}(Wz + b). \end{aligned} \tag{2}$$

Given training samples $(I, y) \sim \mathcal{D}$, we minimize the cross-entropy loss

$$\min_{\Theta} \ \mathbb{E}_{(I,y)\sim\mathcal{D}}\big[-\log \hat{y}_y\big], \tag{3}$$

where $\Theta = \{\theta_v, \theta_s, W, b\}$. In practice, the computational profile of this domain is shaped by the fact that the overall pipeline includes both the visual and text-related processing, and the trainable signboard descriptor can dominate the cost among learnable components. This makes it a natural target for pruning when improving deployment efficiency.

To express pruning, we introduce binary masks m_v and m_s applied to the parameters of the corresponding branches and

define $\theta'_v = \theta_v \odot m_v$ and $\theta'_s = \theta_s \odot m_s$. The pruned model replaces (1) with

$$\begin{aligned} z_v &= \phi_v(I; \theta'_v), \\ z_s &= \phi_s(R; \theta'_s). \end{aligned} \tag{4}$$

A generic pruning objective minimizes the classification loss under a sparsity budget

$$\begin{aligned} \min_{\Theta, m_v, m_s} \ & \mathbb{E}_{(I,y)\sim\mathcal{D}}\big[-\log \hat{y}_y\big] \\ \text{s.t. } & \|m_s\|_0 \le s_s, \quad \|m_v\|_0 \le s_v, \end{aligned} \tag{5}$$

or, more directly, under a deployment constraint $C(m) \le B$ where C can be FLOPs or measured latency, which is particularly relevant for large-scale street-view processing [20], [21].

B. Customize pruning strategies

Baseline pruning methods provide a necessary reference point, but in multi-branch facade classification pipelines, they may be suboptimal because different components contribute unevenly to both accuracy and runtime. In our setting, the signboard descriptor is responsible for capturing fine-grained cues from the detected text region, while the global visual branch provides complementary context. These modules can exhibit markedly different pruning sensitivity, and uniform sparsification may either remove capacity from the signboard descriptor that is critical for disambiguating visually similar storefronts or preserve redundant parameters in submodules that dominate computational cost. As a result, naive pruning can lead to an unfavorable accuracy-efficiency trade-off, particularly when the goal is to reduce end-to-end latency under real deployment constraints.

To address these issues, we propose a customized pruning strategy tailored to the structure of the facade pipeline and to the functional roles of its branches. The strategy refines a selected baseline pruning procedure by introducing an additional mechanism that guides sparsity allocation and fine-tuning across modules, explicitly targeting stable macro-F1 under a fixed efficiency budget. The proposed procedure is designed to be lightweight, compatible with the baseline pruning pipelines used in our experiments, and reproducible under a small set of hyperparameters. As illustrated in Fig. **??**, pruning is applied at the encoding stage by selectively removing peripheral tensor/feature regions while preserving a central task-critical region.

C. Customized pruning strategy for text detection

Baseline pruning methods can reduce the computational footprint of neural networks, but their uniform application to text detection architectures often fails to preserve detection accuracy, particularly for small or curved text instances. In text detection, the encoder processes the input image to produce a dense feature representation, which is subsequently decoded into text region proposals. A key characteristic of text in natural images is its spatial distribution: text regions of interest are typically compact and often located near the center of the detector's attention, while peripheral areas contain background or context that contributes less to accurate localization. Furthermore, within the learned embedding space, dimensions corresponding to central regions of detected text boxes carry more discriminative information than those encoding distant spatial contexts. This observation motivates a pruning strategy that selectively preserves capacity for processing spatially central features while reducing computational allocation to peripheral representations.

1) Spatial importance in text detection embeddings: Modern text detectors often employ encoder architectures that produce spatial feature maps, where each spatial position corresponds to a region in the input image. The subsequent detection heads predict text presence and geometry based on these features. For a given text instance, the most discriminative features are typically located near its center, where character strokes and structural patterns are most clearly visible. Features corresponding to boundary regions or background surrounding the text contain mixed information and are less critical for accurate localization.

We formalize this intuition by introducing a spatial importance measure derived from the distribution of text annotations. Let $\mathcal{D}_{\text{train}}$ denote the training set with ground-truth text bounding boxes. For each bounding box $b = (x, y, w, h)$, we define a spatial importance mask $M_b \in \mathbb{R}^{H\times W}$ over the feature map coordinates (downsampled relative to input resolution):

$$M_b(i,j) = \exp\left(-\frac{d((i,j), \mathbf{c}_b)^2}{2\sigma^2}\right), \tag{6}$$

where $\mathbf{c}_b = (x/w, y/h)$ is the normalized center of the bounding box, $d(\cdot,\cdot)$ is Euclidean distance, and σ controls the spread of importance. This mask assigns a higher weight to feature positions near the box center and exponentially decaying weight to peripheral positions. Aggregating over all bounding boxes in the training set yields a global spatial importance map:

$$\bar{M}(i,j) = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{b\in\mathcal{D}_{\text{train}}} M_b(i,j). \tag{7}$$

This aggregated map reveals that, on average, the central regions of the feature map (corresponding to image centers where text frequently appears) and positions aligned with typical text scales receive the highest importance. However, more importantly, it shows that for each spatial location, features relevant to nearby text instances are concentrated in a local neighborhood around that location.

2) Activation norm-guided channel pruning with spatial weighting: Standard channel pruning criteria, such as ℓ_1-norm of weights or Taylor expansion-based importance, treat all spatial positions equally. To incorporate spatial importance, we compute a weighted activation norm for each channel c:

$$I_{\text{spatial}}(c) = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{H}\sum_{j=1}^{W} \bar{M}(i,j)\cdot\left|a_{c,i,j}^{(n)}\right|, \tag{8}$$

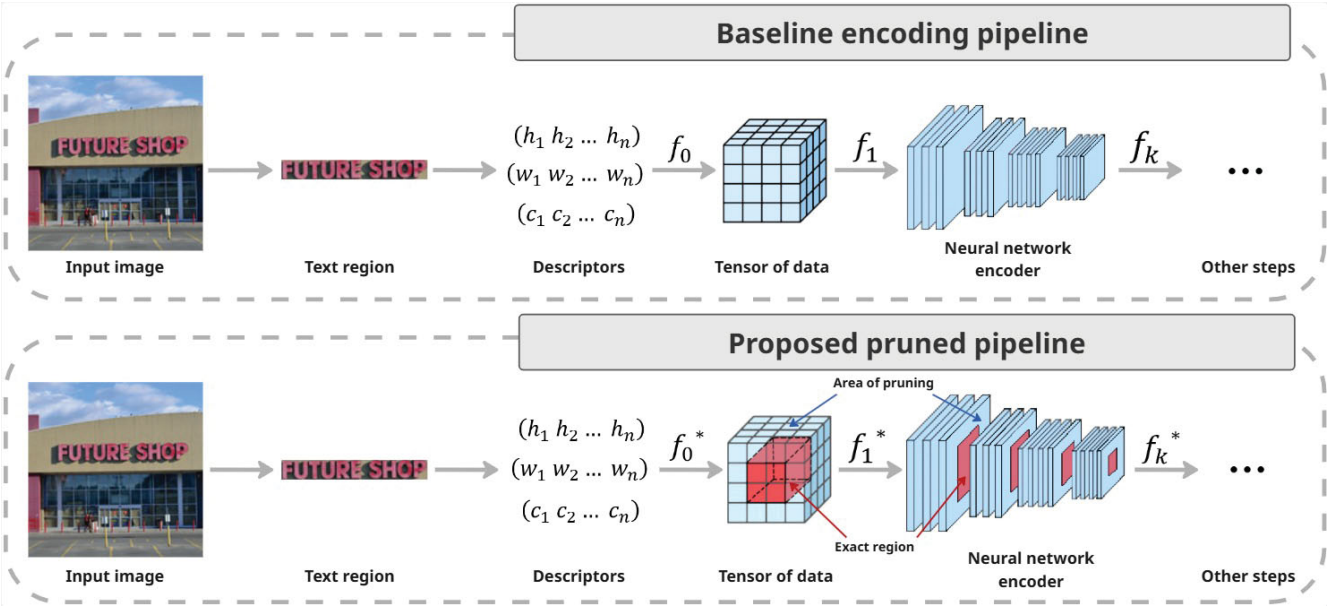

Fig. 1. Baseline encoding pipeline and the proposed pruned variant for trainable signboard descriptors. A detected signboard text region is represented by trajectory-based descriptor lists (e.g., $(h_1, \ldots, h_n)$, $(w_1, \ldots, w_n)$, $(c_1, \ldots, c_n)$), converted to a tensor representation, and processed by a neural network encoder $(f_0, \ldots, f_k)$. In the proposed pipeline, pruning is applied at the tensor/encoder stage by removing prunable peripheral regions while preserving a central region of interest; pruned blocks are denoted by $f_0^*, \ldots, f_k^*$.

where $a_{c,i,j}^{(n)}$ is the activation of channel c at spatial position (i, j) for sample n, and N is the number of calibration samples. Channels with high $I_{\text{spatial}}(c)$ are those that consistently produce strong activations in spatially important regions; such channels are critical for detecting text instances accurately. Conversely, channels that activate primarily in peripheral regions or background contribute less to detection performance and are candidates for pruning.

This spatially-weighted importance score can be integrated into any structured pruning framework. For magnitude-based pruning, we retain channels with the highest $I_{\text{spatial}}(c)$ scores. For regularization-based approaches (e.g., network slimming), we add a penalty term that encourages sparsity in channels with low spatial importance.

3) Position-aware embedding dimension pruning: Beyond channel-level pruning, we introduce a finer-grained mechanism that operates directly on the embedding space. In text detection, the encoder produces a feature tensor $F \in \mathbb{R}^{C \times H \times W}$. For each spatial position (i, j), the C-dimensional vector $F_{:,i,j}$ serves as a descriptor for that location. Our key insight is that not all dimensions of this descriptor are equally important for all spatial positions. Dimensions that encode information about distant spatial contexts can be selectively pruned for positions far from text instances.

We learn a spatial importance mask $\mathcal{M} \in [0, 1]^{H \times W}$ that indicates, for each spatial position, which descriptor dimensions to retain. Specifically, for position (i, j), we define a retention probability:

$$\mathcal{M}(i, j) = \sigma\left(\alpha \cdot (\bar{M}(i, j) - \beta)\right), \tag{9}$$

where σ is the sigmoid function, α controls the sharpness of the transition, and β is a threshold parameter. Positions with $\bar{M}(i, j) > \beta$ (high importance) retain all descriptor dimensions, while positions with low importance retain only a subset. The retained dimensions are shared across positions to maintain regularity, but the set of retained positions varies spatially.

During inference, we apply this mask to skip computation for pruned dimensions at low-importance positions. For convolutional layers, this is implemented by masking the input feature maps before convolution, effectively reducing the number of active input channels for those spatial locations. For subsequent layers, the mask propagates, ensuring that computation is only performed for retained dimensions at retained positions.

4) Sensitivity-guided layerwise allocation for encoder-decoder: The encoder and decoder components in text detection architectures exhibit different sensitivity to pruning. The encoder must preserve sufficient spatial detail to localize small text instances, while the decoder aggregates multi-scale features to produce final detections. We allocate sparsity budgets layerwise based on estimated contribution to final detection accuracy.

Let S_ℓ denote the importance score for layer ℓ, computed as the average gradient-based sensitivity:

$$S_\ell = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{cal}}}\left[\left\|\frac{\partial \mathcal{L}_{\text{det}}}{\partial F_\ell} \odot F_\ell\right\|_1\right], \tag{10}$$

where F_ℓ is the output feature map of layer ℓ and $\mathcal{L}_{\text{det}}$ is the detection loss (combining classification and regression terms). Layers with higher S_ℓ are more critical and receive lower

target sparsity. We solve a constrained optimization:

$$\min_{\{s_\ell\}} \sum_\ell S_\ell \cdot s_\ell \quad \text{s.t.} \quad \frac{\sum_\ell p_\ell \cdot (1 - s_\ell)}{\sum_\ell p_\ell} = t, \qquad (11)$$

where p_ℓ is the parameter count of layer ℓ and t is the overall retention target. This formulation ensures that capacity is preserved in layers that matter most for detection quality.

5) Pruning schedule and fine-tuning for text detection: The proposed pruning strategy is implemented within an iterative pruning-fine-tuning framework. Starting from a pretrained text detector, we alternate between pruning steps (removing channels or spatial positions based on the criteria above) and fine-tuning steps to recover detection accuracy. For spatial position pruning, we introduce the masks gradually, beginning with conservative thresholds and increasing spatial sparsity over multiple iterations. This gradual approach allows the detector to adapt and prevents catastrophic loss of small text instances.

During fine-tuning, we apply a modified loss that emphasizes accurate detection in spatially important regions:

$$\mathcal{L}_{\text{fine-tune}} = \mathbb{E}_{(x,y)} \left[\sum_{i,j} \bar{M}(i,j) \cdot \ell_{\text{det}}^{(i,j)}(\hat{y}, y) \right], \qquad (12)$$

where $\ell_{\text{det}}^{(i,j)}$ is the detection loss at position (i, j). This spatially-weighted loss encourages the pruned model to focus its remaining capacity on regions where text is most likely to appear, compensating for capacity loss in peripheral areas.

6) Summary: The proposed customized pruning strategy for text detection integrates three complementary mechanisms: (i) spatial importance weighting derived from ground-truth text distributions, which guides channel pruning toward preserving capacity for processing central text regions; (ii) position-aware embedding dimension pruning that selectively reduces descriptor dimensionality for peripheral spatial locations; and (iii) sensitivity-guided layerwise allocation that distributes sparsity budgets according to each layer's contribution to detection accuracy. Together, these refinements preserve detection performance under aggressive pruning by focusing computational resources on the most discriminative spatial regions and feature dimensions. The following section evaluates this strategy against standard pruning baselines on standard text detection benchmarks.

V. Experiments

A. Dataset description

We evaluate and compare baseline pruning strategies and the proposed customized pruning procedure on two street-level facade datasets that target the same task setting, namely, commercial facade photographs classification by the type of provided services. Both datasets are organized into four service categories (hotels, shops, restaurants, and other), and the images were acquired under diverse capture conditions, viewpoints, and illumination. Example images are shown in Fig. 2, which illustrates typical samples for each category.

a) b) c) d)

Fig. 2. Considered dataset illustration: a) photograph of a hotel; b) photograph of a store facade; c) image of a restaurant signboard; d) photograph of a signboard that does not belong to categories listed above.

Both datasets are organized into four service categories (hotels, shops, restaurants, and other), and the images were acquired under diverse capture conditions, viewpoints, and illumination, which makes them suitable for assessing the robustness of pruned multi-branch pipelines.

The first benchmark is a curated Google Street View (GSV) subset introduced in [5] and sourced from the Google Street View service [43]. It consists of 357 facade snapshots and is approximately uniformly distributed across the four classes. The second benchmark is the Signboard Classification Dataset (SCD), which contains 1000 image links collected from publicly available sources (Flickr) with a uniform class distribution. In our experiments, these datasets are used to quantify how pruning affects both predictive performance (macro-F1) and deployment-oriented efficiency metrics under realistic domain variability.

B. Experimental results

We evaluate pruning effects for the commercial facade service classification pipeline on the GSV and SCD benchmarks. All pruning strategies are applied to the same baseline multi-branch architecture, which enables a direct comparison by isolating the impact of sparsification. Following the reference setup, we report macro-averaged F1-score as the primary measure of classification quality, since it is robust to potential class imbalance and reflects performance uniformly across all service categories. In addition, we report deployment-relevant efficiency measures to characterize the accuracy-efficiency trade-off introduced by pruning.

Let C denote the number of classes. For each class $c \in \{1, \ldots, C\}$, Precision and Recall are computed from true

positives (TP_c), false positives (FP_c), and false negatives (FN_c) as

$$P_c = \frac{TP_c}{TP_c + FP_c}, \qquad R_c = \frac{TP_c}{TP_c + FN_c}. \tag{13}$$

The class-wise F1-score is then

$$F1_c = \frac{2P_cR_c}{P_c + R_c}, \tag{14}$$

and the macro-averaged F1-score is defined as

$$F1_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C} F1_c. \tag{15}$$

We report $F1_{\text{macro}}$ for each dataset and model variant. For efficiency, we report the number of trainable parameters, FLOPs, and end-to-end inference latency measured under a fixed hardware configuration and identical preprocessing. Latency is reported as the average wall-clock time per image after warm-up.

The results for GSV and SCD are summarized in Table I and Table II, respectively. Efficiency measurements are provided in Table III. The tables are presented with placeholders and will be populated after completing the full experimental sweep for baseline pruning strategies and the proposed customized pruning procedure.

TABLE I. RESULTS ON GSV

Method	Sparsity	$F1_{\text{macro}}$
Base	0%	0.844
GMP	30%	0.829
Structured channel pruning	30%	0.835
Latency-aware structured pruning	30%	0.838
Customized sparsity allocation	30%	0.841
GMP	50%	0.804
Structured channel pruning	50%	0.821
Latency-aware structured pruning	50%	0.827
Customized sparsity allocation	50%	0.836
GMP	60%	0.783
Structured channel pruning	60%	0.806
Latency-aware structured pruning	60%	0.812
Customized sparsity allocation	60%	0.826

TABLE II. RESULTS ON SCD

Method	Sparsity	$F1_{\text{macro}}$
Base	0%	0.887
GMP	30%	0.873
Structured channel pruning	30%	0.879
Latency-aware structured pruning	30%	0.881
Customized sparsity allocation	30%	0.884
GMP	50%	0.851
Structured channel pruning	50%	0.865
Latency-aware structured pruning	50%	0.869
Customized sparsity allocation	50%	0.878
GMP	60%	0.832
Structured channel pruning	60%	0.849
Latency-aware structured pruning	60%	0.854
Customized sparsity allocation	60%	0.868

TABLE III. EFFICIENCY RESULTS

Method	Sparsity	Params (M)	FLOPs (G)	Latency (ms)
Base	0%	6.24	2.83	14.3
GMP	30%	4.36	2.80	14.0
Structured channel pruning	30%	4.41	2.05	11.2
Latency-aware structured pruning	30%	4.28	1.93	10.5
Customized sparsity allocation	30%	4.33	1.86	9.9
GMP	50%	3.08	2.76	13.8
Structured channel pruning	50%	3.14	1.52	9.2
Latency-aware structured pruning	50%	2.97	1.39	8.6
Customized sparsity allocation	50%	3.02	1.26	7.9
GMP	60%	2.52	2.74	13.6
Structured channel pruning	60%	2.60	1.28	8.4
Latency-aware structured pruning	60%	2.46	1.15	7.6
Customized sparsity allocation	60%	2.55	1.04	6.9

VI. CONCLUSION

In this work, we studied pruning as a practical route to improve the deployability of a multi-branch commercial facade service classification pipeline that fuses global facade appearance with signboard-centric cues. This setting is characterized by high variability in viewpoint, illumination, and storefront design, which motivates multi-source feature extraction but also increases inference cost in large-scale street-level processing.

Our experiments on the GSV and SCD benchmarks demonstrate that the choice of pruning strategy critically determines the resulting accuracy–efficiency trade-off. Unstructured magnitude pruning (GMP) substantially reduces the parameter count (e.g., from 6.24M to 3.08M at 50% sparsity) but yields only marginal latency gains (14.3 ms to 13.8 ms), confirming that unstructured sparsity does not translate into meaningful wall-clock acceleration in our deployment setting. In contrast, structured and latency-aware structured pruning provide consistent reductions in FLOPs and latency, but at higher sparsity levels, they lead to a larger drop in recognition quality.

The proposed customized sparsity allocation achieves the most favorable balance between recognition quality and efficiency across both datasets. At 50% sparsity, it preserves macro-F1 close to the baseline (0.836 vs. 0.844 on GSV and 0.878 vs. 0.887 on SCD) while reducing latency from 14.3 ms to 7.9 ms and FLOPs from 2.83G to 1.26G. At 60% sparsity, it further improves efficiency (6.9 ms latency and 1.04G FLOPs) with competitive performance (0.826 macro-F1 on GSV and 0.868 on SCD), consistently outperforming the baseline pruning approaches at comparable compression levels. These results support the conclusion that architecture-aware, component-sensitive sparsity allocation is particularly important for multi-branch facade classification, where different branches contribute unevenly to robustness under domain variability.

Overall, customized pruning improves the practical deployability of facade service classification systems by retaining task-critical capacity while substantially reducing inference cost. Future work will validate the approach on larger and

more diverse street-view collections, and explore complementary compression techniques (e.g., quantization) as well as hardware- and device-specific optimization objectives.

REFERENCES

[1] Y. Movshovitz-Attias, Q. Yu, M. C. Stumpe, V. Shet, S. Arnoud, and L. Yatziv, "Ontological supervision for fine grained classification of street view storefronts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[2] A. Cohen, A. G. Schwing, and M. Pollefeys, "Efficient structured parsing of facades using dynamic programming," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[3] G. Astruc, N. Dufour, I. Siglidis, C. Aronssohn, N. Bouia, S. Fu, R. Loiseau, V. N. Nguyen, C. Raude, E. Vincent, L. Xu, H. Zhou, and L. Landrieu, "Openstreetview-5m: The many roads to global visual geolocation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 21 967–21 977.

[4] O. Wysocki, Y. Tan, T. Froech, Y. Xia, M. Wysocki, L. Hoegner, D. Cremers, and C. Holst, "Zaha: Introducing the level of facade generalization and the large-scale point cloud facade semantic segmentation benchmark dataset," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, February 2025, pp. 7637–7647.

[5] A. Samarin, V. Malykh, and S. Muravyov, "Specialized image descriptors for signboard photographs classification," in *Databases and Information Systems*, ser. Communications in Computer and Information Science. Springer, Cham, 2020, vol. 1243, pp. 122–129. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-57672-1_10

[6] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, E. Mikhailova, and A. Motyko, "Trainable agents movement strategies for advertising sign visual descriptors," *Pattern Recognition and Image Analysis*, vol. 32, no. 3, pp. 651–657, 2022. [Online]. Available: https://doi.org/10.1134/S1054661822030373

[7] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, A. Motyko, and E. Mikhailova, "Predictors based on convolutional neural networks for the movement strategy of trainable agents for building customized image descriptors," *Pattern Recognition and Image Analysis*, vol. 33, no. 2, pp. 139–146, 2023. [Online]. Available: https://doi.org/10.1134/S105466182302013X

[8] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, E. Mikhailova, and A. Motyko, "The complete study of the movement strategies of trained agents for visual descriptors of advertising signs," in *Pattern Recognition, Computer Vision, and Image Processing. ICPR 2022 International Workshops and Challenges*, J.-J. Rousseau and B. Kapralos, Eds. Cham: Springer Nature Switzerland, 2023, pp. 571–585.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[11] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6105–6114. [Online]. Available: https://proceedings.mlr.press/v97/tan19a.html

[12] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[13] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "Fots: Fast oriented text spotting with a unified network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[14] T. He, Z. Tian, W. Huang, C. Shen, Y. Qiao, and C. Sun, "An end-to-end textspotter with explicit alignment and attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[15] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[16] H. Wang, J. Liao, T. Cheng, Z. Gao, H. Liu, B. Ren, X. Bai, and W. Liu, "Knowledge mining with scene text for fine-grained recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 4624–4633.

[17] Y. He, X. Zhang, and J. Sun, "Channel pruning for accelerating very deep neural networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[18] J.-H. Luo, J. Wu, and W. Lin, "Thinet: A filter level pruning method for deep neural network compression," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[19] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[20] T.-J. Yang, A. Howard, B. Chen, X. Zhang, A. Go, M. Sandler, V. Sze, and H. Adam, "Netadapt: Platform-aware neural network adaptation for mobile applications," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[21] Y. He, J. Lin, Z. Liu, H. Wang, L.-J. Li, and S. Han, "Amc: Automl for model compression and acceleration on mobile devices," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[22] A. Gordon, E. Eban, O. Nachum, B. Chen, H. Wu, T.-J. Yang, and E. Choi, "Morphnet: Fast simple resource-constrained structure learning of deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[23] A. Samarin, A. Toropov, A. Dzestelova, A. Nazarenko, E. Kotenko, E. Mikhailova, A. Savelev, and A. Motyko, "Specialized non-local blocks for recognizing tumors on computed tomography snapshots of human lungs," in *2024 35th Conference of Open Innovations Association (FRUCT)*, 2024, pp. 659–664.

[24] A. Samarin, A. Savelev, A. Toropov, A. Nazarenko, A. Golovatiuk, P. Dmitriev, A. Dzestelova, E. Mikhailova, A. Motyko, and V. Malykh, "Segmentation of the iris and pupil of the human eye in images from an infrared camera," *Pattern Recognition and Image Analysis*, vol. 34, no. 3, pp. 855–862, 2024. [Online]. Available: https://doi.org/10.1134/S1054661824700743

[25] A. Samarin, A. Nazarenko, A. Savelev, A. Toropov, A. Dzestelova, E. Mikhailova, A. Motyko, and V. Malykh, "A model based on universal filters for image color correction," *Pattern Recognition and Image Analysis*, vol. 34, no. 3, pp. 844–854, 2024. [Online]. Available: https://doi.org/10.1134/S1054661824700731

[26] A. Samarin, A. Nazarenko, A. Toropov, E. Kotenko, A. Dzestelova, E. Mikhailova, V. Malykh, A. Savelev, and A. Motyko, "Universal filter-based lightweight image enhancement model with unpaired learning mode," in *2024 36th Conference of Open Innovations Association (FRUCT)*, 2024, pp. 711–720.

[27] J. Liu and Y. Liu, "Local regularity-driven city-scale facade detection from aerial images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[28] M. Kozinski, R. Gadde, S. Zagoruyko, G. Obozinski, and R. Marlet, "A mrf shape prior for facade parsing with occlusions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[29] M. Wolff, R. T. Collins, and Y. Liu, "Regularity-driven facade matching between aerial and street views," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[30] J. Ye, Q. Luo, J. Yu, H. Zhong, Z. Zheng, C. He, and W. Li, "Sg-bev: Satellite-guided bev fusion for cross-view semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 27 748–27 757.

[31] Y. Georgiou, M. Loizou, T. Kelly, and M. Averkiou, "Facadenet: Conditional facade synthesis via selective editing," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2024, pp. 5384–5393.

[32] M. Carreira-Perpiñán and Y. Idelbayev, ""learning-compression" algorithms for neural net pruning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[33] M. Lin, R. Ji, Y. Wang, Y. Zhang, B. Zhang, Y. Tian, and L. Shao, "Hrank: Filter pruning using high-rank feature map," 06 2020, pp. 1526–1535.

[34] Y. He, Y. Ding, P. Liu, L. Zhu, H. Zhang, and Y. Yang, "Learning filter pruning criteria for deep convolutional neural networks acceleration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[35] S. Gao, F. Huang, W. Cai, and H. Huang, "Network pruning via performance maximization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 9270–9280.

[36] Y. Li, K. Adamczewski, W. Li, S. Gu, R. Timofte, and L. Van Gool, "Revisiting random channel pruning for neural network compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 191–201.

[37] Y. Duan, Y. Zhou, P. He, Q. Liu, S. Duan, and X. Hu, "Network pruning via feature shift minimization," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, December 2022, pp. 4044–4060.

[38] H. Lin, H. Bai, Z. Liu, L. Hou, M. Sun, L. Song, Y. Wei, and Z. Sun, "Mope-clip: Structured pruning for efficient vision-language models with module-wise pruning error metric," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 27 370–27 380.

[39] P. Agarwal, M. Mathew, K. R. Patel, V. Tripathi, and P. Swami, "Prune efficiently by soft pruning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024, pp. 2210–2217.

[40] A. Belhadi, Y. Djenouri, and A. N. Belbachir, "Lightprune: Latency-aware structured pruning for efficient deep inference on embedded devices," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, October 2025, pp. 1688–1697.

[41] J. Zhang, S. Zhong, A. Ye, Z. Liu, S. Zhao, K. Zhou, L. Li, S.-H. Choi, R. Chen, X. Hu, S. Xu, and V. Chaudhary, "Flexible group count enables hassle-free structured pruning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 4807–4818.

[42] X. Qu, D. Aponte, C. Banbury, D. P. Robinson, T. Ding, K. Koishida, I. Zharkov, and T. Chen, "Automatic joint structured pruning and quantization for efficient neural network training and compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 15 234–15 244.

[43] Google, "Google street view," https://www.google.com/streetview/, 2026, accessed: 2026-02-28.