Composite-Attribute Person Re-Identification via Pose-Guided Disentanglement

Supplementary Material

This supplementary material contains additional details on the dataset construction, annotation pipeline, attribute extraction, and query generation.

1. Overview

Datasets. Our CA-ReID dataset is sourced from two complementary datasets: *Celeb-ReID-Light* (Celeb-ReID-L), a curated subset of Celeb-ReID and *COCAS+ Real2* (Real2), the dataset used by Instruct-ReID for LI-ReID. Celeb-ReID-Light is selected because each identity exhibits *rich intra-identity variation* in clothing and appearance across public sightings of celebrities. Typical clothes-changing ReID datasets focus on view changes (e.g., different camera angles) with limited wardrobe diversity. However, Celeb-ReID-L focuses on celebrities whose outfits, hairstyles, and accessories vary substantially over time. While Celeb-ReID-L lacks camera variation, its strong diversity in clothing and appearance makes it well-suited for testing composed attribute retrieval: from short, generic prompts (e.g., "blue jeans") to fine-grained descriptions (e.g., rare accessories or combined unique clothes).

To cover the complementary axis, we also include Real2, a clothing-template-based ReID benchmark used in LI-ReID. Real2 provides varied camera viewpoints and non-celebrity images, but its text largely comes from template-based GPT-generated *full description captions*, which restrict the long-tail of clothing terms (i.e., fewer open-vocabulary attributes). Pairing both datasets allows us to evaluate both *generic* and *specialized* prompts under different sources of variability.

Annotation modality. For Celeb-ReID-L (images without attributes), we obtain open-vocabulary attributes directly from images using two large vision-language models (LVLMs), with majority voting. For Real2, we parse the provided full description generated captions into the our structured attributes using a text-only large language models (LLMs). Both produce the same JSON covering select attribute groups, which allows consistent query generation across datasets and difficulty levels (Easy/Medium/Hard).

Intended use & licensing. All data are used for research purposes only. Due to image licensing for Real2, we *do not* show sample images from Real2 in the main paper or supplemenrtary material; we report results and show text-side artifacts only.

2. Attribute Categories

We initially annotate seven fine-grained attribute categories to capture the most informative aspects of a person's appearance and the surrounding visual context. These categories are designed to cover major human body regions, together with accessories, belongings, and contextual details that are often important for person understanding and retrieval.

- **Head**: hairstyle and color, headwear (e.g., hats, caps), and face wear (e.g., glasses, sunglasses)
- **Top**: upper body clothing and apparel
- **Bottom**: lower body clothing and apparel
- **Feet**: footwear
- **Accessories***: wearable items and adornments not already assigned to other body regions
- **Belongings***: carried objects and personal items (e.g., shopping bag, backpack, phone)
- **Contextual Details***: pose, actions, and other semantically informative scene cues (e.g., crossed arms, hands in pockets, holding phone)

*For CA-ReID, we combine *accessories*, *belongings*, and *contextual details* into a single **Other** category, except for attributes that naturally belong to a body region (e.g., headwear and face wear), which are grouped under **Head**..

3. Attribute Extraction

We extract open-vocabulary person attributes using two large, open-source VLMs; InternVL3-38B [1] and Qwen2.5-VL-32B-Instruct [2], with the *same* instruction prompt and output format across models for consistency. For Celeb-ReID-Light, since we do not have any captions or attribute labels, we run both VLMs on each image, and choose attributes based off majority voting. The prompt is shown in Fig. 2. For Real2 (captioned images from LI-ReID), we *do not* use a VLM; instead, we keep the original full captions and parse them into the same seven slots with a text-only LLM prompt (shown in Fig. 3). Models use default parameters, temperature is set to 0.7 for VLMs.

Because LLM/MLLM outputs can hallucinate or vary in phrasing, we apply a two-stage normalization. **(1)** We remove fillers and punctuation; normalize colors (e.g., "dark blue"→"navy" when unambiguous); fold synonyms ("tee"→"t-shirt", "slacks"→"trousers"); enforce the slot typing (arrays vs. strings) and empty values as `[]` or `""`. Head/face accessories (caps, hats, glasses, masks, headbands, headphones, earrings) are *only* kept in head and removed from `accessories`. **(2a) CEleb-ReID-L:** For each attribute, we compare InternVL3 vs. Qwen2.5-VL outputs using (i) word-level overlap after regex normalization and (ii) sentence-embedding cosine similarity using all-

[1] https://huggingface.co/OpenGVLab/InternVL3-38B
[2] https://huggingface.co/Qwen/Qwen2.5-VL-32B-Instruct

MiniLM-L6-v2 [3]. If the candidates are semantically close, we keep InternVL3's phrase (initial tests and audits found it more accurate and descriptive); otherwise we flag the sample for human review. **(2b) Real2:** For each attribute, we verify that each extracted phrase appears in the original caption after punctuation/stop-word removal. If extracted attribute does not exist in the original caption, we drop that attribute. In addition, we load the *Instruct-ReID* attribute bank and validate category placement (e.g., "hat" must not appear in **bottom** category; "coat" cannot be in **feet**.

Human audit and fallback policy. We audit roughly 10% of samples at random. Disagreements between VLMs or low-similarity merges are sent to the audit pool. In practice, auditors found that defaulting to InternVL3 on close matches was the correct choice in most cases, so we retain this rule for the remaining data. When an image is missing an entire attribute JSON (about 200 images in each dataset), we annotate it manually.

3.1. Query Generation

We generate text condition queries from per-image attributes. For each image as a query in text set, we find all images from same id and treat them as targetsm by using the target image attribtues to form the condition text. First, we normalize each candidate string by lowercasing, dropping all punctuation except hyphens, removing stopwords {*a, an, the, and, with, of, in, on, person, wearing*}, then whitespace-splitting, lexicographically sorting the tokens. All outputs are formatted as lowercase, comma-separated phrases ending with a period; we keep compact noun phrases (e.g., "white t-shirt", "blue jeans"), preserve hyphens, and omit brand names whenever possible.

Easy/Medium composition. We first collect candidate phrases per category bucket: `head`, `top`, `bottom`, `feet`, and an `other` bucket formed by concatenating `accessories`, `belongings`, `context`.

For easy queries, we want to provide a full description so we pick one attribute from each bucket, and randomly shuffle their order to form a comma seperated sentence, which starts with "wearing" or "with" keywords. Some examples are shown below:

- "with dark wavy hair with fringe, wearing white crop top, black boots, blue jeans, and hairband."
- "with dark ponytail hair, wearing red crop top, multicolored shorts, black heels, and hands on hips."
- "with blonde long hair, wearing blue jeans, black shoes, white shirt, and black handbag."

For medium queries, we *uniformly sample* two distinct buckets, so queries contain two attribtues from different buckets seperatede by keyword '*and*'. Some examples are shown below.

- "wearing white crop top, and blue jeans."
- "wearing white pants with black stripes, and dark medium-length hair."

Hard composition. For Hard we produce one short sentence *per attribute category*. We provide examples per category below with visual examples are shown in Fig. 1.

- **Head:** "with sunglasses", "with grey hat", "with blonde hair tied back"
- **Top:** "wearing white and grey striped crop top", "wearing floral off-shoulder dress", "wearing black blazer"
- **Bottom:** "wearing white and grey striped pants", "wearing ripped blue jeans", "wearing floral skirt"
- **Feet:** "wearing black ankle boots", "wearing high heeled sandals", "wearing white sneakers with orange accents"
- **Accessories:** "with watch", "with blue scarf"
- **Belongings:** "carrying coffee cup", "carrying black handbag", "carrying white tote bag"
- **Context:** "holding hands with someone", "hands in pockets", "waving hands"

Figure 1. Single-attribute query examples (hard setting) from our dataset, per attribute categories: top, bottom, feet, and other (accessories, belongings, context).

[3] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

VLM Prompt

```
You are an expert at identifying human visual attributes from images.
You are shown an image of a single person.
Follow this numbered checklist. Only describe what is clearly visible.
If a detail is not visible/unclear, leave it empty. Do not guess.

1) head_hairstyle_color
   - Hairstyle details (e.g., buzzcut, shoulder-length, ponytail, bald).
   - Hair color (e.g., black, brown, blonde, dyed pink).
   - Include head/face accessories and coverings here: caps/hats/beanies, headbands,
       glasses/sunglasses, face masks, earrings, headphones, etc.
   - If multiple items apply, provide a list of short attribute phrases (e.g.,["short brown hair",
       "black cap", "glasses"]).

2) top_clothing_and_color
   - Upper-body garments with visible colors/patterns (e.g., "white t-shirt","grey coat", "striped
       shirt"). List layered items separately in order from inner to outer.

3) bottom_clothing_and_color
   - Lower-body garments with visible colors/patterns (e.g., "blue jeans", "black trousers", "khaki
       shorts", "checked skirt").

4) footwear_and_color
   - Shoes/boots/sneakers with visible color/pattern (e.g., "black sneakers").
   - If barefoot/unknown, leave empty.

5) accessories
   - Worn items that are not clothing and not already included in (1): watch, bracelet, necklace,
       scarf, belt, gloves, etc.

6) belongings
   - Items carried or worn as gear: backpack, handbag, tote, shopping bag, umbrella, phone, camera,
       suitcase, pet leash, etc.

7) contextual_details
   - Visible cues/actions/details not captured above, and any contextual information about the
       individual: "hand in pocket", "arms crossed","hood up", "rolled sleeves", "waving hand", etc.

Rules:
- Use concise, literal phrases; no attributes not visible in the image.
- Prefer common color words; include patterns when clearly visible (striped, checked).
- Do not repeat head/face accessories in (5) or (6); they belong in (1).
- Return ONLY the JSON object below. No extra text, no comments.
- If there are no attributes for a given category, return an empty list for that category.

Output format (return only this JSON object):
{
  "head_hairstyle_color": [],
  "top_clothing_and_color": [],
  "bottom_clothing_and_color": [],
  "footwear_and_color": [],
  "accessories": [],
  "belongings": [],
  "contextual_details": []
}
```

Figure 2. VLM prompt for attribute extraction from images only.

LLM Prompt

```
You are an expert at extracting structured clothing and appearance attributes from descriptive
captions of people.
Follow this numbered checklist. Use only information explicitly stated in the caption.
If a detail is missing or unclear, leave it empty. Do not guess.

1) head_hairstyle_color
   - Hairstyle (buzzcut, shoulder-length, ponytail, bald).
   - Hair color (black, brown, blonde, dyed pink).
   - Include head/face accessories and coverings here: caps/hats/beanies, headbands,
     glasses/sunglasses, face masks, earrings, headphones, etc.

2) top_clothing_and_color
   - Upper-body garments with colors/patterns mentioned (e.g., "white t-shirt", "grey coat",
     "striped shirt"). List layers separately, inner → outer.

3) bottom_clothing_and_color
   - Lower-body garments with colors/patterns (e.g., "blue jeans", "black trousers",
     "checked skirt").

4) footwear_and_color
   - Shoes/boots/sneakers with color/pattern (e.g., "black sneakers"). Leave empty if not stated.

5) accessories
   - Worn items not in (1): watch, bracelet, necklace, scarf, belt, gloves.

6) belongings
   - Items carried or worn as gear: backpack, handbag, tote, shopping bag, umbrella,
     phone, camera, suitcase, pet leash, etc.

7) contextual_details
   - Short cues explicitly in the caption but not captured above.

Rules:
- Use concise, literal phrases only from the caption; no inference.
- Use same wording as captions. Do not reword.
- Do not duplicate head/face accessories in (5); they belong in (1).
- Return ONLY the JSON object below. No extra text, no comments.

Output format (return only this JSON object):
{
  "head_hairstyle_color": [],
  "top_clothing_and_color": [],
  "bottom_clothing_and_color": [],
  "footwear_and_color": [],
  "accessories": [],
  "belongings": [],
  "contextual_details": []
}
```

Figure 3. LLM prompt for attribute extraction from text captions only.