NEUROSCAPE IMAGING PRIVATE LIMITED

Spum

Physical AI Inference for Biomedical Instruments

Scientific Perception Uniting Mind & Matter

Technical white paper Version 1.0 / 2 November 2026

Prepared by Neuroscape Imaging Private Limited
Vinayak Pathak, Founder and Biomedical Engineer

Abstract

Spum is a local inference engine for language and image models, developed to bring machine intelligence closer to biomedical instruments and their operators. Its public interfaces combine a command-line runtime, a local model-serving interface and a standalone Mac application. This paper describes that product boundary, the role of local inference in physical AI, selected engineering measurements and an evaluation framework for biomedical use. It distinguishes runtime performance from model accuracy and instrument integration from autonomous control. The emphasis is on useful, inspectable workflows: an acquired image or instrument observation becomes model input; an operator receives a response and can examine the underlying evidence. Proprietary implementation methods are outside the scope of this publication.

Keywords Edge inference; biomedical imaging; physical AI; local computing; vision-language models; instrument software.

1 Purpose and product boundary

Biomedical instruments produce information under constraints that differ from a general web application. Images may be large, connectivity may vary, and an operator may need to interpret a result beside the device. Local inference makes it possible to execute a supported model on the host computer without sending each inference request to a remote model service. Model acquisition, software updates and optional external connections remain separate network activities.

Spum supplies the runtime that executes a selected model. Its output depends on that model, its weights, input preparation and generation settings. A faster runtime does not by itself establish better diagnostic accuracy. Likewise, running a medical foundation model does not turn the host application into a validated clinical system.

1.1 Physical AI as an instrument workflow

In this paper, physical AI refers to intelligence connected to observations from a physical instrument and used within an operator's workflow. It need not imply a robot or autonomous actuation. For Spum, the central interaction is observation, local inference and human interpretation. Device control, feedback stability and safety interlocks require their own integration and verification; they are not implied by a successful chat or image response.

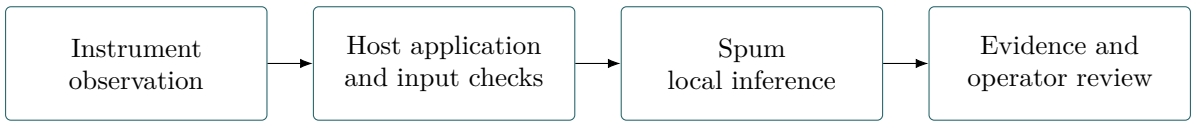


Figure 1: Public workflow boundary. The diagram describes system responsibilities rather than internal engine architecture.

1.2 Public interfaces

The command-line interface supports direct local runs and performance inspection. The local serving interface allows an application to request model output through a familiar chat-completion pattern. The standalone Mac application provides a desktop workflow with conversation history, model management, image attachment, light and dark appearances, and menu-bar access. Availability of a particular input modality depends on the loaded model and the execution path.

Apple silicon is the primary desktop deployment target. Spum also has a separate NVIDIA execution path for supported models. A result obtained on one backend is not evidence that all models, image paths or application features are available on another. Compatibility should be checked for the exact release and model artifact being deployed.

1.3 Data handling

Local model execution can reduce transfers of instrument data to remote inference services. It does not automatically provide access control, de-identification, encryption, retention governance or regulatory compliance. The integrating application must define storage locations, deletion behavior and any permitted external connections. The model's own distribution and use conditions also remain applicable.

2 Engineering measurements and interpretation

The following results reproduce selected project-reported measurements documented with Spum. They are engineering observations, not independent certification or clinical validation. The selected Apple M5 Max comparison used high-performance power mode and medians of five interleaved rounds against llama.cpp with its Metal backend. Each row uses the same named GGUF model and quantization across those two engines. Prefill and decode are separate phases; their rates should not be combined into a single speed claim.

Model and quantization	Prefill (tokens/s)		Decode (tokens/s)	
	Spum	llama.cpp	Spum	llama.cpp
Llama 3.2 1B, Q4_K_M	14,951	13,994	504	436
Qwen 2.5 0.5B, Q8_0	24,707	20,629	623	433
Qwen 2.5 1.5B, Q4_K_M	11,373	8,818	315	293
TinyLlama 1.1B, Q4_K_M	13,255	13,264	440	442

Table 1: Selected project-reported Apple M5 Max results. These figures are specific to the documented model and test configuration.

The Llama 3.2 Q4_K_M decode row was remeasured on 1 November 2026 using greedy generation. The project also reports approximately 455 tokens/s with temperature 0.7 for that model. TinyLlama is essentially level with the comparison engine. These observations support a model-specific performance description, not a claim that Spum is faster for every workload. The model-family literature provides context for the models themselves, rather than validation of Spum's measurements [1].

2.1 Latency is an end-to-end quantity

For an interactive instrument application, a useful latency budget is

$$T_{\text{first response}} = T_{\text{acquire}} + T_{\text{prepare}} + T_{\text{image}} + T_{\text{prefill}} + T_{\text{first token}} + T_{\text{display}}. \tag{1}$$

An engine benchmark may measure only part of this sum. A short arithmetic response, a cached image and a cold image request are different workloads. The image-encoding time should be stated separately from prompt processing and text generation. Repeated use of a cached input must not be presented as first-use image performance.

2.2 Reproducibility requirements

A comparison should identify hardware and power state; software versions; exact model artifact and quantization; prompt and output lengths; sampling settings; warm-up and cache state; concurrency; and the number and order of trials. Report time to first token, sustained decode rate and memory use separately. Disclose any difference in model format, output length or decoding behavior between systems. Archive the input, response and timing record together when sharing an application demonstration.

The table is a compact historical summary. Raw trial distributions and a complete environment manifest are not included in this public paper. Readers should therefore treat the values as project-reported measurements and re-run a matched workload before making a procurement or integration decision.

3 Biomedical and instrument applications

3.1 Image-grounded assistance

Medical vision-language models can connect an image with a textual question. MedGemma is an example of a model family developed for medical image and text understanding [2]. Spum's role is to execute a compatible artifact and return its response locally. The model's published results do not transfer automatically to a quantized artifact, a new patient population or a Spum deployment.

A chest-radiograph demonstration should retain the actual input image, exact prompt, generated response and measured timings. A plausible sentence is not proof that a lesion was recognized correctly. More demanding questions require evidence matched to the task: classification needs reference labels; localization needs spatial annotations; and report generation needs clinically meaningful error assessment. This paper does not claim validated pleural-effusion segmentation or diagnostic performance for Spum.

3.2 Localization and structured outputs

Segmentation provides a spatial output that can be checked against an image. U-Net established a widely used biomedical segmentation approach with an encoder-decoder structure [3]. A segmentation model and a language model can play complementary roles, but their combination still needs end-to-end testing. A language explanation must not be allowed to overwrite an uncertain or failed localization result. Citing U-Net here does not claim that this architecture is shipped in the current Spum release.

3.3 Microcamera arrays and large-area imaging

Multi-camera array microscopy demonstrates how parallel image acquisition can produce large, high-resolution composites, including histopathology specimens [4]. This published work includes Vinayak Pathak as a coauthor and supplies relevant imaging context. It does not constitute a benchmark or validation of Spum.

A local inference layer could support region selection, image review and contextual assistance in such a workflow. Acquisition, calibration, stitching, inference and display have different costs and failure modes. Live stitching accuracy and large-slide throughput require direct measurements on the integrated system; language-model decode rates cannot establish them. These are application opportunities rather than a claim that every component is presently integrated with Spum.

3.4 Portable imaging and interactive instruments

Pocket endoscopy is one instrument context within the broader Neuroscape portfolio. It should be evaluated through image quality, capture stability, operating conditions and task-specific model behavior. Similarly, an interactive piano workflow can use local generation as one component of a physical interface. Musical continuation and user interaction are distinct from evidence of therapeutic efficacy. No PTSD treatment outcome or clinical benefit is established by an inference demonstration.

Across these examples, the unifying engineering objective is to put responsive computation close to the observation while making the output inspectable by the operator. Different instruments need different models, input checks, response formats and acceptance criteria.

4 Evaluation and deployment discipline

A biomedical application needs evidence at three levels. **Runtime correctness** asks whether the engine executes the intended model and preserves the required behavior. **Model validity** asks whether predictions are accurate on representative held-out data. **Workflow validity** asks whether the integrated system is useful and safe for its intended users and operating environment. Passing one level does not establish the others.

Layer	Example evidence	What it does not establish
Runtime	Matched outputs, numerical checks, latency and memory measurements	Clinical correctness of an answer
Model	Patient-disjoint test data, expert labels, subgroup and external evaluation	Usability or safe device control
Workflow	Operator study, failure handling, traceability, instrument tests	Generalization to every site or population

Table 2: Separate evidence requirements for an instrument-facing inference application.

For radiograph localization, CheXlocalize supplies radiologist-annotated regions for evaluation [5]. A rigorous study should freeze the test set, prevent patient overlap across splits, keep tuning separate from testing, and report failures as well as average performance. Dice overlap can be appropriate for a defined segmentation task, but empty masks, negative cases, boundary errors and image orientation need explicit handling. Confidence intervals and external-site evaluation are valuable complements to a single average score. Dataset access and permitted use must be respected independently of software licensing.

5 Scope and availability

Spum is distributed as a local inference product, with a standalone Mac application and command-line installation options linked from its product site. Model weights have separate compatibility and licensing requirements. This paper describes public product behavior and selected documented engineering results as of its publication date. It does not disclose internal source code, execution algorithms, model adaptations, private datasets or unpublished experimental results.

The product and demonstrations described here are research and engineering tools. This paper does not establish authorization for clinical diagnosis, autonomous intervention or treatment. Any clinical deployment requires its own intended-use definition, evidence and applicable review.

References

[1] A. Dubey et al. *The Llama 3 Herd of Models*. 2024. arXiv:2407.21783.

[2] A. Sellergren et al. *MedGemma Technical Report*. 2025; revised 2026. arXiv:2507.05201.

[3] O. Ronneberger, P. Fischer and T. Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. arXiv:1505.04597.

[4] M. Harfouche et al. *Imaging across multiple spatial scales with the multi-camera array microscope*. 2023. arXiv:2212.00027.

[5] A. Saporta et al. *Benchmarking saliency methods for chest X-ray interpretation*. Nature Machine Intelligence, 2022. doi:10.1038/s42256-022-00536-x. Dataset: Stanford AIMI, CheXlocalize.