ECE 281 Final Presentation

# VisionAgent: Fine-Grained Image Editing with LLM Reasoning and Classical Computer Vision

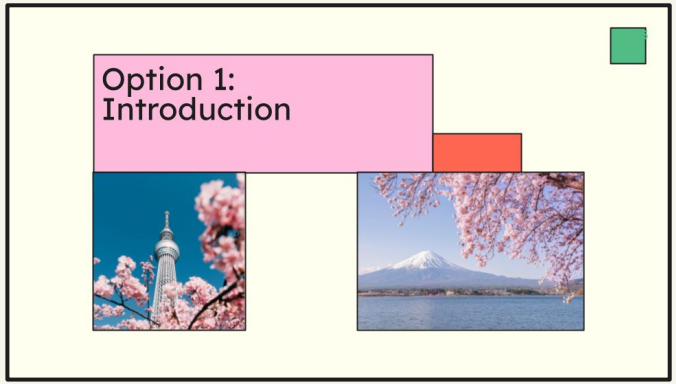
Sawyer Rice, Nik Belle, Dakota Barnes

What happens when you want **precise editing** on your image with **natural language instructions**?

# Current SOTA

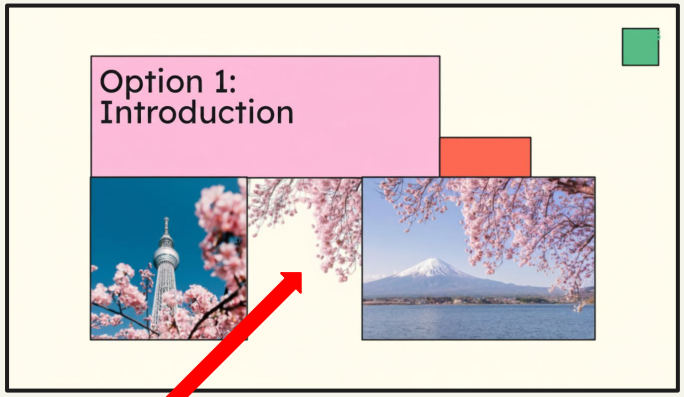
➔ Current SOTA image models change your image in unexpected ways.

*"Move the image of the mountain horizontally so that its left edge is flush with the image of the tower."*




960×540 Original image

*Gemini-2.5-Flash-Image*




What is this???

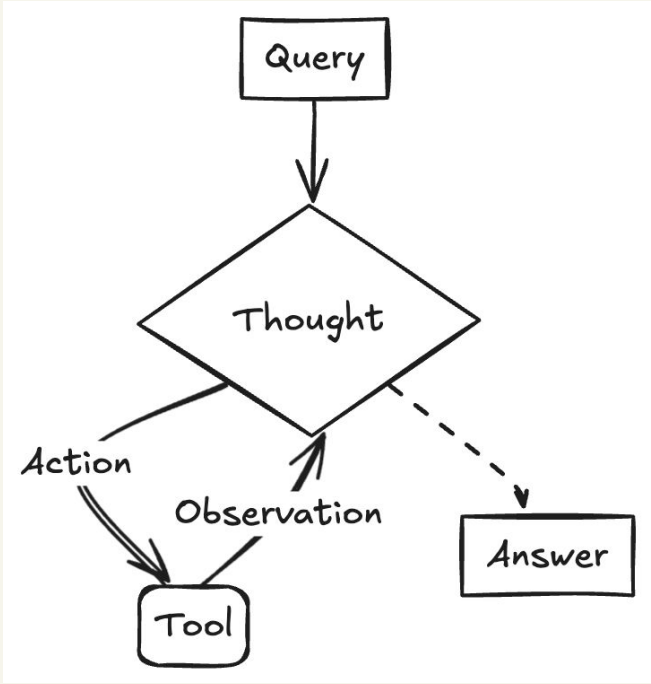
Resized to 1344×768????

# Our Idea

➔ Use an llm based agent with image processing tools to perform precise edits

# What is Opencode?

- Open source coding agent
  - Can read and edit files
  - Can navigate your terminal with bash
  - Similar to Claude Code or Codex
- Able to configure tools, prompts, skills, etc.




ReAct LLM Agent Framework

# Our Agent

Agent Prompt

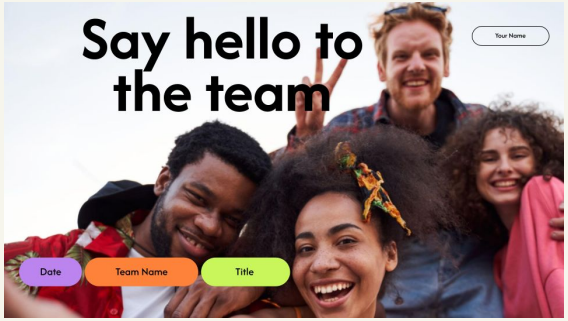
- ‘You have specialized image editing tools available. Do NOT write Python code.‘
- ‘STEP 3 — CLEAN + TRANSFORM: Only clean and transform the objects being edited.’
- ‘When editing MULTIPLE objects (including swaps), follow this exact order:’

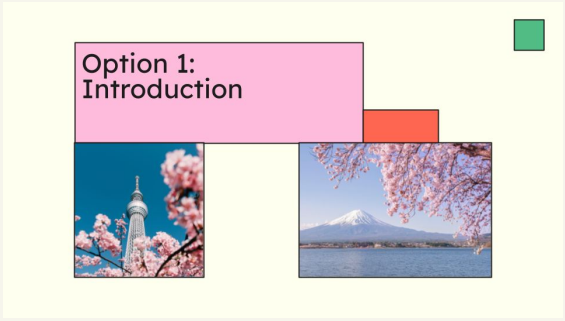
Tools

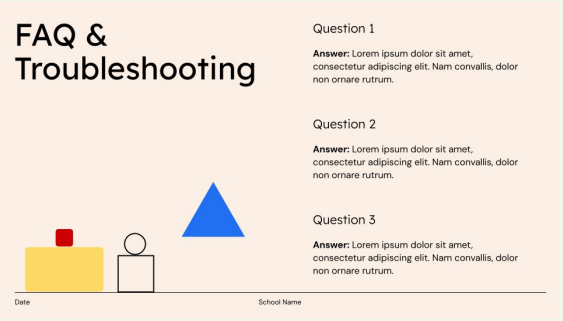
- detect_text (EasyOCR)
- detect_object (Grounding DINO)
- segment_object (SAM)
- create_text_mask
- fill_noise
- diffuse_inpaint (Navier-Stokes)
- translate_object, rotate_object, recolor_pixels, resize_region_simple
- Inspect_region
- save_image

# Custom Benchmark

- ➔ Created a benchmark and a test harness to test our agent, compared to SOTA.
- ➔ 3 Difficulty x 3 Categories x 3 Images (27 Tasks)
- ➔ Created 940x540 Google Slide Images



# Task Taxonomy

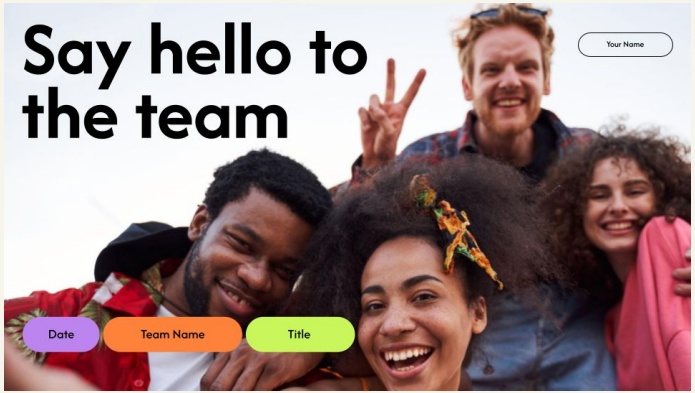
- **Easy**: Identifying and modify object
- **Medium**: Contextual reasoning
- **Hard**: Multi-object/ Multi-step

| | Easy | Medium | Hard |
|---|---|---|---|
| **Recolor** | Change top-left text to blue | Change second word of top-left sentence to blue | Swap colors of first and second words |
| **Resize** | Reduce top-left text by 50% | Extend shorter tab to match longer tab width | Double the size of text inside two label boxes |
| **Move** | Center "Say hello to the team" on the image | Move "Date" tab to left of "Say", align vertically | Move "Team Name" below purple tab, then "Title" to its right |
| **Recolor** | Make the blue triangle red | Make the tallest shape red | Recolor two stacked boxes: smaller blue, larger red |
| **Resize** | Double the red square, anchored left | Scale red square to yellow triangle width | Scale red square and green circle to their base widths |
| **Move** | Flip the blue triangle 180° | Move triangle tip below the second "t" in "troubleshooting" | Swap answers for questions 1 and 2 |
| **Recolor** | Make bottom-left blue box red | Make the box below the mountain photo blue | Recolor two panels: tower panel red, mountain panel blue |
| **Resize** | Extend black box to full page width | Enlarge mountain photo to reach slide bottom | Enlarge both photos to reach slide bottom, anchored top-left |
| **Move** | Move text to bottom-left of pink box | Align mountain photo's left edge with tower photo | Swap left and right photos, preserving top-left corners |

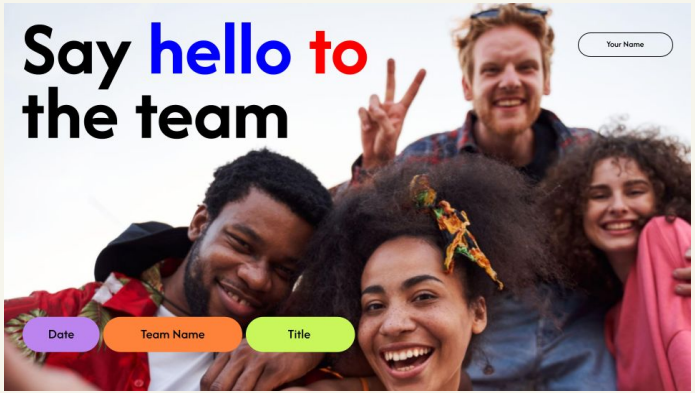
# Task Examples

Prompt: Switch the color of the third word in the sentence in the top left to red (255,0,0) and the second word to blue (0,0,255).

Original Image:




Ground Truth:



# Main Benchmark Metrics

- **Edit-Region LPIPS**: How similar to ground truth within the edit region?
- **Preserved-Region Percent Changed**: How much of the background was altered?
- **Composite Score**: 50-50 weighting between the two metrics above.

# Other Benchmark Metrics

- Full-image LPIPS
- Preserved-region SSIM
- Edit-region SSIM
- Full-image SSIM
- Preserved-region SSIM
- Edit-region MAE
- Preserved-region MAE
- Full-image IoU
- Full-image PSNR

# Agent Comparisons

- Unedited task image
  - **Baseline**
- Autoregressive multi modal
  - **Gpt-Image-1** (OpenAI)
  - **Gemini-2.5-Image-Flash** (Google)
- Diffusion Based
  - **Flux-Kontext-Pro** (Black Forest Labs)
- Agentic
  - **OpenCode Claude Haiku 4.5** (Open Source)
  - **Vision Agent Claude Haiku 4.5** (Ours)

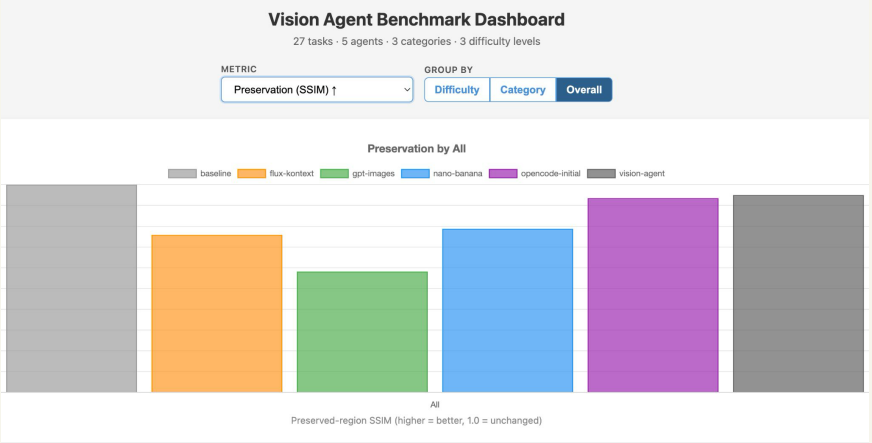
# Benchmark Harness

➔ Run any agent/model on a task and get metrics

| Argument | Description | Default |
|---|---|---|
| `--agent` | Single agent name (folder under `agents/`) | `opencode-initial` |
| `--agent-list` | Path to a text file listing agent names (one per line) | — |
| `--task` | Single task name (folder under `tasks/`). Overrides `-c` / `-d` / `-i`. | `ex_task` |
| `--task-list` | Path to a text file listing task names (one per line) | — |
| `-c`, `--category` | Filter by category: `recolor`, `resize`, `move` | — |
| `-d`, `--difficulty` | Filter by difficulty: `easy`, `medium`, `hard` | — |
| `-i`, `--image` | Filter by image: `img1`, `img2`, `img3` | — |
| `--model` | Model identifier passed to the agent | `anthropic/claude-haiku-4-5` |
| `-n`, `--num-runs` | Number of times to run each agent/task combo | `1` |
| `-t`, `--threads` | Max concurrent runs (parallel execution) | `1` (sequential) |

`--agent` and `--agent-list` are mutually exclusive. Same for `--task` and `--task-list`.

***python3 harness.py <args>***



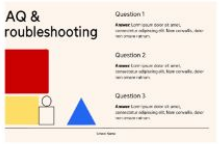
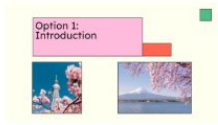
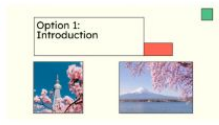
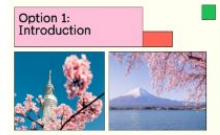
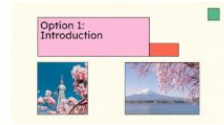
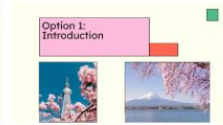
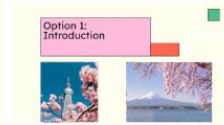
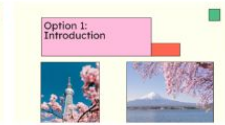
# Results: Overall

- ➔ VisionAgent achieves higher accuracy (LPIPS) compared to alternatives
- ➔ VisionAgent also preserves background pixels better than others (Preserved percent changed)

TABLE II
OVERALL AGENT PERFORMANCE AVERAGED ACROSS ALL TASKS.

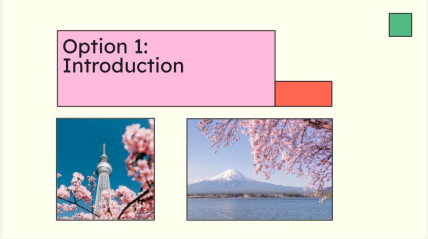
| Agent | LPIPS ↓ | Pres % Δ ↓ | Composite ↑ | Time (s) |
|---|---|---|---|---|
| Baseline (no-edit) | 0.381 | 0.000 | 0.000 | .1 |
| GPT-Image-1 | 0.498 | 50.315 | 0.500 | 30.4 |
| Flux-Kontext-Pro | 0.435 | 22.725 | 0.669 | 12.7 |
| Gemini-2.5-Flash | 0.348 | 18.613 | 0.733 | **9.5** |
| OpenCode Agent | 0.190 | 4.331 | 0.820 | 14.1 |
| VisionAgent (ours) | **0.186** | **0.457** | **0.905** | 36.5 |

# Output Images (Task on X, Agents on Y)

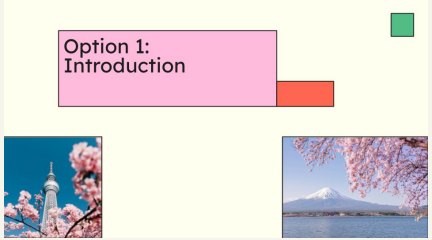
| Prompt | Original / Baseline | flux-kontext | gpt-images | nano-banana | opencode-initial | vision-agent | Ground Truth |
|---|---|---|---|---|---|---|---|
| **Move Easy Img1**<br>Move the "Say hello to the team" text to the center of the image to be centered horizontally and vertically. |  |  |  |  |  |  |  |
| **Resize Medium Img2**<br>Make the red square as wide as the yellow rectangle. Align the bottom edge of the new red square with the top edge of the yellow rectangle. Move the new enlarged red square so that the bottom edge of the red square is touching the top edge of the yellow rectangle. |  |  |  |  |  |  |  |
| **Resize Hard Img3**<br>Increase the size of both the image on the left, and the image on the right, until the images touch the bottom edge of the slide. The bottom of both images should be aligned with the bottom edge of the slide. Make sure the top left corners of both images stay anchored. |  |  |  |  |  |  |  |
| | 940×540 | 1392×752 | 1536×1024 | 1344×768 | 940×540 | 940×540 | 940×540 |

# Results: Per Task Categories

Original:




Ground Truth:




OpenCode:


VisionAgent:

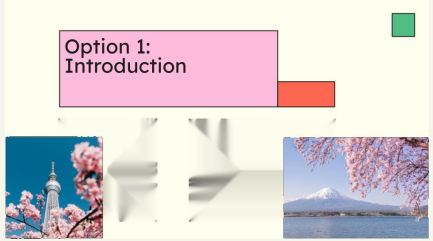



TABLE III
COMPOSITE SCORE BY TASK TYPE.

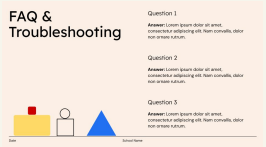
| Agent | Move | Recolor | Resize | Overall |
|---|---|---|---|---|
| Baseline (no-edit) | 0.000 | 0.000 | 0.000 | 0.000 |
| GPT-Image-1 | 0.543 | 0.506 | 0.449 | 0.500 |
| Flux-Kontext-Pro | 0.671 | 0.709 | 0.626 | 0.669 |
| Gemini-2.5-Flash | 0.675 | 0.821 | 0.703 | 0.733 |
| OpenCode Agent | 0.735 | 0.836 | **0.887** | 0.820 |
| VisionAgent (ours) | **0.882** | **0.980** | 0.852 | **0.905** |

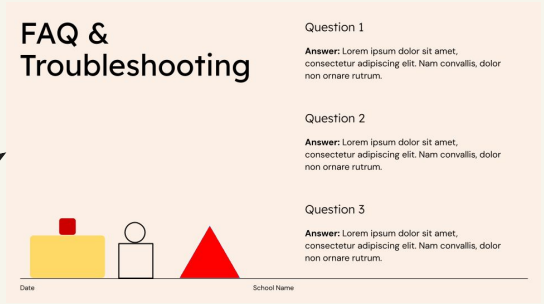
# Interesting Examples

Prompt: “Make the triangle in the middle of the image red (255,0,0).”

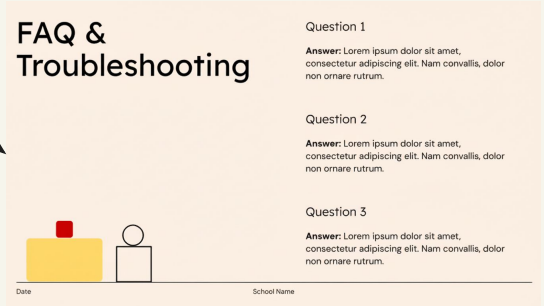
VisionAgent

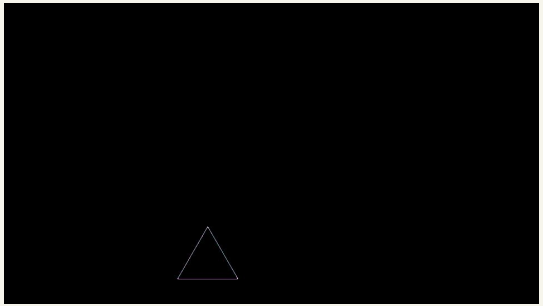
Original



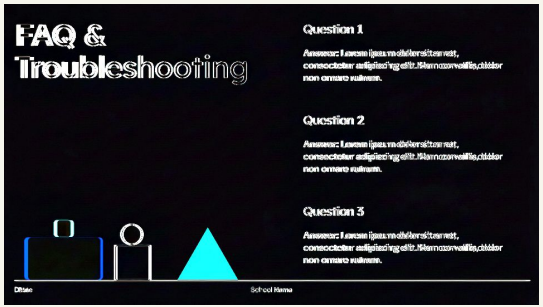




Nano Banana

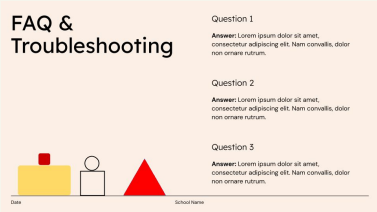


Ground Truth Difference




Ground Truth

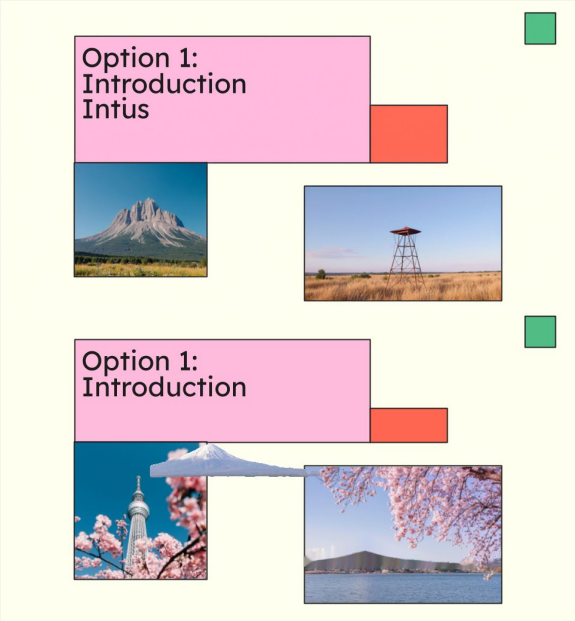


# Interesting Examples

Prompt: “Find the two rectangular images within the slide. There is one image that contains a mountain, and another that contains a tower. Move the entire image that contains the mountain so that the left edge of that entire image is aligned with the right edge of the entire image that contains the tower. Also make sure that the top of the image that contains the mountain is aligned vertically with the bottom of the pink box.”

Flux-Kontext-Pro

VisionAgent




Detection Output:

# Strengths

**VisionAgent**:

- ➔ Background preservation
- ➔ Prompts help provide a workflow
- ➔ Tools give reliable interface to image
- ➔ ReAct loop allows for error diagnosis
- ➔ High explainability

**Benchmark**:

- ➔ Simple task creation and harness
- ➔ Realistic use case
- ➔ Displays clear SOTA limitation
- ➔ Explainable metrics with verbose option
- ➔ Multiple axes to expand on

# Limitations

**VisionAgent**:

- ➔ Runtime (2x-4x slower)
- ➔ Narrow scope of tasks
- ➔ Complex Backgrounds, Navier-Stokes does not suffice
- ➔ Object detection

**Benchmark**:

- ➔ Limited 27 tasks
- ➔ More complex metric weighting would capture more fine grain differences in one score
- ➔ Solid background tasks
- ➔ Simple composite score has pros and cons

# Future Work

- ➔ Expand benchmark:
  - ◆ More complex backgrounds
  - ◆ New task categories
  - ◆ Composite score with more terms
- ➔ Improve tools:
  - ◆ Stronger inpainting methods & word identification
  - ◆ Tools for verifying changes, or adding text
  - ◆ New Google Cli
- ➔ Evaluate with stronger LLM
  - ◆ Many simple tools vs. few powerful tools
  - ◆ Improvement with more freedom?