# Concept Lancet: Image Editing with Compositional Representation Transplant

## Supplementary Material

We organize the supplementary material as follows.
- §6 covers additional details of prior arts on diffusion-based editing to complement those mentioned in §2.
- §7 provides descriptions for collecting the dataset CoLan-150K and implementing concept transplant method CoLan.
- §8 gives extra visualizations on the dataset and the method.
- §9 discusses limitations, future works and societal impacts.
- §10 details the prompting templates used in dataset collection and inference.

## 6. Prior Arts for Diffusion-Based Editing

To generate a new image $\boldsymbol{z}_0$ based on text prompts, diffusion models sample from a standard Gaussian $\boldsymbol{z}_T$ and recursively denoise it through the reverse process [49]:

$$\begin{aligned} \boldsymbol{z}_{t-1} = & \sqrt{\alpha_{t-1}} \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c}) + \sqrt{1-\alpha_{t-1}-\sigma_t^2} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c}) \\ & + \sigma_t \boldsymbol{\epsilon}_t, \text{ with } \boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \boldsymbol{I}). \end{aligned} \tag{5}$$

Here $\boldsymbol{z}_t$ is the denoised image at time $t$, $\boldsymbol{c}$ is the text embedding of the caption of the image to be sampled, $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})$ and $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})$ are two networks that predict the *score function* [50] and the denoised image $\boldsymbol{z}_0$ respectively given $\boldsymbol{c}$ and $\boldsymbol{z}_t$. As we elaborate below, different choices for $\alpha_t, \sigma_t, \boldsymbol{f}_{\boldsymbol{\theta}}$ give rise to a class of diffusion models for editing.

**DDIM-Based Editing.** By choosing $\sigma_t = 0$ and $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c}) = \frac{\boldsymbol{z}_t - \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})}{\sqrt{\alpha_t}}$ for every $t$, the denoising process in (5) yields DDIM sampling [49]. To make sure such a process generates the source image $\boldsymbol{z}_0$ faithfully, one replaces the stand Gaussain $\boldsymbol{z}_T$ with noise computed from a special forward process that iteratively adds deterministic noises, computed via $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\cdot, t, \boldsymbol{c})$, to the source image $\boldsymbol{z}_0$. Some regularization can improve the statistical properties of these noises, resulting in better image editability during the denoising process [40]. Recently, the work of [18] have proposed *Direct Inversion (DI)* to add further guidance, allowing exact recovery of $\boldsymbol{z}_0$ following the source branch and then improving the visual quality of the edited image when concept transfer is imposed.

**Consistency-Model-Based Editing.** Instead of parameterizing $\boldsymbol{f}_{\boldsymbol{\theta}}$ using the learned score $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$, one can learn a separate network for $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})$ to approximate the flow map of the *probablity flow ODE* [50], the deterministic counterpart of DDPM [14] sampling. With the above and the choice of $\sigma_t = \sqrt{1-\alpha_{t-1}}$ for every $t$, the process in (5) gives *Multi-step Consistency Model Sampling* [51], and $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})$ in this case is called the *Consistency Model* [34, 51]. Through a trained consistency model, one can ideally denoise $\boldsymbol{z}_t$ into $\boldsymbol{z}_0$ in one pass of $\boldsymbol{f}_{\boldsymbol{\theta}}$. However, the denoised $\boldsymbol{z}_0^{(t)} := \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c})$ has low quality if $\boldsymbol{z}_t$ is close to a Gaussian, thus a multi-step sampling is adopted to improve the sampled image quality [50]. For the image editing purpose, [65] propose *Virtual Inversion (VI)* that guides the process to sample the source image at every time $t$ in the source branch, i.e., $\boldsymbol{z}_0^{(t)} = \boldsymbol{z}_0, \forall t$.

## 7. Framework Details

**Dataset Collection.** Each concept in the CoLan-150K approximately consists of 30 stimuli. We use GPT-4o (with vision module) [38] for parsing source input and proposing the concepts. After curating all concepts, we use GPT-4o (without vision module) to generate diverse concept stimuli. The instructions for them are shown in §10.

**Concept Transplant.** When constructing the dictionary in the CLIP text embedding space, each concept vector is a sequence of tokens flattened as a single vector of dimension $d = 77 \times 768 = 59136$, where 77 is the maximum number of tokens after padding and 768 is the dimension of token embeddings. For plugging CoLan on the text embedding space of P2P-Zero, we refer to analyzing the process of $c + \Delta c$ in Algorithm 1 of [40]. For plugging CoLan on the text embedding space of InfEdit, we refer to decomposing the embedding of its source branch to solve the coefficients. For plugging CoLan on the score space of InfEdit, we refer to analyzing the $\varepsilon_{\tau_n}^{\text{cons}} + \varepsilon_{\tau_n}^{\text{tgt}} - \varepsilon_{\tau_n}^{\text{src}}$ in Algorithm 2 of [65]. Specifically, given a concept $x$, its direction $\boldsymbol{d}_x$ for concept dictionary in the score space at the time step $t$ is generated as follows:

$$\epsilon_x = \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\cdot; t, \operatorname{RepRead}(E(\boldsymbol{s}_1^x), \dots, E(\boldsymbol{s}_K^x)))$$

where the $\operatorname{RepRead}(\cdot)$ corresponds to the representation reading algorithms described in §3.1.

**Evaluation Detail.** In Table 1, we evaluate all diffusion-based editing baselines with the backbone of Stable Diffusion V1.5 [45], and consistency-based baselines with the Latent Consistency Model [34] (Dreamshaper V7) which is distilled from Stable Diffusion V1.5. The hyperparameter for the sparsity regularizer $\lambda = 0.01$. The null embedding or $\emptyset$ in the paper refers to the CLIP embedding of the empty string. When adding/inserting a target concept, as there is no counterpart described in the source caption, we instruct the VLM to propose a counterpart present in the source image and revise the source caption. The revised dataset will be open-sourced together with all concept stimuli. We use P2P-Zero as the backbone for the representation analysis in CoLan-150K and comparing editing strengths. The experi-

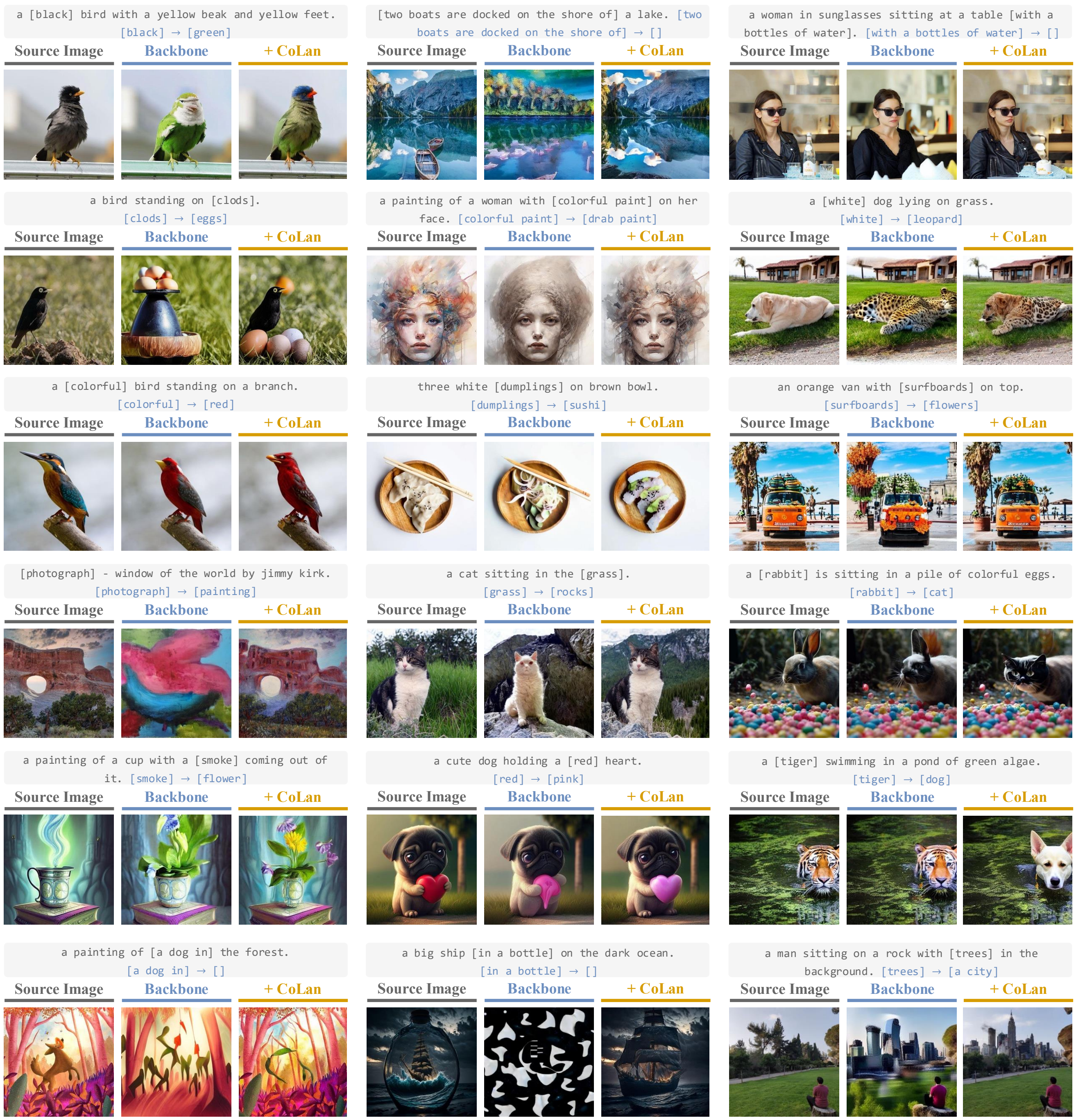


Figure 11. Additional visual comparison of CoLan in the text embedding space of P2P-Zero. We observe that the backbone plugging with CoLan has editing results that visually better align with the task.

ments in §4 are performed on a workstation of 8 NVIDIA A40 GPUs.

**Pipeline.** Algorithm 1 shows the full procedure of our proposed framework CoLan. The first part of the algorithm is to extract a set of concept vectors from the input editing image-text tuples based on § 3.1), followed by the second part where we transplant the target concept via sparse decomposition in § 3.2. In the first part, we instruct a VLM to parse the source input into a set of relevant concepts, and then we instruct an LLM to generate concept stimuli for every concept. Using the concept stimuli, we extract a collection of concept vectors using representation reading from the latent space of our diffusion model. Then, in the second part of CoLan, we decompose the text embedding or diffusion score

**Algorithm 1:** Concept Lancet (CoLan) for Diffusion-based Image Editing

**Input:** Frozen diffusion-based image editing backbone $\mathcal{F}_\theta$, image editing tuples (source prompt, source image, target prompt) $P = \{(\boldsymbol{p}_i, \boldsymbol{q}_i, \boldsymbol{p}'_i)\}_{i=1}^{N_q}$

Parse $P$ with the vision-language model to collect the concepts $X = \mathrm{VLM}(P)$

For each concept $x_i \in X$ : ▷ §3.1: Concept Dictionary Synthesis

Instruct the LLM to synthesize concept stimuli $\{\boldsymbol{s}_j^{x_i}\}_{j=1}^{K} = \mathrm{LLM}(x_i)$

Extract the concept vector $\boldsymbol{d}_{x_i} = \mathrm{RepRead}((\boldsymbol{s}_j^{x_i}\}_{j=1}^{K})$

Stack concept vectors $\{\boldsymbol{d}_{x_i}\}_{i=1}^{N_x}$ as columns of the concept dictionary $\boldsymbol{D}$.

For each source prompt-image pair $(\boldsymbol{p}_i, \boldsymbol{q}_i) \in P$: ▷ §3.2 Concept Transplant via Sparse Decomposition

Encode $\boldsymbol{p}_i$ to the text embedding space or $(\boldsymbol{p}_i, \boldsymbol{q}_i)$ to the diffusion score space as the source representation $\boldsymbol{v}$

Solve for the compositional coefficients that reconstruct the source $\boldsymbol{w}^* = \mathrm{argmin}_{\boldsymbol{w}} \|\boldsymbol{v} - \boldsymbol{D}\boldsymbol{w}\|_2^2 + \lambda \|\boldsymbol{w}\|_1$

Curate a modified dictionary $\boldsymbol{D}'$ by replacing the column of the source concept with that of the target concept

Obtain the edited latent representation as $\boldsymbol{v}' = \boldsymbol{D}'\boldsymbol{w}^* + \boldsymbol{r}$.

Generate the edited image through the image editing backbone $q'_i = \mathcal{F}_\theta(\boldsymbol{v}')$.

**Output:** The edited images $Q' = \{\boldsymbol{q}'_i\}_{i=1}^{N_q}$.


Figure 12. Visualizations of editing results. The first row shows the source images, the second row shows the results with the fixed edit strength of 0.7 for the concept [dog] without CoLan analysis, and the third row shows the edit results with CoLan analysis.

of the source representation using sparse coding techniques. After obtaining the coefficients of each concept vector, we perform a transplant process with the customized operation of removing, adding, or replacing. Finally, we synthesize the edited images with the modified latent representation with the image editing backbone.

## 8. Additional Results

This section provides additional results for CoLan. It includes more editing improvements with baseline models and visualization of concept instances from our CoLan-150K dataset.

**Visual Comparison.** Figure 11 shows additional visualization of the image editing results. The experiment settings follow §4.2. We observe that the editing backbone has a better editing performance after plugging CoLan.

**Concept Grounding.** Figure 13 visualizes the edited images with the extracted concept vectors [watercolor], [dog], and [wearing hat] from the stimuli of our CoLan-150K dataset. We observe that the edited images correctly reflect the semantic meaning of the concepts, which indicates that our concept stimuli successfully ground the concept. Figure 14 further shows additional samples of concepts and their stimuli. Note that there are approximately 30 stimuli per concept, and our figure shows the first three for each concept.

**Edit Strength.** Figure 12 shows the editing results from source images of the cat to the target concept dog without or with CoLan. The synthesis setting follows the Comparing Editing Strengths section in §4 and we fix the edit strength to 0.7 if CoLan is not used. From the second row of the figure, we observe that different source images of the cat require different magnitudes of editing, and simply choosing

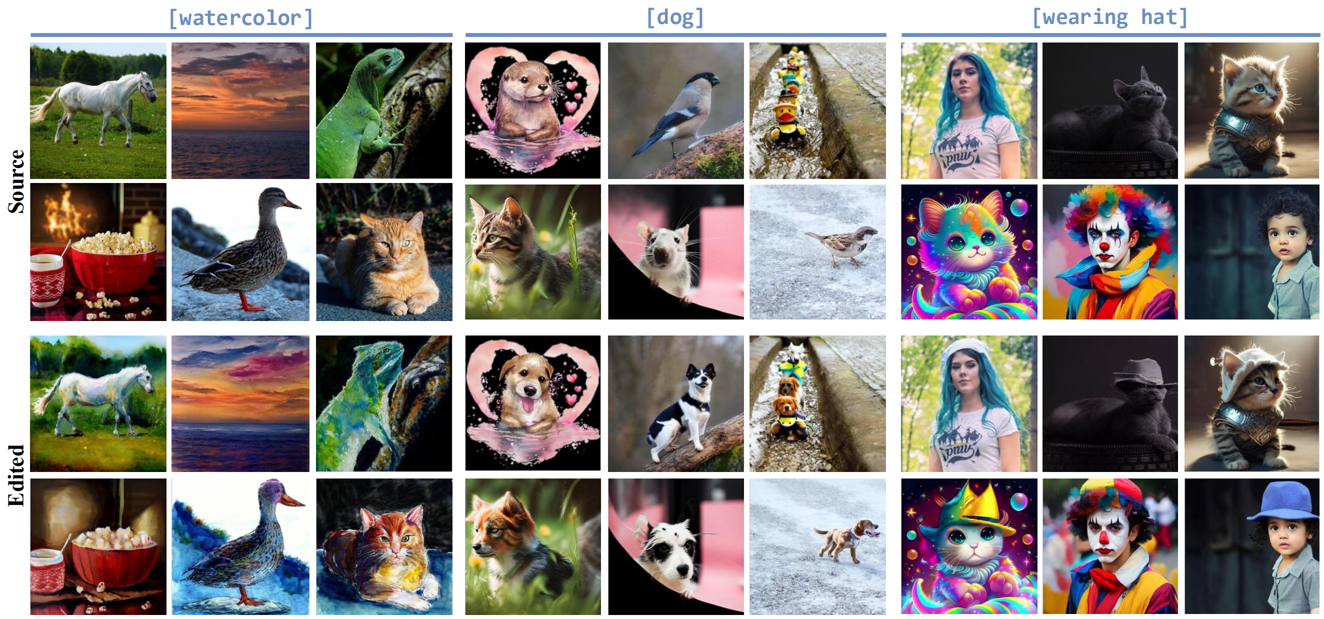

Figure 13. Visualizations of concept grounding for sampled concepts from our CoLan-150K dataset. We observe that the extracted concept vectors from our dataset corresponds to the desired semantics by visualization.

a unified strength for all source images will frequently result in unsatisfactory results for different images (under-edit or over-edit). Then in the third row of the figure, we show that editing with CoLan results in more consistent and reasonable visual results. This is because our framework adaptively estimates the concept composition of each image and solves customized edit strengths for each source image.

## 9. Limitations, Future Works, Societal Impacts

While CoLan demonstrates strong performance for diffusion-based image editing, we elaborate on potential limitations and directions for future work in this section.

**Limitation.** The current framework primarily operates upon diffusion-based backbones with attention-control mechanisms where source concepts correspond to certain regions of interest. It will be challenging to perform spatial manipulations that require editing across different sectors of attention maps. For instance, consider tasks such as moving the cat from right to left or relocating the table to the corner, which shall require non-trivial operations in the attention modules. Another challenge lies in handling numerical modifications, such as changing the number of objects (e.g., changing an image of two cats to have three cats) or composing numerical relations with multiple different objects (e.g., adding two apples to an image of three bananas).

**Future Work.** Future work could explore methods to enhance CoLan's capabilities to handle spatial relationships and global layout modifications while preserving its precise concept manipulation advantages. For numerical editing, it is worthy exploring the bag-of-words effect of CLIP or how the diffusion model shall encode numerical concepts in a way that straightforward manipulation is permitted [60, 63, 67]. The precise mapping between numerical concepts and their representations in the latent space warrants further investigation to enable more sophisticated counting-based edits.

**Societal Impact.** Image editing frameworks with high user-accessibility (through the prompt-based interface) raise considerations about potential misuse. The ability to perform precise conceptual edits could be exploited to create misleading, controversial, or deceptive content. While our framework focuses on enhancing editing quality, future development should incorporate safeguarding against malicious requests and protecting copyrights in content creation.

## 10. Prompting Template

As mentioned in Section 3.1, we instruct the VLM to perform two tasks: rewriting prompts for concept addition or insertion and constructing detailed concept dictionaries. We then instruct the LLM to synthesize concept stimuli. Figure 15 shows the instructions (prompting template) for rewriting captions by identifying source concepts and generating rewritten prompts tailored for image editing tasks. Figure 16 shows the instructions for constructing a comprehensive concept list by parsing multimodal information from the source input. This ensures that the list captures diverse and unique aspects of the source image and prompt. Finally, Figure 17 shows the instructions for generating diverse and contextually rich concept stimuli, which enables the mapping to conceptual representations.

**brown hair**

Brown hair can range in shade from light caramel to deep chocolate, providing a rich variety of options.

Individuals with brown hair may have natural highlights that give depth and dimension to their hair color.

Chestnut brown is a warm, reddish-brown hair color that glows beautifully in sunlight.

**candle**

Candles provide soft, ambient lighting that can create a cozy atmosphere in any room.

A flickering candle casts dancing shadows on the walls, adding a sense of warmth and comfort.

Scented candles are infused with fragrances like lavender or vanilla, helping to create a relaxing environment.

**cloudy sky**

The sky is overcast with thick cumulus clouds, creating a soft, diffuse light over the landscape.

A cloudy sky casts a muted gray hue, affecting the mood of the scene below with an introspective calm.

Patches of blue peek through scattered clouds, hinting at the possibility of a clearing sky.

**desert**

Deserts are characterized by their arid environment and receive less than 25 centimeters of rain annually.

The Sahara Desert is the largest hot desert in the world, stretching across multiple countries in North Africa.

Erosion from wind and water can create dramatic rock formations found in desert environments.

**digital art**

Digital art is created using software tools such as Adobe Photoshop, Corel Painter, or Procreate.

The evolution of digital art has enabled artists to explore new styles and techniques not possible with traditional media.

Artists often use digital tablets and styluses to draw and paint, simulating the feel of natural media.

**elegant**

An elegant ballroom filled with chandeliers and lavish decorations sets the stage for a grand event.

The elegance of a swan gliding gracefully across a serene lake at dusk.

The minimalist design of a Scandinavian interior reflects a refined sense of elegance.

**fireplace**

A fireplace crackles softly, providing both warmth and a cozy ambiance on a cold winter night.

The mantel above the fireplace is often used to display photographs or holiday decorations.

Fireplaces can be fueled by wood, gas, or electricity, each offering distinct characteristics and maintenance needs.

**fluffy**

Fluffy clouds drift lazily across the sky, resembling puffs of cotton wool.

A fluffy white cat purrs contentedly as it curls up on a soft cushion.

Freshly fallen snow creates a fluffy blanket over the landscape.

**garden**

Gardens can be designed to include a variety of plants, including ornamental flowers, shrubs, and trees.

A vegetable garden is cultivated to grow produce such as tomatoes, lettuce, and carrots for home consumption.

A butterfly garden is planted with colorful, nectar-rich flowers like milkweed and lantana to attract butterflies.

**glass container**

Glass containers are typically transparent, allowing for easy viewing of their contents.

A mason jar is a popular type of glass container used for preserving food through canning.

Glass containers are often used to store spices in the kitchen, with airtight seals to maintain freshness.

**historic architecture**

Historic architecture often features intricate carvings and stonework that tell a story of the era in which it was built.

The Roman Colosseum, with its massive stone arches and elliptical structure, is a testament to ancient engineering prowess.

Victorian-era buildings are characterized by their ornate detail, steep gables, and patterned brickwork.

**in the city**

A bustling city street is full of traffic, with cars honking and people rushing by.

The city skyline is dominated by towering skyscrapers, casting long shadows as the sun sets.

Sidewalk cafes in the city are popular spots for people to sip coffee and watch passersby.

**keyboard**

A mechanical keyboard features individual switches for each key, providing tactile feedback and durability.

Wireless keyboards offer portability and convenience by using Bluetooth or RF technology for connectivity.

Ergonomic keyboards are designed to reduce strain on the wrists and hands, often featuring split designs or curved layouts.

**living room**

The living room often serves as the central gathering space for families to relax and entertain guests.

A cozy living room features a large, plush sofa adorned with soft, colorful throw pillows.

The living room is often equipped with a wall-mounted television and a sound system for entertainment.

**looking at the camera**

A child looks directly into the camera, capturing an expression of innocent curiosity.

The bride and groom pose for a photo, both looking directly at the camera with wide smiles.

A group of friends gathers for a selfie, all leaning in and making eye contact with the camera lens.

**marble wall**

Marble walls are often associated with luxury and elegance due to the stone's rich veining and natural beauty.

A marble wall can have a highly polished finish that reflects light and brightens a room.

The unique patterns of veining in marble walls can create a one-of-a-kind visual impact, with swirls and lines varying from slab to slab.

**on the grass**

A toddler crawls on the grass, giggling as they touch the soft blades for the first time.

A couple enjoys a picnic on the grass under the shade of a sprawling oak tree.

Morning dew glistens like tiny pearls on the grass, catching the light of the rising sun.

**purple**

Purple has long been associated with royalty and luxury, stemming from the rarity of the dye historically used to create it.

A field of lavender in bloom paints the landscape a soft purple, with a calming fragrance wafting through the air.

The twilight sky often transitions through shades of purple as the sun sets, creating a peaceful ambiance.

**spaceship**

Spaceships are engineered for travel beyond Earth's atmosphere, equipped with advanced propulsion systems.

A spaceship hovers silently in deep space, its surface gleaming under distant starlight.

The interior of a spaceship often features zero-gravity living quarters and control panels.

**snowflakes**

Snowflakes are intricate ice crystals that form in the atmosphere under cold conditions.

Each snowflake has a unique pattern, with no two snowflakes ever being identical due to their complex formation process.

Snowflakes typically have a hexagonal shape due to the molecular structure of ice.

**table lamp**

Table lamps often come with a fabric shade that diffuses light to create a cozy ambiance.

A touch-activated table lamp allows users to adjust the brightness with a simple tap.

Table lamps can feature adjustable necks, making them perfect for reading or focused tasks.

**tennis ball**

A tennis ball is typically made of a rubber core covered in a fuzzy felt material, giving it a distinctive texture.

Tennis balls are often bright yellow to enhance visibility during play on the court.

The standard diameter of a tennis ball is about 6.7 centimeters, conforming to international regulations.

**vibrant**

The vibrant colors of a sunset cast a warm glow over the horizon.

A vibrant city street bustling with life, full of people, street vendors, and musicians.

A vibrant painting captures the eye with its bold use of color and dynamic brushstrokes.

**wooden**

Wooden furniture often showcases the natural grain and beauty of the wood used.

The creaking of a wooden floor adds character to an old house.

A wooden sculpture carved from oak can highlight the artist's skill and attention to detail.

Figure 14. Additional samples of the concept stimuli from CoLan-150K. Each concept consists of approximately 30 stimuli and this figure samples the first three for a concept.

**Rewriting Captions for Concept Addition/Insertion**

```
You are one of the best experts in Generative Models and Concept Learning in the world. You are very good at designing concept dictionary to research the representation in latent space from CLIP or Score-based Generative Models, which have wide applications in image editing. You are a great expert in understanding and parsing multimodal information from a given image. Now, given a source prompt, a target prompt, and a source image, your task is to rewrite the source prompt for the image editing task. Usually, there is a focused pair of concepts in the source prompt and the target prompt to be edited (e.g., "cat" to "dog"). The source concept is usually annotated in the brackets ("[]") in the source prompt. However, in some editing tasks, there is no clear source concept mentioned in the source prompt. Hence, for these tasks, you are required to comprehend the source image and identify the corresponding source concept. After comprehending the source image, you need to generate a re-written source prompt with a clearly annotated source concept.


Here are two demonstrations:

Source Prompt: a slanted mountain bicycle on the road in front of a building
Target Prompt: a slanted [rusty] mountain bicycle on the road in front of a building
Source Concept: ""
Target Concept: "rusty"
Source Image: (IMG)
Re-written Source Prompt: a slanted [new] mountain bicycle on the road in front of a building


Source Prompt: two birds sitting on a branch
Target Prompt: two [origami] birds sitting on a branch
Source Concept: ""
Target Concept: "origami"
Source Image: (IMG)
Re-written Source Prompt: two [real] birds sitting on a branch


The identified source concept should not be the same as the target concept. The response MUST be with brackets ("[]") around the source concept. You should not use "without" frequently. Try your best to comprehend the image.
You should only output the re-written source prompt. DO NOT print anything else such as "Here are ...", "Sure, ...", "Certainly, ...".
DO NOT print quotation marks unless necessary. Just return the string.

Source Prompt: <input>
Target Prompt: <input>
Source Concept: <input>
Target Concept: <input>
Source Image: <input>
Re-written Source Prompt: <fill the response here>
```

Figure 15. The instructions for rewriting the task of concept addition/insertion with the VLM-found source concept as the counter-part.

**Concept Dictionary Construction**

You are one of the best experts in Generative Models and Concept Learning in the world. You are very good at designing concept dictionary to research the representation in latent space from CLIP or Score-based Generative Models, which have wide applications in image editing. You are a great expert in understanding and parsing multimodal information from a given image. Now, given a source prompt, a target prompt, and a source image, your task is to parse the given information into a concept list. The concept list consists of concepts, attributes, objects, and items that comprehensively describe the source image and cover the source prompt. Your concept list must have at least 15 concepts. As the concept list is for the task of image editing, there is a focused pair of concepts in the source prompt and the target prompt to be edited. The source concept is usually annotated in the bracket ("[]") in the source prompt. You must put the focused concept in the source prompt as the FIRST atom in the concept list. You must NOT put the focused concept in the target prompt in the concept list.

Here are three demonstrations:

Source Prompt: a [round] cake with orange frosting on a wooden plate
Target Prompt: a [square] cake with orange frosting on a wooden plate
Source Concept: "round"
Target Concept: "square"
Source Image: (IMG)
Concept List: ["round", "cake", "orange", "frosting", "wooden", "plate", "swirl", "creamy", "crumbly", "smooth", "rustic", "natural", "muted", "handmade", "warm", "minimalist", "unfrosted", "botanical", "bark", "inviting", "cozy", "textured", "simple", "organic", "earthy", "soft", "classic", "contrasting", "neutral", "clean"]

Source Prompt: a painting of [a dog in] the forest
Target Prompt: a painting of the forest
Source Concept: "a dog in"
Target Concept: ""
Source Image: (IMG)
Concept List: ["a dog in", "painting", "forest", "trees", "leaves", "sunlight", "vibrant colors", "orange hues", "pink trees", "purple plants", "playful", "cartoonish", "nature", "animals", "butterflies", "fantasy", "surreal", "whimsical", "tall trees", "shadows", "depth", "light beams", "foliage", "dynamic", "warm tones", "imaginative", "dreamlike", "motion", "soft textures", "layered composition", "bright atmosphere"]

Source Prompt: blue light, a black and white [cat] is playing with a flower
Target Prompt: blue light, a black and white [dog] is playing with a flower
Source Concept: "cat"
Target Concept: "dog"
Source Image: (IMG)
Concept List: ["cat", "black", "white", "blue light", "flower", "playing", "paws", "stone path", "curious", "whiskers", "small", "fluffy", "outdoor", "pink petals", "focused", "nature", "detailed fur", "green stem", "bright", "youthful", "movement", "natural light", "close-up", "gentle", "exploration", "soft shadows", "grass between stones", "alert", "innocent", "delicate"]

The concepts in the list should not be redundant or repetitive. Each concept in the list represents a unique perspective of objects, styles, and contexts. The response MUST be in Python list format.
You should have at least 15 concepts in the list. You should only output the Python list.
DO NOT print anything else such as "Here are ...", "Sure, ...", "Certainly, ...". Just return the list ["", "", ..., ..., ""].

Source Prompt: <input>
Target Prompt: <input>
Source Concept: <input>
Target Concept: <input>
Source Image: <input>
Concept List: <fill the response here>

Figure 16. The instructions for the VLM to parse the source image-prompt tuple into the concept list for the concept dictionary.

**Concept Stimulus Synthesis**

```
You are one of the best experts in Generative Models and Concept Learning in the world. You are very good at generating
concept stimuli to research the representation in latent space from CLIP or Score-based Generative Models, which have
wide applications in image editing. You are a great expert in providing relevant information and scenarios based on a
given concept. Now, given a concept, your task is to generate 30 (THIRTY) instances of concept stimuli for a given
concept.  As the concept stimuli will be used for the task of image editing, we need comprehensive, diverse, and
accurate descriptions and examples for the concept.

Here are three demonstrations of the concept and its corresponding concept stimuli:

Concept: dog
Concept Stimuli:
[
"Dogs are known for their loyalty and strong bonds with humans.",
"A dog wags its tail excitedly when it sees its owner after a long day.",
"Puppies often chew on objects as a way to explore their environment.",
"The sound of a dog's bark can vary depending on its breed and mood.",
"Dogs rely heavily on their sense of smell, which is far more sensitive than that of humans.",
"A dog runs alongside its owner during a morning jog, full of energy.",
...
]

Concept: cat
Concept Stimuli:
[
"Cats are known for their graceful, stealthy movements.",
"A cat stretches lazily under the warm afternoon sun.",
"Kittens explore their surroundings with curiosity and playfulness.",
"A cat's purring has been shown to have a calming effect on humans.",
"Stray cats often rely on their instincts and sharp senses for survival.",
"The eyes of a cat reflect light in the dark, giving them superior night vision.",
...
]

Concept: cake
Concept Stimuli:
[
"Cakes are often baked in layers and filled with frosting or cream in between each layer.",
"A slice of cake reveals its moist interior, topped with a rich layer of chocolate ganache.",
"Cakes are a common centerpiece for celebrations such as birthdays, weddings, and anniversaries.",
"A cake adorned with fresh berries and whipped cream makes for a light, summery dessert.",
"Cupcakes are miniature cakes baked in individual paper liners and often topped with buttercream frosting.",
"The aroma of a freshly baked vanilla cake fills the kitchen with a warm, sweet scent.",
...
]

The concept stimuli in the list should not be redundant or repetitive. Each stimulus in the list represents a unique
perspective (e.g., styles, contexts, examples, attributes, descriptions, usages) of the concept. The response MUST be
in Python list format.
You should have at least 30 stimuli in the list. You should only output the Python list.
DO NOT print anything else such as "Here are ...", "Sure, ...", "Certainly, ...". Just return the list ["", "", ...,
..., ""].

Concept: <input>
Concept Stimuli: <fill the response here>
```

Figure 17. The instructions for the LLM to generat diverse stimuli given a concept.

## Acknowledgment

This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via a grant of the BENGAL Program and a grant of the HIATUS Program (Contract #2022-22072200005), by the Defense Advanced Research Projects Agency's (DARPA) SciFy program (Agreement No. HR00112520300), and by the Penn Engineering Dean's Fellowship. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

## References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022.

[2] Manuel Brack, Felix Friedrich, Katharia Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinario Passos. Ledits++: Limitless image editing using text-to-image models. In *CVPR*, 2024.

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

[4] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, 2023.

[5] Siyi Chen, Huijie Zhang, Minzhe Guo, Yifu Lu, Peng Wang, and Qing Qu. Exploring low-dimensional subspaces in diffusion models for controllable image editing. *arXiv preprint arXiv:2409.02374*, 2024.

[6] Guillaume Couairon, Matthijs Douze, Matthieu Cord, and Holger Schwenk. Embedding arithmetic of multimodal queries for image retrieval. In *CVPR*, 2022.

[7] Rohit Gandikota, Joanna Materzynska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. *arXiv preprint arXiv:2311.12092*, 2023.

[8] Rohit Gandikota, Joanna Materzyńska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *ICCV*, 2023.

[9] Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Han Hu, Dong Chen, and Baining Guo. Instructdiffusion: A generalist modeling interface for vision tasks. *arXiv preprint arXiv:2309.03895*, 2023.

[10] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. In *IEEE TIP*, 2019.

[11] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *ICLR*, 2023.

[12] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *SIGGRAPH*, 2001.

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[15] Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *ICPR*, 2010.

[16] Wenjing Huang, Shikui Tu, and Lei Xu. Pfb-diff: Progressive feature blending diffusion for text-driven image editing. *arXiv preprint arXiv:2306.16894*, 2023.

[17] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly DDPM noise space: Inversion and manipulations. In *CVPR*, 2024.

[18] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *ICLR*, 2024.

[19] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *CVPR*, 2019.

[20] 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 *CVPR*, 2023.

[21] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *ICCV*, 2023.

[22] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022.

[23] Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Benita Teufel, Marco Bellagente, Minguk Kang, Taesung Park, Jure Leskovec, Jun-Yan Zhu, Li Fei-Fei, Jiajun Wu, Stefano Ermon, and Percy Liang. Holistic evaluation of text-to-image models. In *NeurIPS Datasets and Benchmarks Track*, 2023.

[24] Bowen Li, Zhaoyu Li, Qiwei Du, Jinqi Luo, Wenshan Wang, Yaqi Xie, Simon Stepputtis, Chen Wang, Katia Sycara, Pradeep Ravikumar, Alexander Gray, Xujie Si, and Sebastian Scherer. Logicity: Advancing neuro-symbolic ai with abstract urban simulation. In *NeurIPS Datasets and Benchmarks Track*, 2024.

[25] Hang Li, Chengzhi Shen, Philip Torr, Volker Tresp, and Jindong Gu. Self-discovering interpretable diffusion latent directions for responsible text-to-image generation. In *CVPR*, 2024.

[26] Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Li Erran Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. In *NeurIPS Datasets and Benchmarks Track*, 2024.

[27] Senmao Li, Joost van de Weijer, Taihang Hu, Fahad Shahbaz Khan, Qibin Hou, Yaxing Wang, and Jian Yang. Stylediffusion: Prompt-embedding inversion for text-based editing. *arXiv preprint arXiv:2303.15649*, 2023.

[28] Qiyao Liang, Ziming Liu, Mitchell Ostrow, and Ila R Fiete. How diffusion models learn to factorize and compose. In *NeurIPS*, 2024.

[29] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In *ECCV*, 2022.

[30] Sheng Liu, Haotian Ye, Lei Xing, and James Zou. Reducing hallucinations in vision-language models via latent space steering. *arXiv preprint arXiv:2410.15778*, 2024.

[31] Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

[32] Jinqi Luo, Zhaoning Wang, Chen Henry Wu, Dong Huang, and Fernando De La Torre. Zero-shot model diagnosis. In *CVPR*, 2023.

[33] Jinqi Luo, Tianjiao Ding, Kwan Ho Ryan Chan, Darshan Thaker, Aditya Chattopadhyay, Chris Callison-Burch, and René Vidal. Pace: Parsimonious concept engineering for large language models. In *NeurIPS*, 2024.

[34] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

[35] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic Regularities in Continuous Space Word Representations. In *NAACL HLT*, 2013.

[36] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022.

[37] OpenAI. GPT-4V(ision) System Card, 2023.

[38] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[39] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

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

[41] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In *ICCV*, 2021.

[42] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022.

[43] 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. *arXiv preprint arXiv:2307.01952*, 2023.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[45] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2021.

[46] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[47] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *CVPR*, 2021.

[48] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*, 2015.

[49] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[50] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

[51] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

[52] Nishant Subramani, Nivedita Suresh, and Matthew Peters. Extracting Latent Steering Vectors from Pretrained Language Models. In *ACL Findings*, 2022.

[53] Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. Zerocap: Zero-shot image-to-text generation for visual-semantic arithmetic. In *CVPR*, 2022.

[54] Matthew Trager, Pramuditha Perera, Luca Zancato, Alessandro Achille, Parminder Bhatia, and Stefano Soatto. Linear spaces of meanings: compositional structures in vision-language models. In *ICCV*, 2023.

[55] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *CVPR*, 2022.

[56] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, 2023.

[57] Alexander Matt Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248v3*, 2023.

[58] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. In *IEEE TIP*, 2004.

[59] Zihao Wang, Lin Gui, Jeffrey Negrea, and Victor Veitch. Concept algebra for (score-based) text-controlled generative models. In *NeurIPS*, 2023.

[60] Zhaoning Wang, Ming Li, and Chen Chen. Luciddreaming: Controllable object-centric 3d generation. *arXiv preprint arXiv:2312.00588*, 2023.

[61] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021.

[62] Chen Henry Wu and Fernando De la Torre. Unifying diffusion models' latent space, with applications to cyclediffusion and guidance. *arXiv preprint arXiv:2210.05559*, 2022.

[63] Xindi Wu, Dingli Yu, Yangsibo Huang, Olga Russakovsky, and Sanjeev Arora. Conceptmix: A compositional image generation benchmark with controllable difficulty. In *NeurIPS Datasets and Benchmarks Track*, 2024.

[64] Yongliang Wu, Shiji Zhou, Mingzhuo Yang, Lianzhe Wang, Heng Chang, Wenbo Zhu, Xinting Hu, Xiao Zhou, and Xu Yang. Unlearning concepts in diffusion model via concept domain correction and concept preserving gradient. In *AAAI*, 2025.

[65] Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with natural language. *arXiv preprint arXiv:2312.04965*, 2023.

[66] Chong You, Chun Guang Li, Daniel P Robinson, and Rene Vidal. Oracle Based Active Set Algorithm for Scalable Elastic Net Subspace Clustering. In *CVPR*, 2016.

[67] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

[68] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023.

[69] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[70] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017.

[71] Junhao Zhuang, Yanhong Zeng, Wenran Liu, Chun Yuan, and Kai Chen. A task is worth one word: Learning with task prompts for high-quality versatile image inpainting. In *ECCV*, 2024.

[72] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.