Visual Personalization Turing Test

Supplementary Material

Table 1. Ablation on post count ($\text{VPTT}_{\text{score}}$).

Posts	Base.	Pers. Only	BRAG	VPRAG
2	0.249	0.299	**0.513**	0.507
5	0.232	0.306	0.525	**0.535**
10	0.260	0.338	0.549	**0.578**
20	0.283	0.361	0.569	**0.598**
30	0.295	0.373	0.585	**0.607**

Table 2. Ablation on different components using ($\text{VPTT}_{\text{score}}$-c).

Method	Score	% over BRAG
BRAG (Caption-based)	*0.462*	—
VPRAG	**0.530**	**+14.8%**
Single Post Only	0.113	-75.5%
Random Retrieval	0.501	+8.5%
No Soft Assignment	0.507	+9.8%
No Entropy Selection	0.515	+11.4%
No Category Ranking	0.515	+11.5%

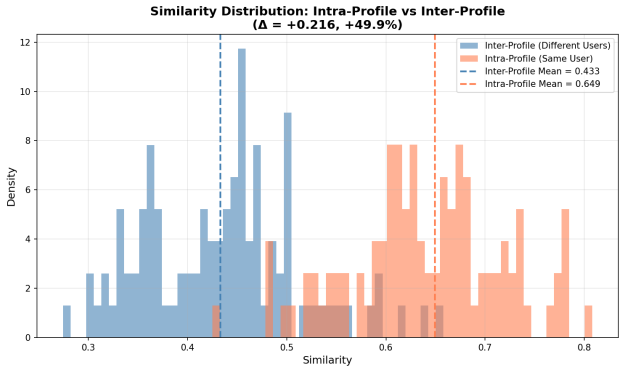

Figure 1. Diversity of outputs across different profiles.

S.1 Additional Details: Formalization of the VPTT Evaluation Protocol

This section provides additional mathematical clarification of the Visual Personalization Turing Test (VPTT) evaluation protocol described in the main paper. The formalization offers a rigorous scientific grounding for the task and mitigates subjective interpretation.

Setup. A persona is defined as $P = \{d, E, C\}$ (demographics, structured visual elements, and caption memory). Given a query p, the personalization system produces a rewritten prompt p' and a generated visual output

$$X = \mathcal{G}(p') \in \mathcal{X},$$

where $\mathcal{G}$ denotes the visual generative model (see Sec.3.2, main paper). For brevity, we write $X \sim G(\cdot \mid p, P)$ to denote the overall personalization pipeline that includes retrieval, prompt rewriting, and generation.

Judge function. As described in the main paper, VPTT evaluates whether X is *indistinguishable from content that the persona might plausibly create or share*. We formalize this via a judge function

$$J : \mathcal{X} \times \mathcal{P} \to [0, 1], \quad J(X, P) = \text{plausibility score}.$$

Human annotators and VLM-based judges provide plausibility judgements on a 0–5 Likert scale, which can be linearly normalized to $[0, 1]$. In large-scale evaluations, we substitute J with the $\text{VPTT}_{\text{score}}$ (Sec. 4, main paper), which serves as a scalable proxy.

Expected VPTT performance. Let μ be the distribution over persona–query pairs. The expected VPTT performance of a generator G is

$$\Pi(G) = \mathbb{E}_{(P,p)\sim\mu}\Big[\mathbb{E}_{X\sim G(\cdot|p,P)}\big[J(X, P)\big]\Big]. \tag{S1}$$

Finite-sample estimator. Using N personas and K queries per persona, we estimate Eq. (S1) as

$$\widehat{\Pi}(G) = \frac{1}{NK}\sum_{i=1}^{N}\sum_{j=1}^{K} J(X_{ij}, P_i), \quad X_{ij} \sim G(\cdot \mid p_{ij}, P_i). \tag{S2}$$

Judging modalities.

- **Human judges:** $J(X, P)$ is the mean normalized Likert score over annotators.
- **VLM judge:** $J(X, P)$ is the calibrated normalized Likert score of the plausibility estimate.
- **Proxy judge ($\text{VPTT}_{\text{score}}$):** for text-scale evaluation, $J(X, P)$ is approximated by $\text{VPTT}_{\text{score}}(p', P)$, shown in Sec. 4 of the main paper to correlate with human judgments.

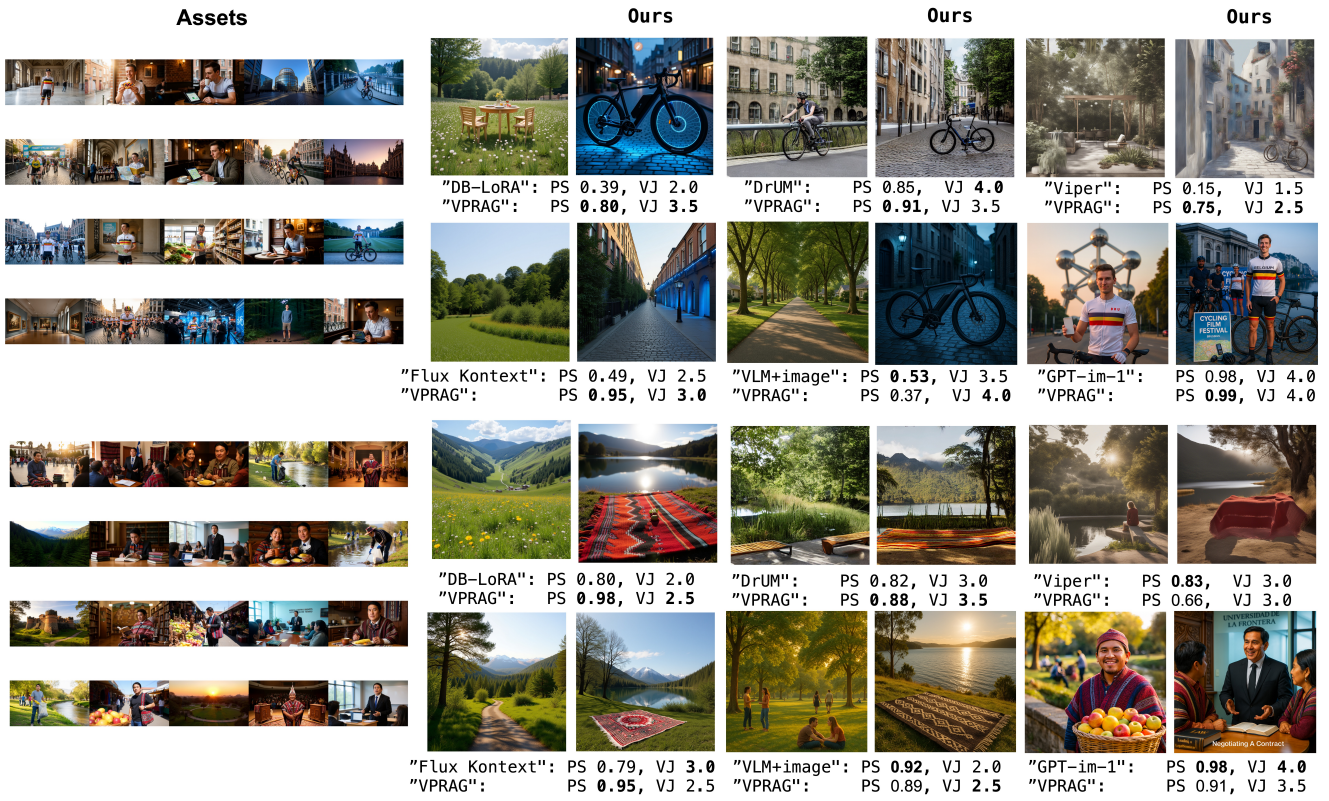

Figure 2. **Comparison to Visual Baselines.** We compare VPRAG (along three columns) against a broad set of visual personalization baselines, including fine-tuning approaches, preference-driven personalization methods, and multimodal LLM (MLLM)–based in-context techniques. Evaluation is conducted using two metrics: the VIPER Proxy Score (PS) [14] and the Gemini VLM Judge (see Sec. 4 in the main paper). Across more challenging and nuanced examples shown in the figure, VPRAG consistently emerges as an efficient and controllable personalization method, performing on par with or outperforming these substantially more expensive baselines.

Metric / Model	Flux DB-LoRA @50	Viper @ 1000	DrUM @1000	Flux Kontext @1000	GPT (VLM + GPT-image-1 @100)	GPT-image-1 @100
PS (Ours)	**0.867** ± 0.173	**0.678** ± 0.269	**0.841** ± 0.139	**0.839** ± 0.184	**0.858** ± 0.180	**0.974** ± 0.019
PS (Other)	0.656 ± 0.232	0.545 ± 0.292	0.757 ± 0.185	0.541 ± 0.297	0.832 ± 0.0.163	0.966 ± 0.047
PS Win % (Ours)	**80.0**	**71.5**	**68.9**	**83.4**	**63.0**	44.0
VJ (Ours)	**3.17** ± 0.70	**2.88** ± 1.13	**3.41** ± 0.57	**3.11** ± 0.73	**3.28** ± 0.75	3.73 ± 0.44
VJ (Other)	1.99 ± 0.67	2.40 ± 1.21	2.89 ± 0.61	2.29 ± 0.98	3.22 ± 0.71	**3.86** ± 0.35
VJ Win % (Ours)	**88.0 (+ 2% ties)**	**61.8 (+15.5% ties)**	**76.4 (+7.1% ties)**	**76.4 (+ 7.9% ties)**	**49.0 (+ 4% ties)**	33.0 (+ 12% ties)

Table 3. Benchmark comparison on VIPER Proxy Score (PS) and VLM Judge Score (VJ). Mean and standard deviation appear on separate lines. Win % reports the percentage of pairwise wins against the compared method for each metric.

1. Ablations and Analysis

Our N-shot robustness study (Table 1 shows that the main bottleneck in the extreme low-data regime ($N = 2$) is Gram–Schmidt reconstruction due to an underconstrained visual basis. Performance recovers quickly: at $N = 5$, VPRAG (**0.535**) overtakes the strong BRAG baseline (0.525), suggesting a minimum viable threshold of ∼5 images. As history increases ($N = 10 \rightarrow 30$), VPRAG consistently widens the gap over baselines, indicating robust gains.

We performed a micro-ablation ($\text{VPTT}_{\text{score}}$-c) on the living room task (see Table 2)). The results confirm that each proposed "heuristic" (e.g., entropy) is essential for preventing mode-collapse/generic outputs. Crucially, the

Figure 3. **Copy-Paste Effect.** The baselines including MLLMs suffer from copy-paste effect where the generations and edits only consider a single or few images of the user assets.

Random Retrieval baseline isolates the impact of retrieval quality, proving that performance gains stem from our retrieval logic rather than the prompt re-writer alone.

Additionally, we show that the text-only deferred renderings do not yield generic outputs with direct evidence. ***Humans distinguish generic conditioning from persona-specific retrieval.*** Compared to *Persona Only* (high-level demographics/interests; the "generic alignment" hypothesis), a blind human study (Table 1 main paper) prefers VPRAG (**54.6%**) over Persona Only (**19.2%**), nearly 3×. ***Implication:*** Such a gap is unlikely if outputs were generic, suggesting deferred renderings encode *persona-specific contextual signatures* (e.g., scene, mood/lighting, composition) recognized under VPTT. Figure 1 further supports this: across 4 prompts, generations show high **intra-profile consistency** but lower **inter-profile similarity**, indicating persistent persona cues rather than collapse to a generic mean.

2. Reproducibility

The version of VPTT-Bench evaluated in this paper was strictly safety-filtered. Due to organizational compliance policies regarding the distribution of synthetic media, the raw visual assets are retained internally. However, the comprehensive generation protocol (Section 6) is provided to allow the community to safely reconstruct the benchmark. These constraints do not affect our findings, as our conclusions rely on aggregate alignment patterns rather than specific images.

3. Limitations and Future Work

Synthetic–Real Gap. Because VPTT-Bench relies on a single family of generator models for producing the synthetic personas, the benchmark inevitably inherits stylistic and cultural biases of those models. This "real-to-sim" gap limits how faithfully the benchmark captures the full diversity of real users. A promising direction is to construct future versions of VPTT-Bench using a heterogeneous ensemble of generators across organizations and training paradigms. We argue, however, that a unified v1 benchmark is an essential step: it moves the field from a data-zero regime to one where controlled, scalable, and privacy-safe personalization research becomes possible.

Image-Only Scope. While our retrieval and alignment mechanisms are modality-agnostic, this work focuses on image generation and image editing. Extending VPTT to videos, 3D assets, and multi-view content is a natural next step, requiring new alignment metrics and temporal-consistency modules.

Scaling Beyond Individuals. Our method currently models single-person personalization. Future work can expand VPTT toward *societal personalization*: simulating communities, subcultures, and collective preference distributions. Such extensions could enable population-level evaluation, community-aware media generation, and product design aligned with specific cohorts.

Enhanced Visual Grounding. Persona assets are currently represented as rich textual "deferred renderings." Future work may couple these with segmentation or detection models to retrieve visual elements directly from user images for opt-in users. This would enable stronger grounding on real visual evidence and more faithful scene composition.

Structure Preservation. Current generators, including those used in VPRAG, do not guarantee preservation of spatial layout during editing. Incorporating structure-aware diffusion models or control modules (e.g., depth/segmentation guidance) may improve fidelity for demanding edit tasks.

Human-in-the-Loop Integration. VPRAG can naturally operate as a "visual copilot": retrieving user-specific cues, proposing edits, and letting the user refine preferences. Iterative preference learning, reinforcement from user feedback, and federated fine-tuning represent compelling next steps.

Real-World Deployment. Although we use synthetic personas for privacy reasons, the same dataset construction pipeline can be inverted to annotate and structure real user data in an opt-in or federated setting. This would enable applying the VPTT Framework directly on real personalization tasks while maintaining strong privacy guarantees.

4. VPTT at scale

4.1. Analogy for Deferred Rendering in VPTT

An analogy for our "deferred rendering" process is an expert film critic. A critic invests considerable effort watching hundreds of movies (the expensive offline alignment) to internalize what makes a script succeed on screen. Once trained, the critic can read a new script and predict whether

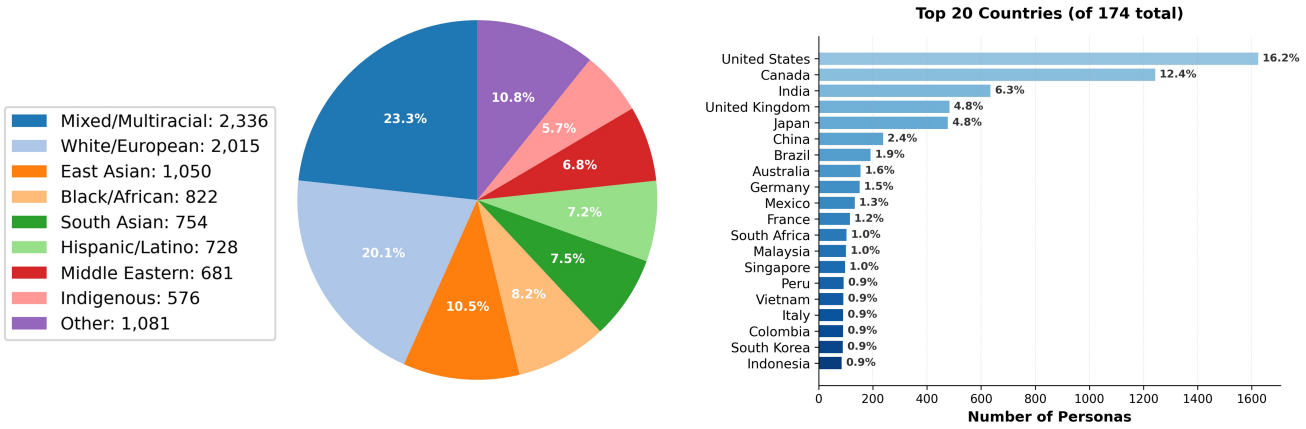

Figure 4. **VPTT-Bench Ethnicity and Location Diversity**. Ethnicity and location diversity of the users in VPTT-Bench

it would make a compelling film *before* spending millions producing it.

Similarly, VPTT evaluates a candidate prompt against a persona's visual identity in text form, first aligning Human/VLM judges/$\text{VPTT}_{\text{score}}$ and using cheap and reliable $\text{VPTT}_{\text{score}}$ *before* generating any images. This enables early rejection of weak generations, reducing costly rendering and accelerating personalization at scale.

4.2. Comparison to Visual Baselines

Returning to our "deferred rendering" analogy, VPTT allows us to evaluate a "script" (a candidate prompt) before producing the "film" (the final generated image). In this context, several existing personalization approaches can be interpreted as *high-budget productions* that must render the entire film before knowing whether it works:

- **Per-user finetuning methods** such as DreamBooth/LoRA [1] require retraining for each identity.
- **Preference-driven generation** systems such as VIPER and DrUM [6, 14] rely on only matching the preferences from images and text.
- **Multimodal LLM pipelines** (e.g., OpenAI GPT-4o VLM + GPT-Image-1 [10, 11], GPT-Image-1 [10], and Flux Kontext [9]) operate as large black-box modules that jointly hallucinate alignment and appearance, but remain difficult to control or steer.

Because these methods must generate/input images to refine alignment, they cannot benefit from early rejection or even privacy safe benchmarks. They thus incur high latency, high cost, and weaker controllability. In contrast, our VPRAG approach evaluates alignment in text first using $\text{VPTT}_{\text{score}}$, requiring no per-user training and no iterative image synthesis.

4.2.1. Evaluation Metrics

For evaluation, we assess generation quality using both automated metrics (VIPER proxy score [14], assigns higher scores to query images that share the preferred visual attributes) and human-aligned VLM judges (Aligned Gemini 2.0 Flash [4], see Sec 4 in the main paper), where judges compare baseline and personalized generations, i.e., using "A preferred outdoor spot" or "A Photo showing my next social media post with my style and content." and the personalized version per profile using VPRAG.

4.2.2. Evaluation Protocol

Flux DB-LoRA@50 We fine-tune FLUX.1-dev [8] using LoRA with rank 16 on attention layers, training for 1000 steps with the Prodigy optimizer and pivotal tuning on CLIP text encoder. Each user's LoRA is trained on 30 gallery images paired with a user-specific trigger word to learn personalized visual styles. Since training LoRAs is expensive, we train this baseline for 50 users to make this evaluation comprehensive.

VIPER@1000 We evaluate VIPER [14], a visual preference optimization baseline that personalizes SDXL [12] by optimizing text-to-image alignment. For each user, VIPER retrieves the top-10 most similar gallery posts given the prompt and uses both the images and captions to compute visual preferences (positive and negative prompts) that guide generation toward user-preferred visual styles. The methods generate images from the test prompt ("A preferred outdoor spot") and the personalized one (VPRAG RAG) using SDXL-base-1.0 [12]. Evaluation is conducted on 1,000 users. Comparison is done against a 5×2 grid of reference images (10 gallery posts/ assets).

DrUM@1000 We evaluate DrUM [6], a baseline that personalizes image generation by conditioning on the user

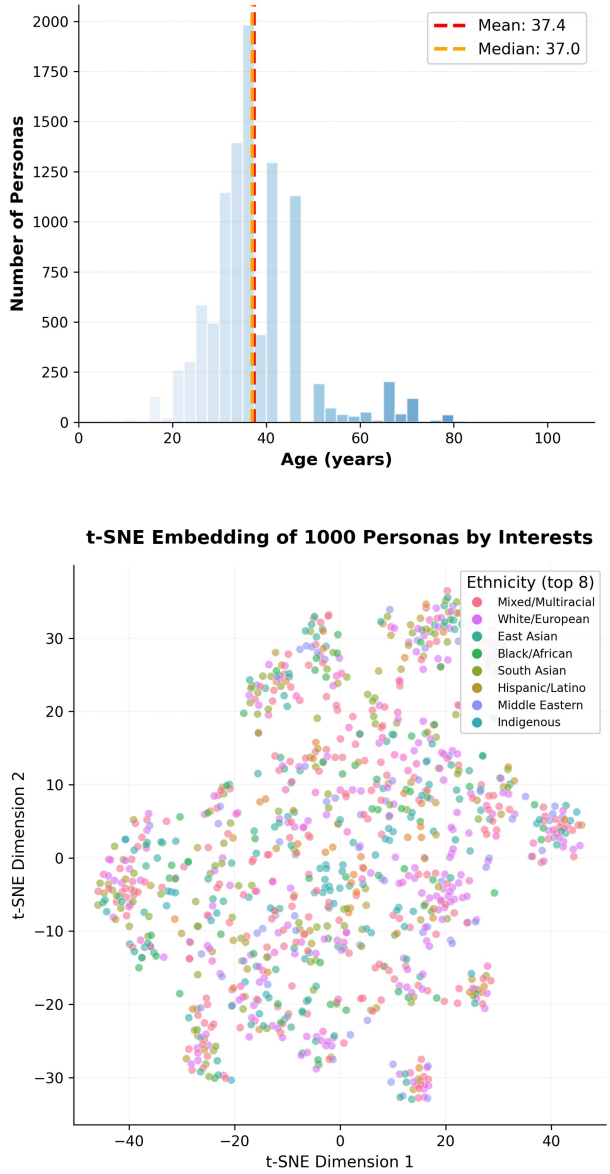

Figure 5. **VPTT-Bench Age and Interest Diversity**. Age Distribution and t-sne visualization (interests) of first 1000 users.

prompt history. For each user, DrUM retrieves the top-5 most similar captions from their gallery posts given the prompt and uses them as reference prompts with personalization strength $\alpha = 0.5$. The methods generate images from the test prompt ("A preferred outdoor spot") and the personalized one (VPRAG RAG) using Stable Diffusion v1.5 [2]. Evaluation is conducted on 1,000 users, where comparison is done using both methods' outputs against a 5×2 grid of reference images (10 gallery posts/ assets).

FLUX Kontext @1000 We evaluate FLUX.1-Kontext-dev [7] in-context learning capability by conditioning generation on a 5×5 grid of 25 reference images from each user's gallery. For each user, we generate two images: one with the base prompt alone and one with a persona-enhanced prompt i.e. VPRAG (both conditioned on the same reference grid), allowing us to assess whether in-context visual conditioning alone is sufficient for personalization. Here we evaluate 1000 users for comprehensive evaluation.

Large Multi-Modal Models We compare VPRAG against OpenAI's GPT-Image-1 [10] with two approaches: (1) **GPT (VLM + GPT-Image-1 @100)** [10, 11] a visual analysis baseline where GPT-4o [11] analyzes a 5×5 grid of 25 user gallery images to extract visual style preferences (foreground, background, materials, objects, lighting, actions, environment, appearance) and generates a refined 3-phrase prompt that incorporates these elements alongside the base prompt ("A preferred outdoor spot"), and (2) **GPT-image-1 @100** visuals generated directly using 5×5 grid of 25 user gallery images by GPT-Image-1 [10]. This evaluation used the unconstrained version of VPRAG with the prompt "A Photo showing my next social media post with my style and content." (Figure. 2 in the main paper). These are compared with the VPRAG augmented generations.

The system prompt for the **GPT (VLM + Image Gen @100)** baseline is:

```
You are an expert at analyzing
visual styles and preferences from
image collections.
```

```
Analyze the provided images and
create a detailed image generation
prompt that will generate a new
image matching both:
1.  The requested text prompt
2.  The visual elements from the
reference images so the generated
image looks like it's from the
same gallery
Focus on visual elements like:
foreground, background, materials,
objects, lighting, actions,
environment, appearance, etc.
Your prompt should be EXACTLY
3 short descriptive phrases
separated by commas.
```

The user prompt of the baseline is:

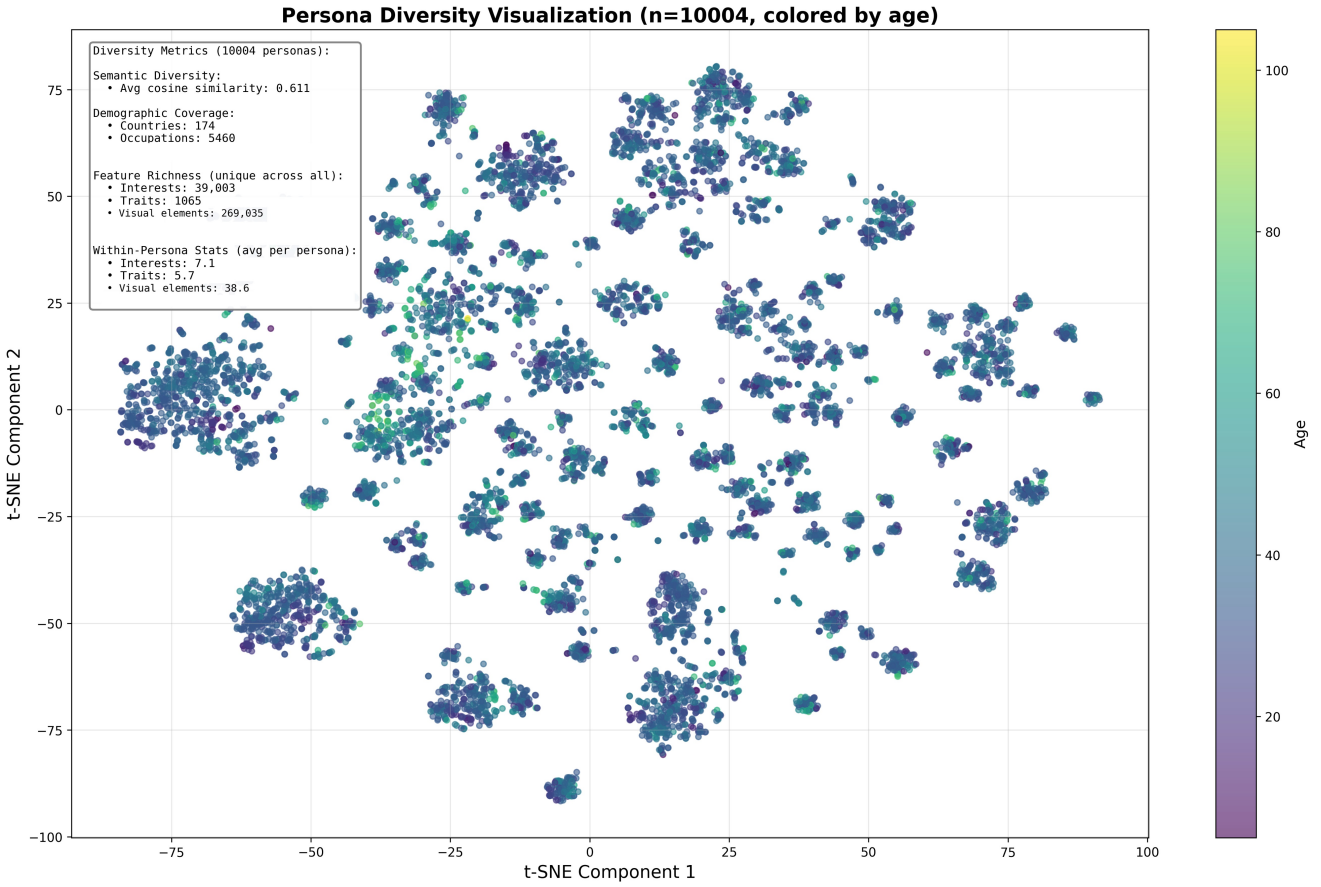

Figure 6. **Diversity of 10K Synthetic Personas.** We visualize the diversity of our 10,004 synthetic personas using t-SNE dimensionality reduction on averaged caption embeddings (OpenAI text-embedding-3-small, 1536-dim) from each persona's 30-image gallery. Points are colored by age. The average pairwise cosine similarity of 0.611 indicates balanced diversity ; personas occupy a shared human aesthetic space while maintaining distinct individual preferences. Our dataset spans 174 countries and 5,460 unique occupations, with 39,003 unique interests and 269,035 visual elements across all personas. Each persona averages 7.1 interests, 5.7 personality traits, and 38.6 visual elements, ensuring rich and diverse personalization signals for image generation models.

```
Here are images from a user's
profile. Analyze the visual
style, color preferences,
composition patterns, and
aesthetic choices in these images.
Pay attention to: foreground
elements, background, materials,
objects, lighting, actions,
environment, and overall
composition.
Create an image generation prompt
for: "base_prompt"
The prompt should incorporate
visual elements from these images
so the generated image feels like
it's part of the same gallery.
IMPORTANT: Your response must be
EXACTLY 3 short phrases separated
by commas.
```

The prompt for the **GPT-image-1 @100** baseline is:

```
Focus on visual elements like:
foreground, background, materials,
objects, lighting, actions,
environment, appearance, etc. in
the images in this grid. Generate
an output image for "base_prompt"
using these visual elements such
that the resultant image also
feels like it belongs to this
profile.
```

The prompt for the **GPT-image-1 @100** VPRAG is:

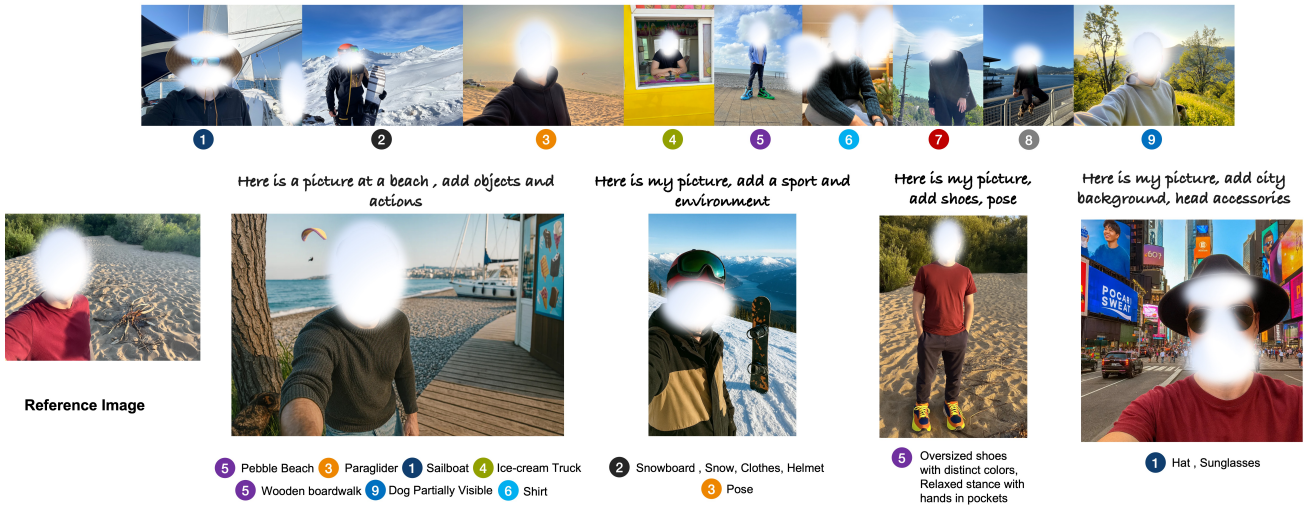

Figure 7. This Figure is only for illustration and not a part of the main dataset. The human figures shown in the sample images are non-author volunteers who provided consent. Their faces and all identifying cues (e.g., location) are fully anonymized.

Figure 8. This Figure is only for illustration and not a part of the main dataset. The human figures shown in the sample images are non-author volunteers who provided consent. Their faces and all identifying cues (e.g., location) are fully anonymized.

> Using the visual style from these reference images, create: "*personalized_prompt*".

4.2.3. Results

Table 3 compares these baselines. Here, our method outperforms all the methods or has comparable performance with the large multi-model models. In Figure 2, we show results on more nuanced and difficult examples, where VPRAG consistently emerges as an efficient and controllable personalization method, performing on par with or outperforming these substantially more expensive baselines. While in-context learning (ICL) approaches such as GPT-4o [11] or GPT-Image-1 [10] can condition generation on a set of reference images, they suffer from two fundamental limitations that our persona-based formulation directly addresses.

First, ICL is not controllable. Without explicit structure, these models frequently copy or closely mimic individual gallery images rather than synthesizing novel content (see Figure 3) from a coherent blend of visual attributes. In our evaluation, penalizing such copy-paste behavior results in a substantial performance drop for the ICL baseline (4.08 $\rightarrow$ 3.86; a 5.4% decline), whereas our persona-enhanced method exhibits far greater robustness (3.83 $\rightarrow$ 3.73; only a 2.6% decline). This indicates that our approach learns to ag-

Figure 9. **Contextual Image Generation and Editing using VPTT-Bench.** Each row shows a distinct user profile: assets and style cues (left), personalized generations (social post, cultural site), and edits (garden, living room) guided by the same persona identity. All images are generated synthetically via our Visual Personalization RAG (VPRAG) by text, which retrieves persona-aligned cues. To show cross model personalization here the assets are generated by QWEN-image-model [15] and generations and edits by Nano-Banana [5] conditioned only on the first image.

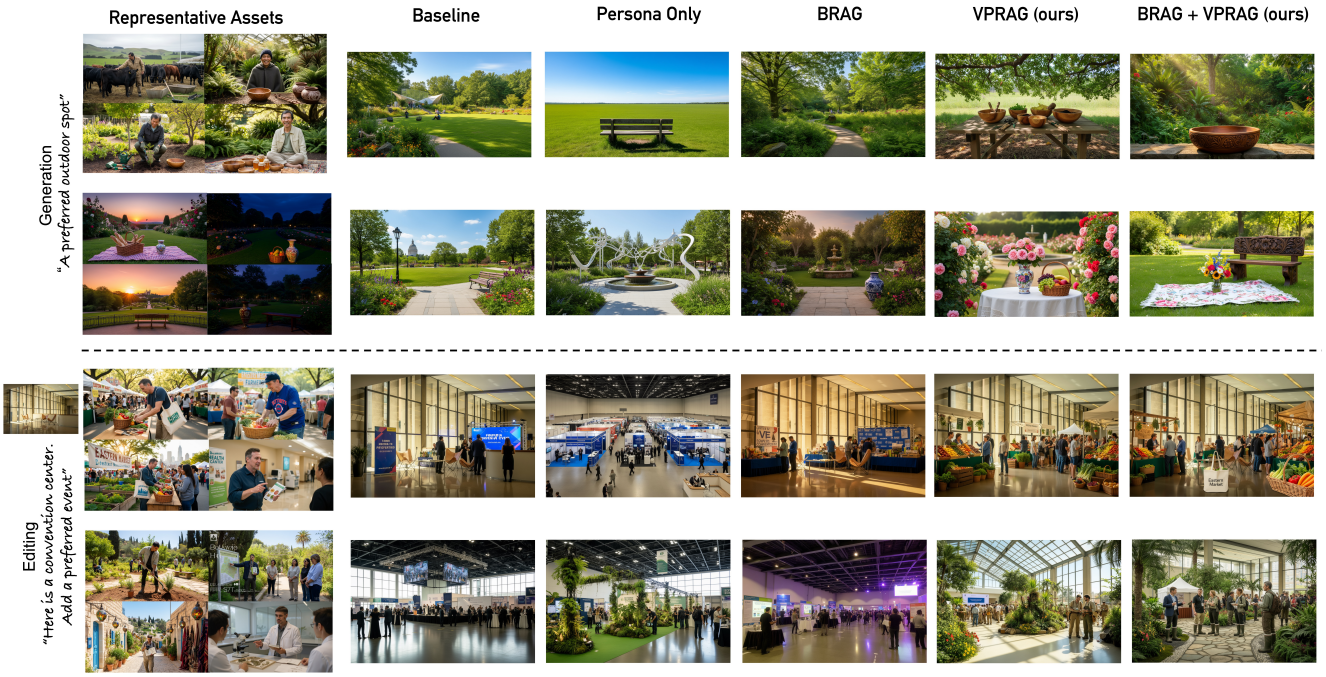

Figure 10. **Qualitative Comparison across Generation and Editing Tasks.** Representative examples from the VPTT-Bench showing outputs from five methods: Baseline, Persona Only, BRAG, VPRAG (ours), and BRAG + VPRAG (ours).

gregate and recompose visual patterns across multiple references instead of replicating isolated scenes. Moreover, ICL provides no principled mechanism for selecting which visual attributes to incorporate: all gallery images are treated uniformly, causing models to focus on salient but potentially irrelevant cues.

Second, ICL is not scalable. Performance degrades as gallery size increases due to context-window constraints and attention dilution, and inference cost grows linearly with the number of reference images ($\mathcal{O}(n)$). This makes evaluation impractical for settings requiring richer user histories or larger galleries.

Third, ICL is not economically viable at scale. At GPT-4o Vision pricing (approximately $0.01 per processed image), a single personalized generation conditioned on a typical gallery (e.g. 25 images) incurs roughly $0.25 in image-token cost alone. A user generating 100 personalized images would therefore cost about $25; serving one million

such users would exceed $25M, excluding text-token fees and overhead. Additionally, these closed-source APIs impose rate limits and quota restrictions, rendering them unsuitable for high-volume, production-scale personalization workloads.

As larger vision models continue to become more democratized and cost-efficient, VPTT will only become more practical and broadly applicable, enabling scaled evaluation across a wider range of personalization tasks.

5. Additional Results

Additional results as an extension to Figure 2 in the main paper are shown in Figure 9. Additionally, more examples , also an extension to Figure 6 in the main paper are shown in Figure 10.

6. VLLM-Bench Construction (Detailed)

6.1. Conceptual Basis: Deferred Rendering

VLLM-Bench conceptualizes text generation as *deferred rendering of visual identity*. Instead of pixels, each profile is expressed through language-level equivalents of visual cues i.e., objects, lighting, actions, background, foreground, materials, appearance, expressions, pose etc., that together represent how a concept would appear in visual media. This abstraction decouples personalization from rendering, enabling scalability, and privacy-preserving use.

Bidirectional Symmetry. The process is fully reversible:

$$\text{images} \rightarrow \text{caption} \rightarrow \text{visual elements} \rightarrow \text{preferences} \rightarrow \text{persona}.$$

In forward mode, we generate structured text; in inverse mode, real user profiles can be captioned and converted into the same structure, enabling safe, text-only adaptation.

6.2. Demographic Generation

6.2.1. Seed Initialization from PersonaHub

We initialize 10K personas from **PersonaHub** [3], which contains ≈200K curated human-authored persona descriptions. Each persona is derived using:

$$\text{seed\_index}_i = \text{hash}(i) \bmod 200{,}000, \tag{1}$$

ensuring deterministic diversity across geography, age, and profession.

6.2.2. Two-Stage Demographic Expansion

Stage 1a: LLM-Based Demographics with Bias Mitigation. Given a seed s, we use Qwen2.5-72B-Instruct to infer country, city, ethnicity, and gender. When confidence in location grounding is low, hash-based diversity re-mapping adjusts sampling with region-specific override rates: India/South Asia (70%), United States (65%), United Kingdom (60%), and Canada (50%). This guarantees balanced representation across 9 ethnicity groups and 60+ authentic cities.

Stage 1b: Persona Completion. Demographic scaffolds are expanded into 20+ attributes, including occupation, education, interests (5–8 domain-specific), social-media tone, and lifestyle traits. Gender is inferred only when explicitly stated in the seed description; otherwise, it is marked as "unknown" to avoid introducing occupational or cultural stereotypes. While residual bias may still propagate through downstream image-generation models, the ***final version of VPTT-Bench will include additional filtering and adjustments to mitigate such demographic biases***.

6.3. Visual Elements and Preference Generation

Each persona contains a structured **visual vocabulary** with 15–20 entries per facet:

- **Foreground:** subjects, actions, objects, body poses;
- **Background:** environments, landmarks, lighting, textures;
- **Atmospheric:** materials, color palette, mood, time of day.

At least 70% of all elements reference culturally authentic motifs drawn from the persona's region (e.g., Seoul Tower, Kashmiri gardens, or Venice canals).

We also generate 15–20 aesthetic and behavioral **preferences** (e.g., "prefers warm lighting," "posts minimalist compositions," "documents festivals") that act as latent conditioning factors. These are then used in feedback simulation part of the method to learn subjective preferences.

6.4. Scenario and Caption Generation

Each persona produces 30 posts in two phases:

1. **Scenario Generation:** We sample 6–8 high-temperature ($\tau = 0.9$) scenarios per batch with diversity constraints across content type (35% activity, 25% appreciation, 25% shared content, 15% selfie), temporal variety (day/night, seasonal), and social context (solo/group).
2. **Caption Synthesis:** For each scenario, the LLM behaves as a vision-language model and produces a 150–250-word caption containing: (i) compositional details, (ii) cultural context, (iii) visible preferences, and (iv) annotated facets (foreground, background, atmosphere).

Each caption is encoded using the `text-embedding-3-small` model (1536 D). Unused elements are pruned to ensure structural consistency.

6.5. Parallelized Generation Pipeline

The dataset is produced on an 8×A100 GPU cluster with **vLLM-optimized Qwen2.5-72B**. Dynamic batching (10–200 profiles) yields 50–150 profiles/hour. Generation of 10K profiles (300K posts) completes in ≈66–200 hours. See Table. 4.

Table 4. **VPTT-Bench Generation Statistics.**

Metric	Value
Total personas	10 000
Posts per persona	30
Total posts	300 000
Mean caption length	187.2 words
Mean visual elements/persona	45.3
Parallel throughput	50–150 profiles/hr

Profiles with fewer than 10 valid posts are excluded. All attributes, embeddings, and metadata are stored in JSONL format.

6.6. Privacy, Scalability, and Extensibility

Because all profiles are text-based, VPTT-Bench operates fully under deferred rendering, guaranteeing privacy and model-agnostic applicability. The dataset can be scaled to millions of profiles or augmented with real-world profiles through inverse captioning:

$$\text{caption} \rightarrow \text{elements} \rightarrow \text{preferences} \rightarrow \text{persona}.$$

This symmetric design ensures both the generative and analytical components of VPTT Framework can operate without any visual exposure, making VPTT-Bench a reusable personalization benchmark.

7. Visual Assets Generation

7.1. Mathematical Face Diversity System

To ensure controlled, globally diverse identity synthesis, we implement a deterministic facial attribute generator producing 97.2M unique combinations. These are then added to the demographics description to first generate a user persona image and then conditioned on this image to generate 30 assets.

Attribute Space. Ten facial attributes with 4–6 discrete options each are defined: face shape, eye shape, eye size, nose type, jawline, cheekbones, lip shape, eyebrow type, face length, and chin shape. For each user ID and age group, we compute:

$$\text{seed} = \text{hash}(\text{user\_id}, \lfloor \tfrac{\text{age}}{10} \rfloor) \bmod 2^{32},$$

and draw attributes $F = \{a_1, \ldots, a_{10}\}$ from the option sets $\{O_i\}$. Modifiers such as age-adapted details (e.g., "bright eyes" vs. "wisdom lines"), expression labels, and photo styles are applied to achieve additional realism.

Combinatorial Diversity.

$$N = \prod_{i=1}^{10} |O_i| \times |A| \times |E| \times |S| \approx 9.72{\times}10^7,$$

where A denotes age modifiers, E expression states, and S photo styles. This formulation ensures reproducible sampling and balanced variation across users.

7.2. Two-Phase Image Generation Pipeline

Phase 1: Persona Base Generation (Text-to-Image). Each persona's base portrait is synthesized using Qwen-Image 2509 [13, 15] diffusion models. Prompts combine demographics and generated facial attributes:

> "photo of a person, {gender}, {race/ethnicity}, {age} years old, works as {occupation}, from {city, country}, {oval face shape}, {almond eyes}, {high cheekbones}, {full lips}, professional portrait, confident expression, natural lighting."

Generation parameters:
- Model: Qwen-Image or Qwen-Image-Edit (vanilla mode)
- Resolution: 1344×768
- Steps: 50, CFG=0.0, Seed=Deterministic per user
- No negative prompts (maximizes diversity)

Phase 2: Post-Specific Editing (Image-to-Image). Each persona's 30 textual posts is rendered by Qwen-Image-Edit-2509. Prompts differ by post type:
- **Activity / Selfie / Shared Content (70%):**

  > "{caption}, wearing {clothing}, with {expression}, {pose}."

- **Appreciation Posts (30%):**

  > Prompt: "{scene_description}" Negative: "person, people, human, face, body, portrait"

Configurations.
- **Standard Mode:** 40 steps, CFG=4.0, ∼15–20 s/image
- **Lightning LoRA Mode:** 4–8 steps, CFG=4.5, ∼3–5 s/image (4× faster)

Pronouns are replaced with "this person" to ensure gender neutrality.

7.3. Parallel Multi-GPU System

An 8×A100 cluster executes both phases in parallel. Models are cached per GPU; tasks are dynamically queued via thread-safe managers to maintain 100% utilization. Phase 1 (base portraits) and Phase 2 (post edits) can run independently or sequentially (Table. 5).

Table 5. Synthetic Image Generation Performance Metrics.

Metric	Value
Throughput (standard)	50–80 images/hour/GPU
Throughput (Lightning)	180–240 images/hour/GPU
Memory footprint	<24 GB/GPU (bfloat16 precision)

8. VPTT-Bench Stats

To illustrate the diversity of the benchmark, Figure 4 presents the distribution of ethnicities and countries of origin. Despite the modest number of samples, the population is highly diverse. Similarly, Figure 5 reports the age distribution of the benchmark and the interests of the first 1,000 users, grouped by ethnicity. At a larger scale, Figure 6 visualizes the averaged caption embeddings of 10K users, highlighting diversity across age groups and visual attributes.

9. VPRAG Algorithm

To formally define the steps used by our VPRAG method, Algorithm 1 shows a compact form of the retrival engine.

10. Real-World Examples

Only for demonstration purposes, we include a small set of real-world example images that illustrate the types of visual inputs supported by our method. These images are not part of the training set, evaluation benchmarks, or any quantitative analysis; they are shown solely to help readers qualitatively understand the range of scenarios in which the system operates.

The human figures (Figures. 7 and 8) shown in the sample images are non-author volunteers who provided informed consent for their anonymized photos to be used for illustration. All faces and identifying features (e.g., facial attributes, backgrounds revealing location) have been fully obscured to preserve privacy. These individuals have no relationship to the authorship of the paper, and their inclusion does not reveal author identity in any way.

These examples highlight the diversity of environments, poses, and visual conditions encountered in typical user-generated content, and demonstrate how the proposed system generalizes across varied real-world scenes.

11. Expanded Tables.

Tables 6, 7, 8, 9, shows the expanded versions of the Table 3 in the main paper. Here we report both $\mathrm{VPTT}_{\mathrm{score}}$-c and $\mathrm{VPTT}_{\mathrm{score}}$ scores showing the results are consistent with the experiments in the main paper. Cohen's d in these table are computed against the Baseline.

Algorithm 1 VPRAG with Optional Feedback Re-ranking

Inputs: query p, persona memory $\mathcal{M} = \{(E_i, c_i)\}_{i=1}^{N}$, categories $\mathcal{C}$, budgets $\{Q^{(k)}\}$, temperature τ, category embeddings $\{\mathbf{u}_k\}$
Embedders: post: $\mathrm{Embed}_{\mathrm{OpenAI}}$, element: $\mathrm{Embed}_{\mathrm{MiniLM}}$
Outputs: re-prompt p', (optional) re-ranked p'^{*}
Post-level retrieval:
$\mathbf{q} \leftarrow \frac{\mathrm{Embed}_{\mathrm{OpenAI}}(p)}{\|\cdot\|_2}$; $\mathbf{v}_i \leftarrow \frac{\mathrm{Embed}_{\mathrm{OpenAI}}(c_i)}{\|\cdot\|_2}$
$w_i \leftarrow \frac{\exp(\mathbf{q}^\top \mathbf{v}_i/\tau)}{\sum_j \exp(\mathbf{q}^\top \mathbf{v}_j/\tau)}$; $H \leftarrow -\sum_i w_i \log w_i$; $n_{\mathrm{eff}} \leftarrow \exp(H)$
$Q \leftarrow \sum_k Q^{(k)}$; $K \leftarrow \min\left(\lfloor n_{\mathrm{eff}} \rfloor, 2Q\right)$
$\mathcal{I} \leftarrow \mathrm{TopKIndices}(w, K)$
Category priorities & quotas:
$\mathrm{priority}_k \leftarrow \mathbf{q}^\top \mathbf{u}_k$; $\mathcal{C}_{\mathrm{sorted}} \leftarrow \mathrm{SortBy}(\mathrm{priority}_k)$
for $k \in \mathcal{C}_{\mathrm{sorted}}$ **do**
  $c_i^{(k)} \leftarrow |E_i^{(k)}|,\ i \in \mathcal{I}$; $q_i^{(k)} \leftarrow \left\lfloor \frac{w_i c_i^{(k)}}{\sum_{j\in\mathcal{I}} w_j c_j^{(k)}} Q^{(k)} \right\rfloor$
end for
Element ranking (atomic):
$\mathbf{q}_{\mathrm{elm}} \leftarrow \frac{\mathrm{Embed}_{\mathrm{MiniLM}}(p)}{\|\cdot\|_2}$; $\mathcal{E}_p \leftarrow \emptyset$
for $k \in \mathcal{C}_{\mathrm{sorted}}$ **do**
  for $i \in \mathcal{I}$ **do**
    if $q_i^{(k)} = 0$ **then** continue
    $\mathcal{S}_{i,k} \leftarrow \mathrm{TopK}\left(E_i^{(k)},\ q_i^{(k)};\ \cos(\mathrm{Embed}_{\mathrm{MiniLM}}(\cdot), \mathbf{q}_{\mathrm{elm}})\right)$
    $\mathcal{E}_p \leftarrow \mathcal{E}_p \cup \mathcal{S}_{i,k}$
  end for
end for
Compose: $p' \leftarrow f_{\mathrm{compose}}(p, \mathcal{E}_p)$ {or $f_{\mathrm{compose}}(p, \mathcal{S}_p, \mathcal{E}_p)$ if $\mathcal{S}_p$ is precomputed}
(Optional) feedback re-ranking:
Train small f_θ on few profiles: $(p', \mathcal{P}) \mapsto s_{\mathrm{VLM}} \in [0, 1]$
At inference, sample $\{p'_m\}_{m=1}^{M}$ and pick $p'^{*} = \arg\max_m f_\theta(\mathrm{Embed}(p'_m), \mathrm{Embed}(\mathcal{P}))$
return p' (or p'^{*})

12. User Study Protocol

To measure how well generated images align with an individual's visual style, we conducted a human evaluation following the VPTT. Each task presented annotators with a 10–image gallery representing a user's typical aesthetics, environments, lighting preferences, clothing patterns, and recurring visual motifs. Alongside the gallery, annotators viewed a 2×2 grid of generated images (Methods A–D). Participants rated each generated image independently using a slider ranging from 0 to 5, guided strictly by visual similarity to the user's gallery rather than image quality, personal preference, or cross-method comparison.

This setup allowed us to isolate whether a generated sample *belonged to the same visual world* as the user's posts. Annotators were trained to focus on concrete sig-

nals such as objects, materials, environments, appearance patterns, lighting, and cultural or stylistic markers. By collecting similarity judgments across thousands of examples, we obtained a fine-grained human signal for the plausibility and consistency of personalization across diverse prompts and visual domains. Here is a concise form of the instructions:

Annotation Instructions

Rate each generated image from 0 to 5 based on:

-- Objects & Materials: Are key objects or textures similar to those in the gallery?
-- Environments & Settings: Are backgrounds or locations consistent with the user's style?
-- Appearance Patterns: Clothing style, color palette, accessories, poses?
-- Lighting & Atmosphere: Similar mood, time of day, natural/white lighting?
-- Cultural / Style Markers: Recurring themes, sports references, regional dress, etc.

Do NOT: rate based on image quality, personal preference, comparison across methods, or prompt correctness.

Score Guide:

5 = Excellent match (fits naturally in user's gallery)
4--4.5 = Good match
3--3.5 = Moderate similarity
2--2.5 = Weak similarity
1--1.5 = Minimal similarity
0--0.5 = No similarity at all

12.1. VLM Judge for Automatic Persona Evaluation

To complement the human user study, we use a vision–language model (VLM) as an automatic judge that approximates the same visual-similarity protocol. For each user in the *generation* split, we first construct a *profile canvas* by tiling up to 10 of their posts into a 5×2 grid, with each post numbered. We then construct a *methods canvas* by arranging the five generated images from different methods horizontally and assigning them blind labels A–E via a user-specific but deterministic shuffle. The VLM receives the baseline textutal generation prompt, the profile canvas, and the methods canvas as inputs, and is asked to score each of A–E independently on a 0–5 scale based purely on visual similarity to the gallery, mirroring the human instructions.

For the *editing* split, the setup is identical except that we additionally provide the original input image that was edited. The same VLM judge prompt structure is used, but the user message explicitly refers to an editing task and includes the editing prompt. In both cases, we query either GPT-4o Vision or Gemini-2.5-Pro (to remove the model bias for the generations by 4o-mini or Gemini-2.5-Pro) with a fixed system instruction and a task-specific user instruction. The model returns natural-language lines that we parse into per-method scores in $[0, 5]$ (with 0.5 increments) and short explanations.

System prompt (VLM judge):

You are an expert visual judge evaluating AI-generated images for visual similarity to a user's persona.
Evaluate how well each generated image captures the VISUAL ELEMENTS from the persona's posts.
EVALUATION CRITERIA (Visual Similarity):
1. **Objects & Materials**: Same objects, materials, textures visible in posts?
2. **Environment & Setting**: Similar locations, backgrounds, environments? 3. **Appearance Patterns**: Similar clothing styles, colors, expressions?
4. **Lighting & Atmosphere**: Similar lighting, mood, atmosphere? 5. **Colors & Composition**: Similar color palettes and visual composition?
6. **Cultural/Style Markers**: Similar cultural elements, aesthetic style?
SCORING (0-5 scale, use 0.5 increments): - **5.0**: Excellent visual similarity - Captures most key visual elements from posts - **4.0-4.5**: Good visual similarity - Several key visual elements present

- **3.0-3.5**: Moderate visual similarity - Some visual elements recognizable - **2.0-2.5**: Weak visual similarity - Few visual elements match - **1.0-1.5**: Minimal visual similarity - Barely any visual connection - **0.0-0.5**: No visual similarity - Completely different visual style **IMPORTANT**: Focus on VISUAL similarity, not just conceptual alignment.

User prompt (generation tasks):

TASK: Evaluate these 5 generated images for visual similarity to the persona's posts
GENERATION PROMPT: "{base_prompt}"
IMAGE 1 (Reference - Profile Context): Grid of selected posts from the persona's gallery (numbered).
IMAGE 2 (Generated Images to Evaluate): 5 generated images labeled A through E (left to right). Each was generated using a different approach (you don't know which approach was used for which image).
YOUR TASK: For EACH image (A, B, C, D, E), score 0-5 based on: - How similar are the VISUAL ELEMENTS (objects, environment, appearance, lighting, colors)? - Do the generated images look like they could belong to the same persona's gallery? - Are there recognizable visual patterns from the posts?
Respond in this EXACT format (one line per image): A: Score=X.X - [1-2 sentence explanation of why this score, focusing on specific visual elements] B: Score=Y.Y - [1-2 sentence explanation of why this score, focusing on specific visual elements] C: Score=Z.Z - [1-2 sentence explanation of why this score, focusing on specific visual elements] D: Score=W.W - [1-2 sentence explanation of why this score, focusing on specific visual elements] E: Score=V.V - [1-2 sentence explanation of why this score, focusing on specific visual elements]

User prompt (editing tasks):

TASK: Evaluate these 5 edited images for visual similarity to the persona's posts
EDITING PROMPT: "{base_prompt}"
IMAGE 1 (Input Image): The original input image that was edited.
IMAGE 2 (Reference - Profile Context): Grid of selected posts from the persona's gallery (numbered).
IMAGE 3 (Edited Images to Evaluate): 5 edited images labeled A through E (left to right). Each was edited using a different approach (you don't know which approach was used for which image).
YOUR TASK: For EACH image (A, B, C, D, E), score 0-5 based on: - How similar are the VISUAL ELEMENTS (objects, environment, appearance, lighting, colors)? - Do the edited images look like they could belong to the same persona's gallery? - Are there recognizable visual patterns from the posts?
Respond in this EXACT format (one line per image): A: Score=X.X - [1-2 sentence explanation of why this score, focusing on specific visual elements] B: Score=Y.Y - [1-2 sentence explanation of why this score, focusing on specific visual elements] C: Score=Z.Z - [1-2 sentence explanation of why this score, focusing on specific visual elements]

```
D: Score=W.W - [1-2 sentence
explanation of why this score,
focusing on specific visual
elements] E: Score=V.V - [1-2
sentence explanation of why this
score, focusing on specific visual
elements]
```

13. Implementation Details

For Table 3 in the main paper, we use Qwen2.5-7B-Instruct via vLLM for text generation ($T = 0.1$, top-p $= 0.9$, max tokens $= 256$, seed $= 42$). For GPT-4o-mini, we use $T = 0.1$, seed $= 42$. We use Gemini-2.5-Pro with $T = 0.7$, top-p $= 0.95$. This temperature is chosen as we noticed that lower temperatures tend to truncate the text. Our soft assignment mechanism computes post-level attention weights via softmax with temperature $\tau = 0.1$.

For the VLM judges in the main paper, we use Gemini-2.5-Pro with temperature $T = 0.0$, top-p $p = 0.95$, and maximum output tokens of 5000. Another variant is the GPT-4o Vision with temperature $T = 0.0$, and seed = 42.

For the feedback network, we train a lightweight cross-attention transformer to predict user-prompt alignment scores. The model takes text-embedding-3-small (1536-dim) embeddings of user profiles and prompts as input, projecting them to a 128-dimensional hidden space. Cross-attention with 4 heads allows the profile representation to attend to prompt features, followed by a feed-forward network (128 $\rightarrow$ 256 $\rightarrow$ 128) with residual connections and layer normalization. The final prediction head (256 $\rightarrow$ 128 $\rightarrow$ 64 $\rightarrow$ 1) outputs scores in [0,1] via sigmoid activation. We train with AdamW (lr=0.001, weight decay=0.05) using MSE loss, with dropout=0.2 for regularization and early stopping (patience=10).

References

[1] Low-rank adaptation for fast text-to-image diffusion fine-tuning. https://github.com/cloneofsimo/lora, 2022. 4

[2] Stability AI. stable-diffusion-v1-5/stable-diffusion-v1-5. *https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5*, 2022. 5

[3] Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas, 2025. 9

[4] Gemini. Gemini 2.0 flash. *https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash*, 2025. 4

[5] Google. Nanobanan. *https://aistudio.google.com/models/gemini-2-5-flash-image*, 2025. 8

[6] Hyungjin Kim, Seokho Ahn, and Young-Duk Seo. Draw your mind: Personalized generation via condition-level modeling in text-to-image diffusion models, 2025. 4

[7] BlackForest Labs. black-forest-labs/flux.1-kontext-dev. *https://huggingface.co/black-forest-labs/FLUX.1-Kontext-dev*, 2025. 5

[8] BlackForest Labs. black-forest-labs/flux.1-dev. *https://huggingface.co/black-forest-labs/FLUX.1-dev*, 2025. 4

[9] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 4

[10] OPENAI. Gpt-image-1. *https://openai.com/*, 2025. 4, 5, 7

[11] OpenAI et al. Gpt-4o system card, 2024. 4, 5, 7

[12] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. In *ICLR*. OpenReview.net, 2024. 4

[13] Qwen. Qwen-image-edit-2509. *https://huggingface.co/Qwen/Qwen-Image-Edit-2509*, 2025. 10

[14] Sogand Salehi, Mahdi Shafiei, Teresa Yeo, Roman Bachmann, and Amir Zamir. ViPer: Visual personalization of generative models via individual preference learning. *arXiv preprint arXiv:2407.17365*, 2024. 2, 4

[15] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. 8, 10

Table 6. Method Comparison Across LLMs (Cultural Site Prompt)

LLM	Method	Individual Metrics				$VPTT_{score}$-c (Uniform Weights)			$VPTT_{score}$ (Novelty Adjusted)		
		PA	GS	CP	NV	Score	Win%	d *(vs base)*	Score	Win%	d *(vs base)*
4o-mini	Baseline	0.150	0.375	0.589	–	0.371	0.0%	–	0.319	0.0%	–
	Persona Only	0.364	0.429	0.630	–	0.474	0.1%	4.18	0.391	0.0%	3.62
	BRAG	0.316	0.597	0.674	0.858	0.529	8.7%	4.40	0.616	11.5%	10.94
	VPRAG (Ours)	0.401	0.591	0.660	**0.900**	0.551	17.9%	5.76	0.635	31.8%	**13.19**
	Comb. VPRAG+BRAG (Ours)	**0.419**	**0.641**	**0.686**	0.821	**0.582**	**73.3%**	**5.95**	**0.646**	**56.8%**	12.73
Qwen	Baseline	0.150	0.375	0.589	–	0.371	0.0%	–	0.319	0.0%	–
	Persona Only	0.325	0.417	0.627	–	0.456	0.1%	3.45	0.378	0.0%	3.00
	BRAG	0.395	**0.670**	**0.683**	0.441	**0.583**	**48.6%**	5.38	0.573	14.5%	6.34
	VPRAG (Ours)	0.378	0.587	0.656	**0.863**	0.540	11.5%	5.35	**0.621**	**62.0%**	**12.24**
	Comb. VPRAG+BRAG (Ours)	**0.414**	0.649	0.678	0.544	0.580	39.9%	**5.43**	0.590	23.5%	6.93
Gemini	Baseline	0.150	0.375	0.589	–	0.371	0.0%	–	0.319	0.0%	–
	Persona Only	0.278	0.407	0.614	–	0.433	0.1%	2.23	0.362	0.0%	2.00
	BRAG	0.286	0.606	0.647	0.775	0.513	18.2%	3.12	0.588	11.6%	7.82
	VPRAG (Ours)	**0.359**	0.597	0.656	**0.893**	0.537	31.9%	4.69	**0.626**	**58.0%**	**11.30**
	Comb. VPRAG+BRAG (Ours)	0.349	**0.635**	**0.667**	0.768	**0.550**	**49.8%**	**4.70**	0.614	30.4%	9.46

Table 7. Method Comparison Across LLMs (Social Media Post Prompt)

LLM	Method	Individual Metrics				$VPTT_{score}$-c(Uniform Weights)			$VPTT_{score}$(Novelty Adjusted)		
		PA	GS	CP	NV	Score	Win%	d *(vs base)*	Score	Win%	d *(vs base)*
4o-mini	Baseline	0.174	0.322	0.602	–	0.366	0.0%	–	0.312	0.0%	–
	Persona Only	0.428	0.437	0.659	–	0.508	0.9%	5.85	0.414	0.0%	5.34
	BRAG	0.434	**0.583**	**0.707**	0.828	0.574	35.3%	5.45	0.639	29.1%	11.65
	VPRAG (Ours)	**0.451**	0.566	0.685	**0.899**	0.567	27.2%	**6.17**	**0.645**	**40.5%**	**13.30**
	Comb. VPRAG+BRAG (Ours)	0.448	0.581	0.703	0.837	**0.578**	**36.6%**	6.09	0.643	30.4%	12.64
Qwen	Baseline	0.174	0.322	0.602	–	0.366	0.0%	–	0.312	0.0%	–
	Persona Only	0.384	0.422	0.656	–	0.488	0.0%	4.66	0.400	0.0%	4.33
	BRAG	**0.516**	**0.697**	**0.707**	0.323	**0.640**	**81.6%**	**8.50**	0.589	14.7%	7.23
	VPRAG (Ours)	0.448	0.583	0.685	**0.854**	0.572	6.1%	6.03	**0.641**	**59.3%**	**12.52**
	Comb. VPRAG+BRAG (Ours)	0.456	0.600	0.702	0.663	0.586	12.3%	5.68	0.614	26.0%	8.54
Gemini	Baseline	0.174	0.322	0.602	–	0.366	0.0%	–	0.312	0.0%	–
	Persona Only	0.371	0.424	0.647	–	0.481	0.0%	4.43	0.396	0.0%	4.11
	BRAG	**0.497**	**0.638**	0.694	0.724	**0.610**	**60.7%**	**6.96**	0.644	39.0%	11.78
	VPRAG (Ours)	0.396	0.544	0.674	**0.894**	0.538	3.7%	4.91	0.623	13.9%	11.66
	Comb. VPRAG+BRAG (Ours)	0.477	0.610	**0.699**	0.808	0.595	35.5%	6.25	**0.650**	**47.1%**	**12.10**

Table 8. Method Comparison Across LLMs (Empty Living Room Prompt)

LLM	Method	Individual Metrics				$VPTT_{score}$-c (Uniform Weights)			$VPTT_{score}$ (Novelty Adjusted)		
		PA	GS	CP	NV	Score	Win%	d *(vs base)*	Score	Win%	d *(vs base)*
4o-mini	Baseline	0.115	0.350	0.571	–	0.346	0.0%	–	0.299	0.0%	–
	Persona Only	0.356	0.418	0.635	–	0.470	4.0%	5.15	0.387	0.0%	4.56
	BRAG	0.264	0.533	0.617	**0.928**	0.472	5.1%	3.80	0.584	8.9%	11.35
	VPRAG (Ours)	**0.379**	**0.560**	**0.654**	0.908	**0.531**	**81.4%**	**5.68**	**0.622**	**77.7%**	**13.07**
	Comb. VPRAG+BRAG (Ours)	0.306	0.540	0.630	0.924	0.492	9.4%	4.51	0.597	13.5%	12.24
Qwen	Baseline	0.115	0.350	0.571	–	0.346	0.0%	–	0.299	0.0%	–
	Persona Only	0.357	0.396	0.630	–	0.461	0.3%	5.14	0.379	0.0%	4.41
	BRAG	0.339	0.547	**0.658**	0.819	0.514	18.9%	4.56	0.593	13.8%	10.67
	VPRAG (Ours)	**0.416**	**0.583**	0.657	**0.873**	**0.552**	**66.7%**	**6.45**	**0.630**	**76.6%**	**13.54**
	Comb. VPRAG+BRAG (Ours)	0.348	0.540	0.657	0.828	0.515	14.1%	4.31	0.594	9.7%	10.68
Gemini	Baseline	0.115	0.350	0.571	–	0.346	0.0%	–	0.299	0.0%	–
	Persona Only	0.333	0.400	0.630	–	0.454	3.6%	4.29	0.375	0.0%	3.80
	BRAG	0.292	0.529	0.619	0.915	0.480	15.8%	3.92	0.586	16.2%	11.07
	VPRAG (Ours)	0.322	0.526	0.639	**0.937**	0.496	30.1%	4.66	0.601	37.8%	**12.48**
	Comb. VPRAG+BRAG (Ours)	**0.346**	**0.537**	**0.643**	0.913	**0.508**	**50.4%**	**4.93**	**0.606**	**45.9%**	12.42

Table 9. Method Comparison Across LLMs (Garden Editing Prompt)

LLM	Method	Individual Metrics				$\mathrm{VPTT}_{\mathrm{score}}$-c (Uniform Weights)			$\mathrm{VPTT}_{\mathrm{score}}$ (Novelty Adjusted)		
		PA	**GS**	**CP**	**NV**	**Score**	**Win%**	**d** *(vs base)*	**Score**	**Win%**	**d** *(vs base)*
GPT-4o-mini	Baseline	0.131	0.364	0.588	–	0.361	0.0%	–	0.312	0.0%	–
	Persona Only	0.358	0.407	0.622	–	0.462	0.2%	3.59	0.380	0.0%	2.98
	BRAG	0.311	**0.594**	0.650	0.867	0.518	15.4%	4.33	0.609	17.0%	10.57
	VPRAG (Ours)	**0.403**	0.582	0.650	**0.901**	**0.545**	**46.2%**	**5.12**	**0.630**	**52.6%**	**11.47**
	Comb. VPRAG+BRAG (Ours)	0.378	0.588	**0.663**	0.860	0.543	38.2%	4.83	0.623	30.5%	10.92
Qwen	Baseline	0.131	0.364	0.588	–	0.361	0.0%	–	0.312	0.0%	–
	Persona Only	0.349	0.407	0.618	–	0.458	0.2%	2.93	0.377	0.0%	2.52
	BRAG	0.377	**0.625**	0.668	0.549	0.557	27.4%	4.90	0.573	12.0%	6.85
	VPRAG (Ours)	0.403	0.583	0.653	**0.858**	0.546	20.9%	**5.08**	**0.623**	**68.6%**	**11.05**
	Comb. VPRAG+BRAG (Ours)	**0.419**	0.613	**0.678**	0.529	**0.570**	**51.6%**	4.50	0.577	19.3%	6.39
Gemini	Baseline	0.131	0.364	0.588	–	0.361	0.0%	–	0.312	0.0%	–
	Persona Only	0.313	0.403	0.615	–	0.444	0.6%	2.97	0.368	0.0%	2.49
	BRAG	0.251	0.563	0.639	0.850	0.484	14.8%	2.88	0.581	14.8%	8.52
	VPRAG (Ours)	**0.340**	0.544	0.651	**0.913**	0.511	31.0%	3.94	**0.609**	**46.2%**	**10.24**
	Comb. VPRAG+BRAG (Ours)	0.336	**0.582**	**0.667**	0.822	**0.528**	**53.6%**	**4.04**	0.606	39.0%	9.66