Resource-Adaptivity Beyond the Model: Sensor Control for Quantized On-Device Vision

Hongjun Suh [1] **Woojin Jang** [1] **Hyung-Sin Kim** [1]

Abstract

Resource-adaptive inference is typically framed as a model-side problem: reduce precision, prune computation, route inputs, or shrink architectures. For camera-based on-device vision, however, the sensor is also a controllable resource that determines the input evidence before inference. We study this interaction on datasets with environment and sensor shifts by evaluating a diverse set of vision models under full-precision and quantized inference across auto-exposure and adaptive sensor policies. Our results show that adaptive sensing often yields larger gains than increasing model size or spending more bits on it: in many cases, compressed models with adaptive sensing outperform higher-precision models using auto-exposure. These findings suggest that resource-adaptive inference for embodied vision should evaluate model–sensor pairs and treat sensing as part of the resource budget. We further show that a simple entropy-based model-free policy can recover much of this benefit without scoring every candidate image with the target model.

1. Introduction

Efficient on-device deployment of neural networks has become a critical goal, and a wide range of resource-adaptive inference techniques have been proposed to this end: quantization reduces weight and activation precision (Gholami et al., 2022), pruning removes computation (Han et al., 2015), dynamic routing selects cheaper subnetworks per input (Shazeer et al., 2017), and early-exit or cascaded systems spend more compute only on hard inputs (Teerapittayanon et al., 2016; Bolukbasi et al., 2017). These approaches share a *model-centric* view: the input is treated as fixed, and only the computation that processes that input is adapted.

[1]Graduate School of Data Science, Seoul National University, Seoul, South Korea. Correspondence to: Hyung-Sin Kim <hyungkim@snu.ac.kr>.

Presented at the ICML 2026 Workshop "Resource-Adaptive Foundation Model Inference". Copyright 2026 by the author(s).

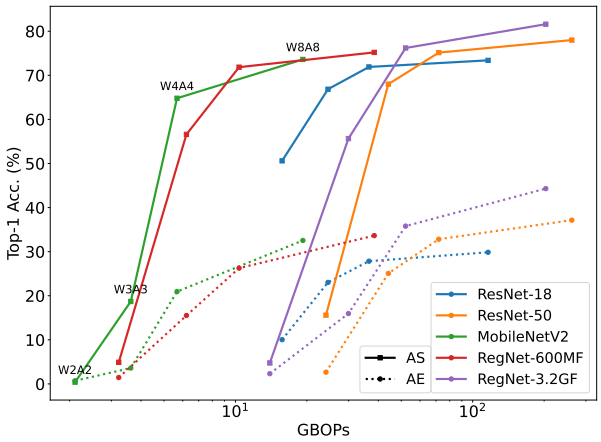

Figure 1. Accuracy on ImageNet-ES (Baek et al., 2024) across bit-widths and sensor policies. 'AS' refers to entropy-based adaptive sensing, and 'AE' refers to auto-exposure. Adaptive sensing shifts the PTQ frontier. Models are quantized with BRECQ (Li et al.).

This view is natural for cloud inference, but incomplete for embodied and on-device vision. For camera-based devices, the camera sensor itself determines what visual evidence is acquired *before* any neural computation begins. Sensor parameters such as exposure, gain, shutter speed, and aperture directly shape the input distribution that the model sees. Under challenging illumination or sensor conditions, a model may fail not because its architecture is too small or its arithmetic precision is too low, but because the captured image is poorly matched to the model (Baek et al., 2026). In such cases, allocating more bits or parameters provides only marginal benefit if useful visual evidence has already been degraded during acquisition.

Two resource axes: model-side and sensor-side. This motivates a different view of resource adaptivity for embodied vision: the system must allocate resources along *two axes*, not one.

Existing efficient inference methods allocate resources on the model side through precision, model capacity, or compute budget. For camera-based AI, however, sensing forms a second resource axis: the system can adapt exposure, gain, shutter speed, or aperture before inference begins. These

axes are coupled because acquisition quality directly shapes the input distribution seen by the model.

This coupling is important because the image that looks best to a human is not necessarily the image that is easiest for a neural network to classify (Baek et al., 2026; Choi et al., 2026), and a smaller or lower-precision model may perform well when the sensor provides a more model-friendly input. Recent physically-grounded benchmarks (Baek et al., 2024; Baek et al.) support this view: under real environmental and camera-sensor shifts, robustness is a property of the *model–sensor pair*, not of the model alone. Yet prior adaptive-sensing work has focused almost exclusively on full-precision models (Wong et al., 2024; Baek et al.; Paul et al., 2025), while practical on-device deployment relies heavily on quantized inference (Jacob et al., 2018; Krishnamoorthi, 2018). Therefore, it remains unclear how sensor adaptation interacts with quantized inference.

Our question and findings. We ask a simple question: *when both model-side and sensor-side resources are controllable, which resource matters more?*

We evaluate a diverse set of CNN and transformer models on ImageNet-ES (Baek et al., 2024) and ImageNet-ES Diverse (Baek et al.) under full-precision and quantized inference. For each model, we compare conventional auto-exposure (AE) with adaptive sensor control, allowing us to evaluate model precision, model scale, and sensor policy on a shared accuracy–resource frontier. Our results show that adaptive sensing can dominate marginal model-side scaling: under physical covariate shift, adapting the sensor often yields larger gains than scaling up the model or spending more bits on it, and in many cases compressed models with adaptive sensing outperform models at higher bit-widths—including full precision—using auto-exposure.

Contributions. Our contributions are:

- We reframe efficient on-device vision as a *model–sensor* resource allocation problem, arguing that resource-adaptive vision should be evaluated over *model–sensor frontiers*, where efficiency depends both on how the model processes inputs and how the system acquires them.
- We empirically show that under physical covariate shift, adaptive sensing can outperform increasing model precision, and that this trend holds across both CNN and transformer architectures.
- We introduce a simple model-free entropy-based sensing policy that recovers much of the adaptive-sensing benefit without invoking the target model on every candidate capture, providing a lightweight alternative whose cost does not scale with model size or bit-width.

Our broader message is simple: we should stop treating edge devices only as resource-poor versions of cloud servers. Embodied devices have unique resources, and for vision systems, sensors may be the most important one.

2. Related Work

Quantization. Quantization is a central technique for deploying vision models on resource-constrained devices, and methods generally fall into two families (Nagel et al., 2021; Gholami et al., 2022). Post-training quantization (PTQ) converts a pretrained high-precision network into a fixed-point one with little or no retraining, using only a small calibration set; representative methods include BRECQ (Li et al.) for CNNs and APHQ (Wu et al., 2025) for transformer architectures. Quantization-aware training (QAT) methods such as LSQ (Esser et al.) simulate quantization during training and typically reach better accuracy at low bit-widths, at the cost of longer training and labeled data. Most quantization evaluations assume a fixed input distribution and measure accuracy as precision is reduced from full precision down to lower-bit settings. For camera-based AI, this view is incomplete: under physical covariate shift, quantization error interacts with acquisition quality, motivating evaluation of quantized models as model–sensor pairs rather than as isolated networks.

Physical covariate shift and adaptive sensing. ImageNet-ES (Baek et al., 2024) introduces a physically-grounded robustness benchmark with covariate shifts induced by real environmental and camera-sensor factors (e.g., lighting, exposure, gain), rather than digital perturbations. Building on this setting, Lens (Baek et al.) shows that adaptive camera control can improve accuracy by selecting sensor parameters preferred by the model, demonstrating that robustness under physical covariate shift is a property of a model–sensor pair. Prior systems have explored adaptive sensing in real-world pipelines: CamTuner (Paul et al., 2025) uses reinforcement learning to tune camera parameters, while MadEye (Wong et al., 2024) adapts camera viewpoints (pan–tilt–zoom) to improve task performance. However, these works do not study the interaction between adaptive sensing and model efficiency. Our work extends this view to quantized on-device vision and asks whether adaptive sensing changes the relative value of model-side compression.

3. Experimental Setup

We design our experiments to isolate the interaction between model-side compression and sensor-side adaptation. Our goal is not to propose a new quantization method, but to evaluate whether adaptive sensing changes the accuracy–resource frontier of compressed on-device vision models

under physical covariate shift. We also suggest a deliberately minimal model-free sensing policy to test whether the sensing effect persists under compression.

3.1. Evaluation task and dataset

We evaluate image classification on ImageNet-ES (Baek et al., 2024) and ImageNet-ES Diverse (Baek et al.) under physical covariate shift induced by camera acquisition conditions. Each scene or source image is captured under multiple camera sensor settings, producing a set of candidate images that differ in exposure, gain, shutter speed, aperture, and lighting-dependent image quality. This setting allows us to compare a conventional camera pipeline, which provides a single auto-exposed image, against sensor policies that choose among multiple acquisition settings.

3.2. Models

We evaluate five CNNs (ResNet-18/50 (He et al., 2016), MobileNetV2 (Sandler et al., 2018), RegNetX-600MF, RegNetX-3.2GF (Radosavovic et al., 2020)) and seven transformer models from the ViT (Dosovitskiy et al.), DeiT (Touvron et al., 2021), and Swin (Liu et al., 2021) families.

3.3. Quantization

Our main experiments use PTQ with BRECQ (Li et al.) for CNNs and APHQ (Wu et al., 2025) for transformers. We evaluate FP32, W8A8, W4A4, W3A3, and W2A2. As supporting evidence, we additionally evaluate LSQ-based QAT (Esser et al.) on ResNet-18 and ResNet-50. Quantization settings follow the respective official implementations.

3.4. Sensor policies

For each model and quantization configuration, we compare sensor policies that differ in how they choose the image used for inference.

Auto-exposure. Auto-exposure is the default camera baseline: a conventional human-oriented pipeline that automatically selects sensor parameters to produce visually acceptable images. In most deployed systems, the vision model receives this image without controlling the acquisition process.

Adaptive sensing. Let s denote a scene and let $\mathcal{P} = \{p_1, \dots, p_N\}$ denote the set of available camera parameter settings. Capturing scene s with setting p produces an image $x_{s,p}$. The adaptive policy selects a sensor setting for each scene using a quality signal. For a scene s, the policy evaluates the candidate set $\mathcal{P}$ and selects

$$\hat{p}(s) = \operatorname*{argmax}_{p \in \mathcal{P}} Q(x_{s,p}),$$

where $Q(\cdot)$ is a quality estimator defined on the captured image $x_{s,p}$. We consider two choices of Q.

Adaptive Sensing 1: The first is a model-based signal,

$$Q_{\text{conf}}(x_{s,p}; M) = \max_c p_M(c \mid x_{s,p}),$$

the maximum softmax probability produced by model M, also known as Lens (Baek et al.). This requires running inference on all candidate images, increasing computational cost in terms of BitOPs (Bit OPerations) (Van Baalen et al., 2020).

Adaptive Sensing 2: As a lightweight alternative, we consider a model-free quality signal based on image entropy. For each candidate image $x_{s,p}$, we compute the Shannon entropy of its grayscale intensity distribution:

$$H(x_{s,p}) = -\sum_{i=1}^{L} p_i \log p_i,$$

where p_i is the empirical probability of intensity level i estimated from the image histogram, and L is the number of discrete grayscale levels. The quality score is then defined as

$$Q_{\text{ent}}(x_{s,p}) = H(x_{s,p}).$$

This tends to favor sensor settings that produce images with higher contrast without requiring access to the downstream model.

In all cases, the adaptive policy changes only which captured image is passed to the downstream model; the model architecture and quantization configuration remain fixed. This isolates the interaction between sensing and compression under a shared candidate-setting protocol.

4. Results

We plot classification accuracy against computational cost, where each point corresponds to a model–bit-width–sensor configuration and BOPs (Bit OPerations) are measured as the number of MACs (Multiply-Accumulate operations) scaled by the weight and activation bit-widths for a single forward pass (Van Baalen et al., 2020). Figure 1 summarizes the core qualitative result on the physical covariate shift dataset ImageNet-ES (Baek et al., 2024): under PTQ, Adaptive Sensing 2 shifts the accuracy–BOPs frontier upward relative to auto-exposure.

4.1. PTQ results across CNN and transformer architectures

Tables 1 and 2 extend the comparison beyond the illustrative frontier. The most important observation is that adaptive sensing can dominate marginal bit-width scaling. In several settings, a lower-precision PTQ model with adaptive

Table 1. Comparison of the top-1 accuracy (%) with different quantization bit-widths and sensor policies on CNN models quantized with BRECQ (Li et al.). 'IN-ES' refers to the 'ImageNet-ES' dataset (Baek et al., 2024), and 'IN-ES-D' refers to the 'ImageNet-ES Diverse' dataset (Baek et al.).

Dataset	Sensor Policy	Bits (W/A)	ResNet-18	ResNet-50	MobileNetV2	RegNet-600MF	RegNet-3.2GF
IN-ES	Auto-Exposure	32/32	30.00	37.34	33.25	33.55	44.22
		8/8	29.86	37.15	32.54	33.64	44.30
		4/4	27.84	32.82	20.94	26.26	35.80
		3/3	23.03	25.07	3.60	15.52	15.96
		2/2	10.04	2.66	**0.71**	1.45	2.32
	Adaptive Sensing 1 (Baek et al.)	32/32	72.55	**78.90**	73.15	75.25	80.85
		8/8	72.50	**78.50**	73.40	**75.70**	80.65
		4/4	**72.10**	**77.15**	64.30	**72.55**	75.60
		3/3	**67.25**	**70.10**	13.50	56.10	48.10
		2/2	47.00	14.00	0.45	3.25	**4.85**
	Adaptive Sensing 2 (Ours)	32/32	**73.50**	77.90	**73.35**	**75.50**	**81.75**
		8/8	**73.40**	78.00	**73.60**	75.20	**81.60**
		4/4	71.90	75.15	**64.80**	71.85	**76.20**
		3/3	66.85	68.00	**18.70**	**56.60**	**55.65**
		2/2	**50.60**	**15.60**	0.40	**4.90**	4.75
IN-ES-D	Auto-Exposure	32/32	13.61	18.98	16.57	15.84	18.30
		8/8	13.69	18.99	15.41	15.90	18.32
		4/4	13.36	17.73	5.65	12.19	14.98
		3/3	11.03	13.33	1.88	5.10	6.92
		2/2	4.82	1.62	0.43	0.87	1.05
	Adaptive Sensing 1 (Baek et al.)	32/32	**35.63**	**42.85**	**35.37**	**37.05**	42.87
		8/8	**35.53**	**42.75**	**34.45**	**36.82**	43.15
		4/4	**33.45**	**39.92**	14.25	30.93	35.10
		3/3	**29.08**	24.97	2.13	12.67	6.40
		2/2	12.57	1.22	**0.55**	1.18	0.85
	Adaptive Sensing 2 (Ours)	32/32	34.67	42.07	35.17	36.30	**43.18**
		8/8	34.60	42.22	33.68	36.25	**43.45**
		4/4	33.38	38.67	**18.62**	**31.40**	**35.68**
		3/3	28.55	**29.78**	**4.30**	**14.55**	**15.17**
		2/2	**15.02**	**3.07**	0.45	**1.25**	**1.47**

sensing outperforms a higher-precision PTQ model using auto-exposure. In particular, 3-bit adaptive models often exceed higher precision auto-exposure models including full precision. While most 2-bit models degrade sharply, ResNet-18 and several transformer models (e.g., DeiT-Base, Swin-Base) retain substantially more accuracy than other configurations.

4.2. Confidence-based vs. entropy-based adaptive sensing

We compare two adaptive sensing policies that differ in how candidate images are scored. Adaptive Sensing 1 follows Lens (Baek et al.) and uses the target model's maximum softmax confidence, while Adaptive Sensing 2 uses pixel entropy as a model-free quality estimator. Thus, the former is model-specific, whereas the latter selects a capture without querying the downstream model.

Across CNNs and transformers, entropy-based sensing is often competitive with confidence-based sensing and sometimes outperforms it. This suggests that the benefit of adaptive sensing is not tied to a single model-confidence heuristic. Instead, selecting a more favorable capture before inference can already recover a large fraction of the accuracy lost under auto-exposure.

Entropy is not intended as a direct measure of semantic relevance. Rather, under the exposure and illumination shifts present in ImageNet-ES and ImageNet-ES Diverse, it serves as a simple proxy for usable visual evidence. Extremely under-exposed images tend to collapse intensity distributions and exhibit low entropy, whereas well-exposed images generally contain richer contrast and intensity diversity. However, entropy can fail when high entropy arises from sensor noise, background clutter, or other task-irrelevant image content rather than object information. The comparison

Table 2. Comparison of the top-1 accuracy (%) with different quantization bit-widths and sensor policies on transformer models quantized with APHQ (Wu et al., 2025).

Dataset	Sensor Policy	Bits (W/A)	ViT-Small	ViT-Base	DeiT-Tiny	DeiT-Small	DeiT-Base	Swin-Small	Swin-Base
IN-ES	Auto-Exposure	32/32	23.59	37.73	47.61	57.17	64.98	68.24	71.01
		8/8	30.44	37.76	44.41	55.54	63.12	58.16	59.35
		4/4	19.05	31.80	36.91	49.70	58.90	60.98	64.66
		3/3	8.29	17.31	24.88	37.50	50.56	47.19	52.55
		2/2	0.59	1.34	5.11	9.43	15.94	10.24	16.11
	Adaptive Sensing 1 (Baek et al.)	32/32	**77.50**	**83.05**	75.20	82.70	85.25	87.50	88.25
		8/8	**79.75**	**85.60**	73.15	82.55	84.55	82.95	84.35
		4/4	**72.20**	**80.55**	**70.20**	80.50	83.40	85.20	87.30
		3/3	**53.55**	**69.30**	**61.60**	71.40	80.25	81.80	80.75
		2/2	**5.30**	**16.85**	**20.85**	**35.15**	**50.85**	35.85	38.95
	Adaptive Sensing 2 (Ours)	32/32	69.85	81.70	**77.25**	**84.70**	**86.35**	**89.55**	**90.00**
		8/8	69.25	83.15	**75.55**	**83.90**	**86.25**	**84.25**	**85.50**
		4/4	56.40	72.25	69.85	**81.85**	**84.50**	**87.25**	**88.40**
		3/3	30.50	51.05	59.50	**71.70**	**80.80**	**82.50**	**82.70**
		2/2	1.60	5.60	17.95	34.65	48.05	**39.40**	**45.10**
IN-ES-D	Auto-Exposure	32/32	26.01	30.90	21.46	31.14	34.76	39.80	41.88
		8/8	28.37	34.58	20.15	29.51	34.45	31.55	30.55
		4/4	20.18	28.16	18.00	26.77	31.75	33.81	36.33
		3/3	12.86	15.25	12.73	18.33	26.32	24.24	25.81
		2/2	2.13	3.04	3.62	5.71	7.30	6.06	6.06
	Adaptive Sensing 1 (Baek et al.)	32/32	**57.70**	62.22	44.67	55.70	60.67	65.88	68.65
		8/8	**55.90**	65.32	42.22	52.92	60.45	57.17	60.07
		4/4	**48.73**	57.13	37.00	49.37	56.48	61.13	64.55
		3/3	**30.05**	40.15	27.12	38.80	50.00	48.15	50.45
		2/2	**3.83**	5.92	6.95	**12.90**	18.43	10.73	11.75
	Adaptive Sensing 2 (Ours)	32/32	56.75	**66.67**	**47.92**	**58.48**	**62.15**	**67.03**	**70.78**
		8/8	54.62	**67.60**	**45.52**	**55.82**	**61.70**	**57.62**	**60.88**
		4/4	46.47	**58.27**	**40.60**	**52.42**	**57.92**	**62.40**	**66.38**
		3/3	28.65	**40.25**	**29.02**	**40.78**	**50.40**	**48.72**	**51.80**
		2/2	3.47	**6.18**	**7.12**	12.58	**18.47**	**13.40**	**13.93**

also reveals a practical tradeoff. Confidence-based sensing can exploit the target model's own preferences, but its quality estimate may degrade when aggressive PTQ distorts the model's confidence landscape. Entropy-based sensing is less model-dependent and provides a simple model-free alternative. Appendix B further analyzes failure modes of both sensing policies under different candidate subsets.

4.3. Supporting QAT results

Table 3 shows that the adaptive-sensing trend also holds under a different quantization pipeline. Notably, QAT models retain strong performance across sub-8-bit regimes, including 2-bit, suggesting that quantization-aware training in the source domain improves robustness under both physical covariate shift and extreme quantization. We further observe that confidence-based adaptive sensing generally outperforms entropy-based sensing in the QAT setting. A possible explanation is that QAT-trained models learn to utilize their limited bit-width more effectively during training, producing more reliable confidence estimates. As a result, model-based selection provides a stronger signal for choosing sensor parameters, whereas in PTQ models, less stable predictions reduce the advantage of confidence-based scoring.

4.4. Latency and system-level tradeoffs

Table 4 reports measured inference latency on a Jetson Orin Nano together with an estimated total latency model for adaptive sensing. We model total latency as

$$L_{\text{total}} = L_{\text{capture}} + L_{\text{inference}} + L_{\text{score}} + \alpha,$$

where capture latency is approximated using shutter speed, inference latency is measured directly on-device, score computation corresponds to candidate selection, and α captures additional deployment overheads not directly measured in our setup, such as parameter switching and data movement. Because the auto-exposure images in both datasets do not provide a fixed shutter speed value, we approximate capture latency using the average shutter speed of the candidate parameter set. Score computation of the evaluated adaptive sensing policies is generally trivial compared to capture and inference cost.

Table 3. Comparison of the top-1 accuracy (%) with different quantization bit-widths and sensor policies on ResNet models (He et al., 2016) quantized with LSQ (Esser et al.).

Dataset	Sensor Policy	Bits (W/A)	ResNet-18	ResNet-50
IN-ES	Auto-Exposure	32/32	15.06	20.43
		8/8	12.86	12.56
		4/4	13.05	13.93
		2/2	11.03	11.97
	Adaptive Sensing 1 (Baek et al.)	32/32	**59.50**	66.35
		8/8	**58.45**	**60.55**
		4/4	**57.85**	**60.60**
		2/2	**55.60**	**59.30**
	Adaptive Sensing 2 (Ours)	32/32	58.10	**67.55**
		8/8	55.90	60.30
		4/4	56.05	58.40
		2/2	52.55	55.65
IN-ES-D	Auto-Exposure	32/32	11.08	13.46
		8/8	11.77	11.81
		4/4	10.59	11.73
		2/2	8.66	9.68
	Adaptive Sensing 1 (Baek et al.)	32/32	**28.25**	29.83
		8/8	**29.27**	**30.17**
		4/4	**28.07**	**32.05**
		2/2	**23.65**	**27.07**
	Adaptive Sensing 2 (Ours)	32/32	25.98	**30.35**
		8/8	26.42	25.85
		4/4	24.73	29.52
		2/2	19.58	23.40

The results reveal a systems-level distinction between the two adaptive sensing policies. Confidence-based sensing (Adaptive Sensing 1) requires evaluating all candidate captures with the downstream model, causing inference to become the dominant cost, especially for higher-capacity FP32 models. In contrast, entropy-based sensing (Adaptive Sensing 2) performs only a single downstream inference and replaces model-based scoring with lightweight image-statistic computation, shifting the dominant cost toward image acquisition rather than neural-network evaluation. The relative importance of these costs depends on the deployment regime: inference may dominate on resource-constrained edge devices, while capture latency can dominate under long-exposure or high-throughput settings. We additionally explored smaller candidate subsets grouped by shutter speed as a practical latency-reduction strategy and report these results in the Appendix B.

5. Conclusion and Limitations

Limitations. Our low-bit experiments use simulated quantization in PyTorch. While this allows controlled evaluation of the interaction between sensing and quantization, not all 2-bit, 3-bit, or 4-bit configurations directly map to optimized edge runtimes; commodity deployment stacks such as TensorRT provide substantially stronger support for FP16 and INT8 inference than for W4A4, W3A3, or W2A2 deployment. In addition, our latency analysis combines measured inference latency with an estimated total latency model based on shutter speed and candidate evaluation rather than a complete capture-to-deployment pipeline. Lastly, our evaluation focuses on image classification. Extending sensor-aware quantization analysis to detection, segmentation, tracking, and multimodal perception remains important future work.

Table 4. Estimated end-to-end latency (s) on **Jetson Orin Nano** for ResNet-18/50 under FP and INT8 inference. AE refers to Auto-Exposure, AS1 refers to Adaptive Sensing 1, and AS2 refers to Adaptive Sensing 2.

			Time (s)		
Model	Bits	Policy	Capture	Inference	Total
ResNet-18	32	AE	0.09	0.11	0.20
		AS1	2.41	5.93	8.34
		AS2	2.41	0.11	2.52
	8	AE	0.09	0.11	0.20
		AS1	2.41	2.34	4.75
		AS2	2.41	0.11	2.52
ResNet-50	32	AE	0.09	0.22	0.31
		AS1	2.41	22.28	24.69
		AS2	2.41	0.21	2.62
	8	AE	0.09	0.25	0.34
		AS1	2.41	5.90	8.31
		AS2	2.41	0.25	2.66

Conclusion. We studied adaptive sensing as a resource for on-device vision under physical covariate shift. Across CNN and transformer architectures, adaptive sensing substantially improves the accuracy of quantized models, and in many cases lower-precision models with adaptive sensing outperform higher-precision or full-precision models using auto-exposure. These findings suggest that resource adaptivity for camera-based AI should include how inputs are acquired, not only how they are processed. More broadly, our results indicate that under physical covariate shift, better sensing may be more valuable than spending additional bits on the model.

References

Baek, E., Han, S.-h., Gong, T., and Kim, H.-S. Adaptive camera sensor for vision models. In *The Thirteenth International Conference on Learning Representations*.

Baek, E., Park, K., Kim, J., and Kim, H.-S. Unexplored faces of robustness and out-of-distribution: Covariate shifts in environment and sensor domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22294–22303, 2024.

Baek, E., Park, K., Ko, J., Oh, M.-h., Gong, T., and Kim, H.-S. Position: Ai should sense better, not just scale bigger: Adaptive sensing as a paradigm shift. *Advances in Neural Information Processing Systems*, 38, 2026.

Bolukbasi, T., Wang, J., Dekel, O., and Saligrama, V. Adaptive neural networks for efficient inference. In *International conference on machine learning*, pp. 527–536. PMLR, 2017.

Choi, Y. R., Park, S., and Kim, H.-S. [emerging ideas] artificial tripartite intelligence: A bio-inspired, sensor-first architecture for physical ai. In *Proceedings of the 24th Annual International Conference on Mobile Systems, Applications and Services*, pp. 839–853, 2026.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Esser, S. K., McKinstry, J. L., Bablani, D., Appuswamy, R., and Modha, D. S. Learned step size quantization. In *International Conference on Learning Representations*.

Gholami, A., Kim, S., Dong, Z., Yao, Z., Mahoney, M. W., and Keutzer, K. A survey of quantization methods for efficient neural network inference. In *Low-power computer vision*, pp. 291–326. Chapman and Hall/CRC, 2022.

Han, S., Pool, J., Tran, J., and Dally, W. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., and Kalenichenko, D. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2704–2713, 2018.

Krishnamoorthi, R. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*, 2018.

Li, Y., Gong, R., Tan, X., Yang, Y., Hu, P., Zhang, Q., Yu, F., Wang, W., and Gu, S. Brecq: Pushing the limit of post-training quantization by block reconstruction. In *International Conference on Learning Representations*.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Nagel, M., Fournarakis, M., Amjad, R. A., Bondarenko, Y., Van Baalen, M., and Blankevoort, T. A white paper on neural network quantization. *arXiv preprint arXiv:2106.08295*, 2021.

Paul, S., Rao, K., Coviello, G., Sankaradas, M., Hu, Y. C., and Chakradhar, S. T. Camtuner: Adaptive video analytics pipelines via real-time automated camera parameter tuning. *IEEE Transactions on Mobile Computing*, 2025.

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., and Dollár, P. Designing network design spaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10428–10436, 2020.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., and Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Teerapittayanon, S., McDanel, B., and Kung, H.-T. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd international conference on pattern recognition (ICPR)*, pp. 2464–2469. IEEE, 2016.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

Van Baalen, M., Louizos, C., Nagel, M., Amjad, R. A., Wang, Y., Blankevoort, T., and Welling, M. Bayesian bits: Unifying quantization and pruning. *Advances in neural information processing systems*, 33:5741–5752, 2020.

Wong, M., Ramanujam, M., Balakrishnan, G., and Netravali, R. {MadEye}: Boosting live video analytics accuracy with adaptive camera configurations. In *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, pp. 549–568, 2024.

Wu, Z., Zhang, J., Chen, J., Guo, J., Huang, D., and Wang, Y. Aphq-vit: Post-training quantization with average perturbation hessian based reconstruction for vision transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9686–9695, 2025.

A. Experimental details

A.1. Hardware

For system latency measurements, we utilized 4 out of the 6 CPU cores on the Jetson Orin Nano. We perform 100 warm-up runs and compute the average latency over 100 runs.

A.2. Quantization

For deployment with integer-only inference, we apply post-training static quantization using `torch.ao.quantization` with the `qnnpack` backend, which is required for ARM-based Jetson hardware. Convolution and linear layers are individually wrapped with `QuantStub/DeQuantStub` pairs to define the quantization boundaries.

B. Latency breakdown details

This appendix expands on the system-level analysis in Section 4.4 by reporting the components of total latency separately and by examining how reducing the candidate set affects both capture latency and accuracy. Three pieces of measurement combine into the total-latency model used in the main text: per-batch inference latency on the target device (Table 5), per-group capture latency derived from shutter speed (Table 6), and downstream accuracy as a function of the candidate subset (Tables 7–8). Together these decompose the cost of adaptive sensing along the same model-side and sensor-side axes used throughout the paper.

Table 5. Inference latency (s) on Jetson Orin Nano for ResNet-18 and ResNet-50 under FP and INT8 inference across batch sizes.

	ResNet-18		ResNet-50	
Batch size	FP32	INT8	FP32	INT8
1	0.11	0.11	0.22	0.25
9	0.87	0.78	1.76	1.80
18	4.88	1.56	18.38	3.99
27	5.93	2.34	22.28	5.90

B.1. Inference latency across batch sizes (model side)

Table 5 reports inference latency on Jetson Orin Nano for ResNet-18 and ResNet-50 at FP32 and INT8 across the batch sizes that correspond to the candidate-set sizes used in our adaptive sensing setting. The benefit of spending fewer bits on the model grows with batch size: at batch 1, INT8 provides little or no latency advantage over FP32, while at larger batch sizes the benefit becomes substantial, particularly for ResNet-50. This suggests that the practical value of quantization depends not only on model precision, but also on deployment configuration and workload characteristics.

This behavior matters for the system-level comparison between sensor policies. Confidence-based sensing (Adaptive Sensing 1) runs the downstream model on every candidate capture and therefore directly inherits this batch-scaling cost, while entropy-based sensing (Adaptive Sensing 2) only runs the model once and stays at the batch-1 column regardless of candidate set size. As a result, the deployment advantage of lower-precision inference becomes increasingly important as the number of candidate captures grows.

Table 6. Capture latency (s) for shutter-speed-grouped candidate subsets. Groups F (Fast), M (Medium), and S (Slow) denote shutter-speed buckets, and combinations such as F+M denote unions of buckets. N_{capture} is the number of candidate captures and L_{capture} (s) is the total capture latency.

Group	N_{capture}	L_{capture} (s)
F	9	0.009
M	9	0.150
S	9	2.250
F+M	18	0.159
M+S	18	2.400
F+M+S	27	2.409

B.2. Capture latency for shutter-speed groups (sensor side)

Table 6 reports capture latency L_{capture} for candidate subsets grouped by shutter speed: Fast(F), Medium(M), Slow(S). Capture latency is essentially additive in shutter time, so the cost is dominated by the slowest bucket in the union. The Slow group alone accounts for 2.250 s out of the 2.409 s spent capturing the full F+M+S set, while Fast and Medium together contribute only 0.159 s. Unlike the model-side axis, where reducing inference cost requires either compressing the network or shrinking the architecture, this axis offers a different lever: trimming the candidates that the sensor needs to physically expose for.

B.3. Shutter-speed-grouped subsets

Tables 7 and 8 report top-1 accuracy of ResNet-18 and ResNet-50 (BRECQ) under each shutter-speed subset. Contrary to the intuition that larger candidate pools should always improve adaptive sensing, adding more captures does not necessarily improve accuracy. On IN-ES, the full F+M+S set is rarely the best: the Medium bucket alone often matches or exceeds it for confidence-based sensing (e.g., ResNet-18: 74.05 at M vs. 72.55 at F+M+S, FP32), while the Slow bucket is frequently best for entropy-based sensing. On IN-ES-D the pattern collapses onto a single bucket — Slow dominates throughout and Fast drops to near-random performance (3–5%) — so the candidate set must include Slow under low-light shift.

Table 7. Top-1 accuracy (%) of ResNet-18 quantized with BRECQ (Li et al.) on candidate subsets grouped by shutter speed. F, M, S denote Fast, Medium, and Slow shutter-speed subsets; F+M, M+S, and F+M+S denote their unions, with F+M+S corresponding to the full candidate set used in Table 1.

Dataset	Sensor Policy	Bits (W/A)	F	M	S	F+M	M+S	F+M+S
IN-ES	Adaptive Sensing 1 (Baek et al.)	32/32	71.90	74.05	72.50	72.50	73.55	72.55
		8/8	72.25	73.90	72.20	72.95	73.35	72.50
	Adaptive Sensing 2 (Ours)	32/32	72.10	74.20	75.05	73.05	74.25	73.50
		8/8	72.10	73.60	75.05	72.95	74.00	73.40
IN-ES-D	Adaptive Sensing 1 (Baek et al.)	32/32	2.95	26.13	37.30	25.80	35.67	35.63
		8/8	2.95	26.02	37.18	25.67	35.55	35.53
	Adaptive Sensing 2 (Ours)	32/32	3.72	26.15	35.75	26.13	34.67	34.67
		8/8	3.78	26.13	35.75	26.12	34.60	34.60

Table 8. Top-1 accuracy (%) of ResNet-50 quantized with BRECQ (Li et al.) on candidate subsets grouped by shutter speed.

Dataset	Sensor Policy	Bits (W/A)	F	M	S	F+M	M+S	F+M+S
IN-ES	Adaptive Sensing 1 (Baek et al.)	32/32	79.35	79.50	77.10	79.15	79.30	78.90
		8/8	78.90	79.40	76.95	79.10	78.90	78.50
	Adaptive Sensing 2 (Ours)	32/32	78.05	77.45	79.75	77.30	78.55	77.90
		8/8	78.50	77.60	79.45	77.40	78.55	78.00
IN-ES-D	Adaptive Sensing 1 (Baek et al.)	32/32	4.20	31.77	44.40	31.28	43.10	42.85
		8/8	4.05	32.00	44.18	31.63	42.90	42.75
	Adaptive Sensing 2 (Ours)	32/32	5.27	32.90	43.03	32.88	42.07	42.07
		8/8	5.47	32.65	43.18	32.63	42.22	42.22

This non-monotonicity arises because both sensing policies select a maximum under heuristics that are not calibrated predictors of correctness. Under confidence-based sensing, poorly matched captures can attract spuriously high-confidence predictions, particularly under PTQ where quantization perturbs the confidence landscape. Under entropy-based sensing, low-light captures may exhibit elevated sensor noise that inflates pixel entropy despite containing weaker semantic signal. Adding candidates therefore helps only when the additional captures are well matched to the operating regime of the selection statistic. INT8 tracks FP32 closely throughout, suggesting that within this setting the subset structure is largely independent of model-side bit allocation.

These patterns combine with the latency components in Tables 5 and 6 to define practical operating points. Confidence-based sensing scales with N on both capture and inference cost, while entropy-based sensing scales primarily with capture cost because downstream inference remains fixed at batch 1. More broadly, these results suggest that the candidate set itself should be treated as a tunable part of the model–sensor pair rather than as a fixed protocol reused across environments.