ScaleLSD: Scalable Deep Line Segment Detection Streamlined

Zeran Ke[1] Bin Tan[2] Xianwei Zheng[3] Yujun Shen[2] Tianfu Wu[4] Nan Xue[†1,2]
[1]School of Computer Science, Wuhan University [2]Ant Group
[3]LIESMARS, Wuhan University [4]Department of ECE, NC State University

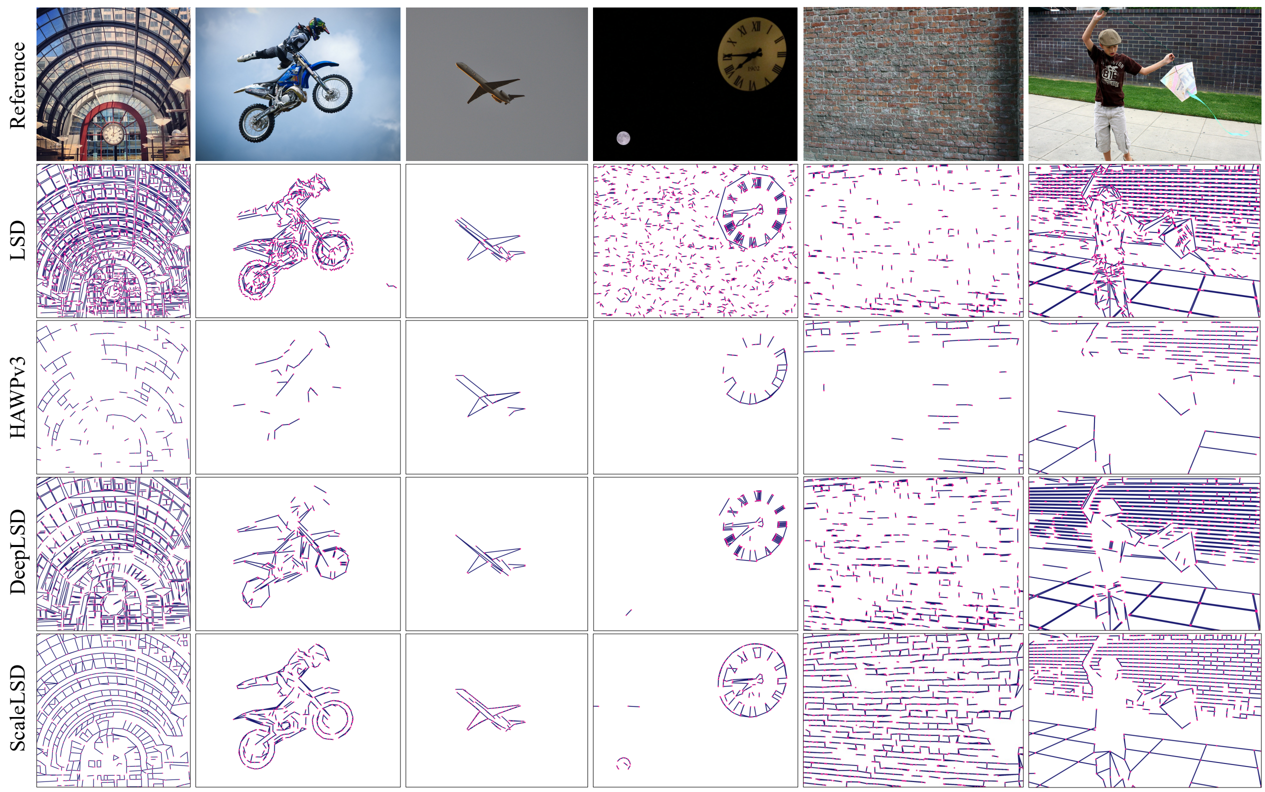


Figure 1. Our ScaleLSD handles a wide range of images, depicting their geometric structures (including the curves, object contours, repeated patterns, and structural regularities) by self-supervised learning of line segment detection from 10M unlabeled images.

Abstract

This paper studies the problem of Line Segment Detection (LSD) for the characterization of line geometry in images, with the aim of learning a domain-agnostic robust LSD model that works well for any natural images. With the focus of scalable self-supervised learning of LSD, we revisit and streamline the fundamental designs of (deep and non-deep) LSD approaches to have a high-performing and efficient LSD learner, dubbed as `ScaleLSD`*, for the curation of line geometry at scale from over* $10M$ *unlabeled real-world images. Our* `ScaleLSD` *works very well to detect much more number of line segments from any natural images even than the pioneered non-deep LSD approach, having a more complete and accurate geometric characterization of images using line segments. Experimentally, our proposed* `ScaleLSD` *is comprehensively testified under zero-shot protocols in detection performance, single-view 3D geometry estimation, two-view line segment matching, and multiview 3D line mapping, all with excellent performance obtained. Based on the thorough evaluation, our* `ScaleLSD` *is observed to be the first deep approach that outperforms the pioneered non-deep LSD in all aspects we have tested, significantly expanding and reinforcing the versatility of the line geometry of images.*

[†]Corresponding author.

1. Introduction

Boundaries are among the most versatile elements in natural images, as low-complexity composable primitives to depict the complicated geometric shapes [22], their spatial and topological relationships [45], as well as the shape-related high-level semantics [10, 14, 21, 25, 42, 44, 52, 53] and semantic structures in natural scenes [12, 56]. There has been a vast body of literature [3, 4, 13, 15, 28, 40, 41, 43, 47] on the computational characterization of boundaries in images at different levels of representation (including corner points, edges, line segments, curves, and contours), first via directly modeling image gradients for a long period, and then transitioned into learning paradigms empowered by deep neural networks and (labor-intensive) annotated datasets. In this paper, we are interested in line segment detection for the geometric characterization of images (Fig. 1), which is useful for many downstream 3D vision tasks due to the parsimoniousness and expressivity of line segments.

Recent studies of deep learning based LSD have been largely driven by meticulously annotated line segments of the Wireframe dataset [16]. Featured by their non-local and vectorized boundary structures, the 5K training data from the Wireframe dataset have enabled and spurred the development of deep line segment detectors in supervised learning settings [17, 46, 48, 49, 51, 57], often with goals to address the locality issue remained in the classical LSD [40]. However, these supervised learning LSD methods struggled with limited generalization to natural images in the wild, which not be easily addressed via scaling up due to the label-intensive and error-prone process of annotating line segments in natural images. Self-supervised learning (SSL) approaches for LSD [30, 31, 51] underscored the limitations of human-annotated labels, and improved the generalizability of LSD over fully supervised counterparts. Nevertheless, the classical LSD [40] often has a higher recall rate than SSL LSD approaches [31, 51].

Our aim in this paper is to devise a method capable of autonomously "defining" boundary line geometries by harnessing image data at scale, to tackle the generalization issue in self-supervised learning of LSD. We hypothesize that existing self-supervised LSD approaches might be limited mainly by their training scales using only thousands of images, but realize that the automatic labeling pipelines in state-of-the-art SSL approaches for LSD, HAWPv3 [51], SOLD^2 [30] and DeepLSD [31], *have scalability issues*.

In both HAWPv3 [51] and SOLD^2 [30], the homographic adaptation schema in the labeling pipeline often leads to low recall rates of line segments in unlabeled images, prohibiting effective large-scale SSL that entails sufficient high-quality pseudo labels. To address the low recall rate issue, DeepLSD [31] exploits the local meaningful alignment schema proposed in the classical LSD [40] in the pseudo-label generation, but unavoidably inherits its locality issue, and experimentally converges to the performance of the classical LSD [40] for downstream tasks as we shall show in experiments (see Fig. 1 for qualitative comparisons).

Scalability entails simplicity in SSL with large-scale unlabeled data, as witnessed by the recent unprecedented progress made in natural language understanding and mid-to-high-level computer vision tasks in the literature. After carefully revisiting state-of-the-art SSL based LSD approaches [30, 31, 51] with the simplicity principle in mind, **we streamline those approaches with three key observations and design choices highlighted**:

- The holistic attraction (HAT) field representations [48, 49, 51] have great potential in SSL of LSD, and predicting the HAT field precisely from images can simplify the LSD modeling.
- Image attributes, inductive biases, and meticulous designs of the classical LSD [40] facilitate the self-supervised learning by inducing a super-efficient and high-recall pseudo labeling pipeline, working well in the integration with the HAT field learning at scale.
- Expressive Transformer [38] based backbones are critical for "ingesting" large-scale data.

We present the `ScaleLSD` method, which works well for any natural images after training with 10M unlabeled images sourced from the SAM-1B dataset [20] (Fig. 1). In our experiments, we showcase the final model of our `ScaleLSD` significantly advanced detection performance measured by the repeatability rate on several data collections that are all different from the training distribution. We further demonstrate that a comprehensive and accurate characterization of line geometry facilitates all the downstream 3D vision tasks, of single-view vanishing point estimation, two-view line segment matching, and multiview 3D line reconstruction, all obtaining state-of-the-art performance.

2. Related Work

Traditional handcraft line segment detection methods [2, 36, 39] primarily rely on low-level image feature processing. Transformer-based fully supervised methods [18, 46] eliminate traditional edge detection steps and directly regress line endpoints using a Transformer decoder. For the deep learning based approaches, the focus of LSD has been shifted from fully supervised learning [7, 11, 17, 24, 27, 46, 48–51, 54, 55, 57] on the Wireframe dataset [16] to self-supervised approaches to address generalization issues of deep LSD models.

Self-supervised LSD Learning. The development of self-supervised LSD learning revealed that the human-annotated line geometry in real-world images contains biases, often leading to suboptimal performance in downstream 3D vision tasks such as vanishing point estimation [8] and multi-view 3D line reconstruction [14, 25, 52]. SOLD^2 [30]

presented the first automatic line geometry labeling process, which took advantage of the inherent generalization ability of boundaries to annotate line segments in a sim-to-real pipeline, in which the homographic adaptation scheme was shown to be useful to eliminate erroneous detection results by averaging multiple inference results up to random homographic warping of unlabeled images. Follow-up studies improved the efficiency and effectiveness of homographic adaptation for better self-supervised learning models [31, 37, 51]. In our study, we found, the cost of homographic adaptation schema for erroneous detection filtering is the completeness during the pseudo label generation for large-scale data, which limits the self-supervised learning of LSD at a small-scale scenario. Our presented method further demonstrate that the homographic adaptation scheme is not necessary for better self-supervised learning of LSD.

Attraction Field Representations. The recent self-supervised LSD methods [31, 51] were benefited from attraction field representations [48, 49] that parameterize sparse line segments using dense fields. DeepLSD [31] further demonstrated that the classic LSD approach [40] facilitates self-supervised LSD learning, but it extensively relies on the local alignment scheme proposed in the LSD [40]. Our proposed work is inspired by DeepLSD [31], but finds a different role of the classic LSD in self-supervised learning, in which LSD [40] is leveraged for rectifying prediction errors during the learning of holistic attraction fields, allowing large-scale self-supervised learning of LSD.

3. Approach

In this section, we first present background on the HAT field representation [49, 51] and the direction / level-line field in the classica LSD [40] to be self-contained. We then present details of our streamlined formulation of `ScaleLSD` (Fig. 3a) on top of HAWPv3 [51], followed by details of pseudo line segment label generation (Fig. 3b).

3.1. Background on Line Segment Representation

The HAT field representation [49, 51] lifts line segments to attraction regions (Fig. 2), which depicts the full geometry of the line segment set defined on the discrete image grid using a rather dense number of pixels in a dense representation. Formally, For a set of line segments $\mathcal{L} = \{\ddot{l}_i = (\mathbf{x}_i^0, \mathbf{x}_i^1)\}_{i=1}^N$ defined on an $H \times W$ image grid, the HAT field maps the set $\mathcal{L}$ in a 4-component field $\mathcal{H}(\mathbf{p}) = (d(\mathbf{p}), \theta(\mathbf{p}), \alpha(\mathbf{p}), \beta(\mathbf{p}))$ in which each (foreground) pixel $\mathbf{p}$ is assigned to its perpendicularly closest line segment, where $d(\mathbf{p}) \in (0, +\infty)$ and $\theta(\mathbf{p}) \in (-\pi, \pi]$ measure the perpendicular distance and direction of the line segment respectively, and $\alpha(\mathbf{p}) \in (-\pi/2, 0), \beta(\mathbf{p}) \in (0, \pi/2)$ characterize the two vectors pointing from $\mathbf{p}$ to the two endpoints in the local coordinate frame origin at $\mathbf{p}$ with the direction θ as the x-axis. The two endpoints of a line segment $\ddot{l}(\mathbf{p}) = (\mathbf{x}_\alpha(\mathbf{p}), \mathbf{x}_\beta(\mathbf{p})) \in \mathbb{R}^2 \times \mathbb{R}^2$ defined by the pixel $\mathbf{p}$ is computed from the 4D distance-angle parametrization by,

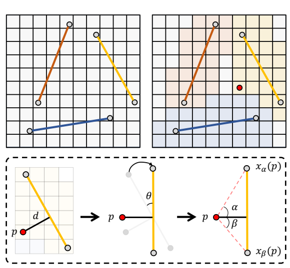


Figure 2. Illustration of the HAT field proposed in [51].

$$\ddot{l}(\mathbf{p}) = d \cdot \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \tan\alpha & \tan\beta \end{bmatrix} + \begin{bmatrix} \mathbf{p}, \mathbf{p} \end{bmatrix}. \quad (1)$$

It is thus straightforward to learn HAT field representation for line segment detection when the ground-truth (GT) label of line segments are available, thus inducing the self-supervised learning of LSD in a pseudo labeling pipeline starting from a bootstrap training using synthetic data with clearly defined GT labels.

The level-line field in the LSD [40] is consistent with the θ field in the HAT field, which is computed by a well-tailored algorithm, and leveraged in our proposed pseudo labeling pipeline to counter the gap between synthetic images and real images.

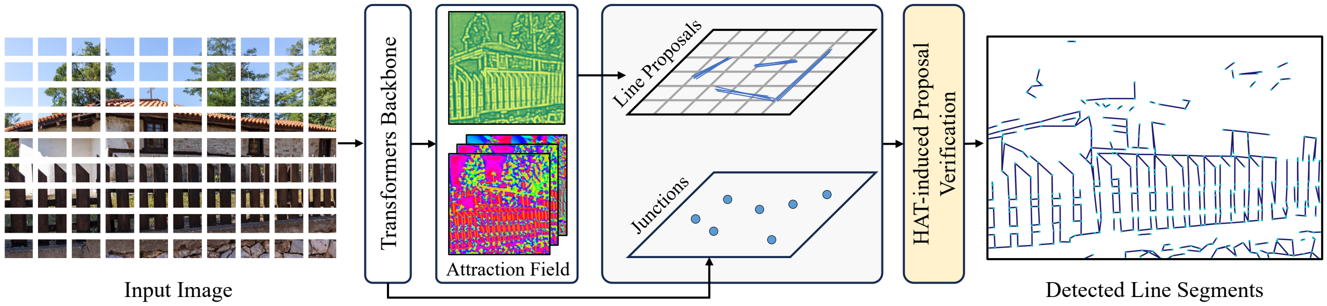
3.2. The Meta Architecture

Fig. 3a illustrates the meta architecture. Compared to HAWPv3 [51], the proposed architecture significantly streamlines designs with a novel method for HAT-induced proposal verification.

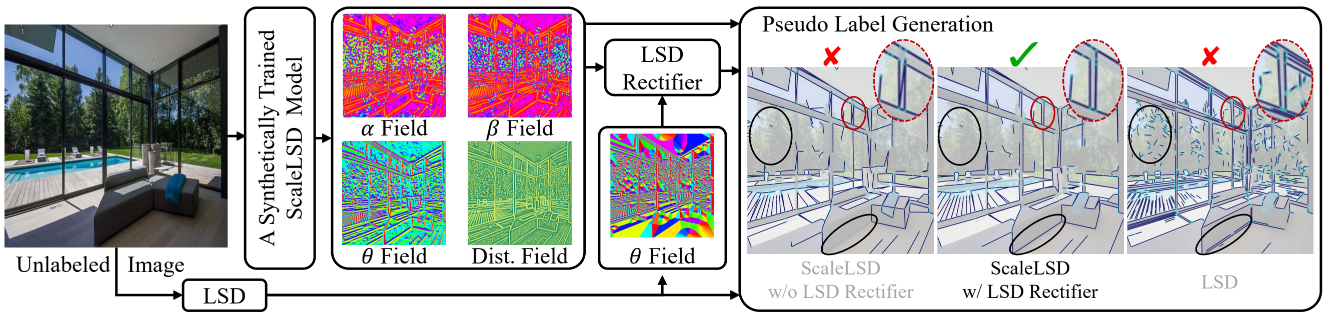
Line Segment Proposal. While the HAT field representation has a clear form to represent each line segment in the set $\mathcal{L}$, the dense field representation, together with the uncertainty in the learning process, often lead to a large number of duplicated proposals for each line segment. As the goal of LSD is to compute a parsimonious and sparse characterization of input images, it is required to sparsely decode the HAT field. To that end, the junctions/endpoints are learned together with the HAT field, leading to the desired sparse decoding scheme. Denoted by $\mathcal{J} = \{\jmath_1, \ldots, \jmath_M\} \subset \mathbb{R}^2$ the set of learned junctions, the sparse decoding scheme first binds the endpoint fields $\mathbf{x}_\alpha(\mathbf{p})$ and $\mathbf{x}_\beta(\mathbf{p})$ by finding their closest junction, indexing each line segment $(\mathbf{x}_\alpha(\mathbf{p}), \mathbf{x}_\beta(\mathbf{p}))$ into $(\imath_\alpha(\mathbf{p}), \imath_\beta(\mathbf{p}))$, where the index mapping $\imath_\alpha(\mathbf{p})$ (or $\imath_\beta(\mathbf{p})$) is defined by

$$\begin{aligned} \imath_\alpha(\mathbf{p}) &= \arg\min_j \|\mathbf{x}_\alpha - \jmath_j\|, \\ \imath_\beta(\mathbf{p}) &= \arg\min_j \|\mathbf{x}_\beta - \jmath_j\|, \end{aligned} \quad (2)$$

$\imath_\alpha, \imath_\beta \in \{0, \ldots, M\}$. Note, when $\imath_\alpha$ (or $\imath_\beta$) becomes 0, it means the minimal distance defined in Eq. (2) is larger than a threshold τ_{dist}, which is set to 10 pixels in our experiments to prune out the outliers in the field prediction. With the index mapping, the line segments in the field with



(a) The architecture overview of the proposed `ScaleLSD` for line segmentation detection.



(b) Illustration of the pseudo label generation pipeline on the real image.

Figure 3. The network architecture and the pseudo label generation in the proposed `ScaleLSD`. Vision Transformer backbones ensure the effectiveness of HAT field learning, thus allowing us to use a HAT-induced verification scheme to decode line segments. In the pseudo label generation, we present an efficient pipeline that use the local line segments by the classical LSD [40] to rectify the network outputs, enabling the large-scale training of LSD with high-quality pseudo labels.

the same index pair (up to the order swapping) are regarded as the same line segment, finally obtaining a sparse set of line segments, each endpoint of which belongs to the set $\mathcal{J}$. Because the endpoint indices are represented in integers, a GPU-builtin implementation yields the unique line segments (and unique index pairs) with little latency.

The HAT-Induced Proposal Verification. The proposal verification were extensively studied to prune out the false detections from the generated proposals for both the classical LSD approach [40] in an a-contrario line verification scheme and the learning-based approaches in the LOI (Line-of-Interest) designs [57] that learns the confidence score of each line proposal according to the ground-truth labels. While LOI-based verification scheme was prevailing in learning-based approaches for end-to-end learning, it poses an issue of label reliability in self-supervised learning, leading to additional designs such as edge map learning and edge-guided verification, as well as the more costly pseudo-label generation schema used in SOLD^2 and HAWPv3.

In our `ScaleLSD`, a novel HAT-induced proposal verification is presented, based on the sparse decoding scheme of HAT field in Eq. (2). Denoted by the sparse set of the junction pairs $\mathcal{I} = \{(\imath_\alpha^1, \imath_\beta^1), \ldots, (\imath_\alpha^K, \imath_\beta^K)\}$, we check the support degree over the field prediction $(\mathbf{x}_\alpha(\mathbf{p}), \mathbf{x}_\beta(\mathbf{p}))$ in

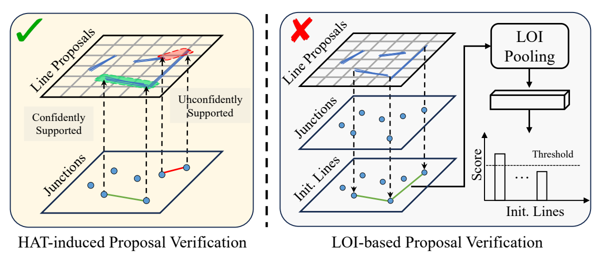


Figure 4. The summary comparison of our proposed HAT-induced proposal verification and the previously used LOI-based proposal verification.

Eq. (1) for each index pair $(\imath_\alpha^k, \imath_\beta^k)$ by

$$\mathrm{Deg}(\imath_\alpha^k, \imath_\beta^k) = \sum \left((\imath_\alpha(\mathbf{p}), \imath_\beta(\mathbf{p})) \sim (\imath_\alpha^k, \imath_\beta^k) \right), \quad (3)$$

where $\sim$ operator returns 1 if the two side of inputs are equal up to permutation otherwise 0. By measuring the proposals using the support degree, the larger number of support pixels in the field prediction, the higher the confidence of learned line segments. Fig. 4 made an illustrative comparison between the proposed HAT-induced and the previously-used LOI verification schema. Because the support degree is measured in the number of pixels, it has better explanation than the classification scores by the feature learning, especially in the outlier-contained self-supervised learning with pseudo labels. In our implementations, we default use the threshold of 10 pixels to filter out the unreliable predictions over the full pipeline of learning.

3.3. The Pseudo Label Generation in ScaleLSD

While HAT-induced proposal verification simplified the learning with white-box and geometrically-meaningful designs, we found that the learning of HAT field itself remains problematic, especially in the bootstrapping phase that was trained on the small-scale synthetic data. We cope with this issue by delving into the classical design of LSD approach [40]. We found, although the LSD approach [40] often produces spurious results in short line segments, its main information source of image gradients are robust and generic to produce reliably line segments when focusing on the orientations, thus bridging the classical design in the learning-based approaches at an appropriate intersection point, leading to an effective design, the LSD-Rectifier for HAT-based self-supervised learning of line segments.

As shown in Fig. 3b, given a seed model trained on the synthetic data, we generate pseudo labels on the real-world images by predicting the HAT fields from the seed model as the main source and the LSD approach [40] as the auxiliary source, then use LSD-Rectifier to replace the θ component from the main source to the LSD-sourced one as a rectified HAT field to predict line segments as the pseudo labels. Because the results by the LSD approach [40] is locally accurate in terms of the line direction, with the proposed LSD-rectifier for pseudo label generation, there is no need to use the computational expensive homographic adaptation schema [30, 49] to filter out the false detection results. The right of Fig. 3b qualitatively compared the pseudo labels generated by different schema, showcasing the effectiveness of the LSD Rectifier.

3.4. Implementation Details

We adopt the transformer-based architecture (ViT-Base) for feature extraction, and employ the DPT [33] head for HAT field of line segments and 2D heatmap of junctions. We maintain the routine of self-supervised learning by the "synthetic-to-real" process of training [9, 30, 51]. The loss functions and training details are provided in the supplementary material.

Training Datasets. Three different datasets are used for training our models. The synthetic dataset consists of 8 simple primitives and 2k images for each primitive, yields 16k samples for training. The Wireframe dataset is augmented by flipping and rotation operations to yield 20k samples for training. The extensive SA1B dataset contains over 10M images obtained around the world and finally yields over 10M samples for training. See more details in the supplementary material.

Training Recipes. We use the ADAM optimizer [19] for training all models. In the synthetic training stage, we train a preheating model on the synthetic dataset for 10 epochs, and we set the learning rate is initialized as 4e-4 and is divided by 10 at the 7th epoch. Then this synthetic model is used to annotate pseudo labels for unlabeled images of realistic dataset. In the real training stage, we separately train our model from scratch on the Wireframe dataset for 30 epochs and on the SA1B dataset for 6 epochs. For training a base model on the Wireframe dataset, we set the learning rate is initialized as 4e-4 and is divided by 10 at the 25th epoch. For scaling up on the SA1B dataset, we set the learning rate increases linearly from a base value 2e-4 to a max value 1e-3 in the first 2000 training iterations and then decreases from the max value to the base value in the manner of cosine annealing [26].

4. Experiments

In this section, we evaluate our `ScaleLSD` models on four tasks, including detection repeatability, estimation of vanishing points, line segment matching, and 3D line reconstruction. Because our method benefits from large-scale training, the main evaluations are zero-shot. In the final, further analyzes on the HAT-induced proposal verification and pseudo-label generation are reported. For more experimental results, please refer to our supplementary materials.

4.1. Repeatability Scores and Localization Errors

Datasets and metrics. The repeatability scores and localization errors measure the performance of feature detectors for given pairs of images. That is to say, given a pair of images captured for the same thing, we expect a line segment detector to repeatably detect line segments up to the viewpoint or photometric changes. We also include the length repeatability evaluation following ELSED [36]. Here, 4 datasets, HPatches [5], RDNIM [29], YorkUrban [8] and COCO (val-2017) [23] are used for the evaluation. For the HPatches [5] and RDNIM [29] datasets, we use the dataset-provided homographies between the image pairs to compute the repeatability scores and localization errors. Because the YorkUrban [8] and COCO [23] do not have paired images, we follow the protocol used by previous studies [31, 51] to warp images by the random homography warping. The detection results that are within 5 pixels (in terms of structural distance and orthogonal distance) are regarded as the repeatedly detected line segments for the evaluation.

Baselines. Due to the poor generalization on zero-shot evaluation of supervised methods (e.g., HAWPv1/v2 [49, 51], L-CNN [57], etc.), we mainly compare our method with classical LSD [40] and the leading self-supervised learning approaches SOLD^2 [30], HAWPv3 [51] and DeepLSD [31]. The official implementation and model

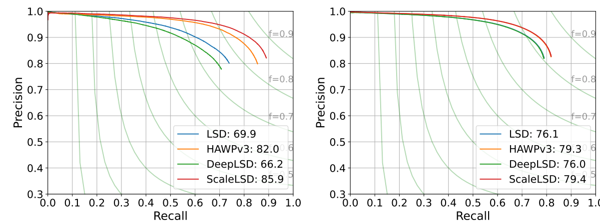
Method	YorkUrban Rep-5 (S) ↑	Loc-5 (S) ↓	Len-5 (S) ↑	Rep-5 (O) ↑	Loc-5 (O) ↓	Len-5 (O) ↑	#Lines/Image	HPatches Rep-5 (S) ↑	Loc-5 (S) ↓	Len-5 (S) ↑	Rep-5 (O) ↑	Loc-5 (O) ↓	Len-5 (O) ↑	#Lines/Image
LSD	0.419	2.123	0.559	0.723	0.959	0.844	591	0.275	2.673	0.264	0.424	1.779	**0.594**	493
SOLD²	0.585	1.918	0.548	0.824	1.097	0.803	196	0.278	2.264	0.251	0.467	1.411	0.460	151
HAWPv3	0.711	1.454	0.687	**0.829**	0.839	0.841	225	0.322	2.314	0.317	0.509	1.572	0.528	149
DeepLSD	0.514	2.199	0.515	0.701	1.054	0.763	310	0.241	2.548	0.228	0.457	1.894	0.493	277
ScaleLSD(Ours)@Wireframe	0.697	1.683	0.714	0.812	0.877	0.847	598	0.337	2.318	0.348	**0.524**	1.624	0.567	499
ScaleLSD(Ours)@SA1B	**0.725**	**1.265**	**0.763**	0.806	**0.768**	**0.849**	**708**	**0.367**	**1.535**	**0.377**	0.515	**1.187**	0.549	**664**

Method	RDNIM Rep-5 (S) ↑	Loc-5 (S) ↓	Len-5 (S) ↑	Rep-5 (O) ↑	Loc-5 (O) ↓	Len-5 (O) ↑	#Lines/Image	COCO Val2017 Rep-5 (S) ↑	Loc-5 (S) ↓	Len-5 (S) ↑	Rep-5 (O) ↑	Loc-5 (O) ↓	Len-5 (O) ↑	#Lines/Image
LSD	0.221	2.766	0.224	0.425	1.733	0.500	433	0.456	2.192	0.386	0.683	1.164	0.637	561
SOLD²	0.241	2.530	0.224	0.421	1.588	0.419	94	0.481	2.233	0.465	0.682	0.956	0.688	83
HAWPv3	0.278	**2.200**	0.268	0.420	**1.496**	0.420	50	0.644	1.614	0.646	0.730	1.107	0.783	99
DeepLSD	0.251	2.661	0.250	0.439	1.639	0.492	152	0.423	2.393	0.423	0.624	1.225	0.678	207
ScaleLSD@Wireframe(Ours)	0.295	2.410	0.299	0.435	1.531	0.465	209	0.636	1.829	0.661	0.749	0.939	0.796	346
ScaleLSD@SA1B(Ours)	**0.337**	2.407	**0.347**	**0.491**	1.510	**0.527**	**540**	**0.666**	**1.540**	**0.699**	**0.764**	**0.909**	**0.809**	**583**

Table 1. The repeatability evaluation results of zero-shot performance on out-of-domain datasets. Numbers with **bold-font** and underline indicate the best and the second best performance on specific metrics. We get the best performance across all datasets and almost all metrics.

Method	YUD+ VP Error ↓	YUD+ AUC ↑	NYU-VP VP Error ↓	NYU-VP AUC ↑
LSD [40]	2.05	82.9 (5.3)	3.29	68.6 (6.3)
TP-LSD [17]	1.73	85.1 (5.0)	3.35	68.0 (4.5)
SOLD² [30]	2.59	75.4 (6.4)	4.46	56.9 (7.6)
HAWPv3 [51]	1.76	84.2 (4.2)	3.35	68.0 (5.7)
DeepLSD [31]	1.63	85.6 (3.6)	3.24	69.1 (6.2)
ScaleLSD@Wireframe(Ours)	1.58	86.6 (1.9)	3.81	63.9 (3.2)
ScaleLSD@SA1B(Ours)	**1.55**	**87.1** (1.1)	**3.18**	**70.4** (1.4)

Table 2. Vanishing points estimation on the YUD+ [8] dataset and the NYU-VP [35] dataset. We make comparisons of all models in term of median VP Error and average AUC (and its standard deviation).



(a) Lines matching (b) Points matching

Figure 5. Comparison of line detectors in the performance of line matcher GlueStick [32] on the ETH3D dataset [34].

weights of those approaches are used. For our `ScaleLSD`, two models trained on the Wireframe and SA1B are evaluated.

Results. As reported in Tab. 1, our `ScaleLSD` trained on the SA1B data gets the best performance across all out-of-domain datasets and almost all metrics and our base Wireframe model also achieves good results. The classical method LSD and the learning-based combined with LSD method DeepLSD can get comparable performance on the first two datasets, only except that DeepLSD apparently outperforms LSD on the challenging RDNIM dataset but LSD is better than DeepLSD on the COCO Val2017 dataset. HAWPv3 gets lower localization errors than ours on the RDNIM dataset but can only detects a few line segments on all datasets. On the whole, our model has the best and most stable detection capability which is in favor for some downstream tasks of image matching and 3D reconstruction.

4.2. Vanishing Points Estimation

Vanishing points (VP) depict infinity under the projective transformations, and play an important role in single-view 3D geometry.

Baselines. We evaluate different line segment detectors (*i.e.*, LSD [40], TP-LSD [17], SOLD^2 [30], HAWPv3 [51], DeepLSD [31] and our `ScaleLSD`) on the VP estimation. We follow DeepLSD [31] to estimate VPs, in which the Progressive-X [6] algorithm is applied to yield vanishing points.

Datasets and Evaluation Metrics. We use YUD+ and NYU-VP datasets for experiments. YUD+ is extended from the YorkUrban [8] dataset and labels up to 8 VPs per image. NYU-VP is adapted from the NYU Depth Dataset V2 [35] and labels 1 to 8 VPS per image. Two metrics are considered, VP Error measures the precision of the estimated VPs in 3D world by the angular error between the directions of the ground-truth VPs and the predicted VPs. AUC means Area Under the Curve (AUC) of the recall curve of the VPs.

Results. As reported in Tab. 2, our base model trained on the structured Wireframe dataset achieves good performance on the YUD+ but drops extremely on the non-Manhattan scenes of the NYU-VP. The scale-up model of our `ScaleLSD` outperforms all baselines in term of VP Error and average AUC (and its standard deviation).

4.3. Line Matching Evaluation

Good line segment detectors are always expected for two-view line segment matching. In this experiment, we feed the detection results into the state-of-the-art line matcher, GlueStick [32] to yield matches from two-view images. The comparisons are made on the ETH3D dataset [34] and we use the precision, recall and F1-score for the resulted point and line matches as the metrics.

Protocol and Results. Because our method focus on the detection, we build the matcher by using SuperPoint [9] as the feature descriptor of junction (or endpoint of line segments) for different detectors. We show the matching precision-recall curves of lines (Fig. 5a) and points

	LSD			HAWPv3			DeepLSD			ScaleLSD (Ours)		
	ACC-L ↓	COMP-L ↓	#Lines	ACC-L ↓	COMP-L ↓	#Lines	ACC-L ↓	COMP-L ↓	#Lines	ACC-L ↓	COMP-L ↓	#Lines
scan16	0.7043	3.0132	1774	0.7898	6.0420	335	0.9242	2.7947	1957	**0.6969**	**2.7162**	**2585**
scan17	0.7961	2.3354	2248	0.8804	5.8212	388	0.9441	**2.2353**	2131	**0.6993**	2.6267	**2867**
scan18	0.8337	2.2196	1995	0.8253	7.0154	287	0.9638	**2.1534**	1894	**0.7357**	2.3008	**2563**
scan19	0.7392	3.2416	1424	0.7110	7.9461	160	0.9614	3.1612	1322	**0.6282**	**2.4352**	**2278**
scan21	0.7890	2.1758	2251	0.8884	5.9821	319	0.9142	**2.0961**	2257	**0.7079**	2.4786	**2757**
scan22	0.7808	2.3884	1863	0.7353	6.8567	281	0.9351	**2.2431**	1948	**0.6593**	2.2951	**2442**
scan24	1.2924	4.0612	1213	**0.7397**	7.7986	246	1.9878	**3.1395**	**1711**	0.8366	4.0756	1624
Avg.	0.8479	2.7765	1824	0.7957	6.7803	288	1.0901	**2.5462**	1888	**0.7091**	2.7040	**2445**

Table 3. Quantitative results of 3D line reconstruction on the DTU [1] dataset for different line segment detectors.

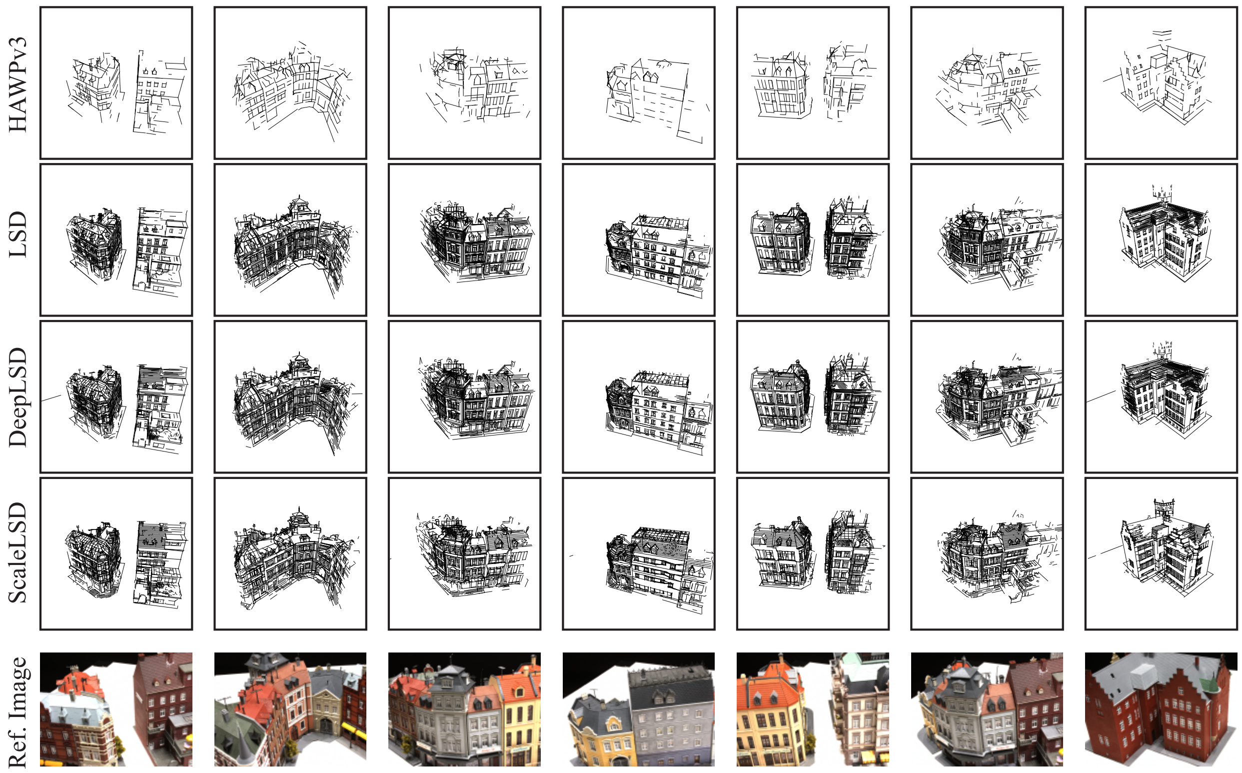


Figure 6. The qualitative comparison of 3D line mapping (by using LiMAP [25]) with different line segment detectors on the building scenes of the DTU dataset [1]. The video results are in the supplementary material.

(Fig. 5b, consists of keypoints and junctions/endpoints) in Fig. 5, and we also attach the average precision (AP) value after each line detector tag of these two sub-figures. It is obviously that our method has significantly exceeded gradient-based methods LSD and DeepLSD and outperforms HAWPv3 greatly in lines matching. For the visualization of line segment matching on challenging cases, please refer to our supplementary material.

4.4. 3D Line Reconstruction

Based on the aforementioned experimental results on detection repeatability, vanishing points estimation and line matching, we move forward to multi-view 3D line reconstruction to evaluate the performance of our `ScaleLSD`.

Protocol and Metrics. The line mapping framework [25] is used in our experiments, which follows a pipeline that sequentially (1) detect line segments and estimate vanishing points from images, (2) match line segments and build line tracks and VP tracks as well, (3) triangulate the line tracks into 3D space using the given camera parameters. 7 scenes of building from the DTU [1] dataset are used for evaluation, in which we compute the ACC and COMP errors between the predicted line segments and the GT point clouds provided by the dataset for each scene. Four detectors, LSD [17], HAWPv3 [51], DeepLSD [31] and our method are evaluated. We additionally report the number of reconstructed 3D line segments as reference for comparison. The detailed evaluation protocol is deferred to supplementary material.

Results. Tab. 3 reports the quantitative evaluation results on the DTU [1] dataset for LiMAP with different detectors. Compared with other detectors, our method obtains the best

ACC scores while keeping reasonable completeness for the 3D line reconstruction. Fig. 6 visualizes the reconstruction results.

LSD-Rectifier	Homo. Adapt.	Struct Rep5 ↑	Struct Loc5 ↓	Orth Rep5 ↑	Orth Loc5 ↓	# Lines / Image	Avg Time[s]	Mem [MB]
		0.397	2.521	0.562	1.688	**80**	**0.381**	**6574**
	✓	<u>0.447</u>	2.452	0.574	1.617	28	3.653	45378
✓		0.445	**2.377**	**0.630**	<u>1.602</u>	<u>78</u>	<u>0.607</u>	<u>6588</u>
✓	✓	**0.473**	<u>2.444</u>	<u>0.621</u>	**1.592**	32	5.811	45492

Table 4. The ablation study of using our LSD-Rectifier and classical Homographic Adaptation for pseudo label generation on the Wireframe dataset. We report the metrics and line numbers to compare the effectiveness and report the average time and space overhead for one batch to compare the efficiency.

Verification LOI-based	Verification HAT-induced	Backbone Hourglass	Backbone DPT	Struct Rep-5 (S) ↑	Struct Loc-5 (S) ↓	Orth Rep-5 (O) ↑	Orth Loc-5 (O) ↓
✓		✓		0.356	2.912	0.629	1.890
✓			✓	0.418	2.763	**0.643**	1.856
	✓	✓		0.263	2.752	0.584	1.852
	✓		✓	**0.445**	**2.377**	0.630	**1.602**

Table 5. The ablation study of using different verifications and backbones for the synthetic bootstrapping stage on the Wireframe dataset.

	Homo. Adapt.										LSD-
iter_num	1	2	3	4	5	6	7	8	9	10	Rectifier
# Lines/Image	35	27	39	33	30	28	33	30	29	28	78
Avg Time[s]	0.702	0.944	1.371	1.562	1.973	2.388	2.693	2.945	3.293	3.653	0.607
Mem[MB]	10588	14400	18214	22042	25856	29956	33812	37666	41522	45378	6588

Table 6. The ablation study of using different number of iteration for Homographic Adaptation for pseudo label generation on the Wireframe dataset.

4.5. Ablation Study

We verify our main designs for line segment detection from two aspects, including the verification of line proposals (Sec. 3.2) and the generation of pseudo labels (Sec. 3.3).

Line Proposals Verification We compare our proposed HAT-induced Proposal Verification with the classical LOI-based verification scheme by testing the learned synthetic models on a hybrid dataset, containing 2,000 images randomly sampled from the Wireframe dataset, the SA1B dataset and the HPatches dataset. We further make the discussion about the impact of CNN-based and transformer-based backbones for the detection performance of these two line proposals verifications. As shown in Tab. 5, compared to HAWPv3 [51] which uses the LOI-based verification scheme, our `ScaleLSD` achieves better results in all metrics, demonstrating the effectiveness of the proposed HAT-induced Proposal Verification. Additionally, LOI-based verification shows negligible scalability of its architecture as the scale-up DPT makes limited improvement of performance relative to the small Hourglass. In contrast, our HAT-induced verification applied with DPT makes significant improvement compared with ones of Hourglass, which shows its promising scalability on applying bigger and powerful backbone for LSD.

Pseudo Labels Generation As discussed in Sec. 3.3, we use LSD-Rectifier for Pseudo Label Generation instead of the commonly used homographic adaptation (Homo.Adap.) scheme [9, 30]. We use the trained synthetic model to compare these two schemes by evaluating the quality of their generated pseudo labels on the Wireframe dataset [16]. As shown in Tab. 4, our LSD-Rectifier strategy achieves comparable repeatability score and localization error, while generating much more line segments than 'Homo. Adap.', which is important for subsequent learning. Besides, our LSD-Rectifier is much faster than 'Homo. Adap.' and is more suitable for large-scale data generation. We set the score threshold for homographic adaptation to 0.75. We also make the ablation study about the impact of iteration number to detected lines number during homographic adaptation in Tab. 6.

5. Conclusion

This paper addressed the problem of line segment detection in self-supervised learning. To tackle the generalization issues persisting in current approaches, typically trained on small-scale datasets of about 20k images, we developed the first model trained using 10M unlabeled data. In designing our method, we critically evaluated prevailing designs, spanning from classical LSD to the recently proposed HAT field representation, streamlining the entire learning pipeline with simple and intuitive designs. Leveraging the powerful scalability inherent in Transformers, we have successfully achieved our goal of generalizable and data-driven line segment detection. This achievement has been demonstrated through various evaluation protocols, including cross-view repeatability, vanishing point estimation, line segment matching and 3D line reconstruction, where we surpassed state-of-the-art performance benchmarks. We believe that our study, which focuses on the symbolic representation of boundary geometry in images, has the potential to offer a parsimonious representation of visual data using a small number of primitives.

Limitations While our method addresses the generalization problem in learning-based line segment detection by utilizing a significantly larger scale of unlabeled data (10M images) compared to prior approaches, we did not fully explore its scalability potential with even larger datasets. Consequently, there remains a risk of under-detecting line segments in testing images. Additionally, while the powerful generalization ability of our method could characterize curves in polylines, our method does not explicitly take curve structures into the modeling process.

Acknowledgement

This work was supported by Ant Group Research Intern Program.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *IJCV*, 120(2):153–168, 2016. 7

[2] Cuneyt Akinlar and Cihan Topal. Edlines: A real-time line segment detector with a false detection control. *Pattern Recognition Letters*, 32(13):1633–1642, 2011. 2

[3] Pablo Arbelaez, Michael Maire, Charless C. Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE TPAMI*, 33(5):898–916, 2011. 2

[4] Dana H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 2

[5] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, pages 5173–5182, 2017. 5

[6] Dániel Baráth and Jiri Matas. Progressive-x: Efficient, anytime, multi-model fitting algorithm. In *ICCV*, pages 3779–3787. IEEE, 2019. 6

[7] Xili Dai, Xiaojun Yuan, Haigang Gong, and Yi Ma. Fully convolutional line parsing. *NeuroComputing*, 2022. 2

[8] Patrick Denis, James H. Elder, and Francisco J. Estrada. Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery. In *ECCV*, pages 197–210, 2008. 2, 5, 6

[9] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, pages 224–236, 2018. 5, 6, 8

[10] Liuyun Duan and Florent Lafarge. Image partitioning into convex polygons. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3119–3127, 2015. 2

[11] Geonmo Gu, Byungsoo Ko, SeoungHyun Go, Sung-Hyun Lee, Jingeun Lee, and Minchul Shin. Towards light-weight and real-time line segment detection. In *AAAI*, 2022. 2

[12] Qi Han, Kai Zhao, Jun Xu, and Ming-Ming Cheng. Deep hough transform for semantic line detection. In *ECCV*, pages 750–766, 2020. 2

[13] Christopher G. Harris and Mike Stephens. A combined corner and edge detector. In *Proceedings of the Alvey Vision Conference (AVC)*, pages 1–6, 1988. 2

[14] Manuel Hofer, Michael Maurer, and Horst Bischof. Efficient 3d scene abstraction using line segments. *Comput. Vis. Image Underst.*, 157:167–178, 2017. 2

[15] Linxi Huan, Nan Xue, Xianwei Zheng, Wei He, Jianya Gong, and Gui-Song Xia. Unmixing convolutional features for crisp edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(10):6602–6609, 2022. 2

[16] Kun Huang, Yifan Wang, Zihan Zhou, Tianjiao Ding, Shenghua Gao, and Yi Ma. Learning to parse wireframes in images of man-made environments. In *CVPR*, pages 626–635, 2018. 2, 8

[17] Siyu Huang, Fangbo Qin, Pengfei Xiong, Ning Ding, Yijia He, and Xiao Liu. TP-LSD: tri-points based line segment detector. In *ECCV*, pages 770–785, 2020. 2, 6, 7

[18] Yasutaka Furukawa Jiacheng Chen, Yiming Qian. Heat: Holistic edge attention transformer for structured reconstruction. In *CVPR*, pages 3856–3865, 2022. 2

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. In *ICCV*, pages 3992–4003. IEEE, 2023. 2

[21] Justin Lazarow, Weijian Xu, and Zhuowen Tu. Instance segmentation with mask-supervised polygonal boundary transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 4372–4381, 2022. 2

[22] Muxingzi Li, Florent Lafarge, and Renaud Marlet. Approximating shapes in images with low-complexity polygons. In *CVPR*, pages 8630–8638. Computer Vision Foundation / IEEE, 2020. 2

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014. 5

[24] Yancong Lin, Silvia L. Pintea, and Jan C. van Gemert. Deep hough-transform line priors. In *ECCV*, pages 323–340, 2020. 2

[25] Shaohui Liu, Yifan Yu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. 3d line mapping revisited. In *CVPR*, pages 21445–21455. IEEE, 2023. 2, 7

[26] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with restarts. *ArXiv*, abs/1608.03983, 2016. 5

[27] Quan Meng, Jiakai Zhang, Qiang Hu, Xuming He, and Jingyi Yu. LGNN: A context-aware line segment detector. In *ACM MM*, pages 4364–4372, 2020. 2

[28] Krystian Mikolajczyk and Cordelia Schmid. Scale & affine invariant interest point detectors. *IJCV*, 60(1):63–86, 2004. 2

[29] Rémi Pautrat, Viktor Larsson, Martin R. Oswald, and Marc Pollefeys. Online invariance selection for local feature descriptors. In *ECCV*, pages 707–724, 2020. 5

[30] Rémi Pautrat, Juan-Ting Lin, Viktor Larsson, Martin R. Oswald, and Marc Pollefeys. SOLD2: self-supervised occlusion-aware line description and detection. In *CVPR*, pages 11368–11378, 2021. 2, 5, 6, 8

[31] Rémi Pautrat, Daniel Barath, Viktor Larsson, Martin R. Oswald, and Marc Pollefeys. Deeplsd: Line segment detection and refinement with deep image gradients. In *CVPR*, pages 17327–17336. IEEE, 2023. 2, 3, 5, 6, 7

[32] Rémi* Pautrat, Iago* Suárez, Yifan Yu, Marc Pollefeys, and Viktor Larsson. GlueStick: Robust image matching by sticking points and lines together. In *ICCV*, pages 9706–9716, 2023. 6

[33] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, pages 12159–12168. IEEE, 2021. 5

[34] Thomas Schops, Johannes L. Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, pages 3260–3269, 2017. 6

[35] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, pages 746–760, 2012. 6

[36] Iago Suárez, José M. Buenaposada, and Luis Baumela. Elsed: Enhanced line segment drawing. *Pattern Recognition*, 127:108619, 2022. 2, 5

[37] Deepak Vasisht, Jayanth Shenoy, and Ranveer Chandra. L2d2: low latency distributed downlink for leo satellites. In *3DV*, page 151–164, 2021. 3

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[39] Rafael Grompone von Gioi, Jérémie Jakubowicz, Jean-Michel Morel, and Gregory Randall. LSD: A fast line segment detector with a false detection control. *IEEE TPAMI*, 32(4):722–732, 2010. 2

[40] R G von Gioi, J Jakubowicz, J M Morel, and G Randall. LSD: A Fast Line Segment Detector with a False Detection Control. *IEEE TPAMI*, 32(4):722–732, 2010. 2, 3, 4, 5, 6

[41] Gui-Song Xia, Julie Delon, and Yann Gousseau. Accurate junction detection and characterization in natural images. *Int. J. Comput. Vis.*, 106(1):31–56, 2014. 2

[42] Gui-Song Xia, Jin Huang, Nan Xue, Qikai Lu, and Xiaoxiang Zhu. Geosay: A geometric saliency for extracting buildings in remote sensing images. *Comput. Vis. Image Underst.*, 186:37–47, 2019. 2

[43] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *IJCV*, 125(1-3):3–18, 2017. 2

[44] Bowen Xu, Jiakun Xu, Nan Xue, and Gui-Song Xia. Accurate polygonal mapping of buildings in satellite imagery. *CoRR*, abs/2208.00609, 2022. 2

[45] Jiakun Xu, Bowen Xu, Gui-Song Xia, Liang Dong, and Nan Xue. Patched line segment learning for vector road mapping. In *AAAI*, pages 6288–6296. AAAI Press, 2024. 2

[46] Yifan Xu, Weijian Xu, David Cheung, and Zhuowen Tu. Line segment detection using transformers without edges. In *CVPR*, pages 4257–4266, 2021. 2

[47] Nan Xue, Gui-Song Xia, Xiang Bai, Liangpei Zhang, and Weiming Shen. Anisotropic-scale junction detection and matching for indoor images. *IEEE Trans. Image Process.*, 27(1):78–91, 2018. 2

[48] Nan Xue, Song Bai, Fudong Wang, Gui-Song Xia, Tianfu Wu, and Liangpei Zhang. Learning attraction field representation for robust line segment detection. In *CVPR*, pages 1595–1603, 2019. 2, 3

[49] Nan Xue, Tianfu Wu, Song Bai, Fudong Wang, Gui-Song Xia, Liangpei Zhang, and Philip H. S. Torr. Holistically-attracted wireframe parsing. In *CVPR*, pages 2785–2794, 2020. 2, 3, 5

[50] Nan Xue, Song Bai, Fudong Wang, Gui-Song Xia, Tianfu Wu, Liangpei Zhang, and Philip H. S. Torr. Learning regional attraction for line segment detection. *IEEE TPAMI*, 43(6):1998–2013, 2021.

[51] Nan Xue, Tianfu Wu, Song Bai, Fu-Dong Wang, Gui-Song Xia, Liangpei Zhang, and Philip H. S. Torr. Holistically-attracted wireframe parsing: From supervised to self-supervised learning. *IEEE TPAMI*, 45(12):14727–14744, 2023. 2, 3, 5, 6, 7, 8

[52] Nan Xue, Bin Tan, Yuxi Xiao, Liang Dong, Gui-Song Xia, Tianfu Wu, and Yujun Shen. Neat: Distilling 3d wireframes from neural attraction fields. In *CVPR*. IEEE, 2024. 2

[53] Yuanwen Yue, Theodora Kontogianni, Konrad Schindler, and Francis Engelmann. Connecting the dots: Floorplan reconstruction using two-level queries. In *CVPR*, pages 845–854. IEEE, 2023. 2

[54] Haotian Zhang, Yicheng Luo, Fangbo Qin, Yijia He, and Xiao Liu. ELSD: efficient line segment detector and descriptor. In *ICCV*, pages 2949–2958, 2021. 2

[55] Ziheng Zhang, Zhengxin Li, Ning Bi, Jia Zheng, Jinlei Wang, Kun Huang, Weixin Luo, Yanyu Xu, and Shenghua Gao. Ppgnet: Learning point-pair graph for line segment detection. In *CVPR*, pages 7105–7114, 2019. 2

[56] Kai Zhao, Qi Han, Chang bin Zhang, Jun Xu, and Ming ming Cheng. Deep hough transform for semantic line detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. 2

[57] Yichao Zhou, Haozhi Qi, and Yi Ma. End-to-end wireframe parsing. In *ICCV*, pages 962–971, 2019. 2, 4, 5