Open Source LLM Speculation (Phi-3.5-Vision) BBS Project

Kazi Islam

Version 1.0

Prepared for Prof David Smith

Date [10-17-2025]

Page of Content

1. Abstract

This document presents the exploratory phase of developing an open-source image-and-text recognition framework for the Balanced Blended Space (BBS) puppet show, focusing on evaluating Phi-3.5-Vision-Instruct as a candidate vision-language model. Phi-3.5-Vision-Instruct is a lightweight multimodal model (~4.2B parameters) that accepts interleaved image-and-text inputs and produces text outputs, with a long 128K-token context and an MIT license that enables local deployment and fine-tuning. The model emphasizes visual reasoning, document understanding/OCR, and multi-image analysis while remaining resource-efficient, making it attractive for small-to-medium creative projects on modest hardware. Our study investigates whether Phi-3.5-Vision-Instruct can learn from paired character/prop images with concise captions to recognize and label new photos in the BBS dataset. We examine architecture, licensing, fine-tuning feasibility, and instruction-following behavior, and we assess suitability by probing responsiveness to visual prompts, text–image alignment accuracy, and generalization to unseen stage scenes, with targeted tests for short vs. long generations within the 128K window. Findings will inform a practical foundation for integrating open-weight vision-language models into artistic and educational media contexts, guiding a lightweight, reproducible recognition workflow tailored to BBS's storytelling environment.

2. Objective

2.1 Primary objective

Determine whether **Phi-3.5-Vision-Instruct** is a suitable, low-cost, open-weight model for BBS image-and-text recognition, and select the deployment configuration (precision, context usage, batching, image resolution) that best balances accuracy, latency, and hardware cost.

2.1.2. Specific objectives

Model Selection
Measure accuracy and efficiency for each setup on BBS tasks—captioning, VQA (answering questions about an image), puppet/prop classification, and multi-image prompts. Track VRAM used, latency (time per response), and throughput (items per second), then choose the smallest variant that still meets our quality target.

Cost & Resource Envelope
Measure what it takes to run the model on regular machines (common GPUs and standard CPUs) and what it costs. For each test, record VRAM/RAM use, the GPU/CPU used, total time from question to answer (wall-clock time), average time per request (latency), and how many items we process per second (throughput). Check how much extra time and memory come from long context windows and from using several images at once.

Openness & Compliance
Confirm permissive licensing and open weights (MIT), and ensure availability of example fine-tuning/configuration artifacts so the pipeline is legally and operationally reusable in educational/creative contexts; record versions, seeds, and reproducibility notes.

3. Background

Phi-3.5-Vision-Instruct is an open-weight vision-language model that looks at images and text together and replies in plain text. It's compact (≈4.2B parameters), supports a long 128K-token context window, and is built to handle everyday multimodal tasks like visual question answering, reading text in images (OCR), chart/table understanding, and general visual reasoning.

The model ships under the MIT license with downloadable weights, so you can run it locally, adapt it, and fine-tune for your use case. Microsoft provides clear usage guidance (chat-style prompts, multi-image formatting) and a cookbook with example fine-tuning recipes, which lowers the setup burden for student and creative projects.

For the BBS puppet show, Phi-3.5-Vision-Instruct can help label rehearsal or show photos by recognizing puppets and props, answering short questions about a scene (VQA), and reading signs or labels (OCR). It can compare multiple images in one prompt and uses multi-crop processing options to capture small details when needed, useful for tiny text or fine costume features while staying light enough for modest hardware.

4. Evaluation of Model Prototypes

Phi-3.5-Vision-Instruct is distributed primarily as a single instruction-tuned checkpoint. For BBS, we evaluate three practical profiles of the same model that serve different needs.

4.1. phi-3.5-vision-instruct — "instruction-tuned default"

What it is: The standard PyTorch/Transformers checkpoint with 128K context and MIT license; accepts interleaved images+text and outputs text.
Best for: Out-of-the-box use on captioning, VQA/OCR, puppet/prop classification, and multi-image prompts.
Pros: Strong instruction following; open weights; widely supported tooling.
Trade-offs: Requires a decent GPU for fastest latency; fewer knobs than a base (non-instruct) model for heavy behavior reshaping.
Typical outputs: Clean, direct answers suitable for pipelines.
Prompting tip: Specify format/length (Example: "JSON with {character, prop, scene}; ≤2 sentences").
Choose this if: You want strong results now with minimal setup.

4.2. phi-3.5-vision-instruct (quantized 8-/4-bit) — “memory-saver”

What it is: The same checkpoint runs with integer quantization to cut VRAM and cost.
Best for: Edge or budget GPUs/CPUs; batching for higher throughput under tight memory.
Pros: Much lower memory use; often similar accuracy for tagging/captioning.
Trade-offs: Small quality/latency variance vs. full-precision; needs careful benchmarking.
Typical outputs: Same as default, possibly slightly shorter/safer due to decoding tweaks.
Prompting tip: Keep decoding conservative (lower temperature/top-p) to stabilize outputs when quantized.
Choose this if: You need to fit within 8–12 GB VRAM or share a single GPU among jobs.

4.3. phi-3.5-vision-instruct-onnx — “deployment-optimized”

What it is: Microsoft’s ONNX Runtime build of the same model for faster, portable inference on CPU/GPU; pairs well with OpenVINO on Intel hardware.
Best for: CPU-first deployments, containerized services, or mixed fleets (some GPU, many CPU).
Pros: Good throughput on commodity CPUs.
Trade-offs: Fewer bleeding-edge features than raw PyTorch; conversion/ops compatibility steps.
Typical outputs: Same as default; performance-tuned serving.
Prompting tip: Standard chat template with explicit schema; pre-tokenize prompts for batch.
Choose this if: You want reliable, low-cost serving without dedicated GPUs.

4.4. Shared traits (all three)

Phi-3.5-Vision-Instruct is released under the MIT license with openly available weights. It supports multimodal interaction, accepting interleaved images and text and producing text responses. The context window is up to approximately 128,000 tokens. The model size is about 4.15 billion parameters.

4.4.1. Quick pick guide

- Need plug-and-play quality now? → **Instruction-tuned default**.
- Tight VRAM/budget or CPU-only? → **Quantized memory-saver**.
- Production serving on mixed hardware? → **ONNX deployment-optimized**.

Recommendation for BBS: Start with Phi-3.5-Vision-Instruct in 8- or 4-bit for tagging, OCR, and short captions.

5. Version History

Version	Created on	Created by
1.0	10/17/2025	Kazi Islam