ConCon-Chi: Concept-Context Chimera Benchmark for Personalized Vision-Language Tasks

Supplementary Material

In this supplementary material we first include additional information about the proposed ConCon-Chi dataset (Sec. 7), then provide additional details about the personalized TIR (Sec. 8) and TIG (Sec. 9) experiments.

7. Concept-Context Chimera Benchmark

We provide additional information about the concepts (Sec. 7.1) and the contexts (Sec. 7.2) in the dataset; then we provide more details about the acquisition and annotation procedure (Sec. 7.3) and the proposed benchmark splits (Sec. 7.4). Finally we provide examples of queries/prompts and associated GT images in the dataset (Sec. 7.5).

7.1. Concepts: Pictures and Descriptions

In Fig. 10 we show all the 20 concepts in ConCon-Chi together with their name and category. Chimeric concepts are indicated with a Bold concept name. Other than the 6 chimeras, 4 other objects are considered "animate" (MYDOLL, MYCLOWN, MYBIRD, MYRAT). This type of concept can combine with contexts that require agency (e.g., "sitting" or "playing") and with inanimate concepts. In Tab. 5 we report the three types of concept descriptions used for the TIR and TIG baselines.

7.2. Contexts: Kinds and Environments

Contexts are grouped into nine kinds:

- **re-contextualization**: the concept appears in a different environment;
- **view**: the concept is photographed by a different view (e.g., back, profile, close-up on details);
- **action**: the concept is carrying out an action (e.g., sitting, lying down);
- **accessorization**: the concept is wearing an accessory (e.g., sunglasses, eye-mask)
- **property modification**: the concept is missing some part (e.g., legs, wings, ears) or some part of the concept is applied to another one;
- **interaction**: the concept is interacting with a person, an object or another concept learned in the same way;
- **rendition**: black and white photography, ink or pencil sketch, baby drawing, blurred painting, dark glow-edge effect.

Contexts can be, and in most of the cases are, tagged with multiple kinds. This is visually depicted in Fig. 11a. From the figure it can be observed that the most frequent kinds are "object interaction" and "recontextualization" which also appear in combination with most of the other kinds. It is also interesting to notice how the 18 contexts tagged as "concept interaction" result in a high number of queries. This is because "concept interaction" contexts allow for numerous concept combinations.

Contexts are also tagged with the environment in which the corresponding ground-truth images were captured. Half of the contexts (48) were captured in a neutral setting (a wooden surface). The other 38 contexts were captured in different environments, coming from two main location types: houses and office spaces. The environment distribution is shown in Fig. 11b.

7.3. Acquisition and Annotation: Procedure Details

Acquisition. We took the images using an iPhone 11 Pro with resolution 4032×3024 and variable focal length. The released pictures were converted to JPEG at a resolution of 1008×756. Part of the renditions was obtained by applying digital effects (ink/pencil sketch, grayscale, blur, dark glow edges) to pictures of 11 concepts manually selected in the image pool. The remaining part contains actual pen drawings of the concepts sketched on paper and later photographed.

Annotation. The annotation process was divided in two steps. First we used an annotation GUI developed by us to review the images and assign to each of them the correct concepts/context combination. This step produced, for each query, a list of ground-truth (GT) images. The distribution of the number of GTs per query is shown in Fig. 12 (Blue bars). At this point each of the images is assigned as GT of only one query. However, since some contexts are more general than others, some images needed to be marked as GT for multiple queries. This is necessary to avoid false negatives in the dataset (images that are correct realization of a certain query but are not marked as its GT). To this end, we automatically assigned additional GT images following context overlaps, from more specific to more general queries (e.g., GT images of the query "black and white photography of X resting on a bedside table" are added as GT images for the query "X resting on a bedside table"). Then, the lists of GT images were further extended by manually checking individual GT images of queries which we knew could overlap with others (e.g., we added several images as GT for "X standing on a wooden surface", where the concept was involved in an action or interaction, but in fact was also standing on the surface). In Fig. 12 we show the resulting number of GT images per query in Magenta.

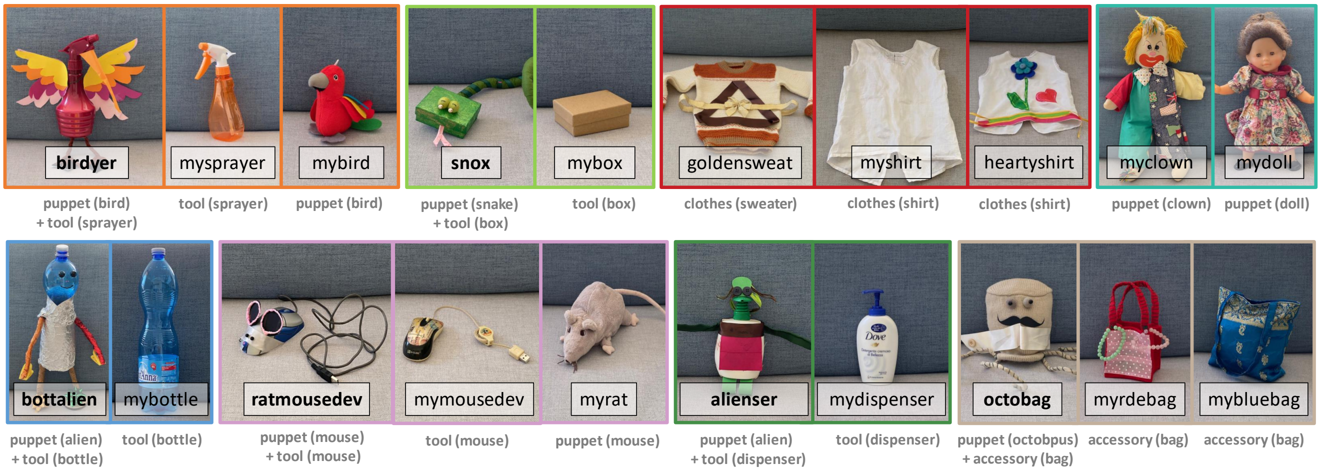


Figure 10. **Concepts in ConCon-Chi.** Images from one of the five training set environments. The concepts are grouped together with their hard negatives and the chimeric ones are indicated in (**Bold**).

CONCEPT	CATEGORIES	DESCRIPTIONS		
		COARSE (1-TOKEN)	RICH	DISCRIMINATIVE
BIRDYER	**puppet (bird) + tool (sprayer)**	puppet	red plastic sprayer bird puppet with colored wings	bird sprayer puppet
MYBIRD	puppet (bird)	plush	red stuffed bird with colored wings	stuffed bird
MYSPRAYER	tool (sprayer)	plastic container (bottle)	orange transparent plastic sprayer with white cap	sprayer
RATMOUSDEV	**puppet (mouse) + tool (mouse)**	puppet	computer mouse rat puppet with black round ears and cable	rat computer mouse puppet
MYRAT	puppet (mouse)	plush	grey stuffed rat with long pink tail	stuffed rat
MYMOUSEDEV	tool (mouse)	computer mouse (device)	black and beige computer mouse with wheel and cable	computer mouse
BOTTALIEN	**puppet (alien) + tool (bottle)**	puppet	blue alien plastic bottle puppet with aluminum foil dress	alien plastic bottle puppet
MYBOTTLE	tool (bottle)	plastic container (bottle)	blue plastic bottle with label	plastic bottle
ALIENSER	**puppet (alien) + tool (dispenser)**	puppet	green alien soap dispenser puppet with pink dress	alien soap dispenser puppet
MYDISPENSER	tool (dispenser)	plastic container (bottle)	white plastic soap dispenser with blue cap	soap dispenser
SNOX	**puppet (snake) + tool (box)**	puppet	green box headed snake puppet with round eyes and pink tongue	snake box headed puppet
MYBOX	tool (box)	box	cardboard box with lid	box
OCTOBAG	**puppet (octopus) + accessory (bag)**	puppet	beige woolen octopus bag with moustache and scarf	octopus bag puppet
MYREDBAG	accessory (bag)	bag	red square handbag with pois and bracelets	red bag
MYBLUEBAG	accessory (bag)	bag	blue square silk handbag with gold decorations	blue bag
MYDOLL	puppet (doll)	doll	doll with floral dress and brown hair	doll
MYCLOWN	puppet (clown)	puppet	rag clown with yellow hair and jumpsuit	clown
GOLDENSWEAT	clothes (sweater)	sweater	cream wool sweater with brown triangle and gold bow	sweater
HEARTYSHIRT	clothes (shirt)	shirt	white sleeveless baby shirt with red heart and blue flower	baby shirt with flower
MYSHIRT	clothes (shirt)	shirt	plain white baby shirt	white baby shirt

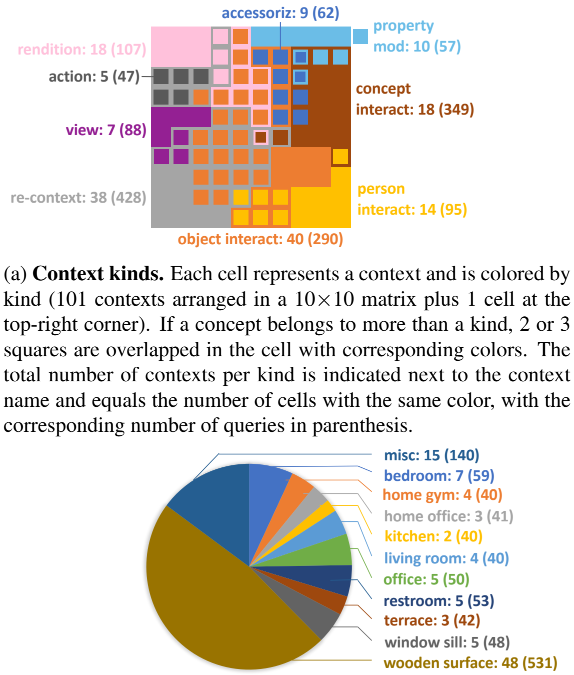
Table 5. **Concept descriptions.** Chimeric concepts (**Bold**) are grouped with their hard negative concepts (Dotted lines).

7.4. Benchmark Splits

We report below the splits that we introduce for the personalized TIR and TIG benchmarks (notice that in the main paper we use only the *train* and *test* splits):

- **train**: A set of 5 images per concept each with a different background. For the experiments with $k = 1$ in Sec. 4 we further split the training set into 5 different splits each with one image per concept.
- **val**: A validation set containing 3 concepts (TOOMOUSE, RATMOUSDEV and MYRAT), composed of the (42) queries that contain any of them (but none of the other concepts), set in the OFFICE or WOODEN SURFACE environments. The associated image pool is the union of the ground-truth images of these queries (165 images). Considering the very few-shot nature of the personalization tasks, we do not use the validation set. However, we define and release one since future methods might exploit it.
- **test**: A set that contains all the dataset minus the training images (1084 queries and 4008 images in the pool).
- **test-unseen**: A set from which we removed the queries that contain any of the 3 concepts used for validation (986 queries and 4008 images in the pool).

See Tab. 1 for the further statistics regarding the validation and the test splits.



(a) **Context kinds.** Each cell represents a context and is colored by kind (101 contexts arranged in a 10×10 matrix plus 1 cell at the top-right corner). If a concept belongs to more than a kind, 2 or 3 squares are overlapped in the cell with corresponding colors. The total number of contexts per kind is indicated next to the context name and equals the number of cells with the same color, with the corresponding number of queries in parenthesis.

(b) **Context environments**. The number of contexts per environment is indicated to the left with the corresponding number of queries in parenthesis (and they respectively sum to 101 and 1084).

Figure 11

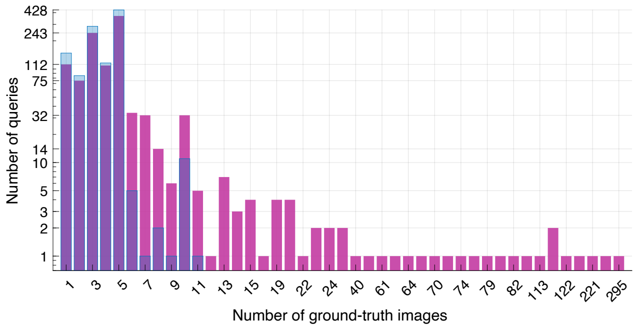


Figure 12. **Ground-truth images per query.** The number of ground-truth images that were acquired to represent a given query is reported in Blue (with transparency), the number of overall images that were annotated as ground-truth for a given query is in Magenta.

7.5. Dataset Examples

In Fig. 13 we show 6 examples of caption-image associations present in the dataset, for each of the 6 chimeric concepts (one concept per row). The caption is used as query in TIR and prompt in TIG tasks. In the same way, in Fig. 14 we report 2 examples of caption-image associations for each of the 14 common concepts (3 concepts per row and 2 in the last row). These two ensembles represent respectively examples of uncommon/novel and common situations, the former more focused on the learning of new-meanings and the latter more focused on the learning of instances of common categories, following related personalization benchmarks.

SPLIT	REFERENCE IMAGES	IMAGE POOL
shirt	2038 - 97	6346 - 164
dress	2017 - 156	3817 - 164
toptee	1961 - 94	5373 - 112

Table 6. Total number of URLs minus (-) the number of broken ones for each split of FashionIQ.

8. Personalized Text-to-Image Retrieval

In this section we provide definition of the TIR metrics adopted in the paper (Sec. 8.1); we report implementation details about the experiment reported in Fig. 5 (Sec. 8.2) and about the methods compared in the TIR benchmark (Sec. 8.3); we report additional metrics for the TIR benchmark (Sec. 8.4) and additional breakdown of performance (Sec. 8.5) and finally some failure cases (Sec. 8.6).

8.1. Metrics Definition

Given a set of queries Q and an image pool I we report below the definition for the adopted metrics:

mean Reciprocal Rank (mRR): the average of the reciprocal rank of the first retrieved ground-truth

$$mRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

where $rank_i$ refers to the rank of the first retrieved ground-truth for the i-th query.

mean Average Precision at k (mAP@k): the average of the AP (Average Precision) at k, among the queries

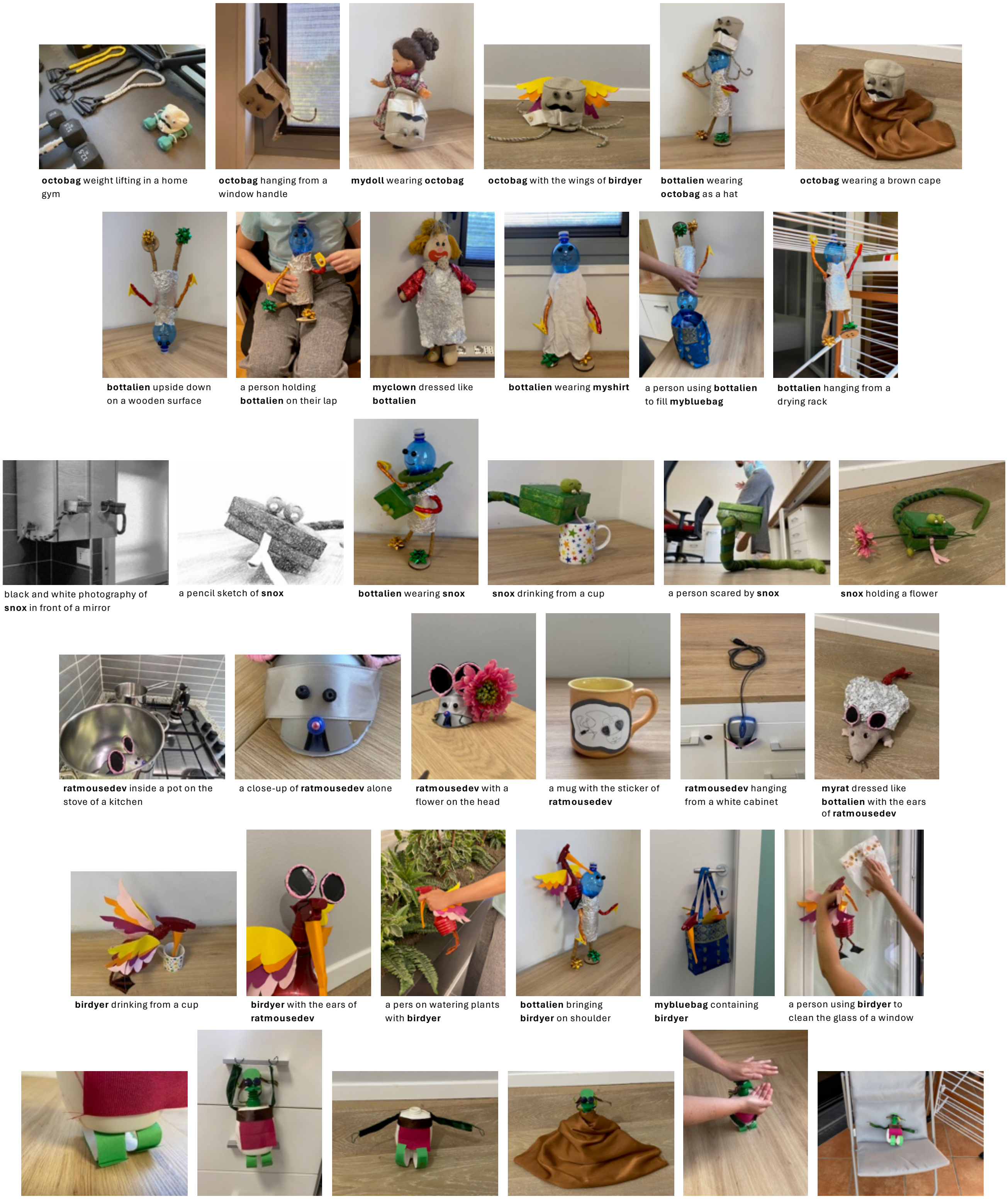
$$mAP@k = \frac{1}{|Q|}\sum_{i=1}^{|Q|} AP_i@k$$

with

$$AP_i@k = \frac{1}{min(k, |GTs_i|)}\sum_{m=1}^{k} P_i@m \times rel_i(m)$$

where $P_i@m$ (precision at m) is the fraction of ground-truth images (for the i-th query) among the top m retrieved ones, $rel_i(m)$ is an indicator function which is 1 if the m-th image in I is a ground-truth for the i-th query, and GTs_i is the set of ground-truth images for the i-th query.

mean Average Precision (mAP): it is the mAP@k with $k = |I|$.



octobag weight lifting in a home gym

octobag hanging from a window handle

mydoll wearing **octobag**

octobag with the wings of **birdyer**

bottalien wearing **octobag** as a hat

octobag wearing a brown cape

bottalien upside down on a wooden surface

a person holding **bottalien** on their lap

myclown dressed like **bottalien**

bottalien wearing **myshirt**

a person using **bottalien** to fill **mybluebag**

bottalien hanging from a drying rack

black and white photography of **snox** in front of a mirror

a pencil sketch of **snox**

bottalien wearing **snox**

snox drinking from a cup

a person scared by **snox**

snox holding a flower

ratmousedev inside a pot on the stove of a kitchen

a close-up of **ratmousedev** alone

ratmousedev with a flower on the head

a mug with the sticker of **ratmousedev**

ratmousedev hanging from a white cabinet

myrat dressed like **bottalien** with the ears of **ratmousedev**

birdyer drinking from a cup

birdyer with the ears of **ratmousedev**

a pers on watering plants with **birdyer**

bottalien bringing **birdyer** on shoulder

mybluebag containing **birdyer**

a person using **birdyer** to clean the glass of a window

the bottom of **alienser**

alienser hanging from a white cabinet

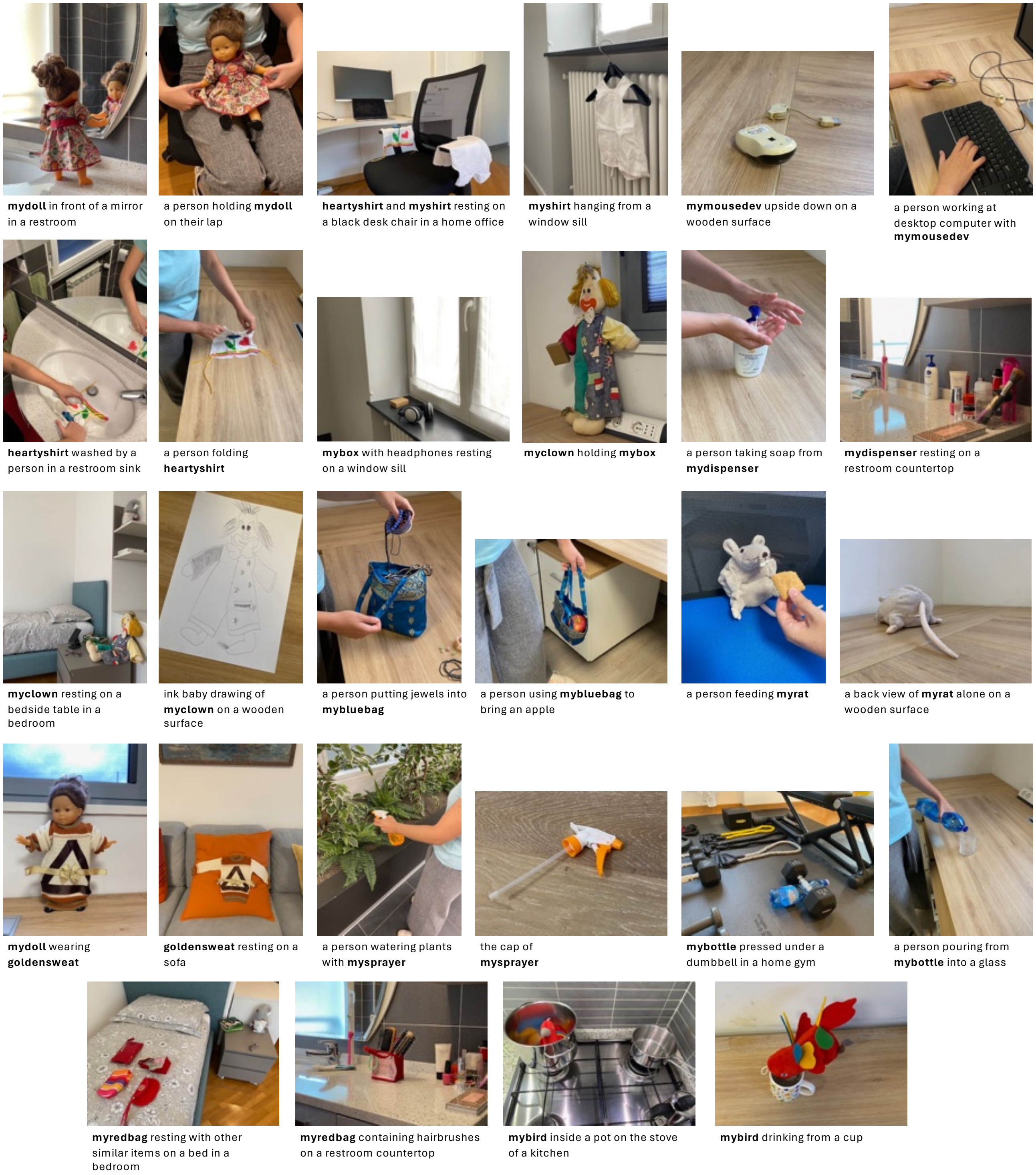
headless **alienser**

alienser wearing a brown cape

a person taking soap from **alienser**

alienser resting on a deckchair

Figure 13. **Examples of caption-image associations involving the 6 chimeric concepts.** One concept per row with six examples per concept. The caption is used as query in TIR and prompt in TIG tasks. The concepts are marked in Bold in the captions. Please note that for simplicity we report only one GT image per caption (and only one caption per image).



mydoll in front of a mirror in a restroom

a person holding **mydoll** on their lap

heartyshirt and **myshirt** resting on a black desk chair in a home office

myshirt hanging from a window sill

mymousedev upside down on a wooden surface

a person working at desktop computer with **mymousedev**

heartyshirt washed by a person in a restroom sink

a person folding **heartyshirt**

mybox with headphones resting on a window sill

myclown holding **mybox**

a person taking soap from **mydispenser**

mydispenser resting on a restroom countertop

myclown resting on a bedside table in a bedroom

ink baby drawing of **myclown** on a wooden surface

a person putting jewels into **mybluebag**

a person using **mybluebag** to bring an apple

a person feeding **myrat**

a back view of **myrat** alone on a wooden surface

mydoll wearing **goldensweat**

goldensweat resting on a sofa

a person watering plants with **mysprayer**

the cap of **mysprayer**

mybottle pressed under a dumbbell in a home gym

a person pouring from **mybottle** into a glass

myredbag resting with other similar items on a bed in a bedroom

myredbag containing hairbrushes on a restroom countertop

mybird inside a pot on the stove of a kitchen

mybird drinking from a cup

Figure 14. **Examples of caption-image associations involving the 14 common concepts.** Three concepts per row and two in the last row. The caption is used as query in TIR and prompt in TIG tasks. The concepts are marked in Bold in the captions. Please note that for simplicity we report only one GT image per caption (and only one caption per image).

	Method	mAP[%]	mAP@5[%]	mAP@10[%]	mRR[%]	R@1[%]	R@5[%]	R@10[%]
k=0	*Coarse*	16.83	11.74	13.13	24.21	14.48	33.49	43.63
	Discriminative	*30.16*	*23.91*	*26.47*	*43.16*	*31.92*	***55.63***	***66.14***
	Rich	*27.65*	*21.62*	*23.57*	*40.58*	*29.98*	*51.75*	*62.55*
k=1	PALAVRA	22.56 ± 1.29	17.58 ± 1.08	18.97 ± 1.18	34.39 ± 1.68	24.59 ± 1.94	44.30 ± 1.51	54.85 ± 1.49
	Pic2Word	25.23 ± 1.20	19.52 ± 1.31	21.30 ± 1.18	37.16 ± 1.76	26.35 ± 1.85	48.06 ± 1.91	58.80 ± 1.67
	SEARLE	28.16 ± 0.55	23.02 ± 0.65	24.60 ± 0.57	41.07 ± 0.92	31.16 ± 0.94	51.72 ± 0.71	60.85 ± 1.06
k=5	PALAVRA	23.59	18.65	20.05	35.99	26.75	45.11	55.08
	Pic2Word	26.39	20.67	22.45	38.62	27.68	50.28	60.61
	SEARLE	**30.74**	**25.51**	**27.11**	**43.83**	**33.49**	55.54	63.84

Table 7. **Personalized TIR benchmark.** Additional metrics for the results reported in Tab. 2.

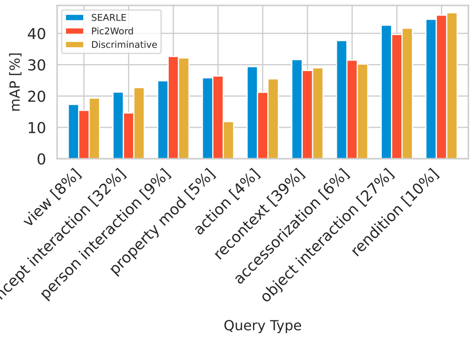


Figure 15. **Performance per query kind.** Breakdown of the mAP performance in Tab. 2 per query kind for the two best performing methods and the best text-based baseline.

Recall at k (R@k): fraction of queries for which the rank of the first retrieved image is smaller than k

$$R@k = \frac{1}{|Q|}\sum_{i=1}^{|Q|} \mathbb{1}_{\{rank_i<k\}}$$

where $\mathbb{1}$ is 1 if $rank_i < k$ and 0 otherwise.

8.2. Concept and Context Bias: Implementation Details

Below we provide some details about the datasets compared in Fig. 5.

For FashionIQ, CIRR, CIRCO we used the validation set as the test set annotations are not publicly available. For ConCon-Chi and PerVL DF2 we used the test set.

The PerVL DF2 dataset provides two types of captions: short and detailed. In the paper, the authors define the detailed captions as the ones that describe extensive context about the image and can facilitate retrieval, while the short ones describe less detail and therefore are more ambiguous. We used the detailed captions for our analysis.

The FashionIQ dataset provides images as lists of URLs. However, we found some of the links were broken, making it impossible to download the original dataset. In Tab. 6 we report, for the validation set, the original number of URLs and the number of images that we couldn't download.

Since FashionIQ provides two relative captions for each reference image, as in [2] (Appendix A), we concatenate them.

8.3. Methods: Implementation Details

For each method we provide a brief description of how it works followed by some details on the choices we took to ensure a fair comparison.

PALAVRA [5]. This method uses a Deep Set function [40] to map the 5 CLIP image embeddings of the concept to a token embedding (inverse mapping). The function is pre-trained on thousands of frequent concepts from COCO [19] and used to provide an initial token value for the concept at hand. This is then fine-tuned with a so-called cycle contrastive loss, such that the CLIP text embedding of a template sentence containing the token is pushed closer to the average CLIP image embedding of the 5 images (cycle term) and farther from the embedding of a same sentence with the concept type replacing the token (contrastive term).

We used the code released by the authors [2] and retrained the inverse mapping following author instructions. When doing so, we replaced the ViT-B/32 CLIP backbone with the larger ViT-L/14 for fair comparison with other methods. When fine-tuning the token embedding, we adopted hyper-parameters suggested by the authors and when using a single image example, we fed the Deep Set function with 5 copies of the single image embedding. As concept types in the fine-tuning, we used the *Coarse* descriptions.

Pic2Word [30] and SEARLE [2]. These are two ZS-CIR methods which, as PALAVRA, learn a inverse mapping function for CLIP. In this case this is a three-layer MLP that maps a CLIP image embedding (input) to a corresponding token embedding (output) and is learned in a self-supervised fashion.

Pic2Word trains the function on 3M images from CC3M[31] with a cycle contrastive loss that uses a given CLIP image embedding as cycle term and others as contrastive term. In SEARLE, first, a token embedding for each of 100K images in the unlabeled test split of ImageNet1K [28] is optimized similarly to PALAVRA's fine-tuning, with a regularization term such that the embedding

[2] https://github.com/NVlabs/PALAVRA

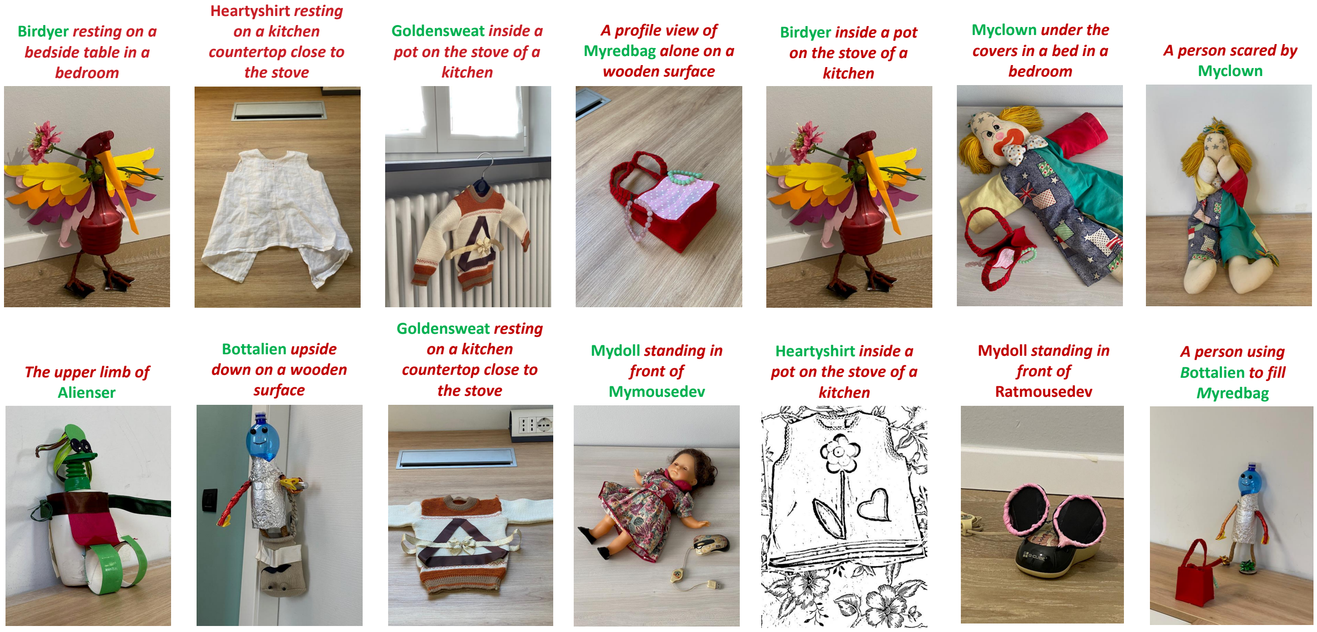

Birdyer resting on a bedside table in a bedroom
Heartyshirt resting on a kitchen countertop close to the stove
Goldensweat inside a pot on the stove of a kitchen
A profile view of Myredbag alone on a wooden surface
Birdyer inside a pot on the stove of a kitchen
Myclown under the covers in a bed in a bedroom
A person scared by Myclown
The upper limb of Alienser
Bottalien upside down on a wooden surface
Goldensweat resting on a kitchen countertop close to the stove
Mydoll standing in front of Mymousedev
Heartyshirt inside a pot on the stove of a kitchen
Mydoll standing in front of Ratmousedev
A person using Bottalien to fill Myredbag

Figure 16. **Retrieval Failure Cases.** First retrieved image for each of the 14 worst performing queries in terms of mAP for SEARLE (Left to Right, Top to Bottom). The part of the query successfully retrieved is highlighted in Green, otherwise in Red.

is also kept close to the textual embeddings of concepts represented in the image. Then, since using this optimization procedure at evaluation time would be computationally expensive, SEARLE learns a mapping network that produces similar embeddings to the ones learnt through back-propagation. This mapping network is trained with a contrastive loss over the 100K optimized tokens, regularized in the same way. At inference time SEARLE architecture is basically identical to Pic2Word.

We use both methods by downloading pre-trained weights of the mapping functions based on CLIP ViT-L14, officially provided by the authors (Pic2Word[3], SEARLE[4]). Differently from ZS-CIR, in our setting 5 concept images are available, thus we average the generated token embeddings to create the concept embedding.

8.4. Additional Benchmark Metrics

To encourage future comparisons on our benchmark, in Tab. 7 we report some additional retrieval metrics for the experiment reported in Tab. 2. In particular we show R@k and mAP@k with k=5, 10.

8.5. Additional Analysis of Results

In Fig. 15 we report the performance per query kind. SEARLE is comparable or better than the baseline across all kinds but "person interaction", where it is outperformed also by Pic2Word, indicating that there may be a limitation specific to its training set. Conversely, the baseline drops heavily on "property modification" and "accessorization".

8.6. TIR Failure Cases

We report in Fig. 16 some failure cases for the top-performing retrieval method, SEARLE. To this end, we sort the queries by decreasing mAP and report the first retrieved image for the last 14 ones. In accordance to the results reported in Tab. 3, on these worst performing queries, SEARLE is almost always able to retrieve the correct concept, but fails to compose it with the correct context.

9. Personalized Text-to-Image Generation

In this section we provide implementation information about the Density and Coverage metrics (Sec. 9.1), and the methods compared in the TIG benchmark (Sec. 9.2); we then report numbers for the FID metrics (Sec. 9.3) and some failure cases (Sec. 9.4).

9.1. Metrics: Implementation Details

To compute Density and Coverage we used the code provided by the authors [23] without substantial modifications[5]. We computed the two metrics on the CLIP embeddings of the real and generated images.

[3] https://github.com/google-research/composed_image_retrieval

[4] https://github.com/miccunifi/SEARLE

[5] https://github.com/clovaai/generative-evaluation-prdc

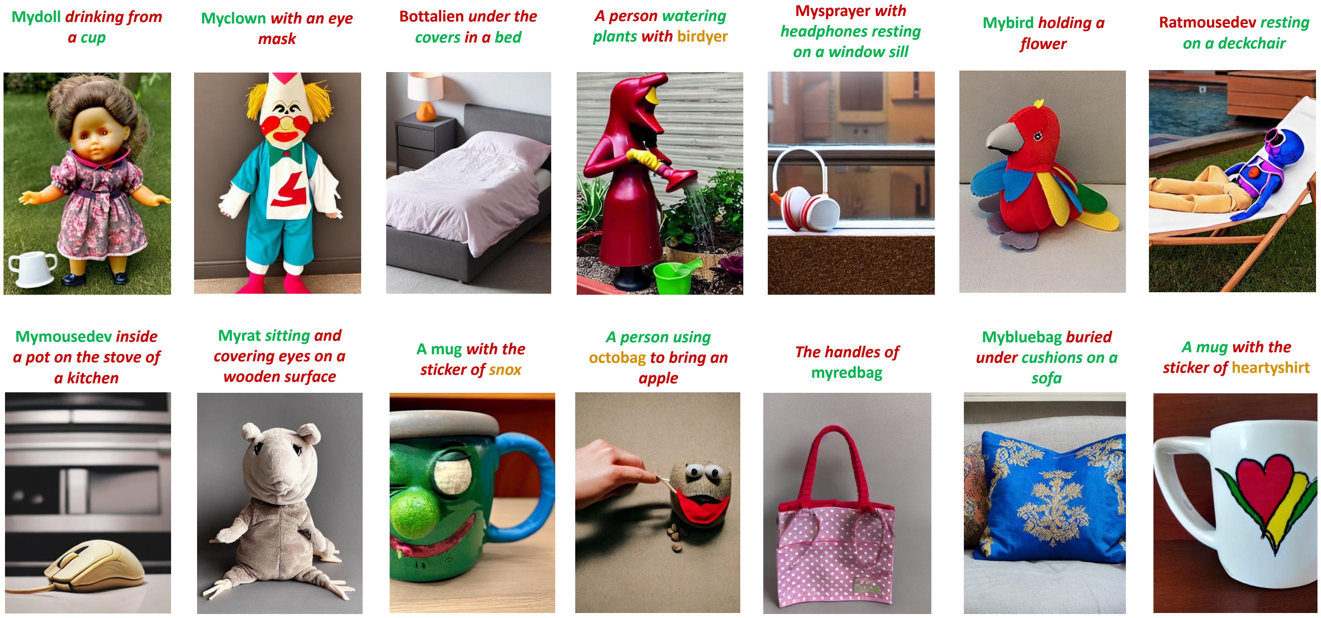

Mydoll drinking from a cup
Myclown with an eye mask
Bottalien under the covers in a bed
A person watering plants with birdyer
Mysprayer with headphones resting on a window sill
Mybird holding a flower
Ratmousedev resting on a deckchair
Mymousedev inside a pot on the stove of a kitchen
Myrat sitting and covering eyes on a wooden surface
A mug with the sticker of snox
A person using octobag to bring an apple
The handles of myredbag
Mybluebag buried under cushions on a sofa
A mug with the sticker of heartyshirt

Figure 17. **Generation Failure Cases.** Some failure cases for DreamBooth. The part of the query successfully generated is highlighted in Green; the part of the query that seems to be contained in the generated image, but is not completely correct, is in Orange; the part that is not represented in the generated images is in Red. While clearly this type of decision can be subjective, we aimed to provide our judgment as additional information.

9.2. Methods: Implementation Details

DreamBooth (DB) [27]. This method fine-tunes a pre-trained TIG model on the example images in order to bind a textual identifier (a chosen rare token followed by the concept class) to the concept appearance. A class-specific prior preservation loss acts as regularizer by forcing the model also to keep generating images of instances of the concept class when the input sentence does not contain the rare token but just the class.

We adopted the implementation provided[6] in the Diffusers library [36], release 0.18.0, with the Stable Diffusion Model (SDM) checkpoint v1.4[7]. We then fine-tuned the text encoder alongside the UNet, using the *Coarse* descriptions at initialization, with a learning rate of 1e-6.

Textual-Inversion (TI) [6]. This method applies an approach similar to PALAVRA and optimizes a token embedding in the vocabulary space of a frozen TIG model (by feeding it with template sentences containing the token and asking it to generate images like the 5 examples). The authors show that a single token suffices to generate a concept faithfully and highlight the advantage of the approach to retain the knowledge of the pre-trained model.

We adopted the officially released code[8], using the 1-token *Coarse* descriptions for the token initialization. We either adopted the Latent Diffusion Model (LDM) [26][9] by following authors' instructions, or, by following more recent work [34], the same SDM used for DB, with the parameters that the authors report for LDM (learning rate of 5e-3) and by applying the same procedure as for DB to select training steps.

Number of steps in the concept optimization. Since we observed performance variations depending on the number of steps, we devised an automatic procedure to choose the optimal number and applied it to both DB and TI. We created a minimal validation set composed of 4 prompts ('a photo of * on a beach', 'a photo of * on the moon', 'a photo of * with a cat', 'a photo of a yellow *'). We avoided using the larger proposed *validation* set since, while we defined it for completeness and possible future uses, we preferred preserving the 5-shot nature of the considered personalization setting.

The score to maximize was chosen empirically among different versions, by comparing the best checkpoints selected automatically, with checkpoints selected manually (by qualitatively comparing the generated images at each validation iteration), for each of the 20 concepts. The best score S resulted being S=min-max(FID_{ctx}), subject to: min-

[6] https://github.com/huggingface/diffusers/tree/main/examples/dreambooth

[7] https://huggingface.co/CompVis/stable-diffusion-v-1-4-original

[8] https://github.com/rinongal/textual_inversion

[9] https://github.com/CompVis/latent-diffusion

	Method	FID_{cpt}	FID_{ctx}
upper bound	concept-only	*92.97*	19.77
	context-only	57.87	*27.92*
	GT images	75.83	25.37
k=0	LDM	70.37	22.35
	SDM	73.35	23.06
k=5	TI (LDM)	70.19	**25.16**
	TI (SDM)	77.25	20.92
	DB (SDM)	**80.42**	23.99

Table 8. **Personalized TIG benchmark**: Fidelity-concept/context metrics on ConCon-Chi.

max(FID_{ctx}) < min-max(FID_{cpt}). Namely, we first scale each metric such that it is in $[0, 1]$ along the optimization (min-max). Then, we maximise the capability to generate an image of the context (we maximize FID_{ctx}), subject to the fact that the generated images are also close to the concept image examples (subject to FID_{ctx}) < FID_{cpt}). This provided slightly better results than the harmonic mean of the FID_{cpt} and a normalized FID_{ctx}, adopted in [34].

While the ground-truth step annotation was possible for DB, we could not easily identify a learning trend in TI. Thus, we validated the procedure on the former method and then applied it also to the latter.

9.3. Benchmark Results: FID Numbers

In Tab. 8 we report the numbers relative to the scatter plot in Fig. 8.

9.4. TIG Failure Cases

In Fig. 17 we report failure cases for the TIG benchmark for the DB method. We first observe that sometimes the method represents the elements mentioned in the prompt, but not in the correct relationship (Top Row picture 1, Bottom Row picture 4). In some cases, the elements in the prompt are "semantically merged" (Top Row picture 4, Bottom Row picture 6). In other cases, some element of the prompt is totally missing (Top Row pictures 2, 3, 6; Bottom Row picture 2) or represented inaccurately (Top Row picture 7, Bottom Row pictures 1, 3, 5, 7).