

# SPoT: Subpixel Placement of Tokens in Vision Transformers

Martine Hjelkrem-Tan<sup>1</sup>, Marius Aasan<sup>1</sup>, Gabriel Y. Arteaga<sup>1</sup>, and Adín Ramírez Rivera<sup>1</sup>

<sup>1</sup> Department of Informatics, University of Oslo

{matan, mariuaas, gabrielya, adinr}@uio.no

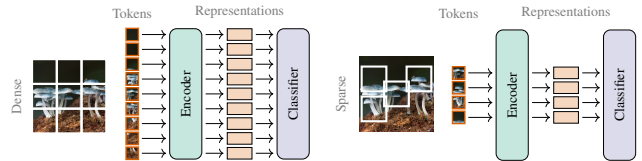
## Abstract

Vision Transformers naturally accommodate sparsity, yet standard tokenization methods confine features to discrete patch grids. This constraint prevents models from fully exploiting sparse regimes, forcing awkward compromises. We propose Subpixel Placement of Tokens (SPoT), a novel tokenization strategy that positions tokens continuously within images, effectively sidestepping grid-based limitations. With our proposed oracle-guided search, we uncover substantial performance gains achievable with ideal subpixel token positioning, drastically reducing the number of tokens necessary for accurate predictions during inference. SPoT provides a new direction for flexible, efficient, and interpretable ViT architectures, redefining sparsity as a strategic advantage rather than an imposed limitation.

## 1. Introduction

Sparsity—the fine art of doing more with less—is an attractive prospect in systems design and modeling. As models grow ever larger, sparse features alleviates the computational demands of a model to provide lower latency, lower memory overhead, and higher throughput—all important properties for real-time applications. Incidentally, sparse selection of features offers inherent interpretability and transparency for increasingly complex models [1]. Clever adaptations of the Vision Transformer (ViT) [2] architecture have shown that this family of models can handle sparse inputs remarkably well [3–7], accelerating inference by selectively processing a reduced subset of the input. Sparsification has even indirectly inspired entirely new paradigms for efficient unsupervised training in the form of masked image modeling (MIM) [8–10].

However, in carefully studying the fine print of Dosovitskiy et al.’s [2] work, one notes an insistence on aligning features with an underlying grid, mirroring the structure inherited by its language counterpart [11, 12] where inputs are

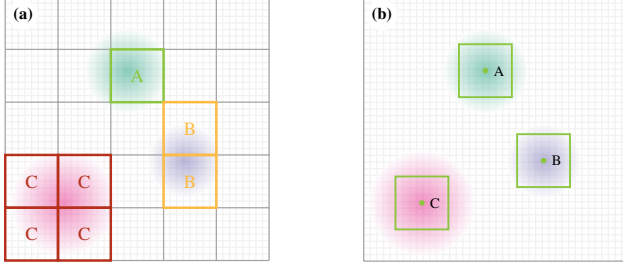


**Figure 1.** (Left) A standard ViT splits the image into a fixed grid of non-overlapping patches. (Right) With SPoT, an adaptively chosen subset of subpixel-precise patches are extracted.

naturally represented as sequences of discrete tokens. This discretization might seem natural—after all, are not pixels fundamentally discrete? Our hypothesis is that this adherence turns sparsity into an awkward dance; forcing the selection of entire tiles, even if the true optimal feature set hides in-between rigid lines. Like eating soup with a fork: possible, but decidedly inefficient and frustrating.

We propose a simple remedy with *Subpixel Placement of Tokens* (SPoT). By allowing patches to occupy continuous subpixel positions instead of constraining features to a discrete grid, we expand our modeling toolkit to include gradient based search and sampling for discovering optimal sparse feature sets. Figure 1 succinctly illustrates the core idea, while the example in Figure 2 shows the limitations of a discrete grid-based approach in reducing tokens while preserving predictive quality. Our contributions include:

- We propose *SPoT*, a novel tokenization framework placing features at continuous subpixel positions, significantly enhancing the robustness and efficiency of ViTs.
- We introduce *Oracle-guided Neighbourhood search* (SPoT-ON) to empirically quantify optimal subpixel positions, showing that carefully selected sparse placements outperform dense grids with only  $\sim 12.5\%$  of the original tokens, and that regions discovered with one model improve performance in another.
- We systematically investigate *spatial priors for subpixel token placement*, and find that dense regimes prefer coverage, while sparse regimes benefit from center bias and saliency-driven priors.



**Figure 2. Grids cannot align every salient region.** (a) A  $5 \times 5$  patch grid (gray) with three optimal region placements for sparse feature selection. The green patch is well aligned (A), yellow straddles two cells (B), and red lies on a corner (C) and leaks into four cells. Translating the grid only swaps which peak is misaligned—one patch is always bad. (b) Our subpixel tokenizer drops fixed-size windows (green squares) directly on each peak, eliminating the alignment trade-off while still allowing conventional grid tokens when they are well aligned.

## 2. Sparse Visual Bag-of-Words and ViTs

At first glance it might seem like a ViT is designed to process an image globally via partitioning an image into patches. However, there is nothing in the transformer architecture that requires a discrete partition. Because self-attention is permutation-invariant, a ViT encoder effectively treats its input tokens as an unordered multiset; a visual bag-of-words, analogous to BERT [12]. This observation suggests we need not restrict tokens to a grid, leading to our formulation of sparse feature selection for ViTs.

We denote a ViT encoder as  $g_\theta: \mathcal{I} \times \Omega \rightarrow \mathbb{R}^d$ , where  $\mathcal{I}$  is a dataset of source images, and  $\Omega$  is a space of positions from which to sample image features. For example, with standard tokenization  $\Omega_{\text{grid}}$  is a fixed, discrete set of non-overlapping square patches tiling the image with a fixed window size on a grid of pixels. The sparse feature selection (SFS) problem can then be formulated as

$$\min_{\phi} \mathbb{E}_{S \sim p_{\phi}} [\mathcal{L}(g_{\theta}(I, S), y)] \text{ s.t. } S \subseteq \Omega, |S| \ll |\Omega|. \quad (1)$$

In other words, we are looking for a probability distribution  $p_{\phi}$  over subsets of  $\Omega$  that minimizes a loss function  $\mathcal{L}^1$ . We note that for the discrete non-overlapping case of  $\Omega_{\text{grid}}$ , there is an implicit assumption that sampling of  $S$  is done without replacement, since sampling the same feature more than once is unlikely to improve model performance. Three specific issues arise from the ViT sparse sampling problem;

1. **Interdependence:** Transformers process tokens as a set. This means that the marginal distribution of one token is dependent on the inclusion of other tokens. Furthermore, the optimal distribution  $p_{\phi}$  for a given image may vary depending on the choice of number of tokens.

<sup>1</sup>Typically standard cross-entropy is used for  $\mathcal{L}$ .

2. **Combinatorial search:** The discrete nature of  $\Omega_{\text{grid}}$  means that selecting a subset of tokens is combinatorial knapsack problem. This makes search difficult and gradient methods intractable, particularly since cardinality-constrained subset selection is NP-hard [13].
3. **Misalignment:** By quantizing patches to a fixed grid, key patterns for discriminating an image could be missed in SFS. Concretely, if the grid imposed by  $\Omega_{\text{grid}}$  is misaligned with key features in the image, SFS could be challenging, as a central shape or texture may be spread over multiple patches, making subset selection more challenging.

These issues hinder efficient optimization of SFS under standard tokenization.

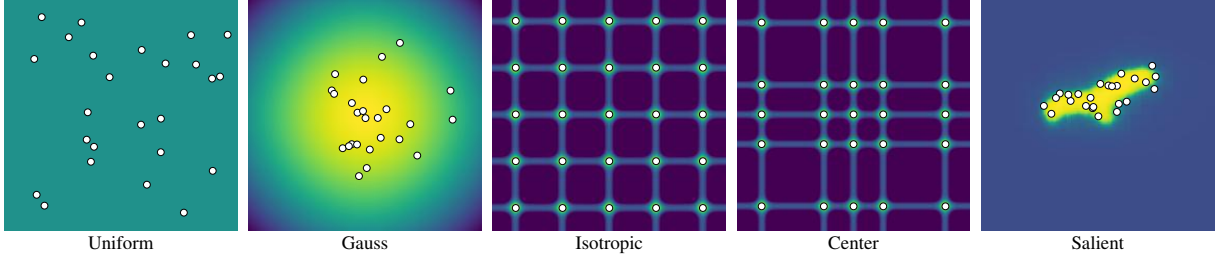
## 3. Subpixel Placement of Tokens: SPoT

We propose a more flexible tokenization scheme to tackle SFS problems in ViTs. Instead of considering  $\Omega$  as a fixed discrete partition, we instead imagine  $\Omega_{\text{subpix}} = [0, H - 1] \times [0, W - 1]$  as a continuous space of subpixel positions from which to select features within a  $H \times W$  image. Put simply, we parametrize a subset of positions  $S = \{s_1, \dots, s_m\}$  as a set of points of interest from which to extract features from within an image. By sampling tokens from continuous subpixel positions, our tokenizer directly addresses the intrinsic *misalignment* issue imposed by traditional grid-based methods, as illustrated in Figure 2. To tackle the *combinatorial search* problem, we use a bilinear interpolation function  $q$  and window size  $k$ , each subpixel position  $s_i = (h, w)$  provides an extracted feature

$$I_q(s_i; k) = I_q(h - \frac{k}{2} : h + \frac{k}{2}, w - \frac{k}{2} : w + \frac{k}{2}). \quad (2)$$

This allows us to formulate SFS as a continuous, probabilistic optimization problem rather than an intractable discrete subset-selection. The key insight is that our novel tokenizer allows us to (1) investigate placing *different priors* on  $p_{\phi}$ , and (2) use *gradient based optimization* to search for  $S$  by way of gradients through  $I_q$ . Since we select  $q$  to be bilinear, its partial derivatives w.r.t.  $s$  exist everywhere except at pixel boundaries, so gradients propagate cleanly back to the placements  $\{s_1, \dots, s_m\}$ . We note that subpixel tokens do not impose any constraint on non-overlapping patches.

Since  $\Omega_{\text{grid}} \subseteq \Omega_{\text{subpix}}$ , patch tokenization is just a special case of our tokenization method. This means that models can be evaluated with the exact same features as a standard patch-based ViT by letting  $S = \Omega_{\text{grid}}$ . Our tokenizer extends existing work showing that more generalized tokenizers can be constructed to be modularly commensurable to standard ViT models, and we adopt their kernelized positional embedding [14].



**Figure 3.** Different sampling priors which can be employed with SPoT. The Sobol prior (not figured) produces uniform quasirandom placements with explicit constraints on coverage.

### 3.1. Spatial Priors

By allowing subpixel freedom in token placement, we lose the implicit spatial prior that discrete grids naturally encode. Hence the shift to a continuous domain raises a dilemma: in sparse regimes, what should guide our choice of token positions? An appropriately selected prior enables efficient sparse representations while preserving performance, whereas an ill-suited one may lead to substantial degradation. We compare several spatial priors, each encoding different assumptions about feature importance and spatial distribution, illustrated in Figure 3:

- **Uniform:** randomly samples locations with no spatial bias, assuming all regions are equally important.
- **Gaussian:** randomly samples locations with a central bias, which encodes a prior belief that subjects are typically centered in images.
- **Sobol:** provides quasirandom sampling aimed at uniform coverage while reducing overlap [15].
- **Isotropic:** deterministically distributes tokens evenly in a subpixel grid, emphasizing coverage.
- **Center:** deterministically distributes tokens evenly with slight central-bias, commonly seen in classification datasets [16].
- **Salient:** encodes object-centric bias by placing tokens based on regions identified as visually salient from a pre-trained saliency model [17].

Moreover, we highlight that a prior could be learned to suit a particular problem. However, we note that interdependency requires a more complex parametric family than a univariate spatial distribution. We therefore leave this step as future work, and focus on the proposal of the continuous positions of tokens and its evaluation.

### 3.2. Oracle Neighborhoods: SPoT-ON

In addition to investigating spatial priors, we also look to directly explore differentiable optimization for token placement. To probe for ideal choices of  $S = \{s_1, \dots, s_m\}$ , we optimize a constrained version of the SFS problem (1) directly for each image. We freeze the encoder  $g_\theta$ , and directly apply gradient search to optimize

$$\arg \min_S [\mathcal{L}(g_\theta(I, S), y)] \text{ s.t. } S \subseteq \Omega_{\text{subpix}}, |S| = m \quad (3)$$

for a set number of tokens  $m$  with initial positions  $S^0 \sim p_\phi$  sampled from a chosen prior  $p_\phi$ . This provides an *Oracle Neighborhood* (ON) adjustments of the initial placements for SPoT. SPoT-ON reveals ideal locations for classifying each image, which allows us to ascertain the existence of an optimal set of positions  $S$  for each image, and estimate an upper bound on performance gain from effective token sampling. We specify that SPoT-ON incurs a higher computational cost for classification, and is *not intended as a practical solution for inference*. Rather, it acts as a tool for analyzing the nature of sparse ViTs, demonstrating the potential of learnable token positions.

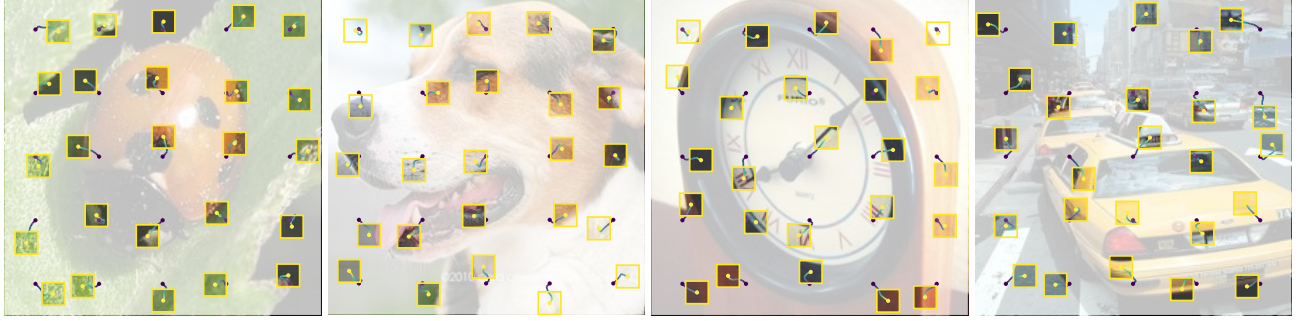
## 4. Experiments: Case Studies

We examine SPoT by adapting two standard ViT models [18], trained on ImageNet-21k and ImageNet-1k (CLS-IN21k, CLS-IN1k) and a self-supervised Masked Autoencoder (MAE-IN1k) [8]. All models utilize the ViT-B/16 architecture [2]. Supervised models initialize weights from TIMM [19], while MAE uses official *pre-trained* weights. To integrate our subpixel tokenizer, each model undergoes a 50-epoch *retrofitting* step on ImageNet-1k [20], after which the MAE-based model is further *fine-tuned* for classification over 100 epochs, aligning with its original protocol [8]. Additional details in Section 5.2. We explore the effectiveness and properties of our approach through four targeted case studies addressing grid limitations, object-centric priors, oracle guidance preferences, and transferability of guided placements.

### 4.1. Are Grids an Inherent Limitation of ViTs?

In our first case study, we investigate whether moving away from fixed-grid token representations toward subpixel placements in continuous space enhances representational quality. Traditional grid-based representations restrict tokens to fixed intervals, which often require multiple patches to cover important features adequately, as illustrated in Figure 2. Subpixel placement, on the other hand, allows tokens to align precisely with these features, potentially enabling more efficient representations with a sparse set of tokens.

To investigate grid versus off-grid representations, we



**Figure 4.** Illustration of oracle placements with 25 tokens with SPoT-ON. By optimizing our oracle-neighborhood search (3) all the way through the model, the oracle discovers optimal placement of points, yielding an accuracy of 90.9% on ImageNet1k with only  $\sim 12.5\%$  of the tokens. Trajectories are colored starting with **dark purple** for initial points, with endpoints colored **bright yellow**.

**Table 1.** Accuracy of grid-constrained and off-grid patch representations in extreme sparse setting with 12.5% of tokens. The grid-based configuration mimics the discrete patch selection of standard ViTs. The off-grid configuration permits subpixel placement in continuous space. Results demonstrate that allowing continuous positioning enhances representational quality under sparse token regimes. We also report the performance of the initially sampled points without oracle supervision (SPoT).

Method	lr	Steps	Acc@1 (%)
SPoT	–	–	61.7
<b>Grid</b>			
SPoT-ON	3e-3	5	66.2
SPoT-ON	1e-2	10	74.0
<b>Subpixel</b>			
SPoT-ON	3e-3	5	<b>90.2</b>
SPoT-ON	1e-2	10	<b>90.9</b>

design an experiment using SPoT-ON to directly compare continuous subpixel placement with discrete, grid-based positioning, all under a fixed token budget of 12.5% of the standard 196 in ViT-B/16 architectures. For the discrete setting, the learned subpixel positions were mapped to their nearest locations on a standard  $14 \times 14$  token grid, mimicking a conventional ViT configuration. We consider two optimization configurations: one with a learning rate of  $3 \times 10^{-3}$  over 5 optimization steps, and another with a higher learning rate of  $1 \times 10^{-2}$  over 10 steps.

The results in Table 1 clearly demonstrate the advantage of subpixel placement, which achieves at least a 16.9 percentage point improvement in accuracy over the grid-constrained method. Interestingly, increasing both the learning rate and the number of optimization steps allows the grid-based approach to discover more effective token positions. Nevertheless, the constraints of discrete, grid-based positioning hinders performance, even under more aggressive optimization. The consistent performance gains highlights significant benefits of continuous, subpixel token placement in resource constrained settings.

**Finding 1:** Off-grid token placement enables greater flexibility and yields substantially better performance than grid-based approaches under sparse token settings.

## 4.2. Do Object-Centric Priors Improve Predictions?

To investigate spatial priors and how they interact with oracle supervision in the sparse setting, we initialize our adaptive sampler using the spatial priors introduced Section 3.1. Each initialization defines the starting coordinates  $S^0$  of token placements, which are subsequently refined by SPoT-ON to minimize classification loss with a learning rate of  $3 \times 10^{-3}$  over 5 optimization steps.

Table 2 reports the resulting downstream accuracy for both supervised and self-supervised backbones in the sparse setting with a token budget of 25 tokens (12.5% of original). We observe that sampling from saliency heatmaps yields the highest performance both out-of-the-box and after oracle supervision. This aligns with common intuition, object-centric features are more relevant to the classification task. The center grid prior also shows higher performance, very likely due to the center bias in ImageNet images. Finally, grid-based initializations (e.g., regular grid, center grid, and Sobol) consistently lead to higher accuracy than the uniform random and Gaussian-stochastic priors, as these result in overlapping placements that are less efficient than structured alternatives.

Comparing these results to Table 5, we see that the benefit of object centric sampling disappears for higher token budgets. Instead, the best performing prior is the regular grid, which ensures broad spatial coverage rather than concentrating solely on the object. This reveals a surprising inductive bias—coverage is more critical than object-centricity for classification under high token budgets. We hypothesize that this is because the information provided by object-focused features quickly saturates, and with higher token budgets the model benefits from the broader context provided by even coverage of the image.

**Table 2.** Accuracy (%) for different spatial initialization priors in extreme sparse setting with 25 tokens. We show out-of-the-box SPoT performance (Acc@1) and achievable increase in performance using SPoT-ON (Oracle  $\Delta$ ).

Prior	k-NN (%)	Acc@1 (%)	Oracle $\Delta$
<b>SPoT CLS-IN21k</b>			
Uniform	45.23 $\pm$ 0.10	44.05	$\uparrow$ 34.22
Gaussian	45.27 $\pm$ 0.10	45.22	$\uparrow$ 32.83
Sobol	46.48 $\pm$ 0.20	43.67	$\uparrow$ 35.83
Isotropic	48.19	46.85	$\uparrow$ 34.85
Center	52.18	52.45	$\uparrow$ 31.07
Salient	56.83 $\pm$ 0.26	55.71	$\uparrow$ 31.90
<b>SPoT MAE-IN1k</b>			
Uniform	49.72 $\pm$ 0.13	56.71	$\uparrow$ 31.44
Gaussian	49.49 $\pm$ 0.33	57.58	$\uparrow$ 29.63
Sobol	53.54 $\pm$ 0.39	60.62	$\uparrow$ 29.40
Isotropic	54.56	61.72	$\uparrow$ 29.21
Center	55.61	62.83	$\uparrow$ 26.70
Salient	60.80 $\pm$ 0.08	66.13	$\uparrow$ 26.59

**Finding 2:** Object-centric priors yield higher performance in sparse regimes. In dense regimes, even and structured coverage provides better performance.

### 4.3. Does Oracle Guidance Prefer Salient Regions?

Intuitively, object-centric priors should help a classifier, but do they actually steer our oracle-guided tokenizer? We investigate whether oracle gradient search in SPoT-ON moves tokens towards pixels with higher class saliency.

We design an experiment as follows; starting with an isotropic prior, we optimize trajectories  $s^{(0)}, \dots, s^{(t)}$  via oracle gradient search (3). Our goal is to measure the shift of token placements toward higher-saliency regions between the initial position  $s^{(0)}$  and the final position  $s^{(t)}$ . Given a saliency mask  $M$ , we compute a score for placements  $s$  by

$$\text{score}(s) = \frac{1}{k^2} \sum_{i=1}^{k^2} M_q(s; k)_i. \quad (4)$$

The *relative saliency gain* for each trajectory is given by

$$\text{RSG} = \frac{\text{score}(s^{(t)}) - \text{score}(s^{(0)})}{\text{score}(s^{(0)})}. \quad (5)$$

Table 3 shows the result of averaging relative saliency gains over ImageNet1k, showing that there is a slight gain in saliency scores for each of our three models. However, the results are not significant enough to claim that saliency alone guides the placements during the oracle search.

We illustrate four examples with different trajectories in Fig. 4, which sheds more light on the behavior of SPoT-ON. While trajectories are drawn to discriminative regions, such as spots on a ladybug (left) or hands of a clock (center right), other placements seem more arbitrary, and even loop

**Table 3.** Quantitative analysis on object-seeking behaviour with SPoT-ON. For each region, we compute the average saliency in a  $16 \times 16$  window centered on each point. We compare the saliency scores of initial placements compared to oracle placements, and compute the relative saliency gain (RSG) over ImageNet1k.

Backbone	RSG (%)			
	25 Tok.	49 Tok.	100 Tok.	196 Tok.
CLS-IN21k	$\uparrow$ 0.34	$\uparrow$ 0.33	$\uparrow$ 0.47	$\downarrow$ 0.14
CLS-IN1k	$\uparrow$ 0.77	$\uparrow$ 0.77	$\uparrow$ 0.75	$\downarrow$ 1.53
MAE-IN1k	$\uparrow$ 0.49	$\uparrow$ 0.54	$\uparrow$ 0.51	$\downarrow$ 0.08

back on themselves. The oracle often positions tokens close to—rather than on—the object; providing context that self-attention can exploit. Hence, *interdependency* rather than saliency alone, drives the final placements.

**Finding 3:** While oracle gradient search yields a slight bias toward higher-saliency pixels, the results are not highly significant. The results suggest that token interdependency — as opposed to pure object saliency — is a predominant factor for optimal placements.

### 4.4. Do Oracle Guided Placements Transfer?

If discovered token placements with oracle guidance captures structure rather than model-specific quirks, *a placement learned by one model should benefit another*. To test this, we investigate *transferability* of token placements. Given two independently trained models,  $g_A, g_B$ , we independently optimize two feature sets;  $S_A, S_B$  respectively (3). Both feature sets are initialized with the same isotropic spatial prior  $S_0$ . Then, for target labels  $y$ , we compute the difference in accuracy from the initial placements  $S_0$  to the optimized placements with the alternate model, i.e.,

$$\Delta = \mathbb{E}[g_A(I, S_0) = y] - \mathbb{E}[g_A(I, S_B) = y], \quad (6)$$

with symmetrical computations for model  $g_B$  on  $S_A$ . The results can be found in Table 4. Our experiments show that placements discovered with one model transfers to yield improvement in performance in a different, independently trained model.

**Finding 4:** Discovered positions via SPoT-ON generalize between models; a set of placements optimized via one model will improve results with another independently trained model in sparse regimes.

## 5. Extended Experimental Results

We present the performance of SPoT under varying sparsity configurations and compare it against baseline models,

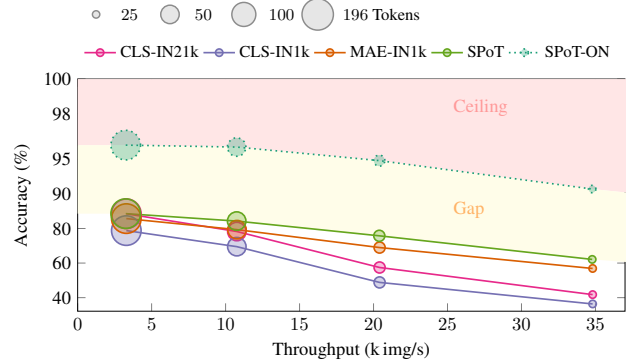
**Table 4.** Transfer properties of SPoT-ON positions between models, in the sparse setting with a 12.5% token budget. We optimize oracle positions using gradient optimization for a source model, initialized with the isotropic prior. We then evaluate the discovered points in an independently trained target model. Each model sees a significant increase in performance, even from points derived from an independent model.

Source	→ Target	Acc@1 (%) (25 Tokens)		
		Original	Transfer	$\Delta$
CLS-IN1k	→ CLS-IN21k	46.85	54.06	$\uparrow 7.21$
MAE-IN1k	→ CLS-IN21k	46.85	56.71	$\uparrow 9.86$
CLS-IN21k	→ CLS-IN1k	33.52	37.91	$\uparrow 4.39$
MAE-IN1k	→ CLS-IN1k	33.52	39.19	$\uparrow 5.67$
CLS-IN21k	→ MAE-IN1k	61.72	68.50	$\uparrow 6.78$
CLS-IN1k	→ MAE-IN1k	61.72	67.81	$\uparrow 6.09$

including the supervised backbones from TIMM [19] and the officially *fine-tuned* MAE model [8]. For clarity, all baselines are denoted as ViT-B/16 in Table 5. To evaluate the baselines under sparsity constraints, we apply Patch-Dropout [3], which randomly drops input patches during inference.

The results in Table 5 reveal several noteworthy observations. First, the self-supervised MAE model [8] consistently outperforms its supervised counterparts under sparse configurations. This advantage is likely due to its pre-training objective, which inherently involves reconstructing inputs from partial observations, thereby fostering robustness to patch dropout. Second, we observe that under full-token conditions, SPoT achieves marginally higher performance than both the supervised and self-supervised baselines. This suggests that even in dense settings, additional gains can be realized by leveraging the flexibility introduced by subpixel representations. Third, as the level of sparsity increases, SPoT consistently surpasses all baselines, regardless of spatial prior. Notably, performance is further improved with priors that promote spatial coverage, compared to stochastic uniform sampling, which demonstrates the importance of an appropriate token placement scheme under sparsity constraints.

In Figure 5 we show image throughput versus accuracy, comparing SPoT with the baselines across varying sparsity levels. As sparsity increases, throughput improves significantly, albeit with an associated trade-off in accuracy. Notably, SPoT achieves the most favorable trade-off, maintaining substantially more of the full-model accuracy while enabling higher throughput than competing approaches. Further, we observe only slight variation in throughput between the models at each sparsity level, indicating that SPoT incurs very minimal computational overhead compared to baselines. We also include our oracle-guided variant SPoT-ON in the figure, which illustrates a ceiling on achievable performance when placements are ideally sampled.



**Figure 5.** We show ImageNet1k accuracy vs. throughput with 5 models at four sparsity levels. The **ceiling** denotes performance unlikely to be achieved given the intrinsic label noise in ImageNet [21]. The **gap** highlights the margin between SPoT with optimal configuration and SPoT-ON, illustrating possible performance gain through better token placement.

## 5.1. Robustness and Sensitivity Analysis

To thoroughly validate the semantic relevance of optimized subpixel token placements, we conduct targeted robustness analyses. Specifically, we evaluate performance under intentionally adversarial conditions—including inverse priors favoring irrelevant regions such as backgrounds or image boundaries, and gradient ascent adversarially maximizing loss or randomized label assignments. Our results in Table 6 demonstrate substantial performance degradation in all these adversarial scenarios, strongly suggesting that our token placement mechanism indeed leverages meaningful semantic cues rather than trivial spatial correlations or easily exploitable priors.

Importantly, the gradient-ascent oracle still receives the correct labels, so its sharp accuracy collapse shows that semantically aligned token positions are indispensable—simply keeping the right supervision is not enough if the tokens are pushed onto irrelevant regions. In contrast, the near-chance performance under label-obfuscation demonstrates that the model does not easily adapt to arbitrary image-label pairings, confirming that our selector grounds predictions in genuine object evidence rather than flexible location-label shortcuts. Background sampling leads to reduced performance; however, we posit that the token set still captures some object edges, providing the model with useful information. This hypothesis is reinforced by the significantly larger performance drop observed when sampling with a strong bias toward image boundaries. Adversarial optimization shows a similarly extreme drop, indicating that good token placements are not trivial.

**Table 5.** Classification top-1 and kNN accuracies for supervised and self-supervised models using different token priors. We find that center-bias in spatial priors is beneficial in sparse regimes, while coverage becomes more important as token budgets increase.

			25 Tokens		49 Tokens		100 Tokens		196 Tokens	
Model	Prior	Oracle	Acc@1	kNN	Acc@1	kNN	Acc@1	kNN	Acc@1	kNN
Backbone: CLS-IN21k										
ViT-B/16	Patch Grid		24.72	27.86	56.29	57.19	78.75	78.77	85.11	83.96
SPoT-B/16	Uniform		44.05	45.23	67.77	66.38	79.64	78.03	83.76	81.85
SPoT-B/16	Gaussian		45.22	45.27	68.64	66.96	79.75	77.74	83.45	81.48
SPoT-B/16	Sobol		43.67	46.48	69.02	68.60	81.63	79.35	84.66	82.62
SPoT-B/16	Isotropic		46.85	48.19	70.61	70.29	82.20	80.73	85.15	83.42
SPoT-B/16	Center		52.45	52.18	69.22	68.16	80.84	78.56	84.01	82.23
SPoT-B/16	Salient	✓	55.71	56.65	72.89	72.38	79.91	80.56	84.56	82.59
SPoT-ON-B/16	Isotropic	✓	81.70	70.65	94.28	88.58	95.97	92.92	96.12	93.52
Backbone: CLS-IN1k										
ViT-B/16	Patch Grid		9.24	12.05	41.05	44.38	71.22	71.41	79.14	77.64
SPoT-B/16	Uniform		29.87	33.88	60.64	60.84	74.44	73.18	79.38	77.36
SPoT-B/16	Gaussian		29.27	33.07	60.47	60.23	74.37	72.82	79.02	77.00
SPoT-B/16	Sobol		30.67	35.23	64.42	63.88	76.45	75.18	79.96	78.17
SPoT-B/16	Isotropic		33.52	37.84	66.18	66.25	77.58	76.29	80.61	79.04
SPoT-B/16	Center		39.91	42.47	63.04	62.65	75.41	73.63	79.32	77.71
SPoT-B/16	Salient	✓	39.83	43.72	66.32	66.00	74.36	75.25	79.54	78.03
SPoT-ON-B/16	Isotropic	✓	73.99	74.42	94.21	90.11	95.79	93.61	96.04	93.97
Backbone: MAE-IN1k										
ViT-B/16	Patch Grid		55.43	48.85	70.69	67.15	79.53	78.41	83.60	82.07
SPoT-B/16	Uniform		56.71	49.72	73.22	65.85	80.53	74.76	82.78	78.21
SPoT-B/16	Gaussian		57.58	49.49	72.51	65.59	80.31	74.52	82.55	77.90
SPoT-B/16	Sobol		60.62	53.54	75.71	68.71	82.19	76.24	83.51	79.09
SPoT-B/16	Isotropic		61.72	54.56	76.84	70.02	82.76	77.24	83.89	79.53
SPoT-B/16	Center		62.83	55.61	74.63	67.31	81.06	75.20	82.97	78.54
SPoT-B/16	Salient	✓	66.13	60.80	77.10	72.24	81.46	77.25	81.64	79.13
SPoT-ON-B/16	Isotropic	✓	90.93	79.73	94.87	87.87	96.09	90.76	96.24	91.28

**Table 6.** Analysis on harmful spatial priors and adversarial oracles in sparse regimes. The background prior samples from inverse saliency maps; the boundary prior samples with image edge bias. Ascent shows accuracy under worst-case token placements, discovered via SPoT-ON. Label obfuscation optimizes placements for randomized labels. Each case is compared with baseline SPoT performance using the isotropic prior—the **performance drop** is shown to the right of each score.

Backbone	Acc@1 (%) (25 Tokens)			
	Backgrd.	Boundary	Ascent	Lab.Obf.
CLS-IN21k	40.80 ↓ 6.85	10.68 ↓ 36.17	13.75 ↓ 33.09	1.17 ↓ 45.69
CLS-IN1k	20.06 ↓ 13.46	4.15 ↓ 29.37	5.90 ↓ 27.82	2.68 ↓ 33.52
MAE-IN1k	31.89 ↓ 29.83	10.83 ↓ 50.89	16.54 ↓ 45.18	1.11 ↓ 60.61

## 5.2. Retrofitting and Finetuning Details

Table 7 details the training configurations for the three backbone variants: CLS-IN1k, CLS-IN21k, and MAE-IN1k. The official TIMM [19] model card names are listed below each corresponding subtable for reference. All protocols use isotropic sampling and run for 50 epochs on  $224 \times 224$  images on the ImageNet-1k dataset [20]. Layer-wise learning rate decay (LLRD) is employed with a slightly more aggressive parameter in the MAE retrofitting, while the MAE finetuning follows the original protocol outlined by He et al. [8], with minor exceptions<sup>2</sup>.

<sup>2</sup>Cosine warmup in lr-scheduler, starting from  $1 \times 10^{-7}$  with peak learning rate of  $1 \times 10^{-3}$ , and gradient clipping set to 3.

## 6. Related Work

Leveraging sparsity to reduce computational overhead is a well-established research direction. Previous work introduced sparsity through masking during pre-training, in self-supervised [8] and language-supervised contexts [22]. Liu et al. [3] applied sparsity at the fine-tuning stage by initially upsampling images and subsequently randomly dropping patches, thus enhancing efficiency and reducing computational complexity. Distinct from training-centric sparsity approaches, our work induces sparsity during inference by retrofitting ViTs with a subpixel tokenizer, significantly improving throughput. Another line of research explores inference-time sparsity via selective pruning to either discard [4–6] or merge [7] tokens based on different heuristics. In contrast, our approach achieves sparsity by sampling rather than selectively pruning tokens during inference.

Recently, other works have explored non-grid based tokenization. One interesting line of research looks to leverage subobject tokenization, which extracts fine-grained segmentations as opposed to square patches [14, 23]. Other works apply learnable clustering into the transformer architecture via cross-attention operators [24, 25], while Nguyen et al. [26] proposed to tokenize each individual pixel. Deformable patches was first proposed in relation to object detection [27], but was further extended to general purpose modeling in ElasticViT [28], which proposed elastic windows as *local augmentations* in standard classification

**Table 7.** Training protocols for retrofitting. We use the same training protocols as He et al. [8] for MAE finetuning.

(a) CLS-IN1k Retrofitting		(b) CLS-IN21k Retrofitting		(c) MAE-IN1k Retrofitting	
config	value	config	value	config	value
sampler	isotropic	sampler	isotropic	sampler	isotropic
batch size	2048	batch size	2048	batch size	4096
epochs	50	epochs	50	epochs	50
dataset	ImageNet1k	dataset	ImageNet1k	dataset	ImageNet1k
img.size	$224 \times 224$	img.size	$224 \times 224$	img.size	$224 \times 224$
loss fn.	CE (0.1 smooth.)	loss fn.	CE (0.1 smooth.)	loss fn.	MSE
optimizer	AdamW	optimizer	AdamW	optimizer	AdamW
momentum	0.9, 0.99	momentum	0.9, 0.99	momentum	0.9, 0.95
lr.sched.	cos.decay (5 w.u.)	lr.sched.	cos.decay (5 w.u.)	lr.sched.	cos.decay (5 w.u.)
lr	$6e-5$	lr	$6e-5$	lr	$3e-3$
dropout path	0.1	dropout path	0.2	dropout path	0
opt. $\epsilon$	$1e-8$	opt. $\epsilon$	$1e-8$	opt. $\epsilon$	$1e-8$
augment	rrc / randaug(15, .5)	augment	rrc / randaug(15, .5)	augment	rrc
mixup $\alpha$	0.8	mixup $\alpha$	0.8	mixup alpha	0.8
cutmix $\alpha$	1.0	cutmix $\alpha$	1.0	cutmix	1.0
llrd	0.65	llrd	0.65	llrd	0.75
vit_base_patch16_224.augreg.in1k		vit_base_patch16_224.augreg2.in21k.ft.in1k		mae.vit_base_patch16.in1k	

**Table 8.** Comparing ElasticViT to SPoT in sparse regimes.

Model	Acc@1 (%) / Number of Tokens								
	39	59	78	98	118	137	157	176	196
ElasticViT	67.17	72.65	75.47	77.47	78.18	79.73	80.81	81.31	82.04
SPoT-MAE-IN1k	<b>71.81</b>	<b>77.86</b>	<b>80.47</b>	<b>81.89</b>	<b>82.55</b>	<b>83.05</b>	<b>83.34</b>	<b>83.52</b>	<b>83.85</b>

tasks. These are defined as stochastic patch perturbations in scale, position, and erasure via patch dropout. Put simply, ElasticViT relaxes the traditional patch grid of ViTs by randomly shifting, rescaling, and dropping patches during training.

ElasticViT differs from SPoT in key aspects. First, we relax the discrete grid assumption not by perturbing existing patch positions, but rather by directly sampling arbitrary continuous-valued points within the image. Conversely, ElasticViT uses discrete pixel positions, and does not adapt a continuous subpixel approach. Second, we do not explicitly train our model to handle sparse inputs; instead, our method’s inherent robustness to sparse token configurations naturally arises from training on continuously sampled points. Nevertheless, comparing our method to ElasticViT is insightful, as their approach is explicitly trained to handle continuous-valued positions and sparse token scenarios. Table 8 compares SPoT with ElasticViT’s officially reported results, demonstrating that SPoT consistently outperforms ElasticViT across all evaluated sparse configurations.

## 7. Conclusion

We proposed SPoT for extracting features at continuous subpixel positions, and used oracle-guided gradient search to probe the nature of optimal token placements and ideal spatial sampling priors. Our case studies showed that the flexibility of continuous off-grid placements improves performance out-of-the-box, especially in sparse token budget settings. SPoT-ON provided an estimate of best-case per-

formance from optimal token placement. Although placements are guided via an oracle, these optimal features *exist independently of how they were discovered*, revealing a performance gap that better informed priors could help bridge. While we focused on analyzing the effects of subpixel tokenization under varying sparsity configurations with different spatial priors, the development of learnable spatial priors is a next step towards narrowing the oracle performance gap. We emphasize that our study with SPoT on different spatial priors focused on classification on ImageNet, and optimal placements may vary for different datasets and downstream tasks.

By enabling continuous token positioning, SPoT facilitates gradient-based optimization of token placement, which can be advantageous in resource-constrained environments where sparsification is beneficial. Although we limit our scope to employ an oracle to determine optimal token placements, exploring oracle-independent strategies represents a compelling direction for future research. Specifically, integrating efficient saliency-driven objectives or heuristics during inference could potentially enhance throughput efficiency while maintaining competitive performance compared to models utilizing a full token budget. Further improvements may also be seen by allowing the model to adjust the patch window size dynamically during training. Moreover, while the scope of this work is towards modeling in sparse regimes, our results indicate that continuous subpixel token placements provide a novel research direction for ViTs on a more general level.

## References

- [1] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *J. R. Stat. Soc., B: Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, 1996. [1](#)
- [2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An Image is Worth  $16 \times 16$  Words: Transformers for Image Recognition at Scale,” in *Inter. Conf. Learn. Represent. (ICLR)*, 2021. [1](#), [3](#)
- [3] Y. Liu, C. Matsoukas, F. Strand, H. Azizpour, and K. Smith, “PatchDropout: Economizing vision transformers using patch dropout,” in *IEEE Wint. Conf. Appl. Comput. Vis. (WACV)*, 2023, pp. 3953–3962. [1](#), [6](#), [7](#)
- [4] X. Chen, Z. Liu, H. Tang, L. Yi, H. Zhao, and S. Han, “Sparsevit: Revisiting activation sparsity for efficient high-resolution vision transformer,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2023, pp. 2061–2070. [7](#)
- [5] Y. Rao, W. Zhao, B. Liu, J. Lu, J. Zhou, and C.-J. Hsieh, “DynamicViT: Efficient vision transformers with dynamic token sparsification,” in *Adv. Neural Inf. Process. Sys. (NeurIPS)*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 13 937–13 949.
- [6] H. Yin, A. Vahdat, J. M. Alvarez, A. Mallya, J. Kautz, and P. Molchanov, “A-ViT: Adaptive tokens for efficient vision transformer,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 10 809–10 818. [7](#)
- [7] D. Bolya, C.-Y. Fu, X. Dai, P. Zhang, C. Feichtenhofer, and J. Hoffman, “Token merging: Your ViT but faster,” in *Inter. Conf. Learn. Represent. (ICLR)*, 2023. [1](#), [7](#)
- [8] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Masked autoencoders are scalable vision learners,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2022, pp. 16 000–16 009. [1](#), [3](#), [6](#), [7](#), [8](#)
- [9] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. L. Yuille, and T. Kong, “Image BERT pre-training with online tokenizer,” in *Inter. Conf. Learn. Represent. (ICLR)*. OpenReview.net, 2022.
- [10] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. HAZIZA, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, “DINOv2: Learning robust visual features without supervision,” *Trans. Mach. Learn. Res.*, 2024. [1](#)
- [11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Adv. Neural Inf. Process. Sys. (NeurIPS)*, vol. 30, 2017. [1](#)
- [12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Conf. North Amer. Ch. Assoc. Comput. Ling. (NAACL)*, J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [1](#), [2](#)
- [13] G. L. Nemhauser and L. A. Wolsey, “Best algorithms for approximating the maximum of a submodular set function,” *Mathematics of Operations Research*, vol. 3, no. 3, pp. 177–188, 1978. [2](#)
- [14] M. Aasan, O. Kolbjørnsen, A. Schistad Solberg, and A. Ramírez Rivera, “A spitting image: Modular super-pixel tokenization in vision transformers,” in *European Conf. Comput. Vis. Wksp. (ECCVW)*, 2024. [2](#), [7](#)
- [15] I. Sobol, “On the distribution of points in a cube and the approximate evaluation of integrals,” *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86–112, 1967. [3](#)
- [16] G. Szabó and A. Horváth, “Mitigating the bias of centered objects in common datasets,” in *IEEE Inter. Conf. Pattern Recog. (ICPR)*, 2022, pp. 4786–4792. [3](#)
- [17] T. Lüddecke and A. S. Ecker, “Image segmentation using text and image prompts,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 7076–7086. [3](#)
- [18] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, “How to train your ViT? data, augmentation, and regularization in vision transformers,” *CoRR*, vol. abs/2106.10270, 2021. [3](#)
- [19] R. Wightman, “Pytorch image models,” 2019. [3](#), [6](#), [7](#)
- [20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*. Ieee, 2009, pp. 248–255. [3](#), [7](#)
- [21] L. Beyer, O. J. Hénaff, A. Kolesnikov, X. Zhai, and A. van den Oord, “Are we done with ImageNet?” *CoRR*, vol. abs/2006.07159, 2020. [6](#)
- [22] Y. Li, H. Fan, R. Hu, C. Feichtenhofer, and K. He, “Scaling language-image pre-training via masking,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2023, pp. 23 390–23 400. [7](#)
- [23] D. Chen, S. Cahyawijaya, J. Liu, B. Wang, and P. Fung, “Subobject-level image tokenization,” *ArXiv*, vol. abs/2402.14327, 2024. [7](#)
- [24] K. Fan, Z. Bai, T. Xiao, T. He, M. Horn, Y. Fu, F. Locatello, and Z. Zhang, “Adaptive slot attention: Object discovery with dynamic slot number,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 23 062–23 071. [7](#)
- [25] S. van Steenkiste, D. Zoran, Y. Yang, Y. Rubanova, R. Kabra, C. Doersch, D. Gokay, E. Pot, K. Greff, D. Hudson *et al.*, “Moving off-the-grid: Scene-grounded video representations,” in *Adv. Neural Inf. Process. Sys. (NeurIPS)*, vol. 37, 2024, pp. 124 319–124 346. [7](#)
- [26] D.-K. Nguyen, M. Assran, U. Jain, M. R. Oswald, C. G. Snoek, and X. Chen, “An image is worth more than  $16 \times 16$  patches: Exploring transformers on individual pixels,” in *Inter. Conf. Learn. Represent. (ICLR)*, 2025. [7](#)
- [27] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, “Vision transformer with deformable attention,” in *IEEE/CVF Inter. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 4794–4803. [7](#)

- [28] A. Pardyl, G. Kurzejamski, J. Olszewski, T. Trzciński, and B. Zieliński, “Beyond grids: exploring elastic input sampling for vision transformers,” in *IEEE Wint. Conf. Appl. Comput. Vis. (WACV)*. IEEE, 2025, pp. 8536–8545. [7](#)