BaSeOpt: Bayesian Image Sensor Layout Optimization for Efficient Vision Systems

Mishal Fatima[1] * **Danail Ignatovski**[1] **Petar Dimitrov Malamov**[1]
Jovita Lukasik[2] **Margret Keuper**[1,3]
[1] University of Mannheim [2] University of Siegen
[3] Max-Planck-Institute for Informatics, Saarland Informatics Campus

Abstract

Conventional camera sensors capture images on a uniform pixel grid, producing redundant data, high memory usage, and costly transmission regardless of the downstream task. We present **BaSeOpt**, a task-aware sensing framework that jointly optimizes sensor layouts and vision models for applications such as semantic segmentation. Instead of uniformly sampling every pixel, BaSeOpt allocates higher resolution to task-critical regions while sparsely sampling less informative areas, reducing acquisition overhead. To search the vast space of possible layouts, we formulate the problem as Bayesian Optimization in the latent space of a Variational Autoencoder trained on candidate layouts, enabling efficient discovery of promising configurations. Experiments demonstrate that BaSeOpt automatically identifies sensor layouts that accelerate data acquisition at the camera level, highlighting the benefits of co-optimizing sensing and inference for efficient vision systems.

1 Introduction

Designing efficient imaging sensors remains a critical challenge in computer vision and robotics. Modern neural networks achieve strong performance on tasks like classification and segmentation, but typically assume access to fully sampled, high-resolution images, an assumption that is costly in terms of memory, bandwidth, and computation, especially for real-world applications such as autonomous vehicles, drones, and edge devices with limited hardware. Traditional sensor designs sample uniformly across the image, even though background regions often provide redundant information while object regions are most important for prediction. This motivates adaptive approaches that allocate resolution selectively: recent work explores compressive sensing strategies that prioritize informative regions [Liu et al., 2024] and differentiable frameworks that jointly optimize sensor layouts and neural network parameters [Sommerhoff et al., 2024].

Optimizing such layouts is a high-dimensional problem, particularly when sensors must adapt to task and environment-specific constraints. Bayesian Optimization (BO) has shown promise in addressing similar challenges in domains ranging from hyperparameter tuning [Snoek et al., 2012] to aerospace [Perlini et al., 2024, Priem et al., 2020] and renewable energy design [Sheikh et al., 2022]. To efficiently explore the combinatorial layout space, candidate sensor configurations can be encoded into a continuous latent space using a Variational Autoencoder (VAE) [Kingma and Welling, 2014]. This representation enables BO to operate in a lower-dimensional, smooth space, allowing for faster convergence and better generalization across environments. By conditioning this optimization on environment-specific priors, the framework identifies layouts that focus resolution on the most informative regions while minimizing unnecessary data acquisition elsewhere.

*Email: mishal.fatima@uni-mannheim.de

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Learning to Sense.

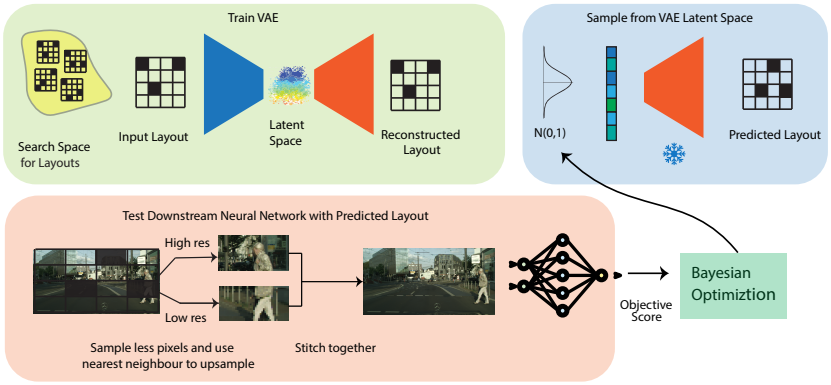

Figure 1: **Proposed Method**: we first train a Variational Autoencoder (VAE) on layout data to learn a compact latent representation. Afterwards, Bayesian Optimization (BO) samples a latent vector from the trained VAE's latent space, which is decoded to generate a candidate layout. Next, we apply this layout to the data and evaluate the target network with pretrained weights, obtaining a performance metric. Finally, Bayesian Optimization (BO) iteratively refines the latent vector sampling based on the observed performance, repeating the process to optimize the layout generation.

Overall, our approach offers a practical way to design task and environment-aware sensors, efficiently combining data acquisition and model computation to build high-performance vision systems for resource-limited settings.

2 Method

We consider a sensor layout comprising H rows and W columns of pixels arranged in a regular rectangular grid. The pixel grid is partitioned into N non-overlapping sub-regions, or *patches*, each of size $h \times w$ pixels, such that

$$H = h \cdot H_p, \quad W = w \cdot W_p, \quad N = H_p \cdot W_p,$$

where H_p and W_p denote the number of patches along the vertical and horizontal directions, respectively. Each patch P_k, for $k \in \{1, \ldots, N\}$, is assigned a resolution mode:

- **High-resolution mode**: all $h \times w$ pixels within P_k are sampled.
- **Low-resolution mode**: only a reduced subset $S_k \subset \{1, \ldots, h\} \times \{1, \ldots, w\}$ of pixels is sampled in a regular grid, with $|S_k| < h \times w$.

In our setting, the resolution mode assignment is *fixed* and does not change over time. This arises from the characteristics of the deployed imaging system, where the sensor is embedded in an environment-specific camera. Thus, the selection of low- and high-resolution patches must be determined prior to deployment and remains constant. The central challenge is to design this fixed spatial allocation to balance spatial coverage and pixel density, ensuring optimal downstream performance under a strict sampling budget.

2.1 Variational Autoencoder for Layout Encoding

To efficiently represent possible sensor layouts and capture dependencies among patches, we employ a *Variational Autoencoder (VAE)* [Kingma and Welling, 2014]. Let $\mathbf{M} \in \{0, 1\}^{H \times W}$ denote a binary layout of the sensor. The VAE introduces a latent variable $\mathbf{z} \in \mathbb{R}^D$ and models the joint distribution:

$$p_\theta(\mathbf{M}, \mathbf{z}) = p_\theta(\mathbf{M} \mid \mathbf{z})\, p(\mathbf{z}),$$

where $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a standard Gaussian prior, and $p_\theta(\mathbf{M} \mid \mathbf{z})$ is the decoder parameterized by θ.

Since the true posterior $p_\theta(\mathbf{z} \mid \mathbf{M})$ is intractable, Kingma and Welling [2014] introduced a variational approximation $q_\phi(\mathbf{z} \mid \mathbf{M})$ (encoder) with parameters ϕ:

$$q_\phi(\mathbf{z} \mid \mathbf{M}) = \mathcal{N}\big(\mathbf{z} \mid \boldsymbol{\mu}_\phi(\mathbf{M}), \mathrm{diag}(\boldsymbol{\sigma}^2_\phi(\mathbf{M}))\big),$$

where $\boldsymbol{\mu}_\phi(\mathbf{M})$ and $\boldsymbol{\sigma}^2_\phi(\mathbf{M})$ are outputs of the encoder network.

The VAE is trained by maximizing the evidence lower bound (ELBO):

$$\mathcal{L}(\theta, \phi; \mathbf{M}) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{M})}\big[\log p_\theta(\mathbf{M} \mid \mathbf{z})\big] - D_{\mathrm{KL}}\big(q_\phi(\mathbf{z} \mid \mathbf{M}) \,||\, p(\mathbf{z})\big),$$

where D_{KL} denotes the Kullback–Leibler divergence.

In our setup, the decoder produces logits that are passed through a sigmoid to generate a probability map. To satisfy the budget constraint, we derive a binary layout: the top-k lowest-probability patches are assigned as low-resolution (value 0), while the remaining patches are high-resolution (value 1). This procedure enforces a clear separation between regions and maintains consistency with the binary layout design.

2.2 Bayesian Optimization in Latent Space with TPE

Bayesian Optimization (BO) is a probabilistic framework for optimizing expensive black-box functions, particularly suitable when evaluations are costly, non-differentiable, or involve complex interactions. In our work, we use the Tree-structured Parzen Estimator [Bergstra et al., 2011] (TPE) sampler in Optuna Akiba et al. [2019], which approximates the objective function using a non-parametric surrogate based on probability densities over the latent vectors. Specifically, TPE models two densities:

$$l(\mathbf{z}) = P(\mathbf{z} \mid \text{good trials}), \quad g(\mathbf{z}) = P(\mathbf{z} \mid \text{bad trials}),$$

and selects the next candidate $\mathbf{z}_{t+1}$ by maximizing the ratio $l(\mathbf{z})/g(\mathbf{z})$. Here, good trials correspond to latent vectors whose decoded layouts achieve high downstream performance, while bad trials correspond to vectors with low performance. TPE determines these dynamically based on a chosen quantile of previously observed objective values.

In our approach, each latent vector $\mathbf{z} \in \mathbb{R}^D$ encodes a candidate sensor layout through a VAE, and the decoder ϕ maps $\mathbf{z}$ to a binary mask (including the deterministic post-processing):

$$\begin{aligned} \phi :&\mathbb{R}^d \to \{0,1\}^{H\times M} \\ &\mathbf{z} \mapsto \mathbf{M}. \end{aligned}$$

The objective function can then be expressed as

$$\mathcal{J}(\mathbf{z}) = \mathcal{J}(\phi(\mathbf{z})),$$

representing the downstream performance of the layout defined by $\phi(\mathbf{z})$. For example, in case of segmentation tasks, $\mathcal{J}(\mathbf{z})$ corresponds to the mean Intersection over Union (mIoU) on the validation set, computed between the ground-truth segmentation masks and the masks predicted by the neural network. TPE iteratively samples latent vectors $\mathbf{z}$, evaluates $\mathcal{J}(\mathbf{z})$, updates the density estimates l and g, and selects subsequent candidates. The optimal layout is finally given by

$$\mathbf{z}^* = \arg\max_{\mathbf{z}\in\mathbb{R}^D} \mathcal{J}(\mathbf{z}).$$

By performing optimization in the continuous latent space defined by the VAE, BO can efficiently explore the space of feasible layouts while inherently capturing dependencies among patches via the decoder ϕ.

3 Experiments

3.1 Binary Layout Generation and VAE Setup

We define a binary layout as an image composed of patches containing uniform values: 1s (high resolution) and 0s (low resolution). The budget is defined as the number of low-resolution patches in a layout. We experiment with budgets of 60, and 80. Cityscapes dataset [Cordts et al., 2016] is a large-scale benchmark for urban scene understanding containing high-resolution images of street scenes from 50 cities, annotated for tasks such as semantic segmentation and having original resolution 1024×2048. We construct layouts of size 128×256, where each patch has dimensions 16×16. This choice ensures that the layouts can later be upsampled by a factor of 8 to match the original Cityscapes resolution. To train the VAE, we generate a search space of 50,000 layouts, using

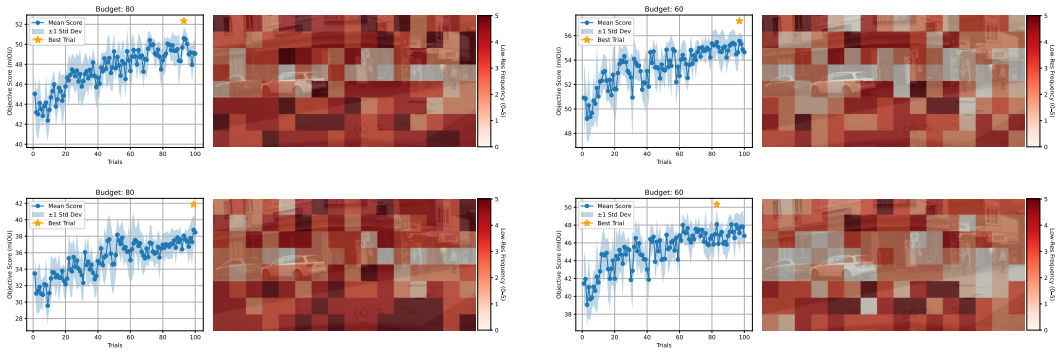

Figure 2: **Top row**: Results when 32×32 pixels are sampled from each 128×128 low-res patch, shown for budgets of 80 and 60. **Bottom row**: Results when 16×16 pixels are sampled from each 128×128 low-res patch, for the same budgets. The color bar indicates the frequency of low-resolution patch occurrences across all five random seeds. Across these seeds, most high-resolution patches tend to cluster near the center of the image, reflecting the typical spatial structure of Cityscapes scenes. The model used for this analysis is SegFormer.

90% for training and the remaining 10% for visualization. The VAE uses a latent dimension of 64 and is trained for 250 epochs. To enhance the learning process, we apply a Gaussian blur to both the layouts and their ground truths. The loss function combines mean squared error, KL divergence, and an additional regularization term that enforces the budget constraint. Specifically, this term computes the difference between the predicted and actual number of zero-valued pixels in the layout, ensuring that the learned layouts adhere to the specified budget.

3.2 Layout Optimization with Pretrained Segmentation Models

After training the VAE, we leverage pretrained segmentation model, **SegFormer** [Xie et al., 2021], trained on the full-resolution Cityscapes dataset, to optimize layout design. Specifically, we sample a latent vector of length 64 from the VAE's latent space and decode it into a layout of size 128×256. The decoded layout is converted to a binary mask by selecting the top-k lowest-probability patches as low-resolution, with the remaining patches treated as high-resolution. The layout is then upsampled by a factor of 8 to match the original Cityscapes resolution, resulting in patches of size 128×128. Next, within each low rsolution 128×128 patch, we sample a subset of pixels in a regular grid pattern and use nearest-neighbor interpolation to upsample them back to the original patch size. The reconstructed RGB images are then passed to the pretrained segmentation model, and the resulting mIoU is used as the optimization objective for Bayesian Optimization. For this purpose, we employ the TPE sampler from Optuna to efficiently explore the latent space.

In Figure 2, we present results for budgets of 60 and 80 (number of low resolution patches in the layout), varying the number of pixels sampled in low-resolution regions. In both cases, a smaller budget yields higher mIoU, while fewer sampled pixels per patch reduce performance. The resulting layouts are intuitive, with higher-resolution patches concentrated in central regions of greatest activity. In future work, we aim to fine-tune the model during each trial to further improve performance on mixed-resolution data while optimizing the layout.

4 Conclusion

In this paper, we have presented a new framework that combines generative modeling with layout optimization to guide resource allocation in complex vision tasks. By leveraging a Variational Autoencoder to explore the latent space of layouts and applying Bayesian Optimization to refine sampling, our approach efficiently identifies configurations that maximize task performance. Our experiments demonstrate that our proposed method consistently discovers meaningful patterns in layout allocation, highlighting its potential to adapt to different problem settings. This work opens avenues for further exploration of generative-optimization strategies in vision and beyond, including extensions to different data modalities and more sophisticated objective criteria.

5 Acknowledgments

The authors acknowledge support by the DFG research unit 5336 “Learning to Sense”.

References

Luyang Liu, Hiroki Nishikawa, Jinjia Zhou, Ittetsu Taniguchi, and Takao Onoye. Computer-vision-oriented adaptive sampling in compressive sensing. *Sensors*, 24(13):4348, 2024.

Hendrik Sommerhoff, Shashank Agnihotri, Mohamed Saleh, Michael Moeller, Margret Keuper, Bhaskar Choubey, and Andreas Kolb. Task driven sensor layouts-joint optimization of pixel layout and network parameters. In *2024 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10. IEEE, 2024.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

Alberto Perlini, Luca Abergo, and Giulio Gori. A hybrid bayesian-adjoint framework for aerodynamic shape optimization. In *AIAA Aviation Forum and Ascend 2024*, page 3753, 2024.

Remy Priem, Hugo Gagnon, Ian Chittick, Stephane Dufresne, Youssef Diouane, and Nathalie Bartoli. An efficient application of bayesian optimization to an industrial mdo framework for aircraft design. In *AIAA aviation 2020 forum*, page 3152, 2020.

Haris Moazam Sheikh, Tess A Callan, Kealan J Hennessy, and Philip S Marcus. Optimization of the shape of a hydrokinetic turbine's draft tube and hub assembly using design-by-morphing with bayesian optimization. *Computer Methods in Applied Mechanics and Engineering*, 401:115654, 2022.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, 2014. URL `https://arxiv.org/abs/1312.6114`.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pages 2623–2631, 2019.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.