The Perceptual Observatory
Characterizing Robustness and Grounding in MLLMs

Supplementary Material

1. Dataset Details

1.1. CELEB

We sample 1,000 celebrity face images for facial feature attribution. Bounding boxes for left/right eyes, nose, and mouth are computed using `MediaPipe`. To validate reliability, the first and second authors manually annotated 10% of images, achieving 98% IoU with `MediaPipe` outputs. Hence, we treat `MediaPipe`-derived boxes as gold annotations.

1.2. WORD

We collect ∼267K unique words across 21 semantic categories (*Computer Science, Cities, People, Food, Politics, Abuse*, etc.). Word length $l \in [2, 10]$ with $\mathbb{E}[l] \approx 4.8$. Each word is rendered under:

$$\mathcal{F} \times \mathcal{C} \times \mathcal{P} \times \mathcal{R},$$

with $\mathcal{F} = \{CourierNew, ..., TimesNewRoman\}$ (fonts), $\mathcal{C} = \{\text{upper}, \text{lower}, \text{camel}\}$ (casings), $\mathcal{P} = \{\text{center}, \text{top}, \text{bottom}\}$ (positions), $\mathcal{R} = \{-45°, 0°, 45°\}$ (rotations). Uniform sampling across these factors produces >1M rendered images overall. Because WORD is procedurally generated, bounding boxes are exactly known.

1.3. Perturbations

We apply two perturbation families:

Linear augmentations ($\mathcal{P}_1$). Implemented with Albumentations [6]. Each image is augmented by sampling from the set

$$\mathcal{M} = \left\{ \begin{array}{l} \texttt{GaussianBlur}(11, 11),\ \texttt{MedianFilter}(21), \\ \texttt{ZoomBlur}([1.05, 1.07]),\ \texttt{ChromaticAberration}(\pm 0.2), \\ \texttt{ISONoise}([0.01, 0.05], [0.1, 0.5]),\ \texttt{RGBShift}(\pm 20), \\ \texttt{Salt\&PepperNoise}([10^{-4}, 10^{-3}]),\ \texttt{GammaLimit}([80, 140]), \\ \texttt{JPEGCompression}([20, 50]),\ \texttt{MultiplicativeNoise}([0.9, 1.1]), \\ \texttt{Sharpen}(\alpha \in [0.3, 0.5]),\ \texttt{GlassBlur}(\sigma = 0.3, \Delta = 2), \\ \texttt{Posterize}(4\ \text{bits}),\ \texttt{MotionBlur}(7, 7), \\ \texttt{GaussianNoise}(\mu = 0,\ \sigma \in [0.05, 0.1]) \end{array} \right\}$$

Thus, $\mathcal{P}_1(x) \sim \mathcal{U}(\mathcal{M})$.

Illusion perturbations ($\mathcal{P}_2$). Following IllusionBench [18], each source image x_i is embedded into a stylized scene using ControlNet [54] with Stable Diffusion [37]. Prompts are composed from:

$$[\texttt{SubjectScene}] \times [\texttt{Style}] \times [\texttt{Light/ColorHighlight}],$$

where representative values are listed below:

Subject Scene	Style	Light/Color Highlight
Museum	Cinematic	Dust Motes
Rainy Alleyway	Gothic	Neon Glow
Forest	Fantasy Art	Golden Hour
Desert Dune	Vintage Photo	Pastel Hues
Medieval Village	Minimalist	Stark Shadows
Ocean	Surrealism	Electric Blue
Sunset Beach	Bioluminescent	Crystal Refraction
Cozy Cottage	Origami	Venetian Blinds
Mountain Range	Dystopian	Hearth Fire
Overgrown Ruins	Abstract	Volumetric Rays
Starry Night	Painting	Smudged Grays
Cloudy	Pixel Art	Pink Cyan

We apply a negative prompt (`glitch, low quality`) to suppress artifacts. Control strengths are dataset-dependent: WORD: $cn\_scale = 1.2$, $guide\_scale = 10.5$, CELEB: $cn\_scale = 3.0$, $guide\_scale = 7.5$.

Each final entry is stored as (x_{ij}, s_j), where s_j encodes the sampled scene, style, and lighting.

Final Dataset Size

For each dataset (CELEB, WORD), we sample 1,000 original images and generate 15 variants with $\mathcal{P}_1$, 15 with $\mathcal{P}_2$, plus the original. This yields:

$$1000 \times (1 + 15 + 15) = 31{,}000 \text{ images per dataset.}$$

In total, the benchmark contains **62,000** images.

2. Prompt Templates

Prompt A: Image Matching Query

```
**INSTRUCTIONS**
You are given 4 images each of size 1024x1024.

**TASK**
Compare the support image with 4 candidate images, and
    ↪ select a single candidate image that best matches
    ↪ the support image.

Return the result as valid JSON with detailed reasoning.

**JSON output format**
‘‘‘json
{
  "reasoning": "Provide a structured explanation based on
    ↪ visual cues. Cite concrete visual evidence and
    ↪ justify your identification.",
  "final_answer": "A" or "B" or "C" or "D"
}
‘‘‘
```

Prompt B: Image Matching Support (Celeb)

```
**CONTEXT**
You are given an image of size 1024x1024 of a famous person
     ↪ . This information serves as a factual reference.
Additionally, you will further be given 4 images, only one
     ↪ of them is generated from this image, where a face
     ↪ might be clearly visible, perturbed, stylized, or
     ↪ blended with the environment (background or object)
     ↪  as a visual illusion.
```

Prompt C: Image Matching Support (Word)

```
**CONTEXT**
You are given an image of size 1024x1024 of a case
     ↪ sensitive sequence of characters, "[WORD_LABEL]".
     ↪ This information serves as a factual reference.
Additionally, you will further be given 4 images, only one
     ↪ of them is generated from this image, where a
     ↪ sequence of characters are clearly written,
     ↪ perturbed, stylized, or blended with the
     ↪ environment (background or object) as a visual
     ↪ illusion.
```

Prompt D: GPG Query

```
**INSTRUCTIONS**
You are given an image of size 1024x1024 that is composed
     ↪ of 4 sub-images arranged in a 2x2 grid as a collage
     ↪ .
Only one of these sub-images is the *source image* from
     ↪ which the support image was generated.

**TASK**
Identify and locate which grid cell contains the source
     ↪ image.
Coordinates mapping:
[0,0] = top-left
[0,1] = top-right
[1,0] = bottom-left
[1,1] = bottom-right

Return the result as valid JSON with detailed reasoning.

**JSON output format**
‘‘‘json
{
  "reasoning": "Provide a structured explanation based on
     ↪ visual cues. Cite concrete visual evidence and
     ↪ justify your identification.",
  "final_answer": "[0,0]" or "[0,1]" or "[1,0]" or "[1,1]"
}
‘‘‘
```

Prompt E: GPG Support (Celeb)

```
**CONTEXT**
You are given an image of size 1024x1024. This image is a
     ↪ visually altered version of some original image
     ↪ such that the source might contain a face of famous
     ↪  person that have been clearly visible, perturbed,
     ↪ stylized, or blended with the environment (
     ↪ background or object) as a visual illusion.
The information serves as the support context.
```

Prompt F: GPG Support (Word)

```
**CONTEXT**
You are given an image of size 1024x1024. This image is a
     ↪ visually altered version of some original image
     ↪ such that the source might contain a sequence of
     ↪ characters that are clearly written, perturbed,
     ↪ stylized, or blended with the environment (
     ↪ background or object) as a visual illusion.
The information serves as the support context.
```

Prompt G: Attribution Support (Celeb)

```
**CONTEXT**
You are given an image of size 1024x1024 of a famous person
     ↪ , "[CELEB_LABEL]". The following text provides
     ↪ context for the key features present in this image,
     ↪  listing each attribute with its precise bounding
     ↪ box coordinates. This information serves as a
     ↪ factual reference.

Attributes:
‘‘‘json
[BBOX]
‘‘‘
```

Prompt H: Attribution Support (Word)

```
**CONTEXT**
You are given an image of size 1024x1024 of a case
     ↪ sensitive sequence of characters, "[WORD_LABEL]".
     ↪ The following text provides context for the
     ↪ sequence visible in this image, defining the
     ↪ characters and their precise bounding box. This
     ↪ information serves as a factual reference.

Attributes:
‘‘‘json
[BBOX]
‘‘‘
```

Prompt I: Attribution Guided Query (Word)

~~~~
**INSTRUCTIONS**
You are given an image of size 1024x1024 that was generated
     ↪  from the above support image. The image contains a
     ↪  sequence of characters clearly written, distorted,
     ↪  stylized, or blended with the environment (
     ↪ background or object) as a visual illusion.

**TASK**
Think and analyze the image carefully. Using the support
     ↪ context as a reference, detect the sequence of
     ↪ characters and provide a single bounding box that
     ↪ encloses all of its characters with detailed
     ↪ reasoning.

Return the result as a valid JSON list. If no sequence is
     ↪ confidently located, return empty list [].

**JSON output format**
```json
{
    "sequence": "The sequence of characters you read and
     ↪ detected",
    "reasoning": "Explain the visual evidence for this
     ↪ detection.",
    "x1": int,
    "y1": int,
    "x2": int,
    "y2": int
}
```
~~~~

Prompt J: Attribution Semi Guided Query (Word)

~~~~
**INSTRUCTIONS**
You are given an image of size 1024x1024 that was generated
     ↪  from the above support image. The image contains a
     ↪  sequence of characters clearly written, distorted,
     ↪  stylized, or blended with the environment (
     ↪ background or object) as a visual illusion.
The support context only consists of top-left corner point
     ↪ of the bounding box.

**TASK**
Think and analyze the image carefully. Using the support
     ↪ context as a reference, detect the sequence of
     ↪ characters and provide a single bounding box that
     ↪ encloses all of its characters with detailed
     ↪ reasoning.

Return the result as a valid JSON list. If no sequence is
     ↪ confidently located, return empty list [].

**JSON output format**
```json
{
    "sequence": "The sequence of characters you read and
     ↪ detected",
    "reasoning": "Explain the visual evidence for this
     ↪ detection.",
    "x1": int,
    "y1": int,
    "x2": int,
    "y2": int
}
```
~~~~

Prompt K: Attribution Guided Query (Celeb)

~~~
**INSTRUCTIONS**
You are given an image of size 1024x1024 that was generated from the above support image. A face is present in this image. The face might be clearly visible, perturbed, stylized, or blended with the environment (background or object) as a visual illusion.

**TASK**
Think and analyze the image carefully. Using the support context as a reference, detect the bounding boxes and provide detailed reasoning for each discernable facial attributes:
1. Left Eye
2. Right Eye
3. Nose
4. Mouth

Return the result as a valid JSON list. Your reasoning must explain how you identified the attribute, and only include the attribute that you can detect. If no attributes are confidently located, return empty list [].

**JSON output format**
```json
{
    "left_eye": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "right_eye": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "nose": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "mouth": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    }
}
```
~~~

Prompt L: Attribution Semi Guided Query (Celeb)

```
**INSTRUCTIONS**
You are given an image of size 1024x1024 that was generated from the above support image. A face is present in this image. The
      ↪ face might be clearly visible, perturbed, stylized, or blended with the environment (background or object) as a visual
      ↪ illusion.
The support context only consists of one key feature’s bounding box.

**TASK**
Think and analyze the image carefully. Using the support context as a reference, detect the bounding boxes and provide detailed
      ↪ reasoning for each discernable facial attributes:
1. Left Eye
2. Right Eye
3. Nose
4. Mouth

Return the result as a valid JSON list. Your reasoning must explain how you identified the attribute, and only include the
      ↪ attribute that you can detect. If no attributes are confidently located, return empty list [].

**JSON output format**
‘‘‘json
{
    "left_eye": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "right_eye": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "nose": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    },
    "mouth": {
        "reasoning": "Explain the visual evidence for this detection.",
        "x1": int,
        "y1": int,
        "x2": int,
        "y2": int
    }
}
‘‘‘
```