# ConceptLab: Creative Generation using Diffusion Prior Constraints

Elad Richardson Kfir Goldberg Yuval Alaluf Daniel Cohen-Or
Tel Aviv University

https://kfirgoldberg.github.io/ConceptLab/

Figure 1. New "pets" generated using ConceptLab. Each pair depicts a learned concept that was optimized to be unique and distinct from existing members of the pet category. Our method can generate a variety of novel concepts from a single broad category.

## Abstract

*Recent text-to-image generative models have enabled us to transform our words into vibrant, captivating imagery. The surge of personalization techniques that has followed has also allowed us to imagine unique concepts in new scenes. However, an intriguing question remains: How can we generate a new, imaginary concept that has never been seen before? In this paper, we present the task of creative text-to-image generation, where we seek to generate new members of a broad category (e.g., generating a pet that differs from all existing pets). We leverage the under-studied Diffusion Prior models and show that the creative generation problem can be formulated as an optimization process over the output space of the diffusion prior, resulting in a set of "prior constraints". To keep our generated concept from converging into existing members, we incorporate a question-answering model that adaptively adds new constraints to the optimization problem, encouraging the model to discover increasingly more unique creations. Finally, we show that our prior constraints can also serve as a strong mixing mechanism allowing us to create hybrids between generated concepts, introducing even more flexibility into the creative process.*

## 1. Introduction

The quest for creative generation in computer graphics has sparked the study of computational creativity [8, 19, 41, 42, 52], which involves algorithms that simulate creative behaviors or try to enhance and augment the human creative process. Thanks to the rapid advancements in powerful text-to-image generative models, we now have an unprecedented ability to transform language into incredible, diverse images [4, 11, 27, 33, 35, 37, 39], opening up new possibilities for generating creative content. Building on these models, recent personalization techniques [14, 15, 22, 36, 50] have also enabled us to create personalized concepts and incorporate them into the generative process. Yet, an interesting question remains: can we use these powerful models to generate a novel creative concept that was not explicitly described to the model?

In this paper, we tackle the task of *creative text-to-image generation* using diffusion models. Specifically, we seek to generate novel and creative members of a given broad category. Consider, for example, the category of all "pets". Here, we would like to find a new concept that visually resembles a pet, but differs from any existing pet. For example, in Figure 1, we show generated concepts that semantically resemble a pet, but do not belong to a specific species. All these results were generated by only specifying the target category without further describing the desired output, resulting in a variety of possible outcomes.

Inspired by token-based personalization [7, 14], we represent our new concept as a token in the text encoder of a pretrained generative model. However, to generate a new concept, we cannot simply apply a standard inversion scheme as we naturally do not have any images depicting the target subject. Instead, we turn to the CLIP vision-language model [32] to help guide our optimization process. In essence, we divide our constraints into a set of


Figure 2. In text-guided generation (top left), an image is created given a free-form text prompt. With personalization methods (bottom left), we can learn new tokens representing a specific concept or subject. Our creative generation method (right) learns tokens that represent novel concepts belonging to a given category (e.g., "a pet" or "a fruit"). The learned concepts are optimized to belong to the broad category while differing from existing members of that cateogry.

positive and negative constraints. The positive constraint is introduced to encourage the generation of images that still match the broad category. Conversely, the negative constraints represent existing members of the category we wish to shift away from. Considering our previous pet example, the positive constraint is defined by the word "pet" while the negative constraints may consist of words such as "cat" and "dog", indicating that we wish to generate a pet that is not a cat nor a dog. Applying these constraints together should ideally encourage the learned concept to reside inside the category, but differ from the specified members.

While conceptually simple, it is not clear how to apply our CLIP-based optimization in practice in the context of diffusion models. First, applying a CLIP loss during the diffusion denoising process requires an approximation of the output image, which was shown to be unstable without applying dedicated augmentations [3], or a dedicated noise-aware CLIP model [27]. Second, we do not have a set of reference images that can be directly denoised during the optimization process, further complicating the process. A key understanding in our approach is that our constraints can be better represented when used with a Diffusion Prior model [33]. Specifically, we show that the output space of the Diffusion Prior serves as a more suitable target space for our optimization task. As such, we optimize our learned token by applying our CLIP constraints over the outputs of the Diffusion Prior, resulting in a set of *"prior constraints"*.

While we now have a working optimization framework, another challenge remains. For our negative constraints, we should ideally specify all existing members of the given category (e.g., all types of pets). However, doing so is cumbersome and not always practical. Instead, we build upon recent question-answering models [23] to iteratively suggest additional category members. This is achieved by dividing the optimization problem into segments. After each segment, we generate an image using our current concept token and then query the question-answering model to textually describe what member of the given category is depicted in the image. This technique allows us to "project" the current concept into the space of existing category members, as each member already has a unique word describing it. The new word is then added to our set of negative constraints, allowing us to gradually shift away from a growing set of category members, resulting in more creative generations.

Finally, we show that our proposed *prior constraints* can also be used to mix up generated concepts and create new hybrids by using a set of positive constraints derived from the generated concepts. This allows us to extend and evolve the newly generated concepts. The flexibility of our prior constraints and iterative optimization scheme is demonstrated using both quantitative and qualitative evaluation, showing its effectiveness for creative generation.

## 2. Related Works

**Text-Guided Synthesis.** Recently, large-scale text-to-image diffusion models [10, 20, 28] have achieved an unprecedented ability to generate high-quality imagery guided by a free-form text prompt [4, 27, 33, 35, 37, 39]. Leveraging these powerful generative models, many have attempted to utilize such models for downstream editing tasks [9, 18, 21, 25, 29, 47]. Most text-guided generation techniques condition the diffusion model directly on embeddings extracting from a pretrained text encoder [3, 5, 6, 18, 31]. In this work, we utilize a Latent Diffusion Model [35] paired with a *Diffusion Prior* model [33, 39] and show its benefits in the context of creative generation.

**Diffusion Prior.** A *Diffusion Prior* model, introduced in Ramesh *et al*. [33], is tasked with mapping an input text embedding to its corresponding image embedding in CLIP's [32] latent space. A decoder is then trained to generate a corresponding image, conditioned on the CLIP image embedding. In Ramesh *et al*. [33] the authors demonstrate that applying the Diffusion Prior and conditioning over the resulting image embeddings attains improved diversity while enabling image variations, interpolations, and editing. Several works have adopted the use of a Diffusion Prior for text-guided video synthesis [13, 43] and 3D generation and texturing [26, 51]. The use of Diffusion Prior for text-guided synthesis is further analyzed in [1, 53].

**Personalization.** In the task of personalization [7, 14], we aim to inject new user-specific concepts into a pretrained generative model. In the context of text-guided synthesis, doing so should allow for the generation of novel images depicting the target subject or artistic style using an input text prompt. To teach the generative model new concepts, current personalization techniques either optimize a set of text embeddings [2, 14, 49], fine-tune the denoising network [22, 36, 46], or train an encoder to map a concept to its textual representation [15, 40, 50]. Deviating from existing personalization literature, we do not aim to teach the generative model a new subject or concept. Instead, we focus on the task of *Creative Generation* and generate new concepts which can be placed in novel scenes, see Figure 2.

**Creative Generation.** A long-standing question in computer graphics centers around whether computers can truly generate creative art [19]. Naturally, generating creative content can be tackled in many different ways. Xu *et al*. [52] propose a set-evolution method for creative 3D shape modeling which aims to offer the user creative shapes that fit his preferences while still offering diversity. Elgammal *et al*. [12] explore creative generation in the context of GANs [17] and learn new styles by maximizing the deviation from existing artistic styles using discriminators. Sbai *et al*. [38] introduce a novel loss encouraging deviation from existing styles found in the training set.

Some works also approach the creative generation task as a composition task, learning and fusing fine-level components into a complete creation. This has been demonstrated across various creative domains including sketching [16] and 3D Modeling [34]. Recently Vinker *et al*. [48] have shown that one can decompose personalized concepts into different visual aspects which can then be joined together in creative ways. We choose to approach creative generation by finding novel concepts that are optimized to match a given category while differing from existing concepts in that category. These novel concepts can then be mixed together resulting in a flexible generation process.

## 3. Preliminaries

We apply our creative generation scheme over the Kandinsky 2 model [39] which combines the Latent Diffusion Model from Rombach *et al*. [35] with a Diffusion Prior model employed in Ramesh *et al*. [33].

**Latent Diffusion Models.** In a Latent Diffusion Model (LDM), the diffusion process is performed within the latent space of an autoencoder. First, an encoder $\mathcal{E}$ is trained to map a given image $x \in \mathcal{X}$ into a latent code $z = \mathcal{E}(x)$ while a decoder $\mathcal{D}$ is simultaneously tasked with reconstructing the original input image such that $\mathcal{D}(\mathcal{E}(x)) \approx x$. Given the autoencoder, a denoising diffusion probabilistic model (DDPM) [20, 44] is trained to produce latent codes within this learned latent space. During the denoising process, the diffusion model can be conditioned on an additional input vector. The DDPM model is trained to minimize the objective given by:

$$\mathcal{L} = \mathbb{E}_{z,y,\varepsilon,t}\left[||\varepsilon - \varepsilon_\theta(z_t, t, c)||_2^2\right]. \tag{1}$$

The denoising network $\varepsilon_\theta$ is tasked with correctly removing the noise $\varepsilon$ added to the latent code $z_t$, given $z_t$, the current timestep $t$, and the conditioning vector $c$.

**Diffusion Prior.** Diffusion models are commonly trained with the conditioning vector $c$ directly derived from the CLIP [32] text encoding of a given text prompt, $y$. In Ramesh *et al*. [33], it was proposed to decompose the generative text-to-image problem into two steps. First, an image embedding is predicted from a given text prompt, using a Diffusion Prior model. Next, the image embedding is fed into a diffusion decoder trained to generate an image conditioned on the image embedding.

Training is typically done in two independent steps. The diffusion decoder is trained using the objective defined in Equation (1) with an image embedding as the conditioning $c$. The Diffusion Prior model, $P_\theta$, is then tasked with directly predicting the denoised image embedding $e$ from a noised embedding $e_t$:

$$\mathcal{L}_{prior} = \mathbb{E}_{e,y,t}\left[||e - P_\theta(e_t, t, y)||_2^2\right]. \tag{2}$$

Once the two models are trained, each on its objective, they can be put together to create a complete text-to-image pipeline. This two-stage approach was shown to improve image diversity, but more importantly from our context, it provides direct access to an intermediate CLIP image embedding and allows introducing constraints directly in that space. We show the output space of the Diffusion Prior to be more effective than applying a constraint on a standard diffusion model or directly on the CLIP text embeddings.

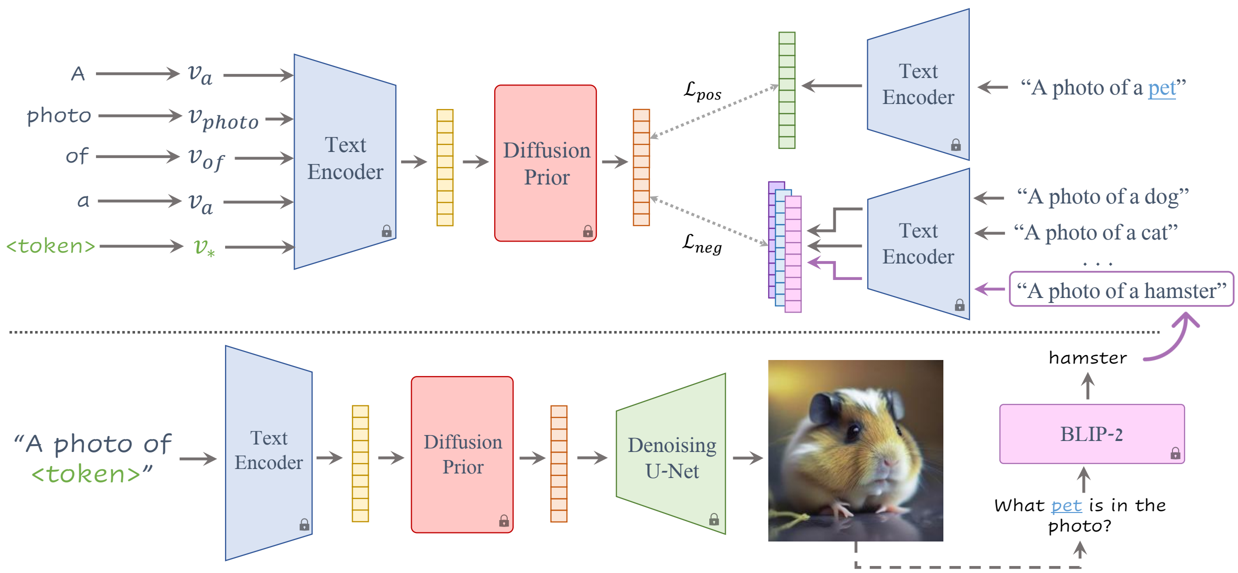



Figure 3. ConceptLab overview. We optimize a single embedding $v_*$ representing the novel concept we wish to generate (e.g., a new type of "pet"). To do so, we compute a set of losses encouraging the learned embedding to be similar to that of a given category while being different from a set of existing members (e.g., a "dog" or a "cat"). To gradually generate more unique creations, during training, we query a pretrained BLIP-2 VQA model [23] to expand the set of negative constraints based on the currently generated novel concept (e.g., we add the token "hamster" to shift our embedding from generating images resembling a "hamster").

## 4. Method

At its core, our method, dubbed ConceptLab, aims to tackle the creative generation task where we wish to learn a token representing a novel, never-before-seen concept belonging to a general category that differs from any existing concepts within that category. Similar to Textual Inversion [14], we do so by optimizing a new embedding vector $v_*$ representing our novel concept in the text conditioning space of a pretrained text-to-image model. As we seek to generate novel concepts that do not exist, optimizing this representation using a reconstruction-based objective is not possible. Instead, we impose a set of constraints over our learned representation where the embedding $v_*$ is optimized to be similar to a given broad category while differing from existing members of that category. As shall be discussed, we choose to apply this optimization scheme using a set of "prior constraints" (see Section 4.1). During training, we gradually expand the set of constraints (see Section 4.2), encouraging the creation of more unique concepts over time. Our complete training scheme is illustrated in Figure 3. At inference, compositions of our novel concept can be generated by adding the new token to an input prompt, see Figures 1, 6 and 7.

### 4.1. Diffusion Prior Constraints

**The Constraints.** We define our prior constraints as a set of losses applied over the output space of a Diffusion Prior model. These constraints are divided into a set of positive constraints $\mathcal{C}_{pos}$ and negative constraints $\mathcal{C}_{neg}$, where each constraint is defined using textual tokens. For example, to generate a new member of the "pet" category, our positive constraints could be simple defined as $\mathcal{C}_{pos} = \{\text{pet}\}$ with $\mathcal{C}_{neg} = \{\text{cat}, \text{dog}, \dots, \text{hamster}\}$ as the negative constraints.

**The Objective.** Given our two sets of constraints, we next define a measurement of similarity between $v_*$ and each constraint. We first incorporate $v_*$ and each constraining word $c$ into the same randomly sampled prompt template $y$ (e.g., "A photo of a $\{\}$", "An oil painting of $\{\}$"). Each such sentence can now be encoded into a CLIP text embedding, an operation we denote as $E_{\text{y}}(c)$, and defines a textual constraint. Given the textual constraints, a simple approach for defining the similarity to $v_*$ would be to compute the similarity between $E_y(v_*)$ and each textual constraint $E_y(c)$. We instead show that it is preferable to pass $E_y(v_*)$ through the Diffusion Prior model before computing the similarity measure. Intuitively, passing a text prompt through the Diffusion Prior results in a specific instance of the prompt. For example, applying the prior on "A photo of a dog" would result in a specific image of a specific dog breed. By passing $E_{\text{y}}(v_*)$ through the prior we encourage all realizations of $v_*$ to align with the textual constraints, resulting in more consistent generations. Conversely, we choose *not* to pass the positive and negative constraints through the Diffusion Prior. This is motivated by the intuition that we want to ideally keep the constraints themselves as broad as possible. That is, instead of applying the constraints over a specific image of a "cat" or "dog", we wish to shift away from the set of all possible "cats" and "dogs".

Thus our loss objective is defined as:

$$
\begin{aligned}
\mathcal{S}(\mathcal{C}, v_*) &= \mathbb{E}_{c\sim\mathcal{C}}\left[\langle E_{\text{y}}(c), P(E_{\text{y}}(v_*))\rangle\right] \\
\mathcal{L} &= \mathcal{S}(\mathcal{C}_{neg}, v_*) + \lambda(1 - \mathcal{S}(\mathcal{C}_{pos}, v_*))
\end{aligned}
\tag{3}
$$

In words, we encourage every sampled image embedding $P(E_{\text{y}}(v_*))$ generated from our learned embedding $v_*$ to distance itself from the text constraints defined by $C_{neg}$ while staying close to those of $C_{pos}$, with $\lambda$ allowing us to control the balance between the two.

**Regularizations.** When the set of constraints becomes large, the penalty for collapsing to a specific member of the constraint becomes increasingly more negligible. To avoid such a collapse, we use an additional objective that measures the *maximal* similarity to the negative constraints:

$$
\mathcal{S}_{max}(\mathcal{C}, v_*) = \max_{c\sim\mathcal{C}}\left(\langle E_{\text{y}}(c), P(E_{\text{y}}(v_*))\rangle\right). \tag{4}
$$

This similarity measure is incorporated into Equation (3), by averaging it with $\mathcal{S}(\mathcal{C}, v_*)$, ensuring that the constraint that is closest to $v_*$ receives a greater penalty.

Finally, we also restrict the similarity measure between two predetermined similarity values to avoid pathological solutions. For example, we empirically find that without such a restriction the model starts to generate text in the image that matches the target category, as a way to obtain high similarity without actually generating the desired concept.

**Using the Constraints.** In the context of creative generation, we set the positive constraints, $C_{pos}$, to contain a single broad category, e.g., {"pet"}, and set the negative constraints either manually, or automatically through our *adaptive negatives* scheme, introduced below. An additional application enabled through our constraints is that of *concept mixing*, where we wish to fuse existing concepts into a single creation. To this end, we can define a set of positive constraints with no negative constraints, see Figure 9.

### 4.2. Adaptive Negatives

Ideally, we would like to apply a large set of negative constraints in order to encourage the generation of truly unique creations. Yet, manually defining a large set of negative constraints is both cumbersome and may not accurately represent the most relevant members of the broad category. To this end, we propose an adaptive scheme to gradually expand the set of negative constraints during training. As illustrated at the bottom of Figure 3, at regular intervals during the optimization process (e.g., 250 steps) we generate an image using our current representation. We then query a pretrained BLIP-2 VQA model [23] and ask the model to identify which member of the broad category is currently present in the image. We then add the resulting instance to the set of negative constraints for the rest of the training. Note that we always incorporate the target category (e.g., "pet") as part of the question (e.g., "What kind of *pet* is in this photo") to encourage the VQA model to respond with members of that category. This adaptive scheme not only shifts the learned concepts away from existing members but also results in diverse creations across different seeds as each training seed may add a different set of negative classes or change the order in which they are added, see Figure 5.

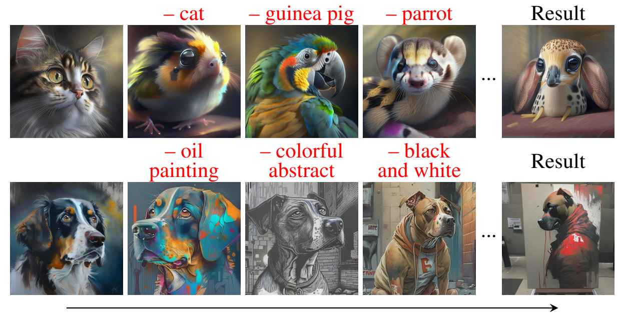



Figure 4. During training, we use BLIP-2 to infer the closest word to our current concept, which is then added to our constraints.

### 4.3. Evolutionary Generation

Building on our prior constraints, we show that one can also fuse generated concepts into a new concept. To perform *concept mixing* over a given set of concepts we first generate a set of images from each concept, creating a set of image constraints, $C_{im}$. Each image is then passed through a CLIP image encoder, $E_{im}(c)$ to create a set of image embeddings. We then apply a modified loss that pushes a learnable concept $v_{mix}$ closer to the given embeddings,

$$
\mathcal{L}_{\text{mix}} = 1 - \mathbb{E}_{c\sim\mathcal{C}_{im}}\left[\langle E_{im}(c), P(E_{\text{y}}(v_{mix}))\rangle\right]. \tag{5}
$$

This objective can be applied over either generated concepts or real images and can also be iteratively applied to create hierarchical generations of creative creatures. An optional weight term can additionally be applied to better control the effect of each concept on the generated output.


Figure 5. Creative generation results obtained across various categories using adaptive negatives with different training seeds.

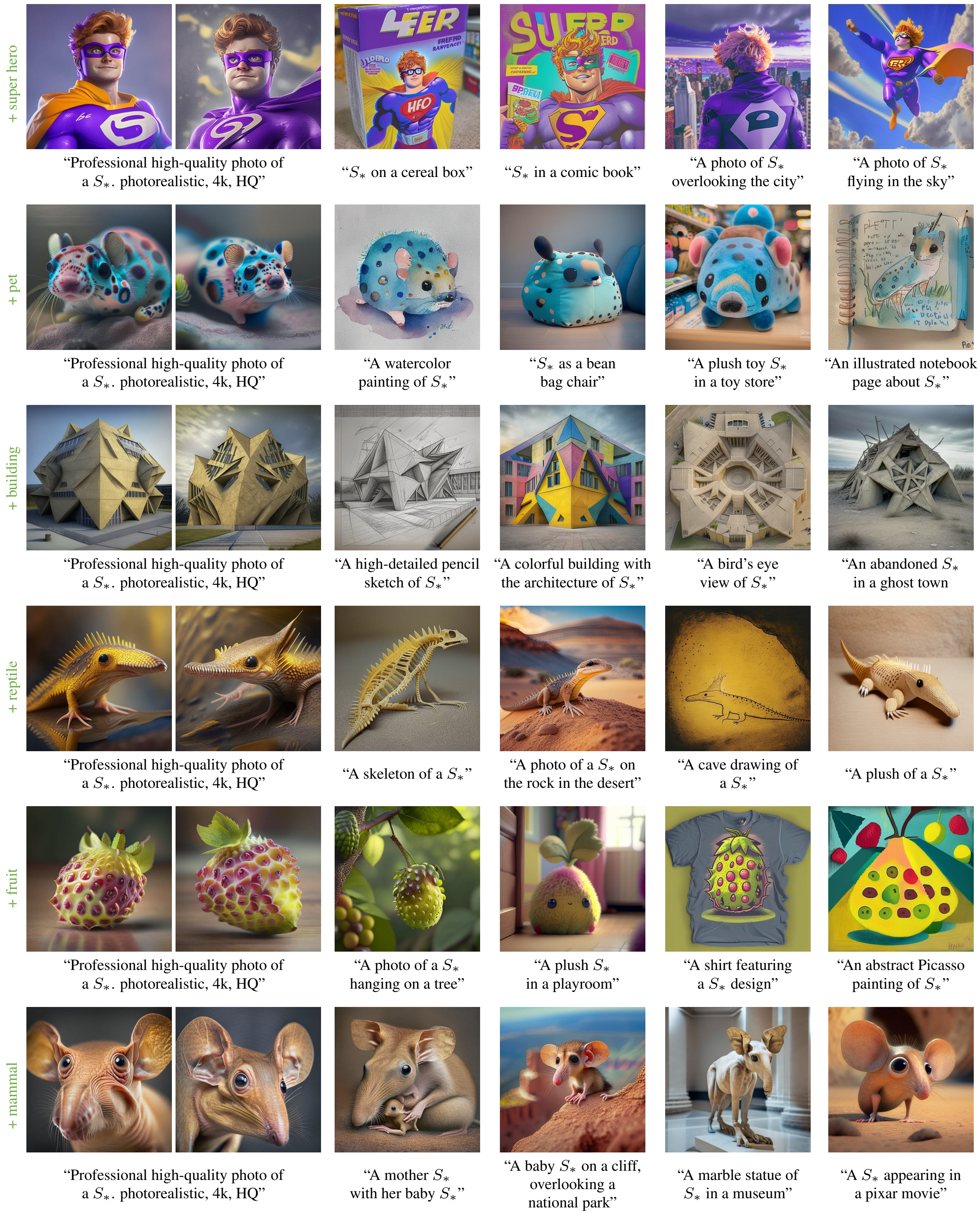
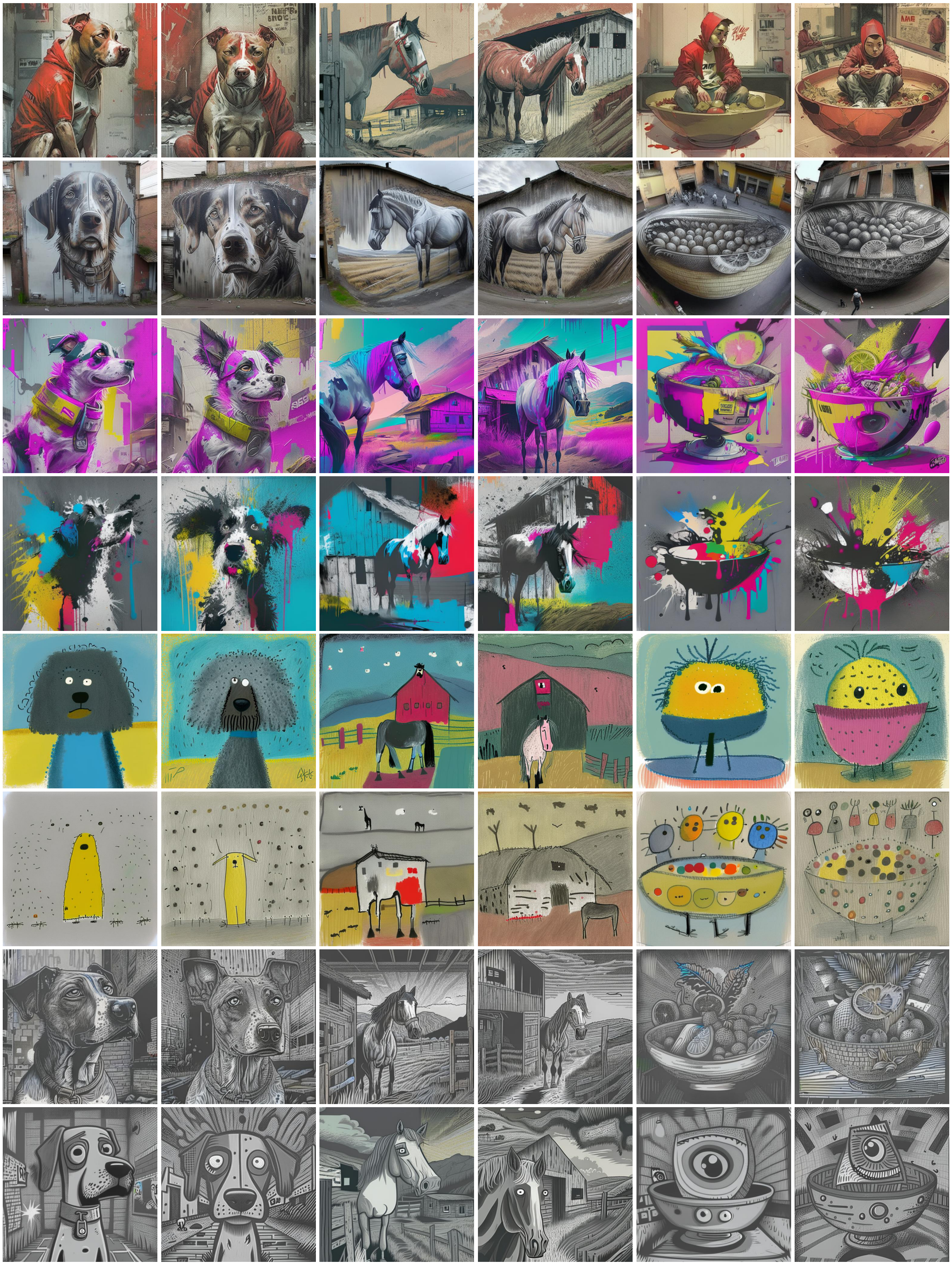



Figure 6. Sample text-guided creative generation results obtained with ConceptLab. The positive concept used for training is shown to the left. All results are obtained using our adaptive negative technique.



“... a dog ...” “... a horse and a barn in a valley ... ” “ ... a bowl of fruit ... ”

Figure 7. Styles suggested by ConceptLab using our artistic prompts with adaptive negatives. $S_*$ is always initialized as “painting”. All prompts start with “a painting of ” and end with “in the style of $S_*$”

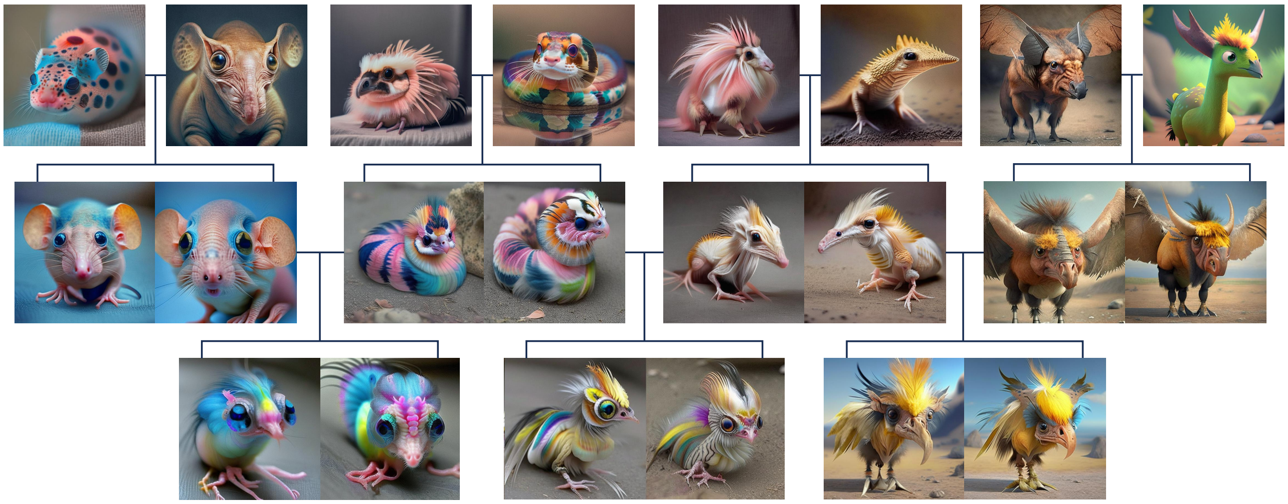


Figure 8. Evolutionary Creative Generation. ConceptLab can be used to mix up generated concepts to iteratively learn new unique creations. In the topmost row, we show concepts learned using our adaptive negatives technique (Section 4.2) followed by concepts obtained using our evolution generation process (Section 4.3).

## 5. Experiments

We now turn to validate the effectiveness of ConceptLab through a series of qualitative and quantitative evaluations.

### 5.1. Results

**Creative Generation.** First, in Figure 5, we demonstrate ConceptLab's ability to learn a wide range of novel creative concepts across various categories. All results are obtained using our adaptive negatives technique, highlighting our ability to generate these diverse concepts simply by varying the training seed.

Next, as demonstrated Figure 6, ConceptLab can place these learned creative concepts in novel scenes. As shown, these generations range from background modifications and artistic styles to imagining new creations resembling the concept. Yet, ConceptLab can go beyond generating new members of an *object* category. In Figure 7 we show how ConceptLab can be used to discover new artistic styles using our adaptive negative technique. Observe how each row captures a unique style while remaining faithful to the guiding text prompt. This further highlights the advantages of our adaptive training scheme which can be applied for a variety of different categories.

**Concept Mixing.** In Figure 9 we show how we can form hybrid concepts by merging unique traits across multiple real concepts using only positive constraints. Observe, for example, the first row where we are able to capture key characteristics of the lobster (e.g., its color and claws) and fuse them with those of a turtle (e.g., its shell). Moreover, in the second row, we are able to fuse three concepts, capturing the body of the snake, the texture of the zebra, and the head

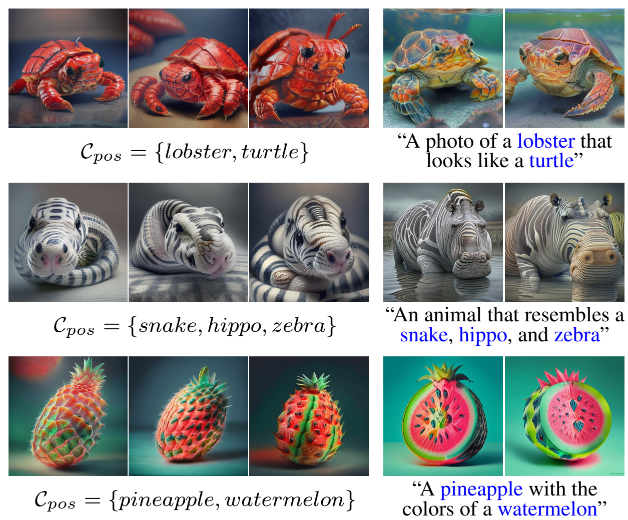



Figure 9. Mixing results obtained with ConceptLab. On the left, we show images generated using a concept learned by ConceptLab using positive constraints. On the right, we show results obtained with Kandinsky using curated prompts that aim to achieve a mixing result using prompt engineering.

of the hippopotamus. To illustrate that learning such combinations of concepts is non-trivial, we attempt to achieve a similar mixture using hand-crafted prompts. As shown on the right-hand side of Figure 9, such prompts fail to capture key aspects of all desired concepts.

**Evolutionary Generation.** We next explore our ability to mix various learned concepts using our evolution generation procedure, as described in Section 4.3. In Figure 8, we show results obtained across multiple "generations" of concepts learned by ConceptLab. For example, consider the

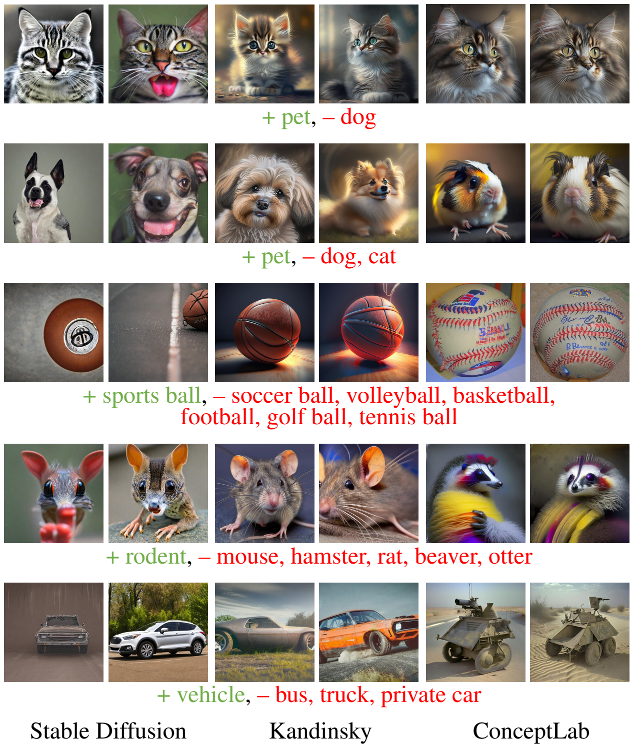



Figure 10. Comparison of ConceptLab and negative prompting. For both Stable Diffusion and Kandinsky, a negative prompt was composed containing all specified classes.

leftmost mixing in the provided family tree. Observe how we are able to fuse the color and general shape of the left parent with the distinct ears of the right parent to obtain a plausible blue-like rat mammal. We can then continue this evolutionary mix-up process across multiple generations as shown in the bottom-most row.

### 5.2. Comparisons

**Evaluation Setup.** While no related work tackles the exact same problem as ConceptLab, a natural baseline arises from the negative prompting technique [24], which has become a prominent technique in text-to-image generation. In the context of creative generation, it can potentially be used to generate novel concepts by defining a negative prompt that includes the negative constraints. We compare ConceptLab to two such baselines. Specifically, we consider both Stable Diffusion 2 [35] and Kandinsky 2.1 [39] and generate images using an input prompt of the form "A photo of a $c_{pos}$" where $c_{pos}$ is our positive token (e.g., "pet") and a negative prompt of the form "A photo of a $c_{neg,1}$,..., A photo of a $c_{neg,k}$" where $c_{neg,1}, \ldots, c_{neg,k}$ are our negative tokens (e.g., "cat", "dog", "hamster"). For Kandinsky, the negative prompt is applied over the Diffusion Prior model and not the Latent Diffusion Model, as it empirically resulted in more favorable results.

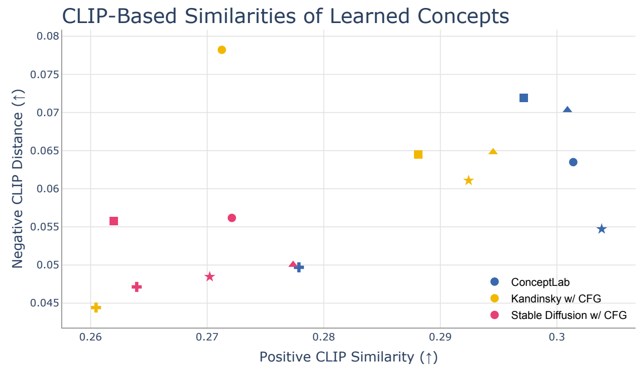



Figure 11. Quantitative evaluation. We compare ConceptLab to Kandinsky [39] and Stable Diffusion [35] with classifier-free guidance using negative prompting. For each, we compute (1) the similarity between the generated images and the positive concept, and (2) the difference between the positive similarity and the maximum negative similarity between the generated images and all negative concepts. Results are averaged across each category separately. The domains are represented by: pet: $\circ$, plant: $\square$, fruit: $\star$, furniture: $+$, musical instrument: $\triangle$.

**Qualitative Comparisons.** In Figure 10 we compare ConceptLab to the training-free baselines. As can be seen, while negative prompting does work when a single constraint is used, the baselines generally do not perform well when faced with multiple constraints. Specifically, even when tasked with generating a "pet" with both "cat" and "dog" explicitly stated in the negative prompt, both approaches tend to generate images of dogs. Conversely, ConceptLab is able to consistently align with both the positive token and negative constraints. We further note that the training-free baselines do not learn a consistent representation of a *specific* concept, and hence do not allow for the same editing capabilities as ConceptLab.

**Quantitative Comparisons.** We now turn to quantitatively evaluate the considered methods using a CLIP-based evaluation scheme. Specifically, we evaluate the ability of each method to (1) capture the positive concept while (2) generating images that do not resemble any of the given negative concepts. We consider five broad categories: pets, plants, fruits, furniture, and musical instruments. For each domain, we consider three different pairs of negative concepts (e.g., "cat" and "dog", "closet" and "bed", etc.) and train ConceptLab using five random seeds for each combination, resulting in a total of $5 \times 3 \times 5 = 75$ learned concepts. For each learned concept, we then generate 32 images using the prompt "A photo of a $S_*$", resulting in 160 images for each positive-negative combination. For Stable Diffusion and Kandinsky, we use negative prompting and generate $32 \times 5 = 160$ images for the same sets of positive and negative concept pairs.

| | Stable Diffusion | Kandinsky | **ConceptLab** |
|---|---|---|---|
| Average Rating (↑) | $1.90 \pm 1.11$ | $1.79 \pm 1.16$ | $\mathbf{3.77 \pm 1.35}$ |

Table 1. User Study. We asked respondents to rate images on a scale of 1 to 5 based on how well they respect a given set of constraints.

We define two measurements that are jointly used to measure and compare the different methods. First, we compute the positive similarity of each concept to the target category by calculating the CLIP-space similarity between the embeddings of all generated images and the text prompt "A photo of a $c_{pos}$", where $c_{pos}$ is our positive concept. Next, we compute a measurement of the distance between the positive constraints and the negative constraints. This is done by first calculating the maximum similarity between the generated images and all negative concepts. We then compute the difference between the previously computed positive similarity and the maximum negative similarity. This measures the method's ability to stay away from negative constraints, while also penalizing predictions that are out of distribution. Consider for example the case where the target concept is a "pet" and the negative constraints are "cat" and "dog", but the generated images resemble a "fruit". The negative similarity between the images and a "cat" and a "dog" would be low, but this is clearly an undesirable solution. Together, the metrics capture both the ability of the method to remain close to the positive class, while distinguishing its concepts from the negative constraints.

The results are illustrated in Figure 11 where we show the average metrics, split across the five categories. As can be seen, ConceptLab consistently outperforms both baselines in positive CLIP similarity across all five domains, indicating that ConceptLab is able to faithfully generate images belonging to the target broad category. Moreover, in terms of our negative distance metric, ConceptLab outperforms Stable Diffusion in all five categories while outperforming Kandinsky in four of the five categories (all except "pet"). This indicates that ConceptLab is able to generate images that belong to the target category, but differ significantly from existing concepts.

**User Study.** We additionally conduct a user study to compare ConceptLab to the negative prompting techniques. We follow the same evaluation setup as above and generate images using each method belonging to five different broad categories. We then asked respondents to rate the images generated by each method based on their ability to both capture the target broad concept category and differ from the specified negative concepts. Respondents were asked to rate each set of results on a scale from 1 to 5. Results are shown in Table 1. In total, we had 30 respondents, for a total of 300 ratings per method. As shown, participants heavily favored ConceptLab when compared to both baselines.


Figure 12. Ablation of applying our constraints in the prior space. For SD-ConceptLab we apply constraints over estimated denoised images. For CLIP-ConceptLab we apply the constraints directly on the text encoder output and only use the prior to generate the final images. To highlight our improved consistency, each concept is presented under two prompts: "A digital cartoon art of ..." on the right, and "A pencil sketch of ..." on the left.

### 5.3. Additional Analysis

**Using the Prior.** We now turn to validate the use of our prior constraints. To this end, we compare ConceptLab to two baselines. First, we consider ConceptLab *without* passing the text encoding through the Diffusion Prior $P$, a method which we call CLIP-ConceptLab, as all loss objectives from Equation (3) are computed over the text conditioning space, $E_y(\cdot)$. Next, we compare to a variant of ConceptLab using Stable Diffusion [35]. Specifically, we collect images of each negative class and apply our CLIP-space constraints between the collected images and denoised images $x_0$ computed throughout training using a DDIM scheduler [45]. We note this is not an existing method but rather our attempt to "implement" ConceptLab with Stable Diffusion, which we call SD-ConceptLab.

The results are illustrated in Figure 12. As can be seen, SD-ConceptLab often fails to align with the constraints, as can be seen in the first two rows, or generates inconsistent images between different prompts featuring the same learned token. While CLIP-ConceptLab usually does a surprisingly good job at respecting the constraints, it tends to be more inconsistent between different prompts. This aligns well with our insight that applying the Diffusion Prior over $E_y(v_*)$ encourages the generated instances of $v_*$ to better uphold the textual constraints.


Figure 13. The effect of the relative weighting of our loss between the positive and negative constraints. For small values of $\lambda$ (i.e., low positive weight), the positive constraint is ignored, while for large weights, the negative constraints are largely ignored.

**Balancing the Constraints.** In Figure 13, we explore the effect of the weighting between the positive and negative constraints as defined in Equation (3). As shown, when a low weight is given to the positive similarity, the resulting images do not align with the target positive category. Conversely, when the weight is too large, the negative constraints are generally ignored, and the resulting images depict existing concepts found in the list of negative concepts. We find that setting $\lambda = 1$ results in images that nicely balance both constraints.

**Similarity Analysis** In Figure 14, we demonstrate how the similarity to different constraints behaves along the optimization process when applying our adaptive negatives scheme. In the upper part of the Figure, we can observe that the similarity to the positive constraint, in this case, "pet", remains relatively constant. Every 250 iterations, a new negative constraint is added based on BLIP-2's predictions, and one can observe how the similarity to the new constraint decreases over time. At the bottom, we present the rendered images from which BLIP-2 inferred the new negative member to add to our list of constraints. Observe the unique concept generated after 1500 steps.

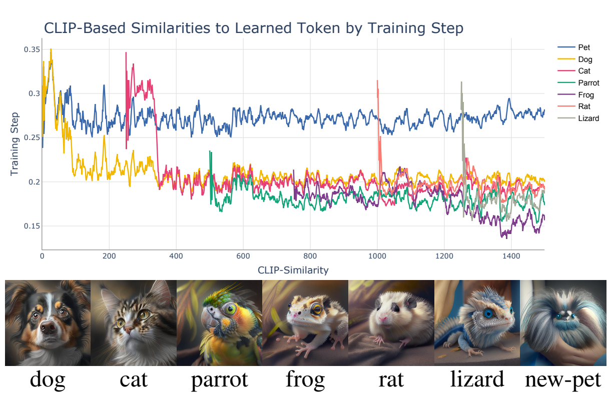



Figure 14. CLIP-based similarity between our learned concept and the positive and negative constraints throughout training.


Figure 15. Attempting to generate our novel generations with Kandinsky 2 [39]. Given an image generated by our method, we use CLIP Interrogator [30] to compose a prompt describing our concept, which is then used to generate an image. For example, the prompt for the rightmost image is: *"a close up of a lizard on a table, inspired by Bob Eggleton, zbrush central contest winner, yellow spiky hair, photoreal, vivid colours. sharp focus. wow!, realistic gold, great pinterest photo, beautiful, photo realistic"*.

**Generated Descriptions** Finally, we try to highlight the unique nature of our generated concepts by comparing them to images generated directly from text prompts. We first pass an image depicting a learned concept to a vision-language model [30] and ask it to compose a prompt corresponding to the input image. We then pass the generated prompt to Kandinsky [39] and generate a corresponding image. As can be seen in Figure 15, although the generated prompt is able to capture the general nature of the concept, its unique details are missing. For example, in the fourth column, the queried models understand that a rat-like animal is present, but fail to capture the unique color and spots present in our learned concept. A similar behavior can be observed with the lizard in the rightmost column where fine-level details of our reptile are not reproduced. One can potentially manually refine each prompt to better represent some of the missing properties of our generated concepts, but this only further highlights the creative nature of our automatically generated concepts.

## 6. Limitations

Our method is generally capable of learning intriguing and novel concepts that follow the given constraints. However, it is important to acknowledge its limitations. First, similar to personalization methods, creating new images with prompts that include the learned concept does not always preserve the concept's properties across different prompts. We illustrate such examples in the first two rows of Figure 16. Second, the optimization process itself does not always yield the desired outcomes. For some classes, such as "airplane" or "fish", ConceptLab struggles to generate creative concepts. We empirically observe that this is often related to negatives generated by BLIP-2. For instance, in some categories, BLIP-2 tends to produce highly specific negatives (e.g., a particular airplane model) that do not serve as a strong constraint. Examples of unsatisfactory results are presented in the bottom row of Figure 16.

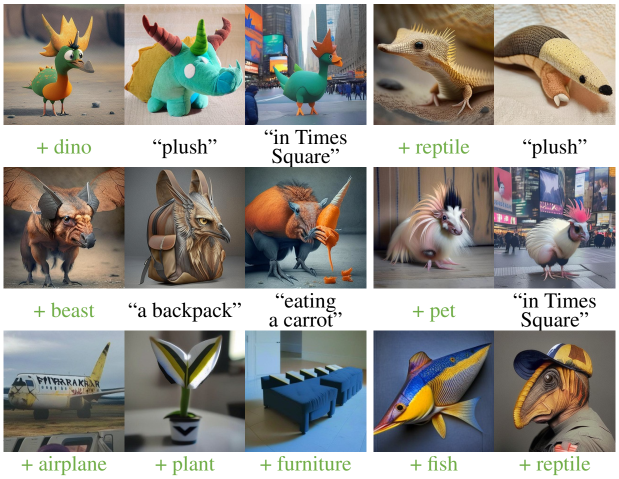



Figure 16. Limitations of our proposed method. Some edits do not respect all the properties of the generated concept, resulting in more generic outputs. Some learned concepts are not creative or do not respect the positive constraint well enough.

## 7. Conclusions

We introduced a novel approach for creative generation using text-to-image diffusion models. Specifically, we proposed to use Diffusion Prior models to learn novel concepts that belong to a given broad category. To optimize our learned concept we introduced "prior constraints", a set of positive and negative constraints applied over the Diffusion Prior output. By integrating a question-answering model into the optimization process we encouraged uniqueness while ensuring distinctness from existing category members. Our experiments demonstrate the effectiveness of our method, producing visually diverse and appealing concepts, and further showcasing the effectiveness of "prior constraints" for concept mixing. We hope that our approach will open up exciting possibilities for generating creative content using text-to-image models.

**Acknowledgements** We would like to give a special thanks to Hao Zhang for inspiring and encouraging us throughout this work. We would also like to thank Gal Metzer and Rinon Gal for their valuable feedback and suggestions. This work was supported by the Israel Science Foundation under Grant No. 2366/16 and Grant No. 2492/20.

## References

[1] Pranav Aggarwal, Hareesh Ravi, Naveen Marri, Sachin Kelkar, Fengbin Chen, Vinh Khuc, Midhun Harikumar, Ritiz Tambi, Sudharshan Reddy Kakumanu, Purvak Lapsiya, et al. Controlled and conditional text to image generation with diffusion prior. *arXiv preprint arXiv:2302.11710*, 2023. 3

[2] Yuval Alaluf, Elad Richardson, Gal Metzer, and Daniel Cohen-Or. A neural space-time representation for text-to-image personalization. *arXiv preprint arXiv:2305.15391*, 2023. 3

[3] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *arXiv preprint arXiv:2206.02779*, 2022. 2

[4] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers, 2023. 1, 2

[5] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 2

[6] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models, 2023. 2

[7] Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XX*, pages 558–577. Springer, 2022. 1, 3

[8] Daniel Cohen-Or and Hao Zhang. From inspired modeling to creative modeling. *The Visual Computer*, 32:7–14, 2016. 1

[9] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 2

[11] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*, 35:16890–16902, 2022. 1

[12] Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms. *arXiv preprint arXiv:1706.07068*, 2017. 3

[13] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. *arXiv preprint arXiv:2302.03011*, 2023. 3

[14] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 3, 4

[15] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Encoder-based domain tuning for fast personalization of text-to-image models, 2023. 1, 3

[16] Songwei Ge, Vedanuj Goswami, Larry Zitnick, and Devi Parikh. Creative sketch generation. In *International Conference on Learning Representations*, 2021. 3

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3

[18] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[19] Aaron Hertzmann. Can computers create art? In *Arts*, page 18. MDPI, 2018. 1, 3

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2, 3

[21] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Conference on Computer Vision and Pattern Recognition 2023*, 2023. 2

[22] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 3

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2, 4, 5

[24] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022. 9

[25] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 2

[26] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8, 2022. 3

[27] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 1, 2

[28] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2

[29] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023. 2

[30] pharmapsychotic. clip-interrogator. https://github.com/pharmapsychotic/clip-interrogator, 2022. 11

[31] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 3

[33] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 2, 3

[34] Warunika Lakmini Ranaweera. Exquimo: An exquisite corpse tool for co-creative 3d shape modeling. 2016. 3

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2, 3, 9, 10

[36] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 3

[37] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 1, 2

[38] Othman Sbai, Mohamed Elhoseiny, Antoine Bordes, Yann LeCun, and Camille Couprie. Design: Design inspiration from generative networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 3

[39] Arseniy Shakhmatov, Anton Razzhigaev, Aleksandr Nikolich, Vladimir Arkhipkin, Igor Pavlov, Andrey Kuznetsov, and Denis Dimitrov. Kandinsky 2. https://github.com/ai-forever/Kandinsky-2, 2022. 1, 2, 3, 9, 11

[40] Jing Shi, Wei Xiong, Zhe Lin, and Hyun Joon Jung. Instantbooth: Personalized text-to-image generation without test-time finetuning, 2023. 3

[41] Karl Sims. Artificial evolution for computer graphics. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, pages 319–328, 1991. 1

[42] Karl Sims. Evolving virtual creatures. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 15–22, 1994. 1

[43] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[44] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[45] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 10

[46] Yoad Tewel, Rinon Gal, Gal Chechik, and Yuval Atzmon. Key-locked rank one editing for text-to-image personalization. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023. 3

[47] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1921–1930, 2023. 2

[48] Yael Vinker, Andrey Voynov, Daniel Cohen-Or, and Ariel Shamir. Concept decomposition for visual exploration and inspiration. *arXiv preprint arXiv:2305.18203*, 2023. 3

[49] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation, 2023. 3

[50] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848*, 2023. 1, 3

[51] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20908–20918, 2023. 3

[52] Kai Xu, Hao Zhang, Daniel Cohen-Or, and Baoquan Chen. Fit and diverse: Set evolution for inspiring 3d shape galleries. *ACM Transactions on Graphics (TOG)*, 31(4):1–10, 2012. 1, 3

[53] Yufan Zhou, Bingchen Liu, Yizhe Zhu, Xiao Yang, Changyou Chen, and Jinhui Xu. Shifted diffusion for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10157–10166, 2023. 3

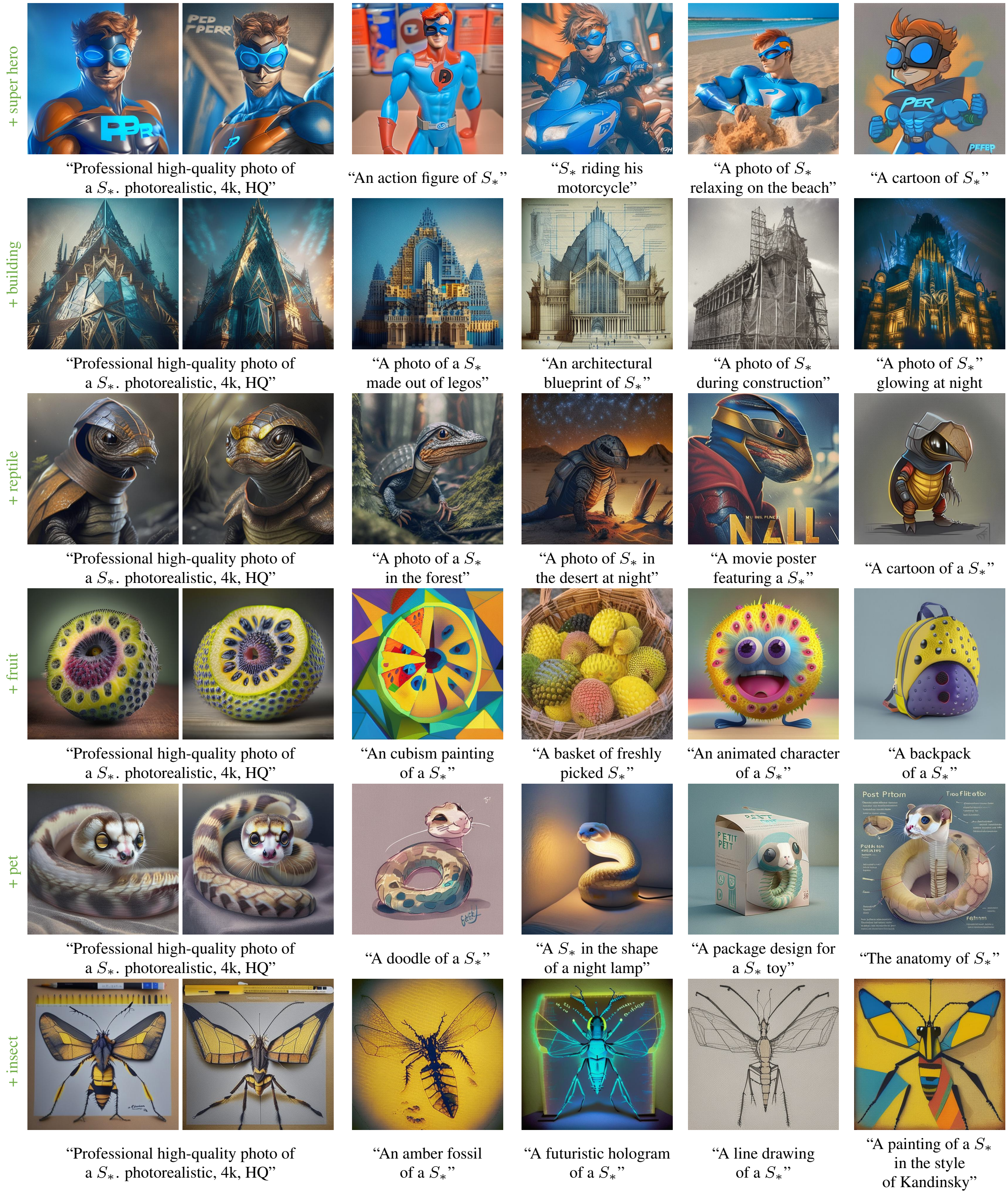



Figure 17. Sample text-guided creative generation results obtained with ConceptLab. The positive concept used for training is shown to the left. All results are obtained using our adaptive negative technique.