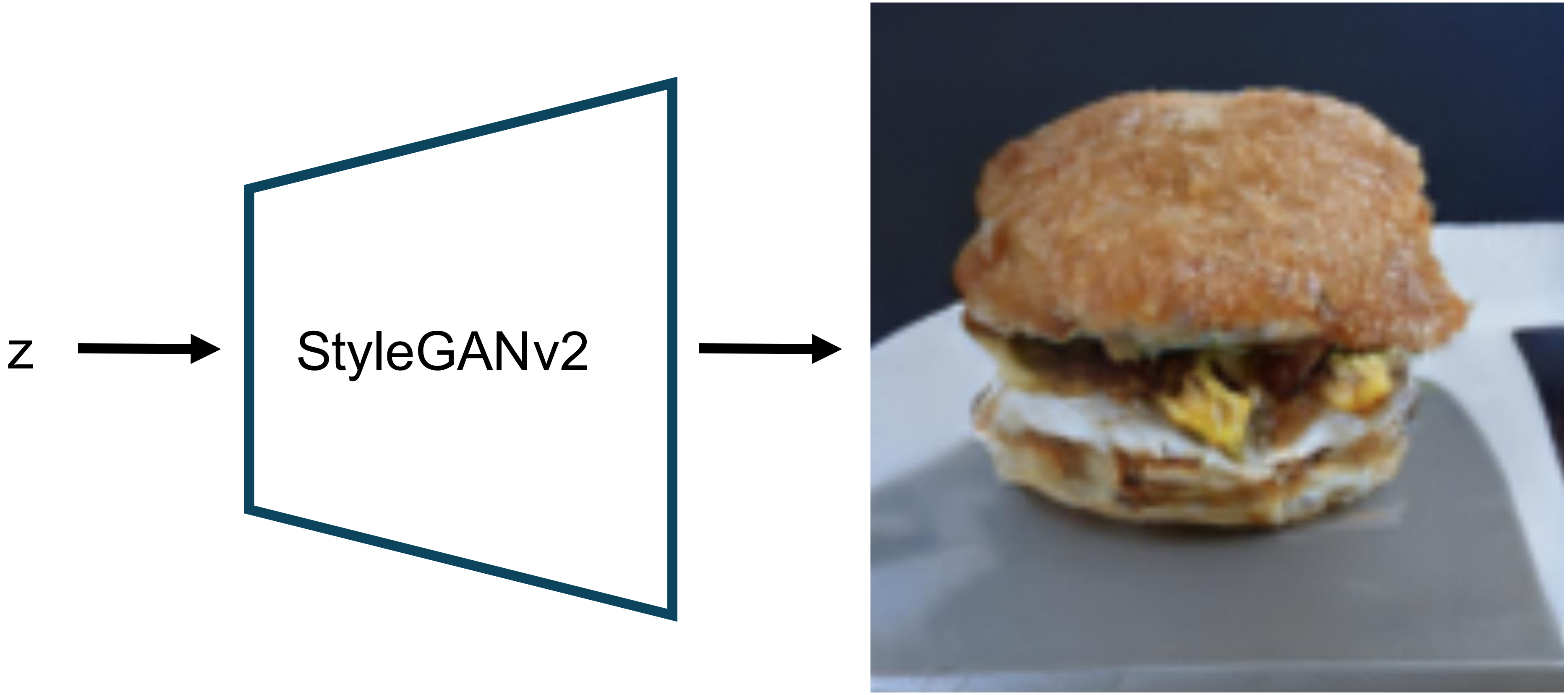
Multimodal tasks (=Vision and Language Tasks)

- Multimodal Classification
- Image-Text Retrieval
- Visual Grounding
- Image Captioning
- Visual Question Answering and Visual Reasoning
- **Text-to-image Generation**

Text-to-Image Generation (GANs)

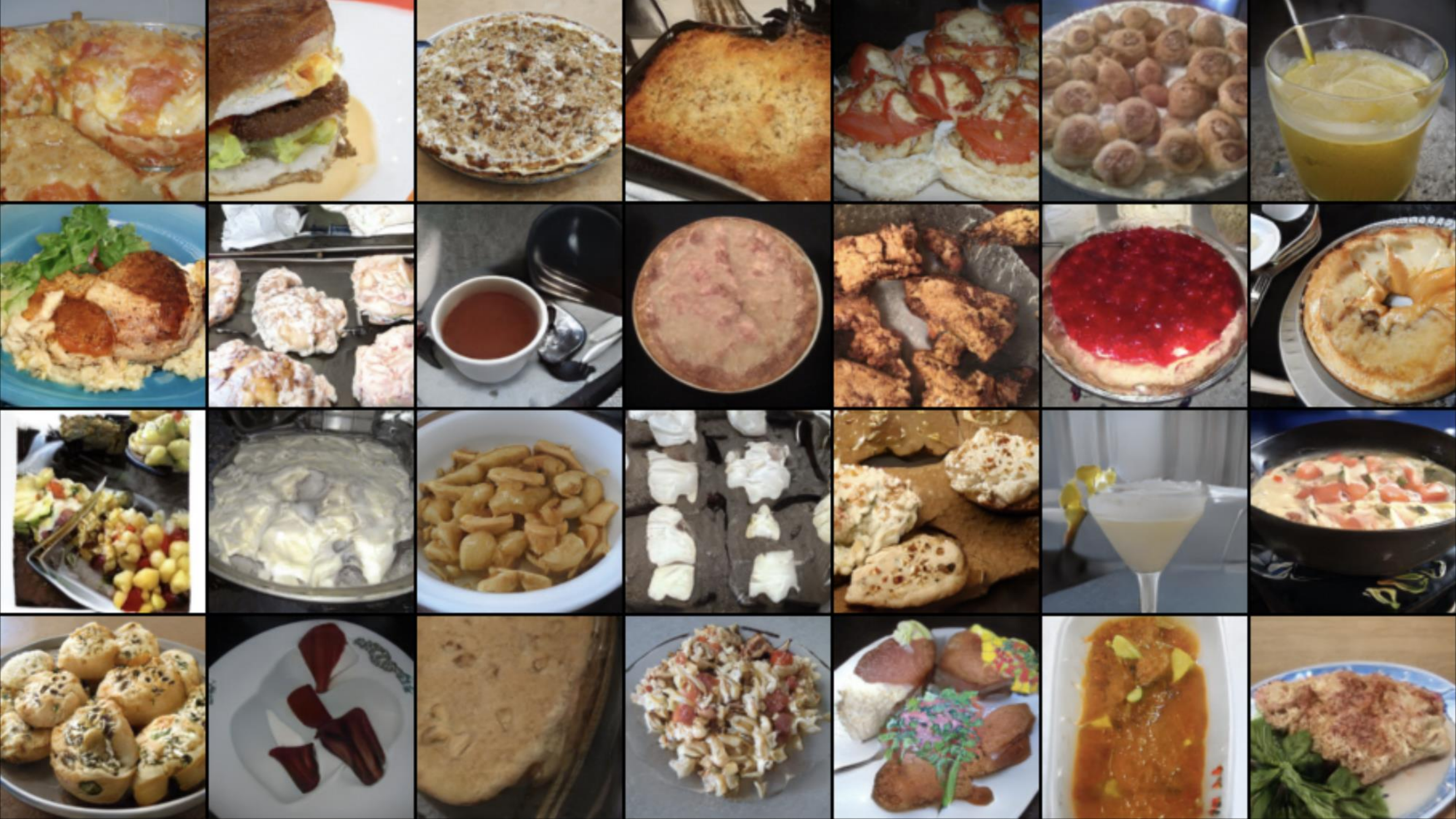
Text-to-Image Generation (GANs)



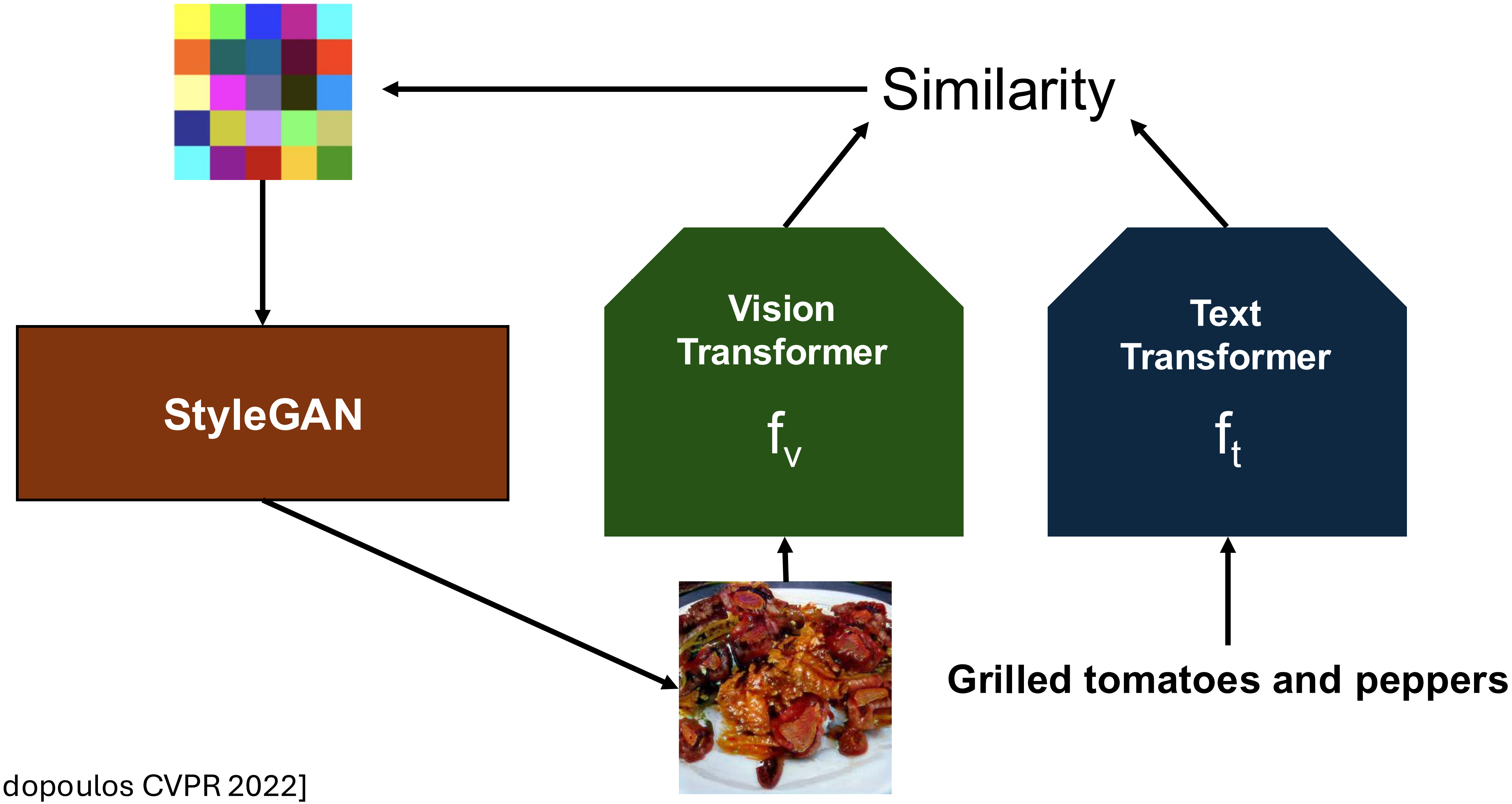
z

StyleGANv2

[Papadopoulos CVPR 2022]



StyleGAN + Vision/Text transformers



Similarity

StyleGAN

Vision Transformer

f_v

Text Transformer

f_t

Grilled tomatoes and peppers

[Papadopoulos CVPR 2022]

Chocolate chip cookies

Vanilla ice cream

Mixed green salad

Homemade chocolate bars

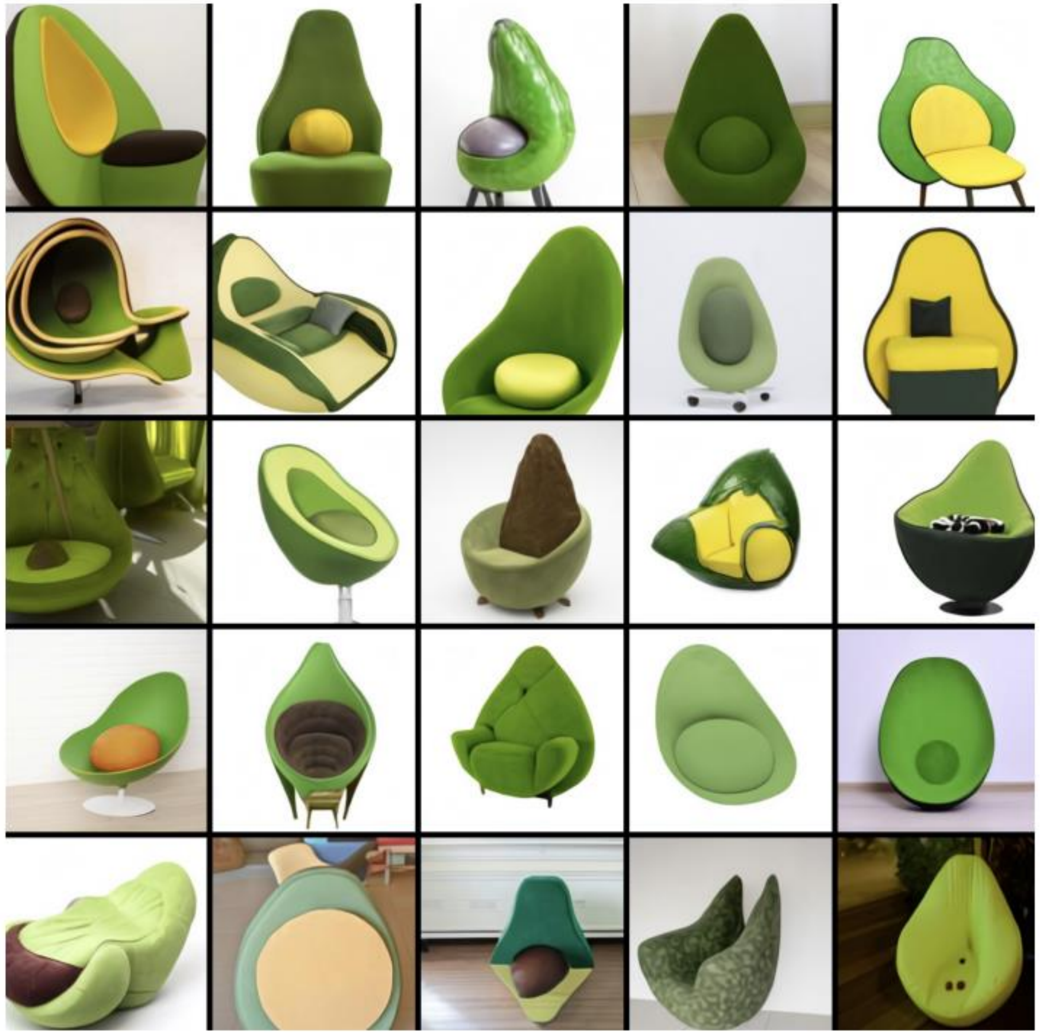
Grilled salmon

Pizza pepperoni

Text-to-Image Generation

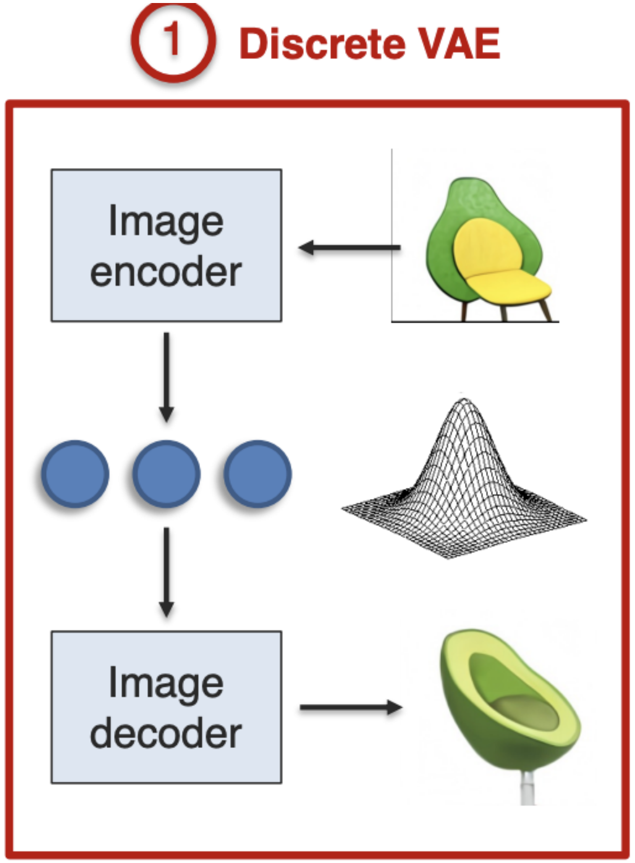
DALL·E: Text-to-image translation at scale

An armchair in the shape of an avocado →



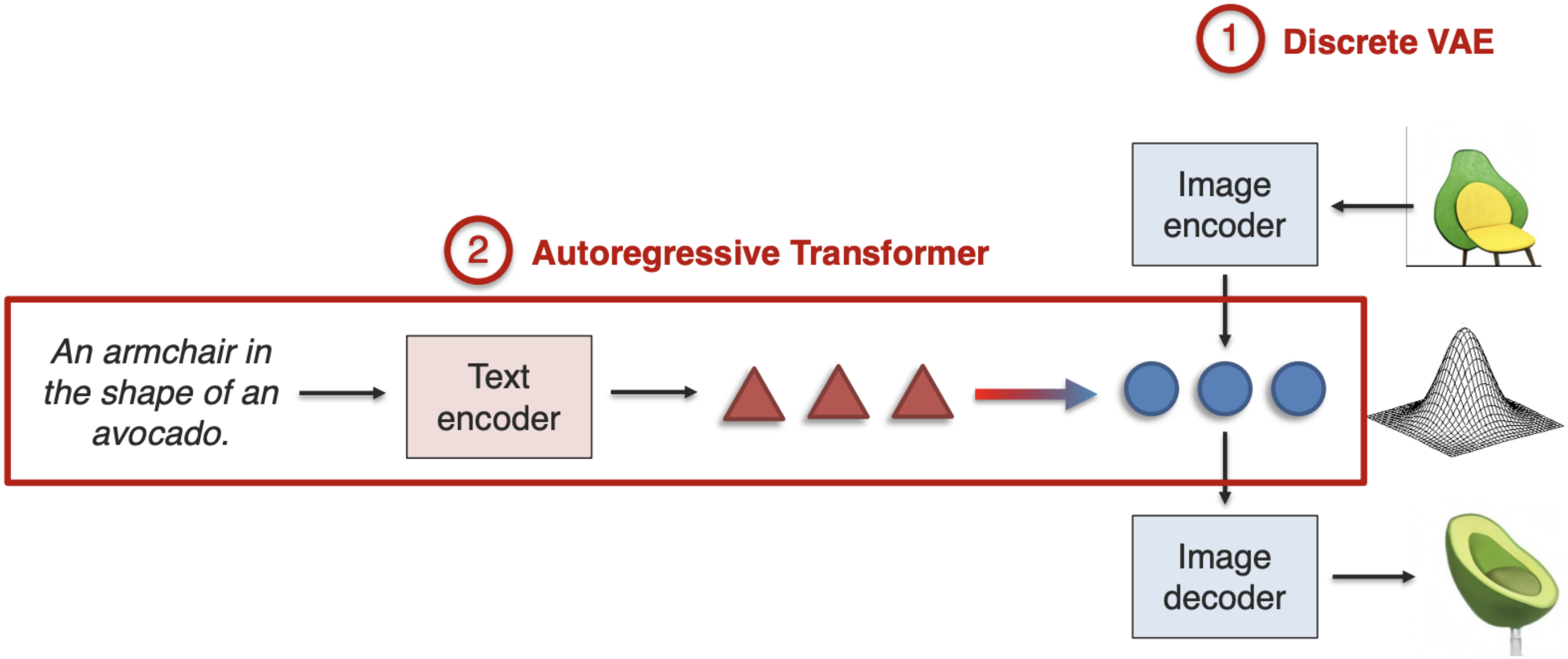
[Ramesh et al., Zero-Shot Text-to-Image Generation. ICML 2021]

DALL-E



[Ramesh et al., Zero-Shot Text-to-Image Generation. ICML 2021]

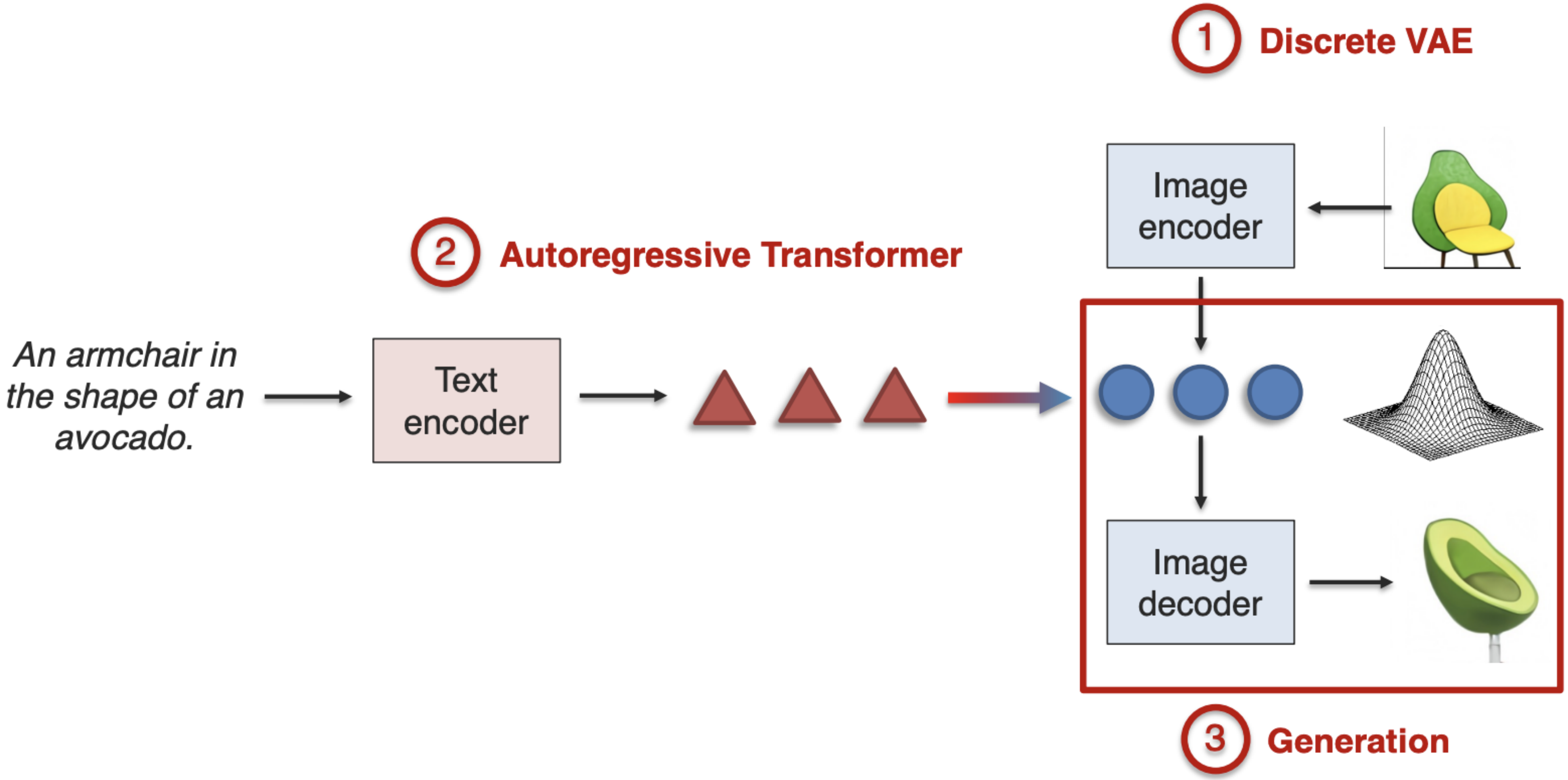
DALL-E



1 Discrete VAE
2 Autoregressive Transformer
An armchair in the shape of an avocado.
Text encoder
Image encoder
Image decoder

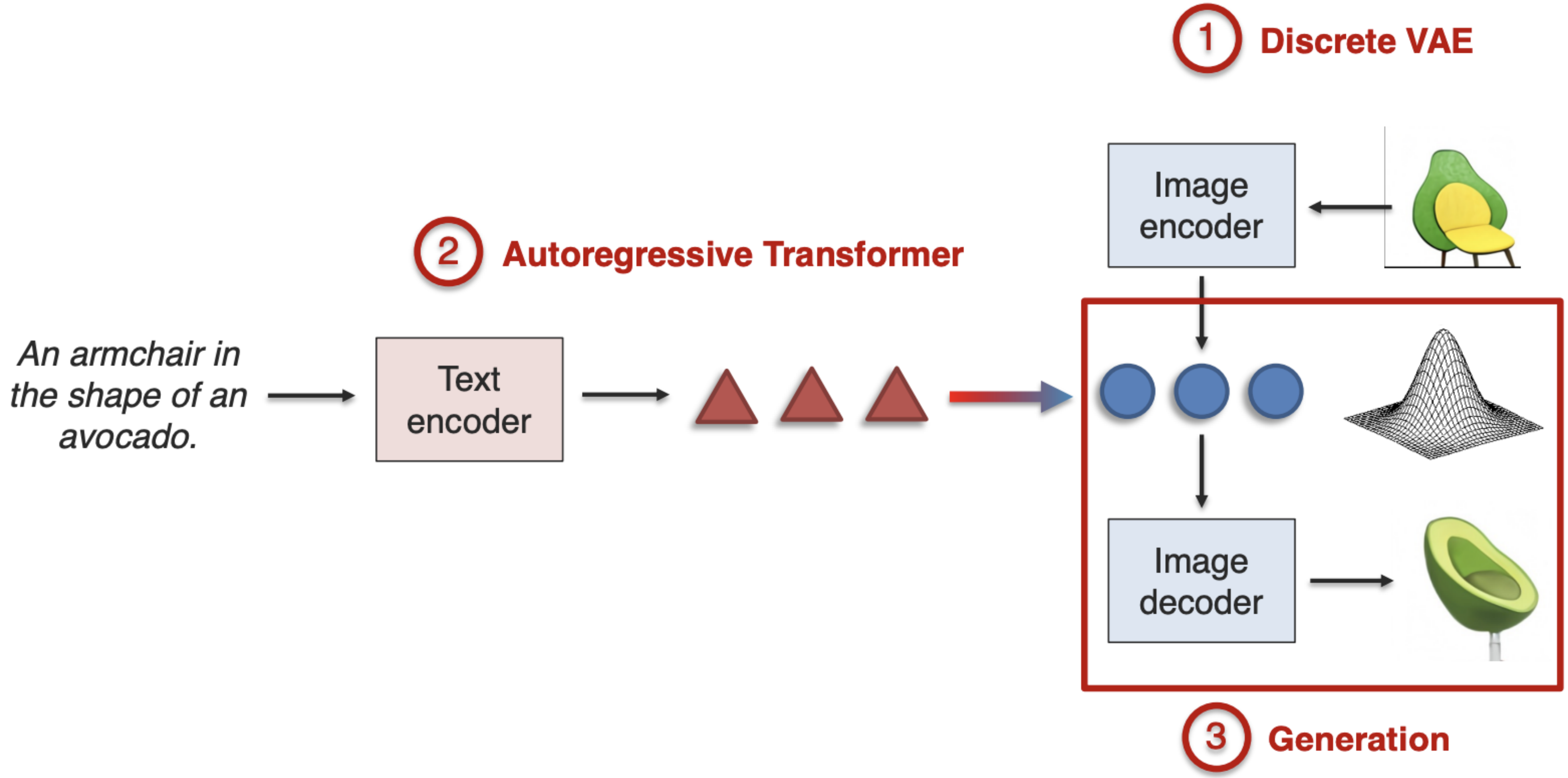
[Ramesh et al., Zero-Shot Text-to-Image Generation. ICML 2021]

DALL-E


1 Discrete VAE
Image encoder
2 Autoregressive Transformer
An armchair in the shape of an avocado.
Text encoder
Image decoder
3 Generation

[Ramesh et al., Zero-Shot Text-to-Image Generation. ICML 2021]

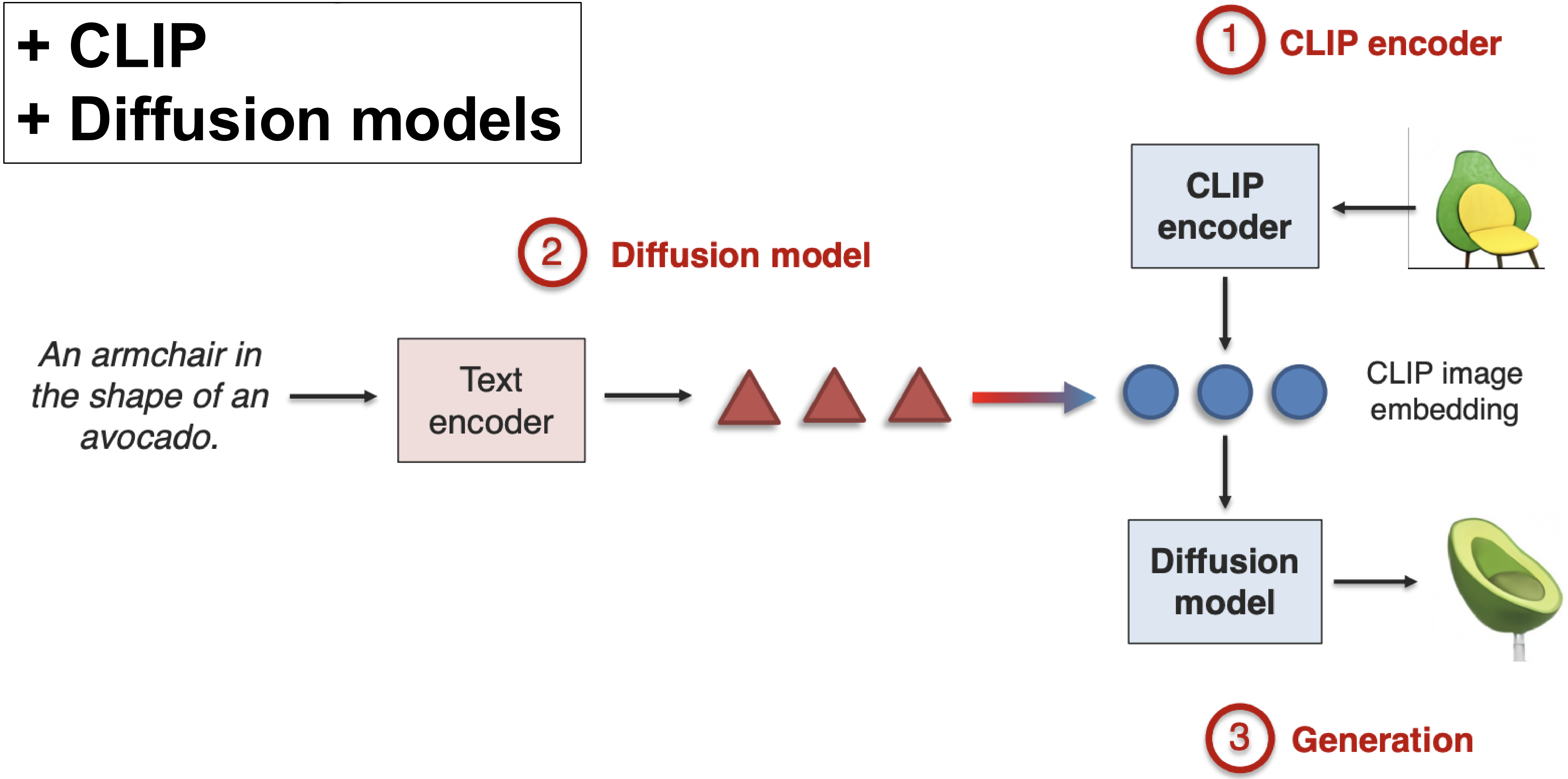
DALL-E


1 Discrete VAE
Image encoder
2 Autoregressive Transformer
An armchair in the shape of an avocado.
Text encoder
Image decoder
3 Generation

[Ramesh et al., Zero-Shot Text-to-Image Generation. ICML 2021]

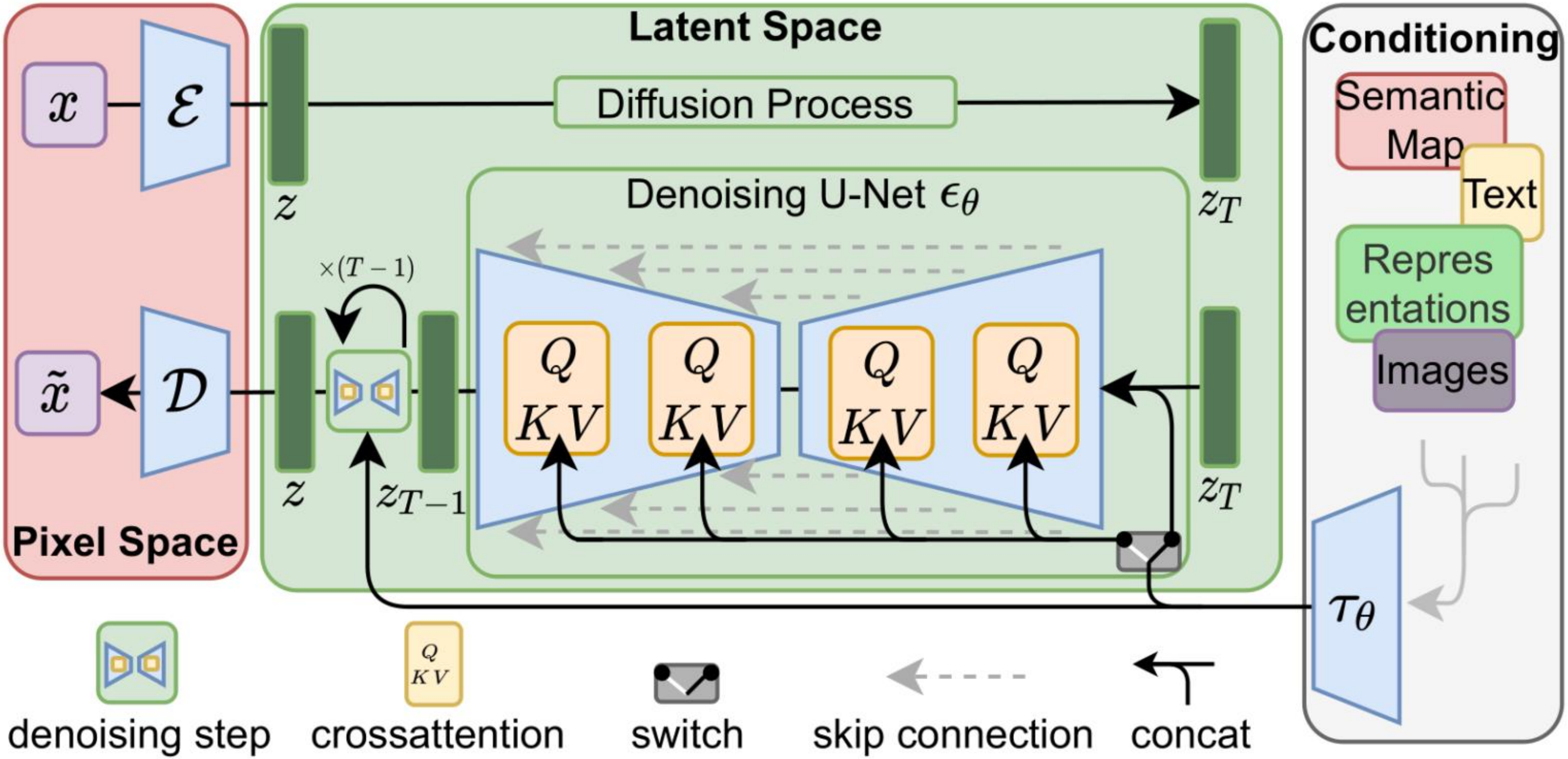
DALL-E2

+ CLIP
+ Diffusion models



[Ramesh et al., Hierarchical Text-Conditional Image Generation with CLIP Latents, arxiv 2022]

Stable Diffusion: Rombach CVPR 2022



Stable Diffusion

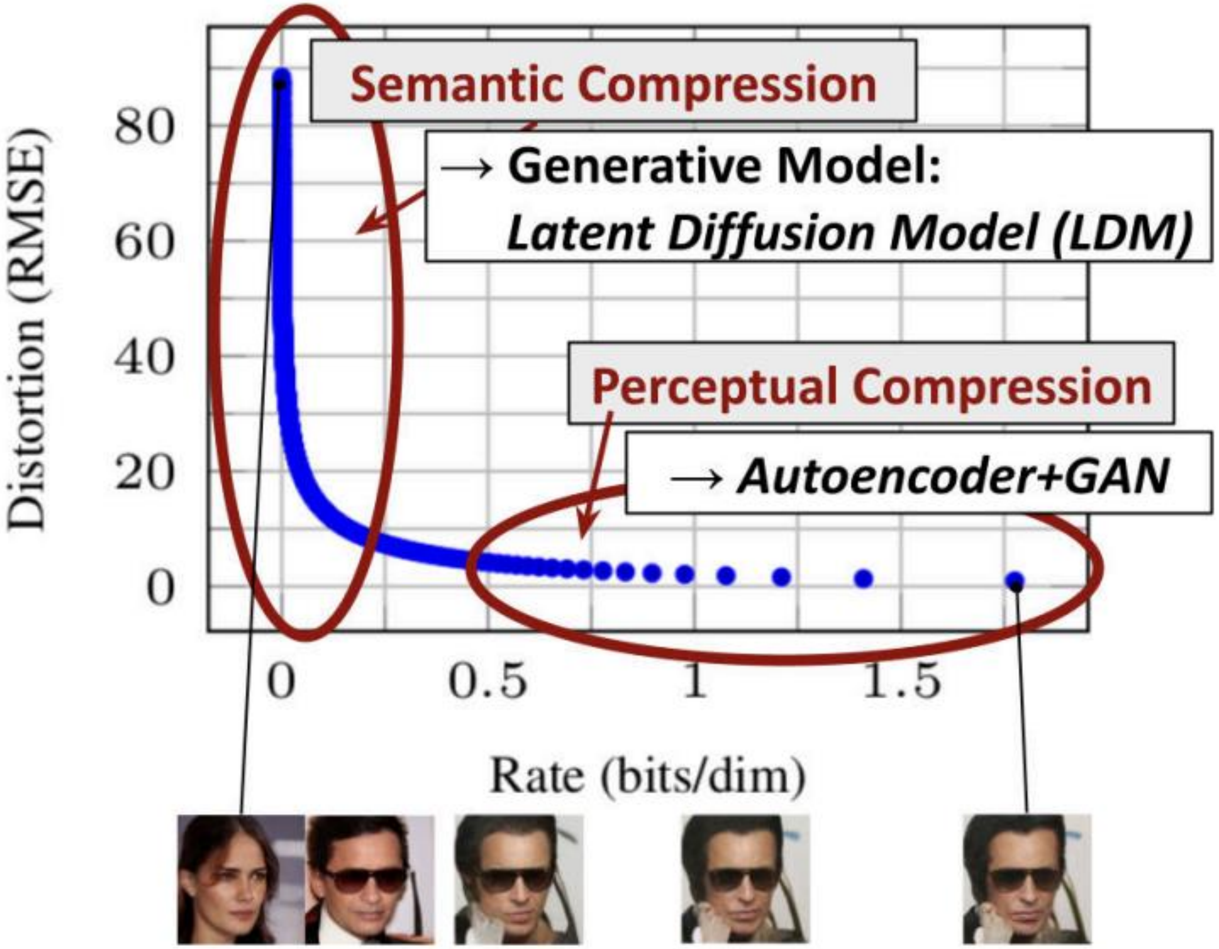
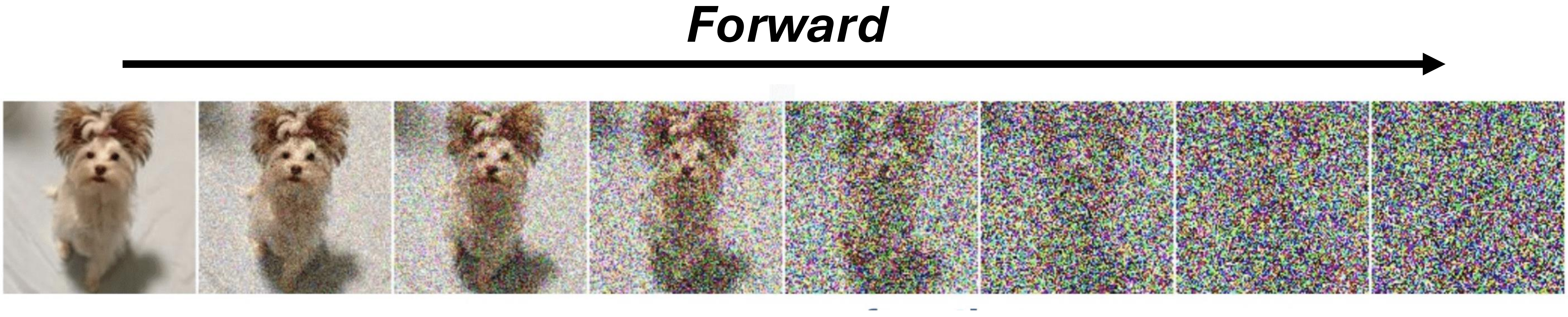


Figure 2. Illustrating perceptual and semantic compression: Most bits of a digital image correspond to imperceptible details. While DMs allow to suppress this semantically meaningless information by minimizing the responsible loss term, gradients (during training) and the neural network backbone (training and inference) still need to be evaluated on all pixels, leading to superfluous computations and unnecessarily expensive optimization and inference. We propose *latent diffusion models (LDMs)* as an effective generative model and a separate mild compression stage that only eliminates imperceptible details. Data and images from [29].

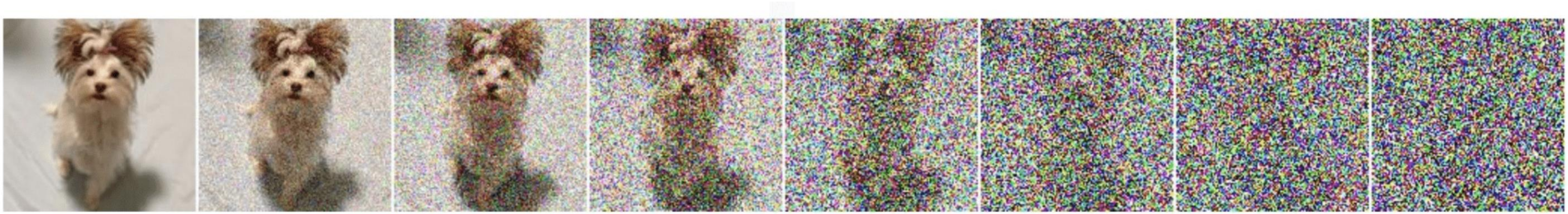
Diffusion to Latent space



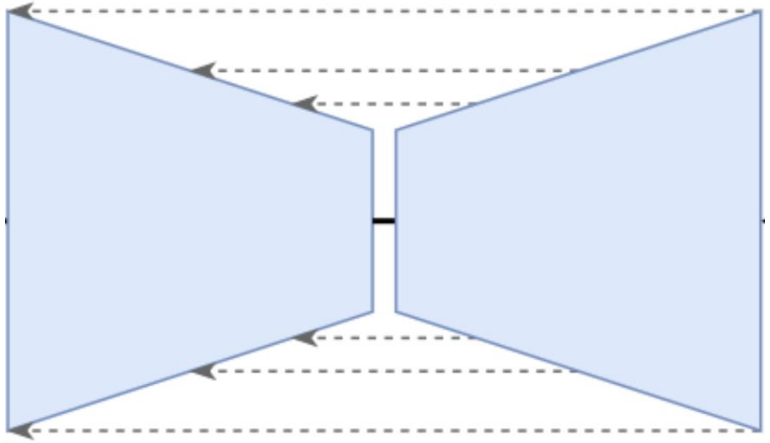
Forward

Diffusion to Latent space

Forward

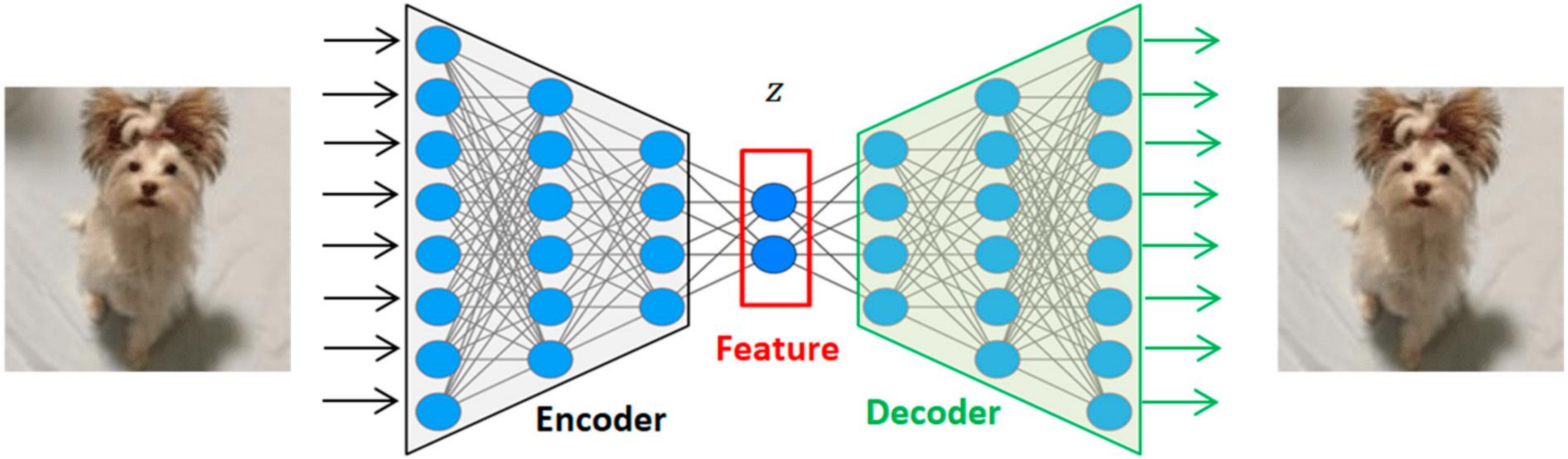


Reverse

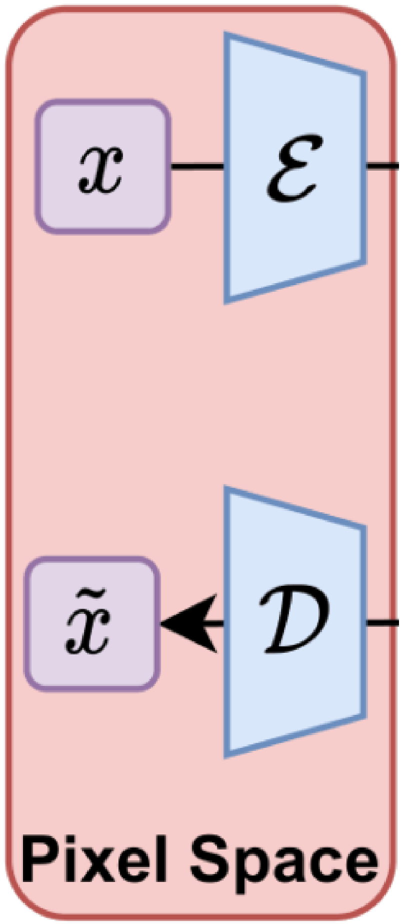


Denoising UNet

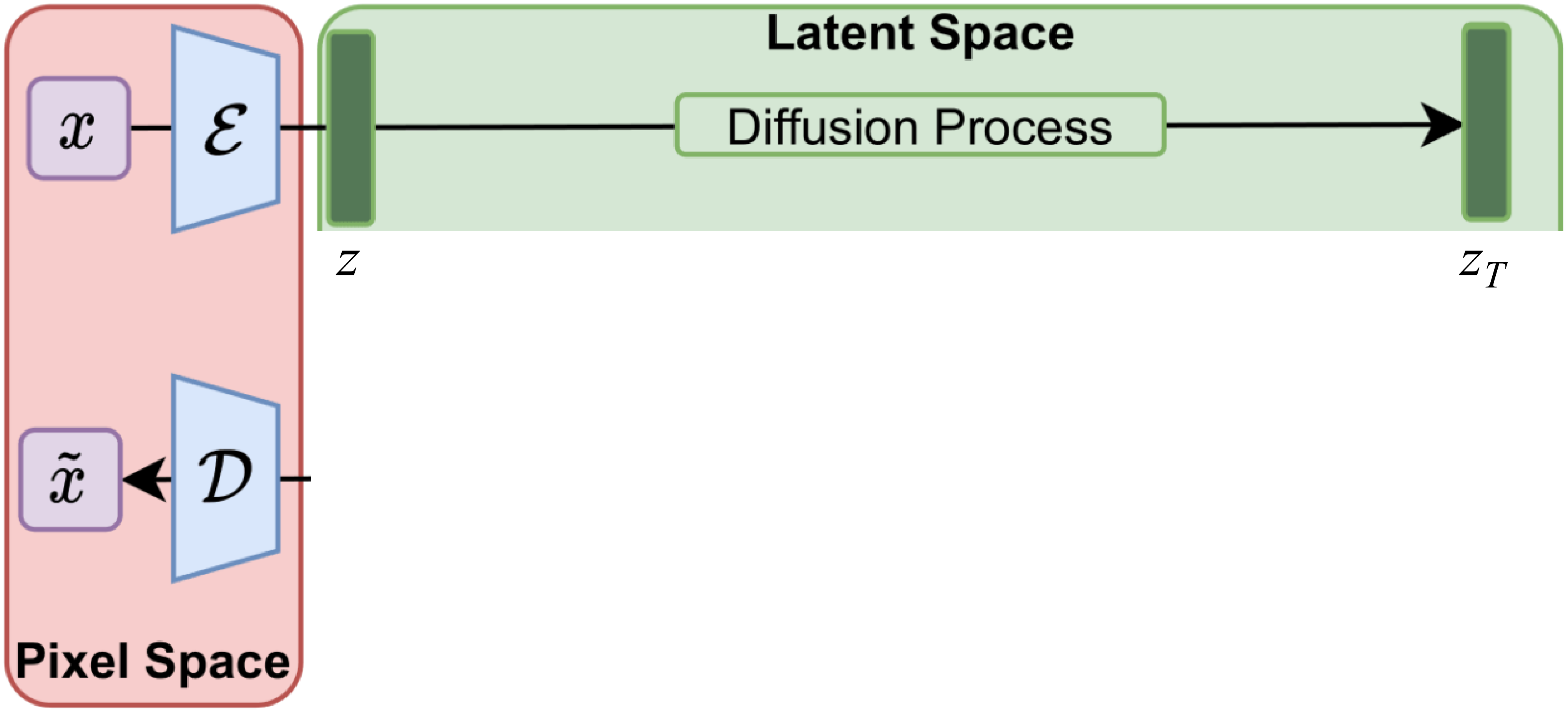
(1) Train an AutoEncoder (AE, VAE, VQVAE, VGGAN)



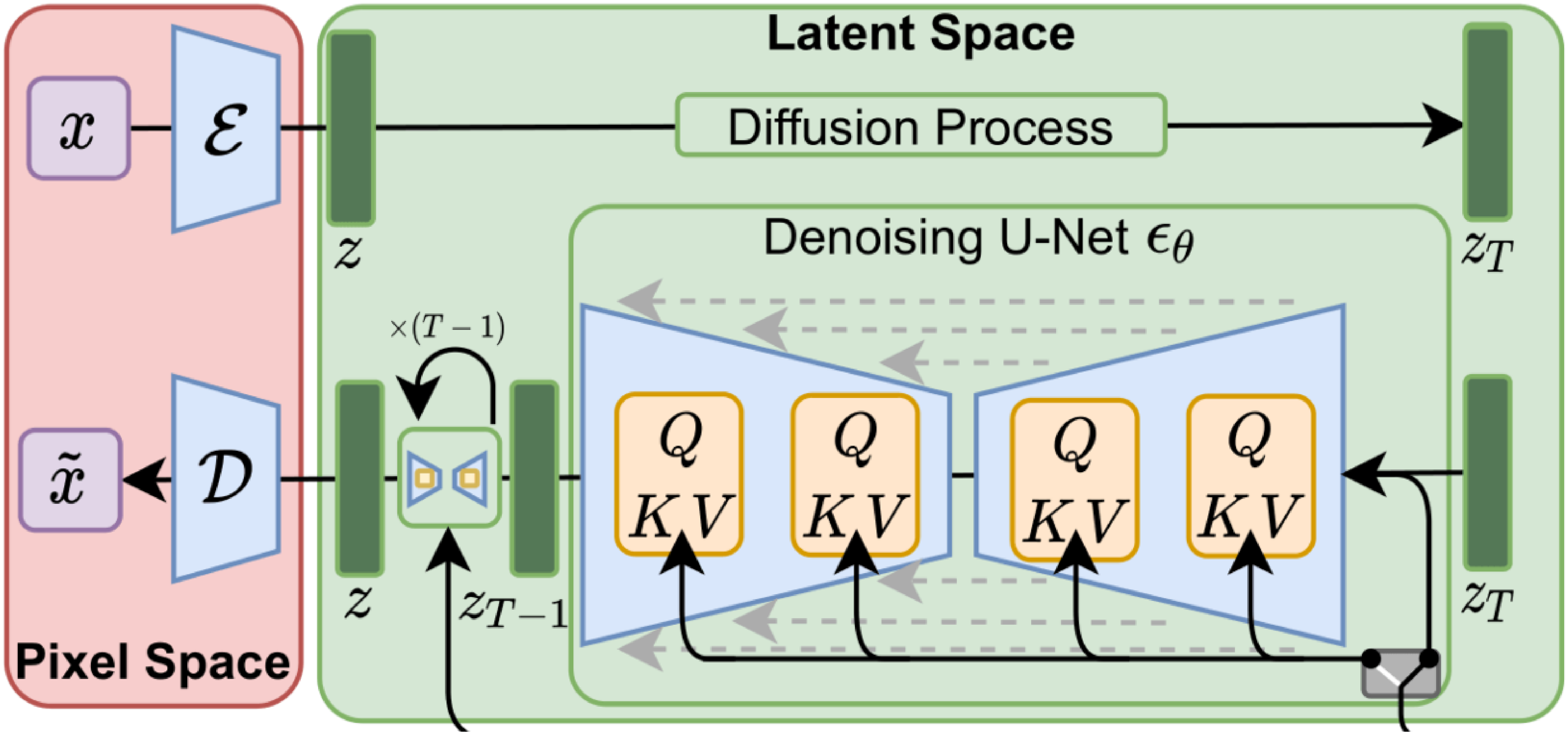
AutoEncoder (AE, VAE, VQVAE, VGGAN)



Latent Diffusion



Latent Diffusion



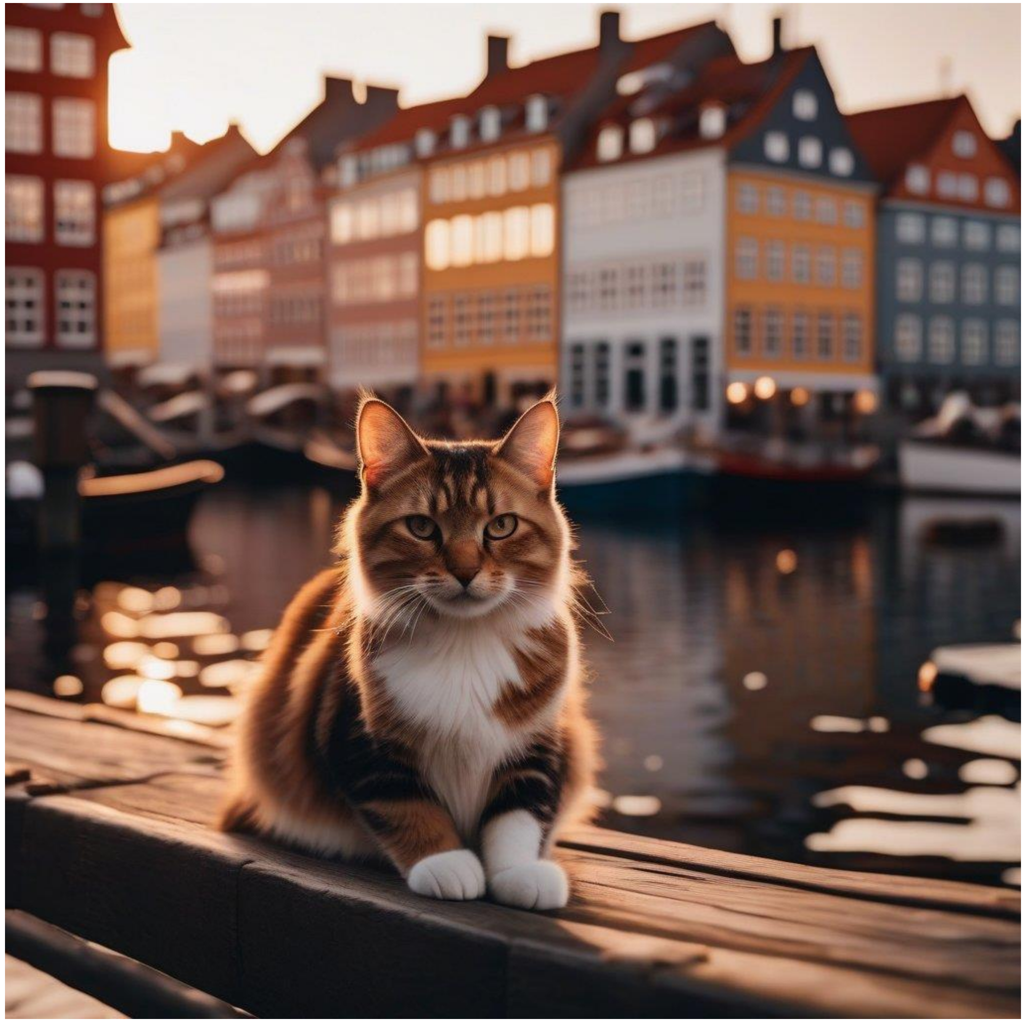
CG/CFG is cool, but I want freedom

CG/CFG is cool, but I want freedom

Text prompts to Image Generation

CG/CFG is cool, but I want freedom

Text prompts to Image Generation



cat in Nyhavn!
Sunny day/
Sunset

CG/CFG is cool, but I want freedom

Text prompts to Image Generation

dog in Norrebro streets!
Sunny day/
Sunset

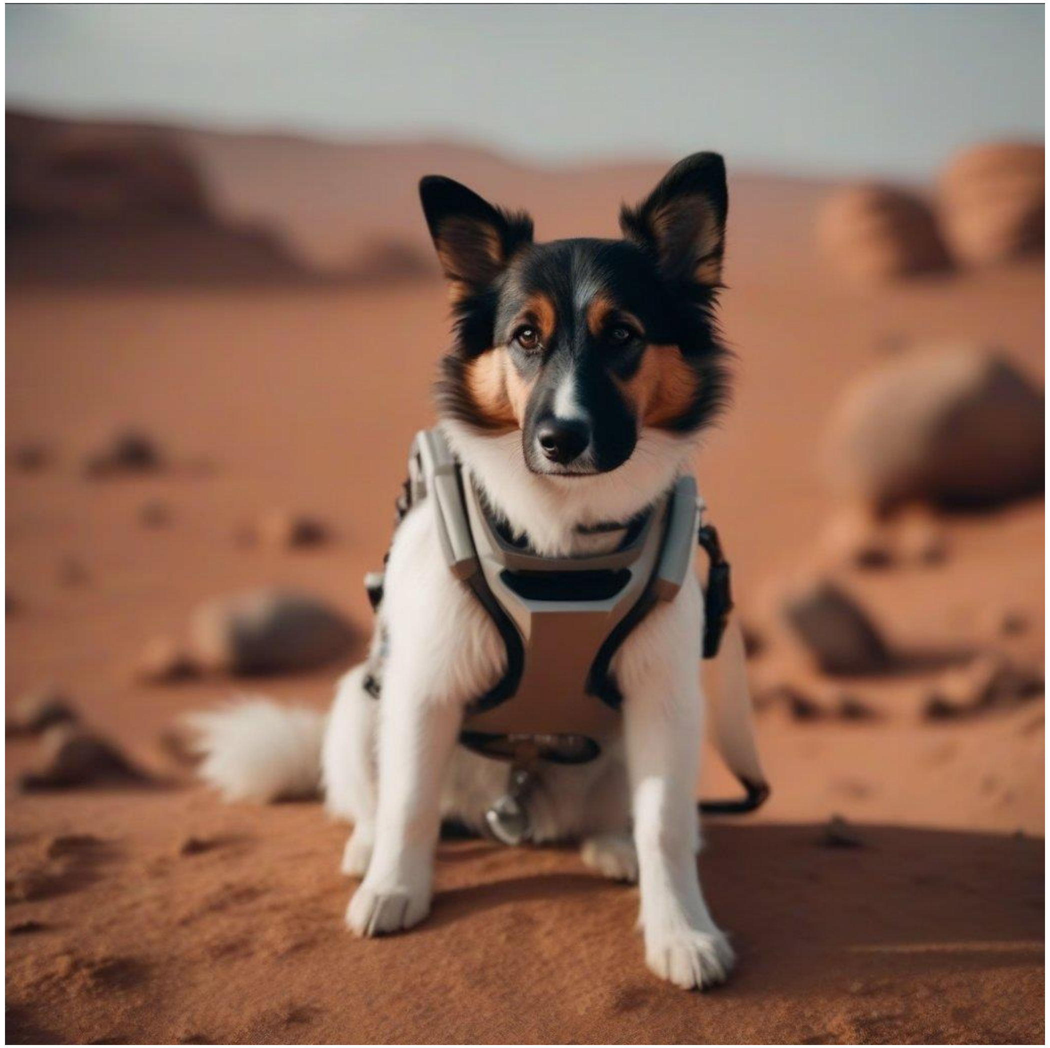
CG/CFG is cool, but I want freedom

Text prompts to Image Generation

Winter! Let it snow ☺

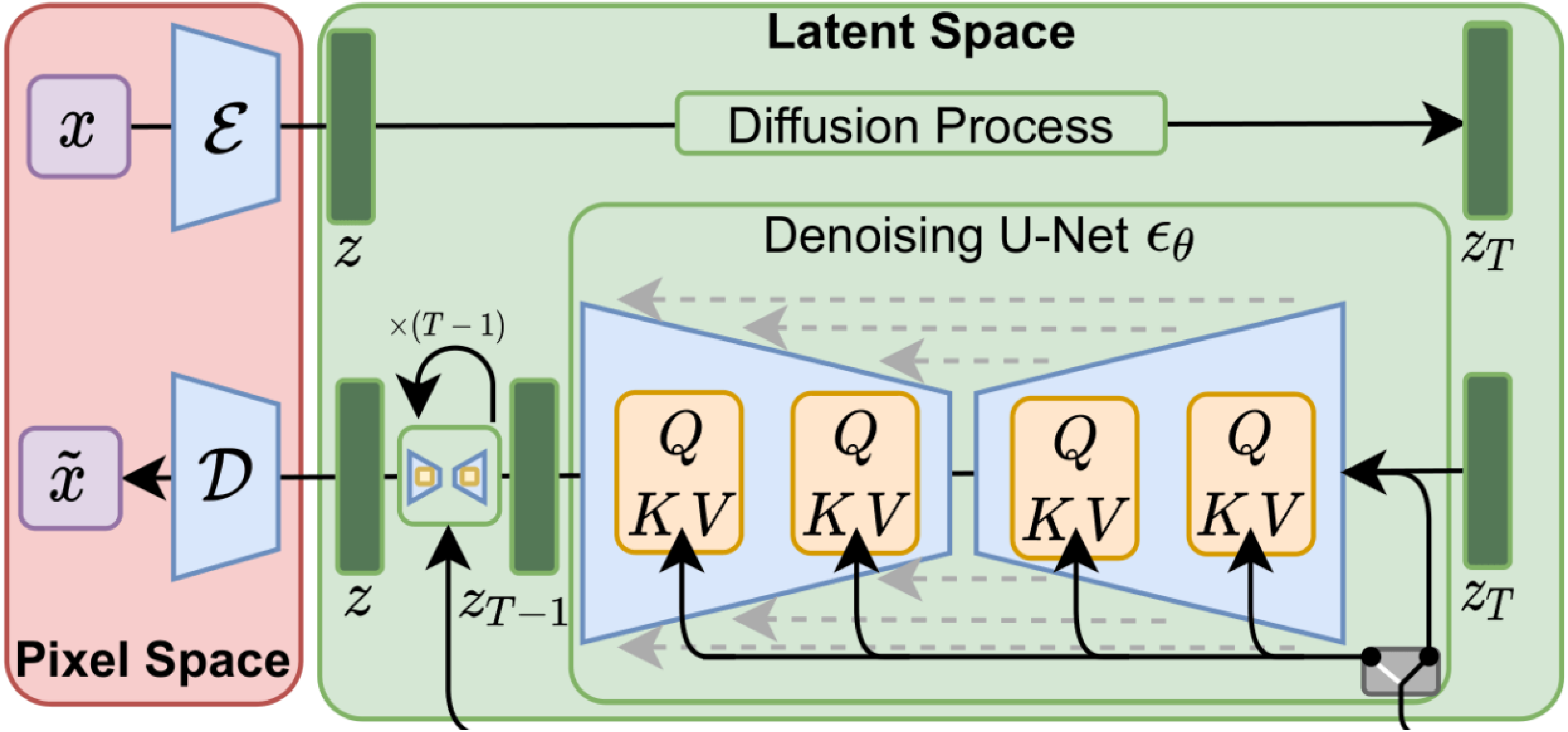
CG/CFG is cool, but I want freedom

Text prompts to Image Generation

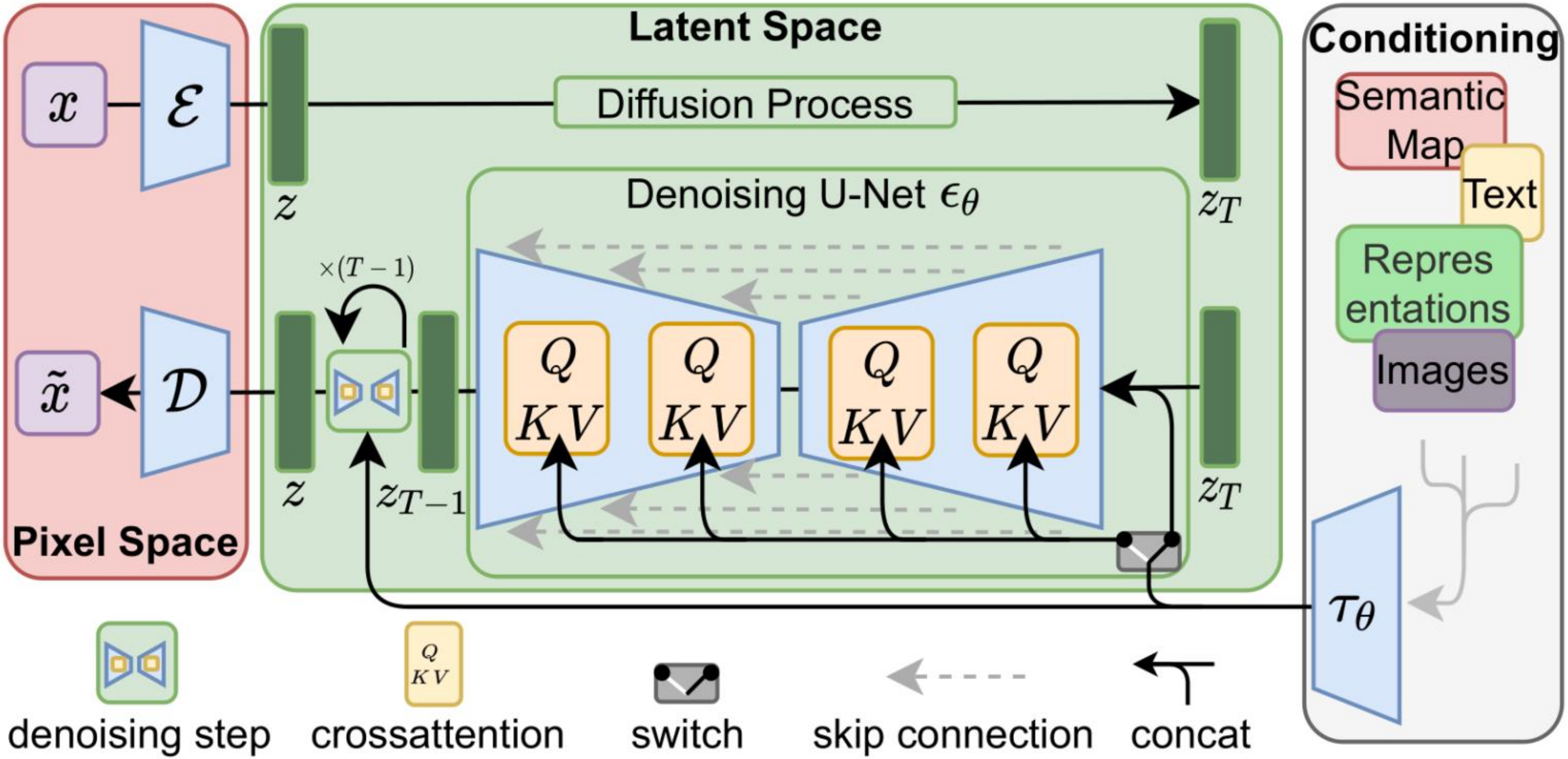


A dog on Mars!! 🤯

Latent Diffusion

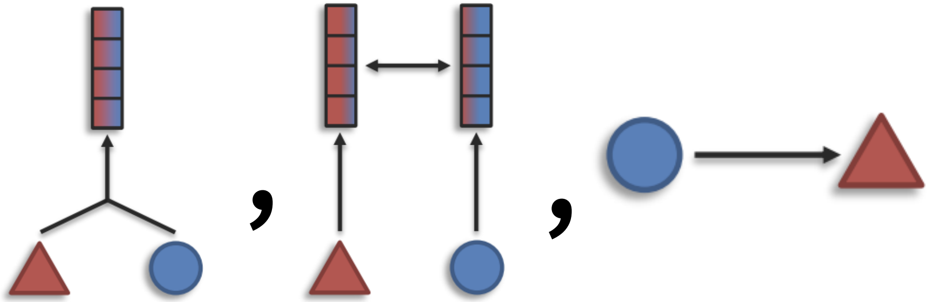


Stable Diffusion: Rombach CVPR 2022



Research work on Multimodal Learning

Cooking Programs
Papadopoulos, Mora, Chepurko, Huang, Ofli, Torralba
CVPR 2022



Precise Image Editing
Schouten, Kaya, Belongie, Papadopoulos
CVPR-W 2025, SCIA 2025

Test-time Scaling for Image Generation
Riise, Kaya, Papadopoulos
Work-in-progress 2025

Test-time Augmentation for MLLMs
Kaya, Elliott, Papadopoulos
Work-in-progress 2025

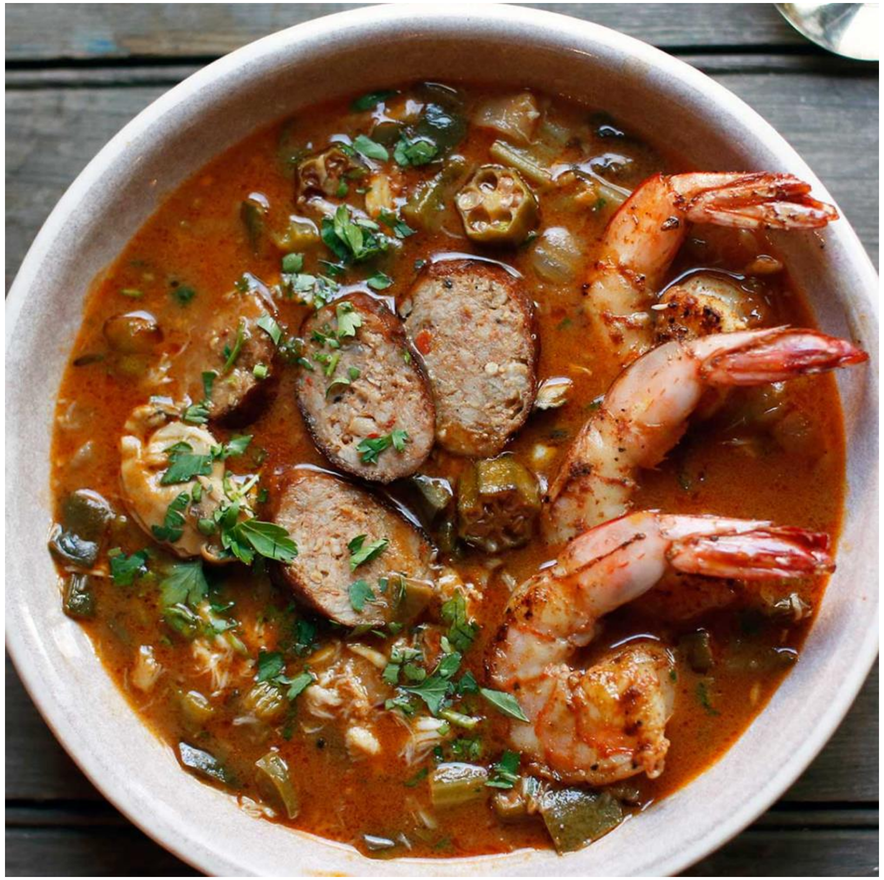
CVPR JUNE 19-24 2022
NEW ORLEANS • LOUISIANA

Learning Program Representations for Food Images and Cooking Recipes

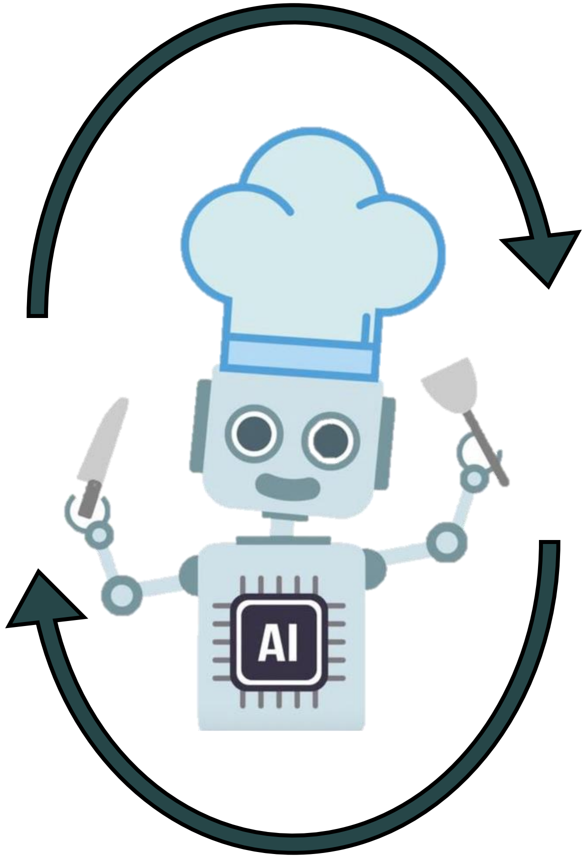
Dim P. Papadopoulos, Enrique Mora, Nadiia Chepurko,
Kuan Wei Huang, Ferda Ofli, Antonio Torralba

Food understanding

New Orleans style Gumbo



look & cook



read & cook

Instructions

Mix the butter and the flour in a large pot and cook for 20 minutes on medium high heat. Stir in the green pepper, onion and celery and cook for 5 minutes. Add garlic and Cajun seasoning. Slowly add stock and simmer gently for 30 minutes. Stir in the sausages. Add the shrimp and simmer for another 5 minutes. Add black pepper and salt to taste. Serve!

Learning programs from food images and recipes

Linguine with Peppers and Sausages

Cook pasta in a large pot of boiling salted water until al dente. Saute sausages in a heavy skillet over medium high heat until light brown, breaking up clumps with back of spoon. Add peppers, onion, and garlic. Saute until tender. Add wine and simmer until liquid is slightly reduced, about 6 minutes. Drain pasta, and add to the skillet. Toss to combine. Serve.

```
h1 = Cook(pasta, tool=pot, time=until al dente)
h2 = Saute(sausages, tool=skillet, temp=medium heat,
           time=until light brown, how=breaking clumps)
h3 = Add(h2, peppers, onions, garlic)
h4 = Saute(h3, time=until tender)
h5 = Add(h4, wine)
h6 = Simmer(h5, time=6 minutes)
h7 = Drain(h1)
h8 = Add(h6, h7)
h9 = Toss(h8, why=to combine)
out = Serve(h9)
return out
```

Learning programs from food images and recipes

Linguine with Peppers and Sausages

Cook pasta in a large pot of boiling salted water until al dente. Saute sausages in a heavy skillet over medium high heat until light brown, breaking up clumps with back of spoon. Add peppers, onion, and garlic. Saute until tender. Add wine and simmer until liquid is slightly reduced, about 6 minutes. Drain pasta, and add to the skillet. Toss to combine. Serve.

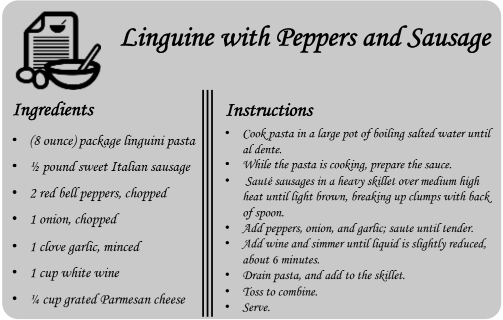
```
h1 = Cook(pasta, tool=pot, time=until al dente)
h2 = Saute(sausages, tool=skillet, temp=medium heat
out = Serve(h5)

return out
```

Cooking actions ➡ Functions

Ingredients ➡ Input variables

Tools, Time, Temperature ➡ Arguments

Learning programs from food images and recipes



Linguine with Peppers and Sausage

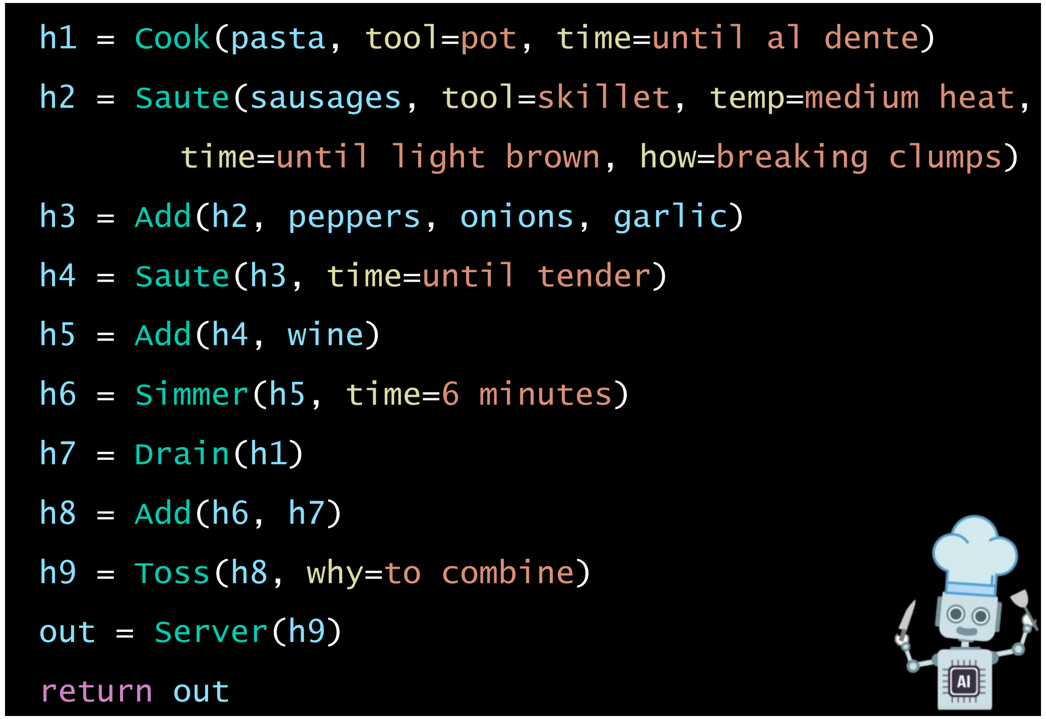
Ingredients

- *(8 ounce) package linguini pasta*
- *½ pound sweet Italian sausage*
- *2 red bell peppers, chopped*
- *1 onion, chopped*
- *1 clove garlic, minced*
- *1 cup white wine*
- *¼ cup grated Parmesan cheese*

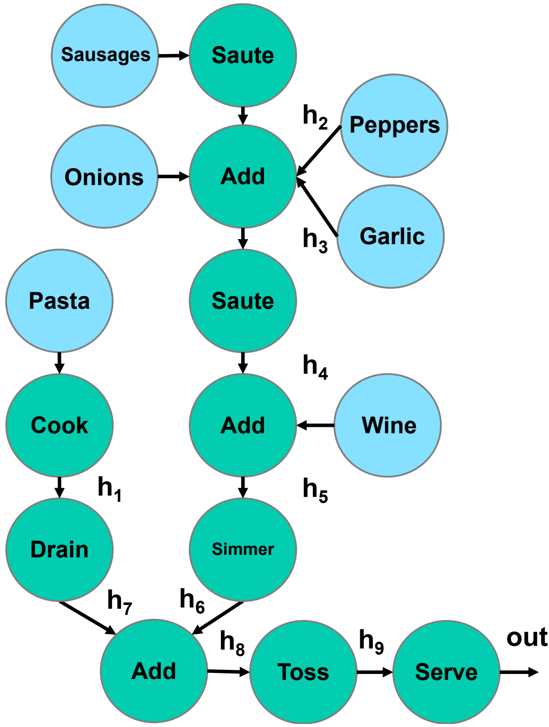
Instructions

- *Cook pasta in a large pot of boiling salted water until al dente.*
- *While the pasta is cooking, prepare the sauce.*
- *Sauté sausages in a heavy skillet over medium high heat until light brown, breaking up clumps with back of spoon.*
- *Add peppers, onion, and garlic; saute until tender.*
- *Add wine and simmer until liquid is slightly reduced, about 6 minutes.*
- *Drain pasta, and add to the skillet.*
- *Toss to combine.*
- *Serve.*

Cooking symbolic program and graph



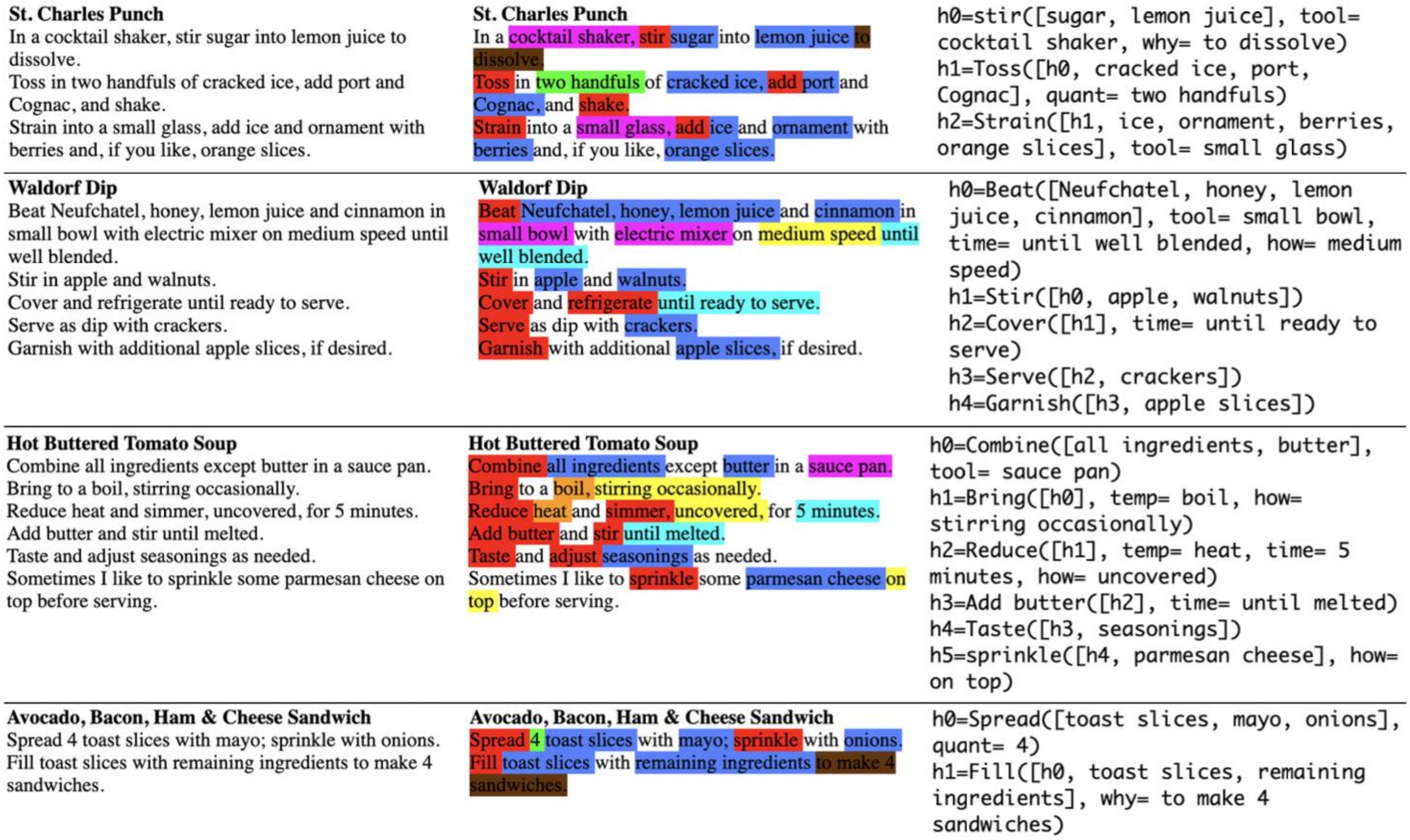
```
h1 = Cook(pasta, tool=pot, time=until al dente)
h2 = Saute(sausages, tool=skillet, temp=medium heat,
      time=until light brown, how=breaking clumps)
h3 = Add(h2, peppers, onions, garlic)
h4 = Saute(h3, time=until tender)
h5 = Add(h4, wine)
h6 = Simmer(h5, time=6 minutes)
h7 = Drain(h1)
h8 = Add(h6, h7)
h9 = Toss(h8, why=to combine)
out = Server(h9)
return out
```



Cooking Programs:

- ✓ non-ambiguous, structured representation
- ✓ capture cooking semantics
- ✓ can be easily manipulated by users
- ✓ can be potentially executed by agents

Crowdsourcing Cooking Programs



St. Charles Punch
In a cocktail shaker, stir sugar into lemon juice to dissolve.
Toss in two handfuls of cracked ice, add port and Cognac, and shake.
Strain into a small glass, add ice and ornament with berries and, if you like, orange slices.

St. Charles Punch
In a cocktail shaker, stir sugar into lemon juice to dissolve.
Toss in two handfuls of cracked ice, add port and Cognac, and shake.
Strain into a small glass, add ice and ornament with berries and, if you like, orange slices.

```
h0=stir([sugar, lemon juice], tool=
cocktail shaker, why= to dissolve)
h1=Toss([h0, cracked ice, port,
Cognac], quant= two handfuls)
h2=Strain([h1, ice, ornament, berries,
orange slices], tool= small glass)
```

Waldorf Dip
Beat Neufchatel, honey, lemon juice and cinnamon in small bowl with electric mixer on medium speed until well blended.
Stir in apple and walnuts.
Cover and refrigerate until ready to serve.
Serve as dip with crackers.
Garnish with additional apple slices, if desired.

Waldorf Dip
Beat Neufchatel, honey, lemon juice and cinnamon in small bowl with electric mixer on medium speed until well blended.
Stir in apple and walnuts.
Cover and refrigerate until ready to serve.
Serve as dip with crackers.
Garnish with additional apple slices, if desired.

```
h0=Beat([Neufchatel, honey, lemon
juice, cinnamon], tool= small bowl,
time= until well blended, how= medium
speed)
h1=Stir([h0, apple, walnuts])
h2=Cover([h1], time= until ready to
serve)
h3=Serve([h2, crackers])
h4=Garnish([h3, apple slices])
```

Hot Buttered Tomato Soup
Combine all ingredients except butter in a sauce pan.
Bring to a boil, stirring occasionally.
Reduce heat and simmer, uncovered, for 5 minutes.
Add butter and stir until melted.
Taste and adjust seasonings as needed.
Sometimes I like to sprinkle some parmesan cheese on top before serving.

Hot Buttered Tomato Soup
Combine all ingredients except butter in a sauce pan.
Bring to a boil, stirring occasionally.
Reduce heat and simmer, uncovered, for 5 minutes.
Add butter and stir until melted.
Taste and adjust seasonings as needed.
Sometimes I like to sprinkle some parmesan cheese on top before serving.

```
h0=Combine([all ingredients, butter],
tool= sauce pan)
h1=Bring([h0], temp= boil, how=
stirring occasionally)
h2=Reduce([h1], temp= heat, time= 5
minutes, how= uncovered)
h3=Add butter([h2], time= until melted)
h4=Taste([h3, seasonings])
h5=sprinkle([h4, parmesan cheese], how=
on top)
```

Avocado, Bacon, Ham & Cheese Sandwich
Spread 4 toast slices with mayo; sprinkle with onions.
Fill toast slices with remaining ingredients to make 4 sandwiches.

Avocado, Bacon, Ham & Cheese Sandwich
Spread 4 toast slices with mayo; sprinkle with onions.
Fill toast slices with remaining ingredients to make 4 sandwiches.

```
h0=Spread([toast slices, mayo, onions],
quant= 4)
h1=Fill([h0, toast slices, remaining
ingredients], why= to make 4
sandwiches)
```

amazon mechanical turk

- Recipe1M
- 3,708 programs
- 42,473 sentences

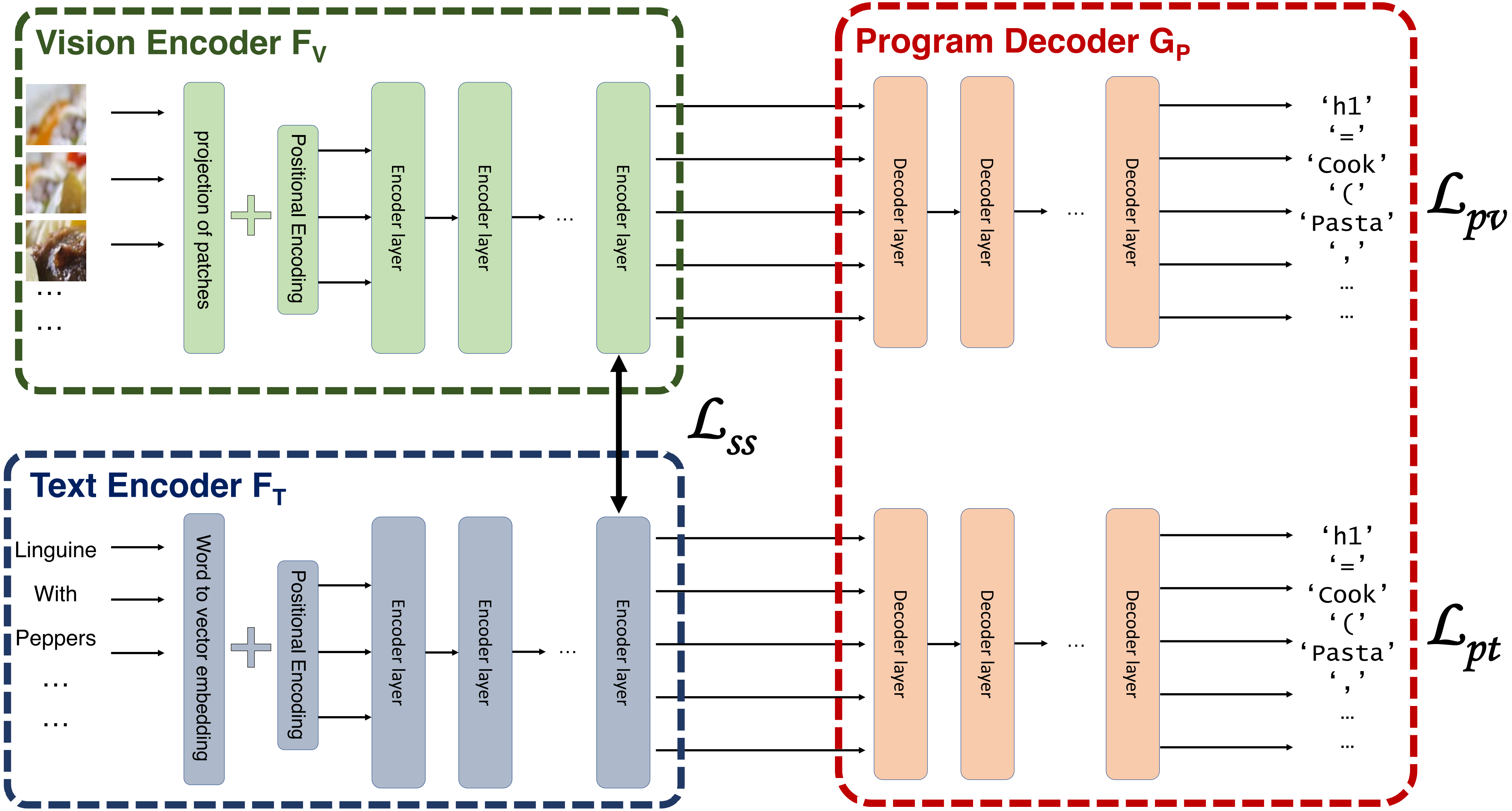
Learning programs from food images and recipes

Image

Text recipe

(Title, Ingredients, Instructions)

Linguine with Peppers and Sausage
(8 ounce) package linguini pasta
½ pound sweet Italian sausage
…
…
¼ cup grated Parmesan cheese
Cook pasta in a large pot.
Sauté sausages in a heavy skillet
over medium high heat
…
…
Serve.



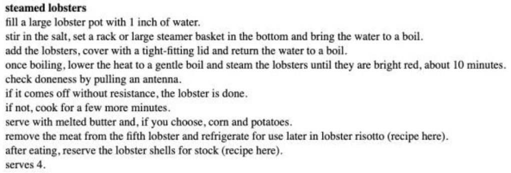
Experiments

- Experiments on **Recipe1M** [Salvador CVPR17]
- **Visual encoder**: ViT-B/16 [Dosovitskiy ICLR21]
- **Text encoder** and **Program decoder**: Transformer [Vaswani NeurIPS17]

(1) Image-to-recipe Retrieval Task

Query Image

Retrieved Recipe

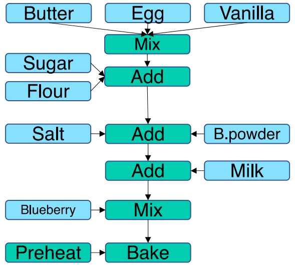


steamed lobsters
fill a large lobster pot with 1 inch of water.
stir in the salt, set a rack or large steamer basket in the bottom and bring the water to a boil.
add the lobsters, cover with a tight-fitting lid and return the water to a boil.
once boiling, lower the heat to a gentle boil and steam the lobsters until they are bright red, about 10 minutes.
check doneness by pulling an antenna.
if it comes off without resistance, the lobster is done.
if not, cook for a few more minutes.
serve with melted butter and, if you choose, corn and potatoes.
remove the meat from the fifth lobster and refrigerate for use later in lobster risotto (recipe here).
after eating, reserve the lobster shells for stock (recipe here).
serves 4.

(2) Program Generation Task

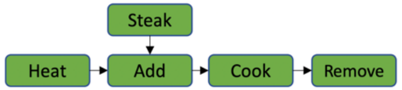
Image

Predicted from Image

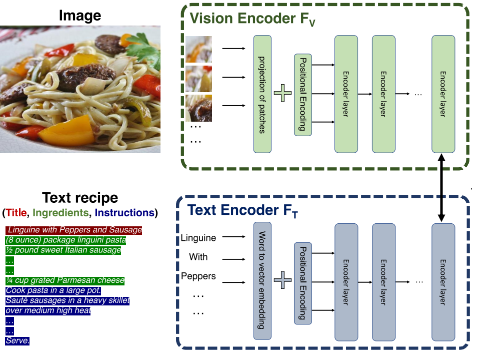


(3) Food Generation

Input program



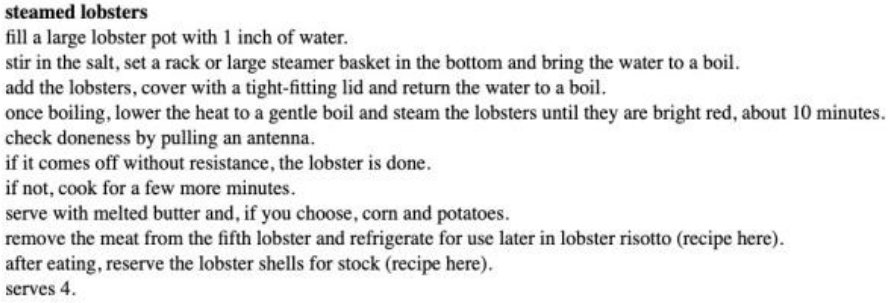
Generated image



(A) Image-to-recipe Retrieval Task

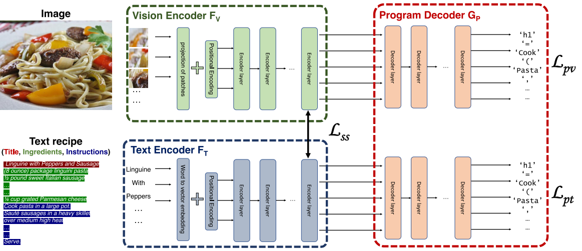
Query Image

Retrieved Recipe



steamed lobsters
fill a large lobster pot with 1 inch of water.
stir in the salt, set a rack or large steamer basket in the bottom and bring the water to a boil.
add the lobsters, cover with a tight-fitting lid and return the water to a boil.
once boiling, lower the heat to a gentle boil and steam the lobsters until they are bright red, about 10 minutes.
check doneness by pulling an antenna.
if it comes off without resistance, the lobster is done.
if not, cook for a few more minutes.
serve with melted butter and, if you choose, corn and potatoes.
remove the meat from the fifth lobster and refrigerate for use later in lobster risotto (recipe here).
after eating, reserve the lobster shells for stock (recipe here).
serves 4.

Method	medR	Recall@1	Recall@5	Recall@10
Salvador CVPR 17	5.2	24.0	51.0	65.0
Chen ACM MM 18	4.6	25.6	53.7	66.9
Zhu CVPR 19	2.0	39.1	71.0	81.7
Fu CVPR 20	2.0	48.2	75.8	83.6
Wang CVPR 19	1.0	51.8	80.2	87.5
Fain arXiv 19	1.0	55.9	82.4	88.7
Salvador CVPR	1.0	63.2	88.3	93.1



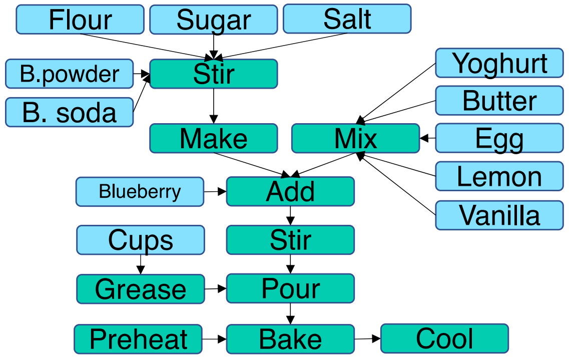
(B) Program Generation Task

Image

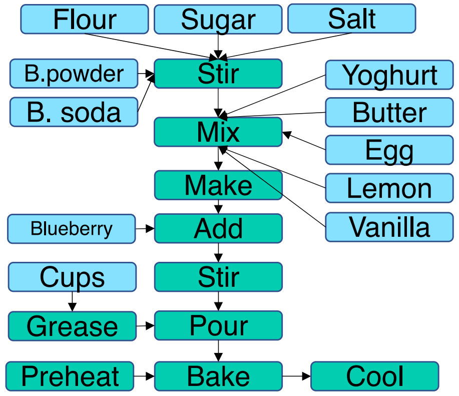
Recipe

Title: lemon blueberry muffins
Ingredients: 2 cups flour, 2/3 cup sugar, 1 teaspoon baking powder, 1 teaspoon baking soda, 1/2 teaspoon salt, … …
Instructions: Preheat oven to 400f (200c). grease muffin cups. stir together flour, 2/3 cup sugar, baking powder, baking soda and salt. separately mix yoghurt, butter, egg, lemon zest and vanilla extract until blended. make a well in the centre of the dry ingredients, add yoghurt mixture and blueberries and stir to combine. pour into muffin cups. bake 20-25 minutes. cool for 5 minutes before eating.

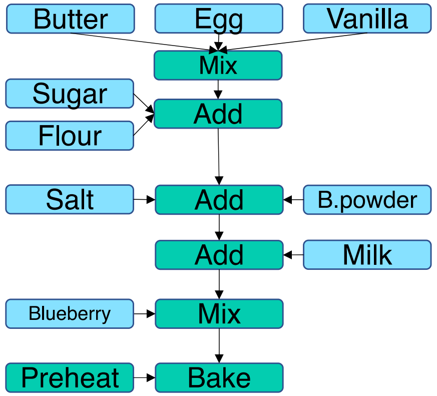
GT



Predicted from Recipe



Predicted from Image

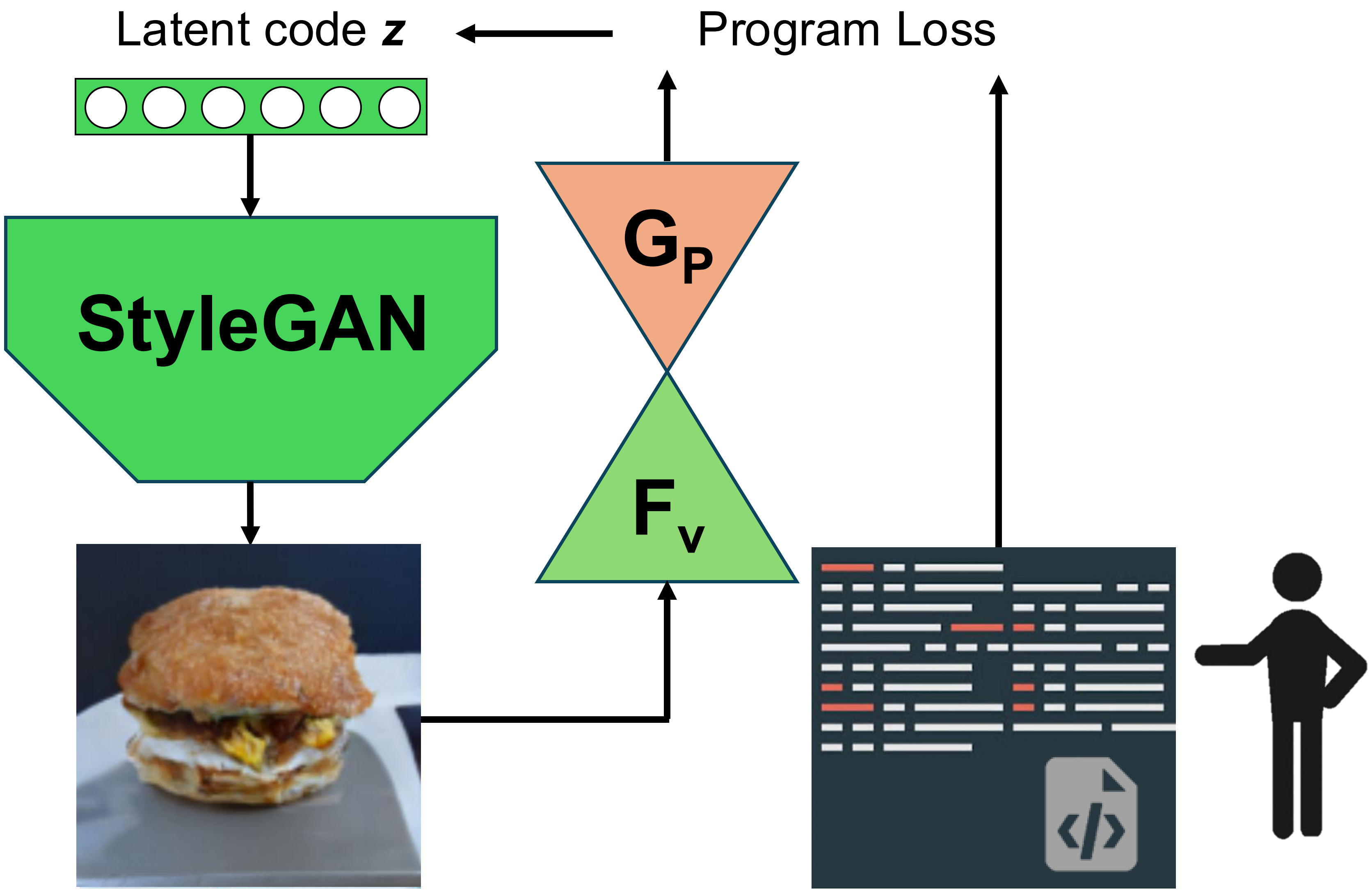


Input: Cooking recipes	Ingredients (F1 ↑)	Actions (F1 ↑)	Tools (F1 ↑)	Full graph* (GED ↓)	Input: Food images	Ingredients (F1 ↑)	Actions (F1 ↑)	Tools (F1 ↑)	Full graph* (GED ↓)
Random recipe	12.4	14.5	14.2	101.5	Random image	12.6	14.6	14.2	102.1
Retrieved recipe	43.4	55.2	74.2	67.1	**Retrieved image**	39.4	51.6	66.9	79.1
Instructions	41.6	49.3	66.6	–	**Instructions**	28.5	38.3	50.5	–
Programs (minCE)	75.5	83.1	84.1	16.8	**Programs (minCE)**	53.5	64.7	78.1	67.2

✓ **Much better results (+10-28% F1 score)** than the top-retrieved image/recipe

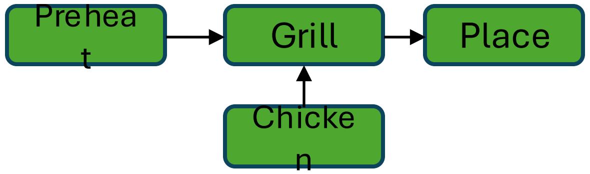
✓ Decoding **program** is **better** than decoding the raw **instructions (+17-34% F1 score)**

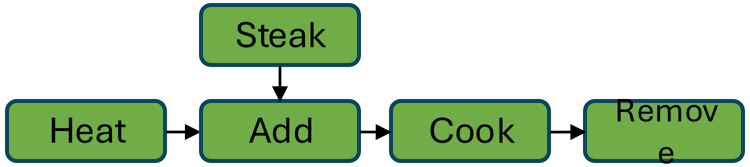
(C) Food Generation



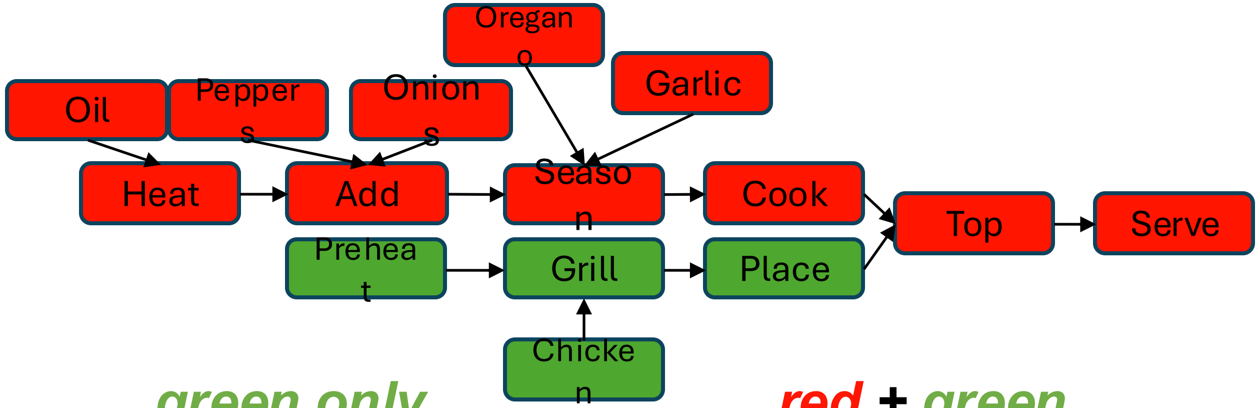
Generated image **Input program:** Grilled tomatoes and red peppers

(C) Food Generation

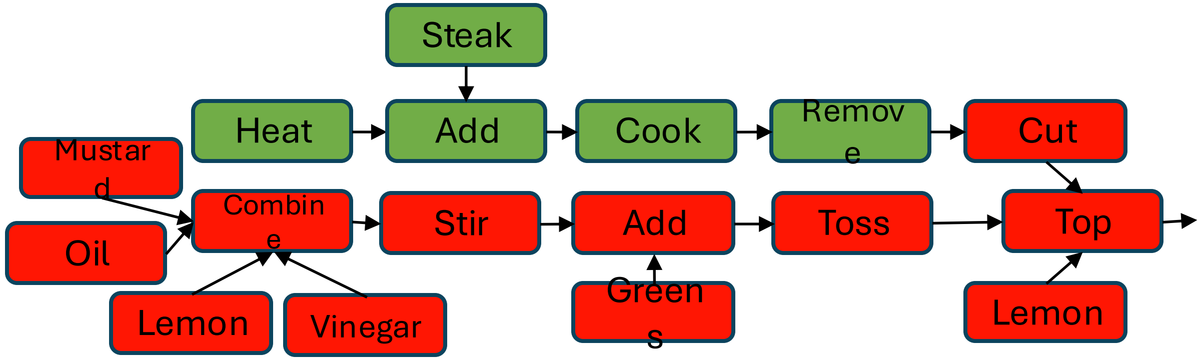




(C) Food Generation

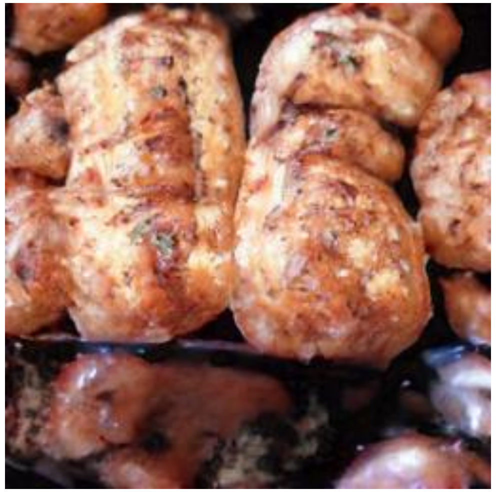


Oregano
Oil
Peppers
Onions
Garlic
Heat
Add
Season
Cook
Top
Serve
Preheat
Grill
Place
Chicken

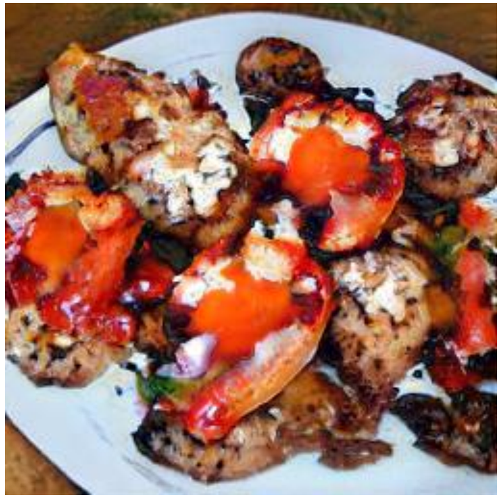


Steak
Heat
Add
Cook
Remove
Cut
Mustard
Combine
Stir
Add
Toss
Top
Oil
Lemon
Vinegar
Greens
Lemon

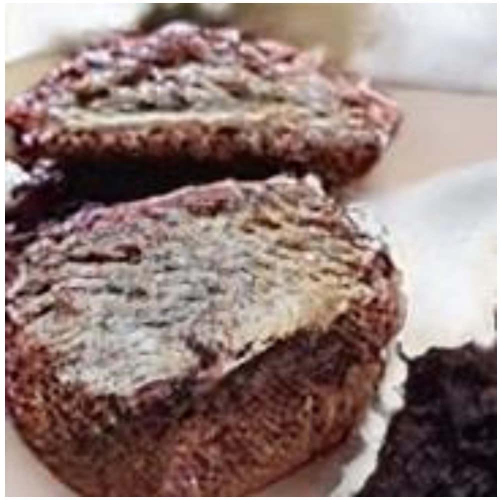
green only



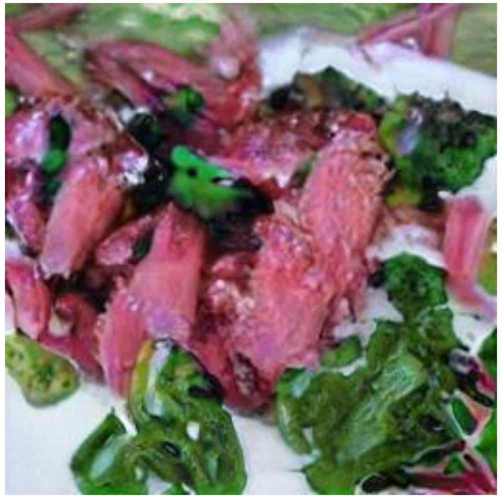
red* + *green



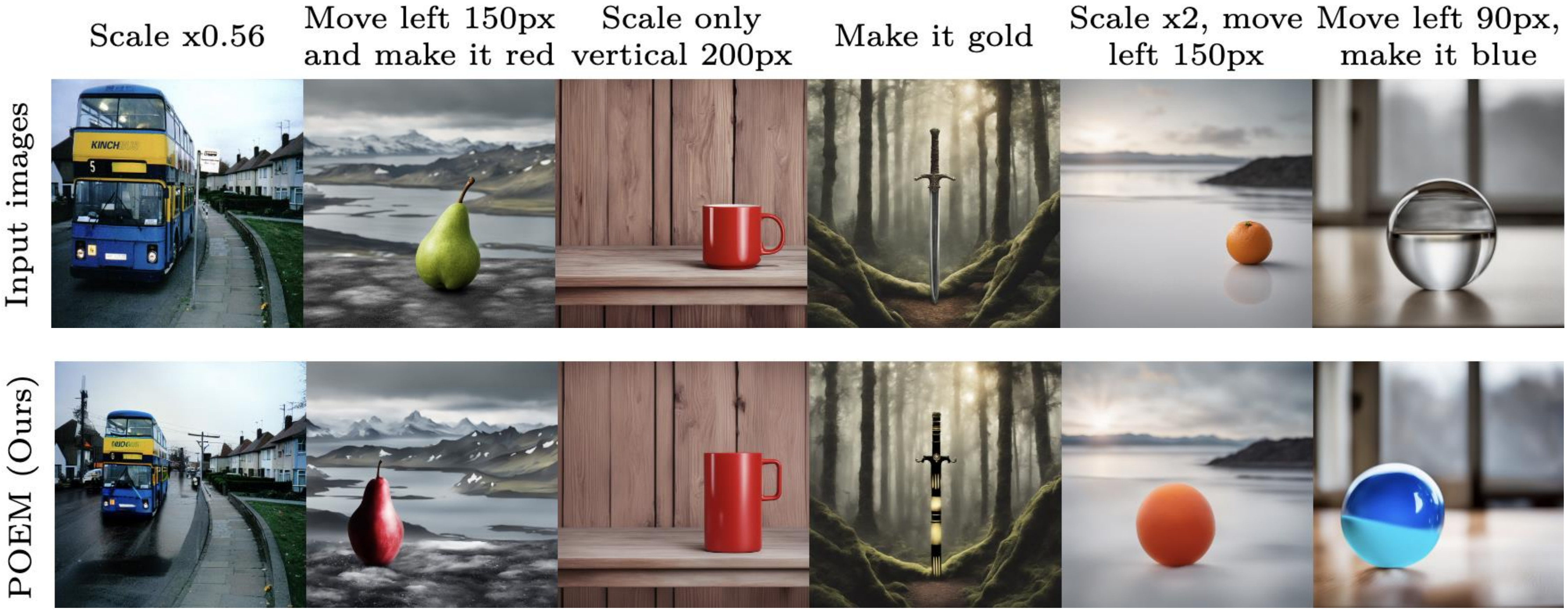
green only



red* + *green



POEM: Precise Object-level Editing


Scale x0.56
Move left 150px and make it red
Scale only vertical 200px
Make it gold
Scale x2, move left 150px
Move left 90px, make it blue
Input images
POEM (Ours)
KINCH
5

[Marco, Onur and Dimitrios, Submitted 2025]

DTU

PIONEER CENTRE FOR AI

CVPR Nashville JUNE 11-15, 2025

SCIA

POEM: Precise Object-level Editing via MLLM control

Marco Schouten, Mehmet Onurcan Kaya,
Serge Belongie, **Dim P. Papadopoulos**

DTU PIONEER CENTRE FOR AI

CVPR Nashville JUNE 11-15, 2025 SCIA

POEM: Precise Object-level Editing via MLLM control

co Schouten, Mehmet Onurcan Kaya,
rge Belongie, **Dim P. Papadopoulos**

Image Generation vs. Image Editing with Diffusion Models

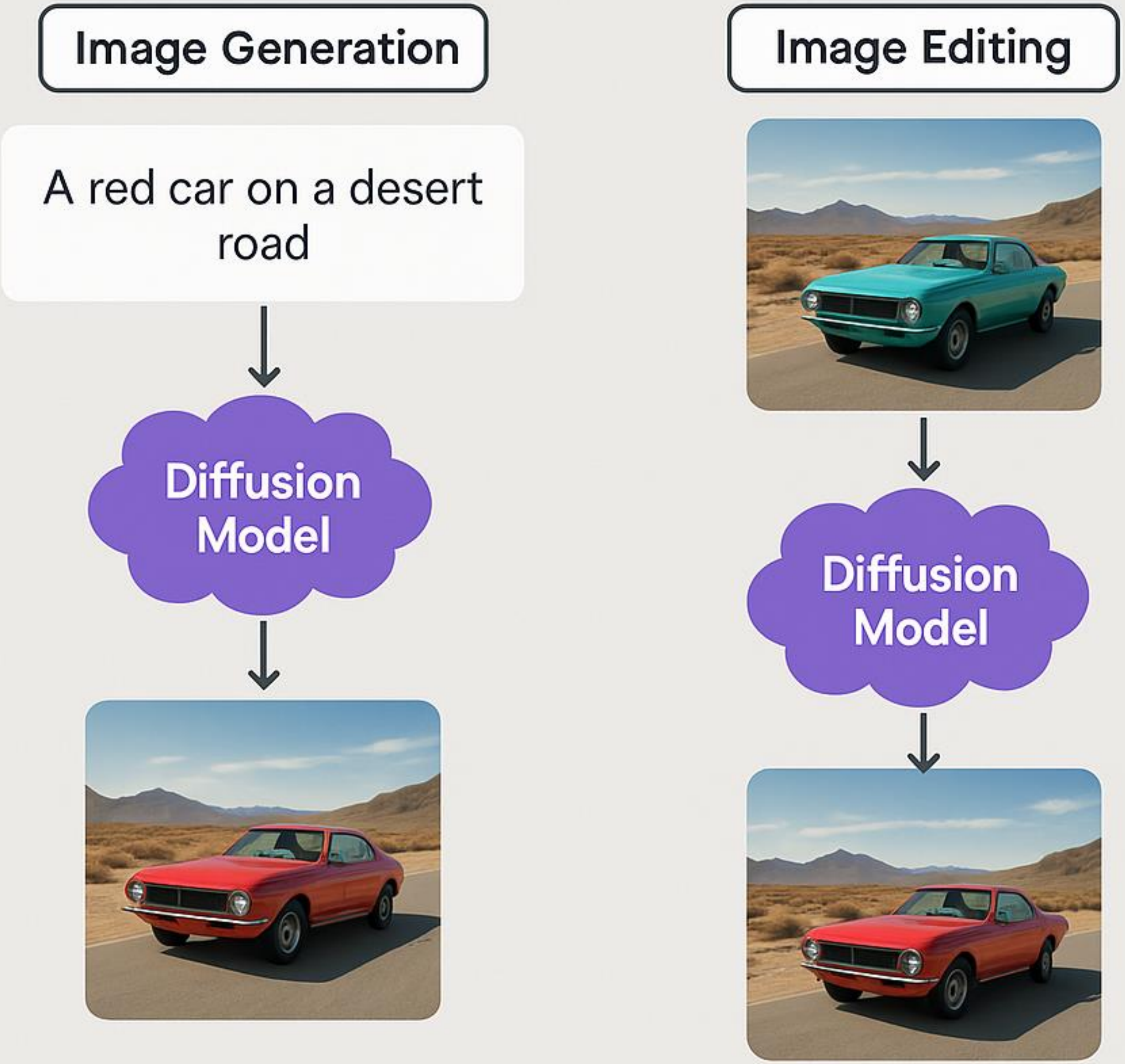


Image Generation

A red car on a desert road

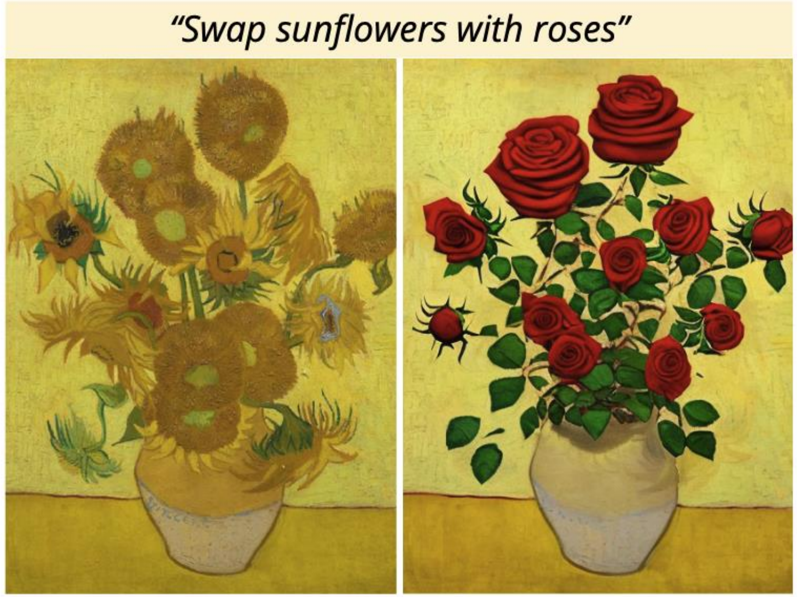
Diffusion Model

Image Editing

Diffusion Model

Image Editing

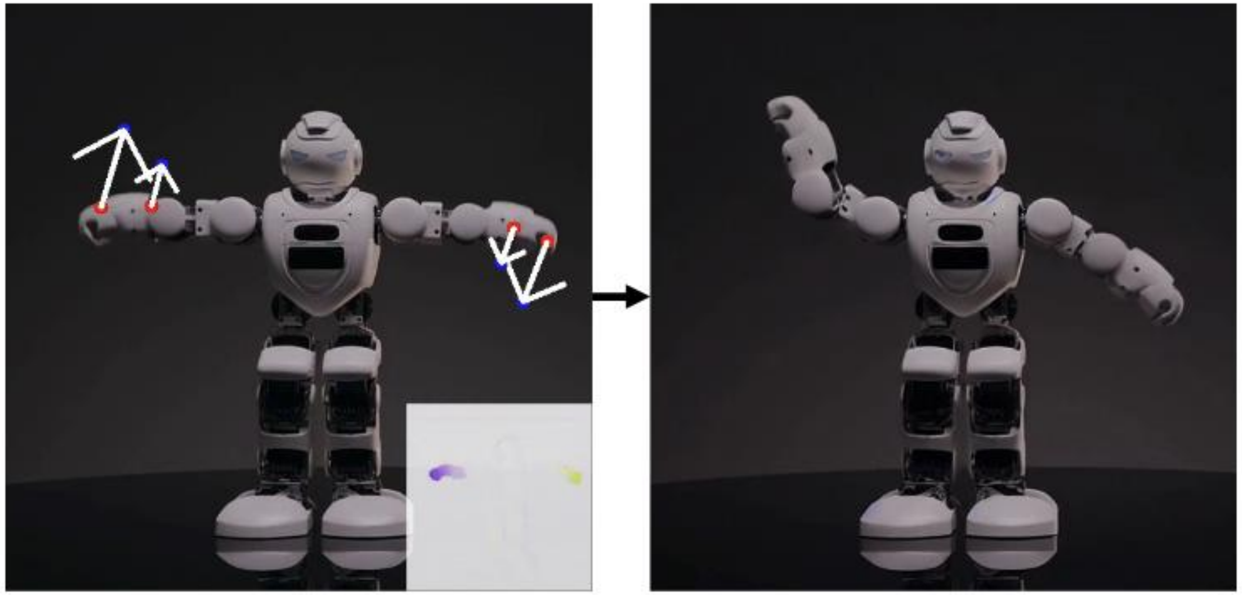
Text instruction-based



"Swap sunflowers with roses"

[Brooks CVPR 2023]

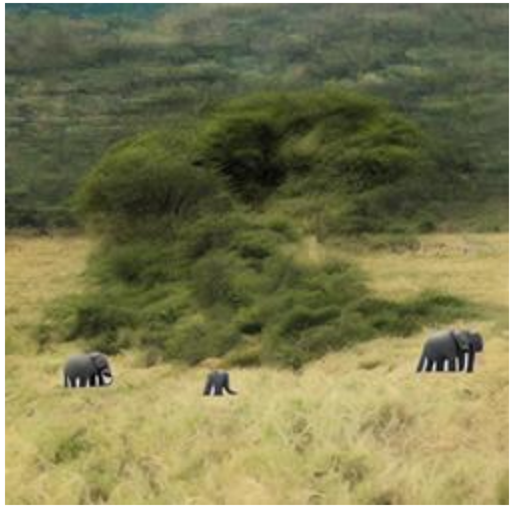
Interaction-based



[Shin CVPR 2024]

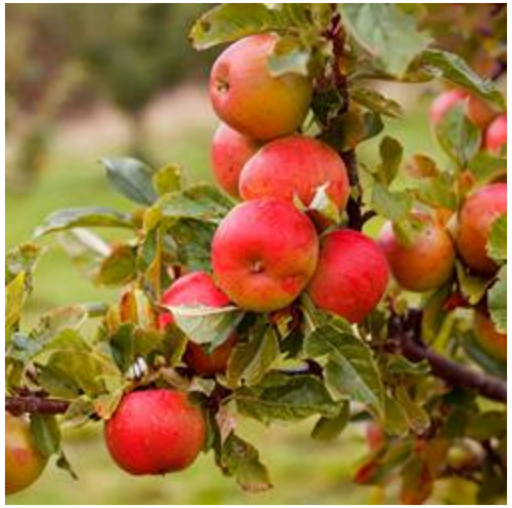
Example: instruction-based image editing

"Make the elephant smaller"



fails at controlling shape

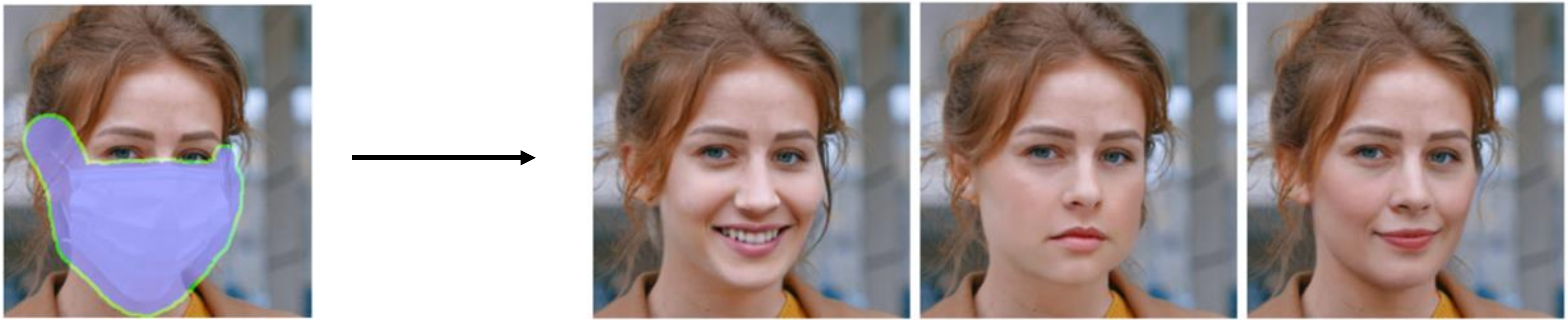
"Make the leaves red"



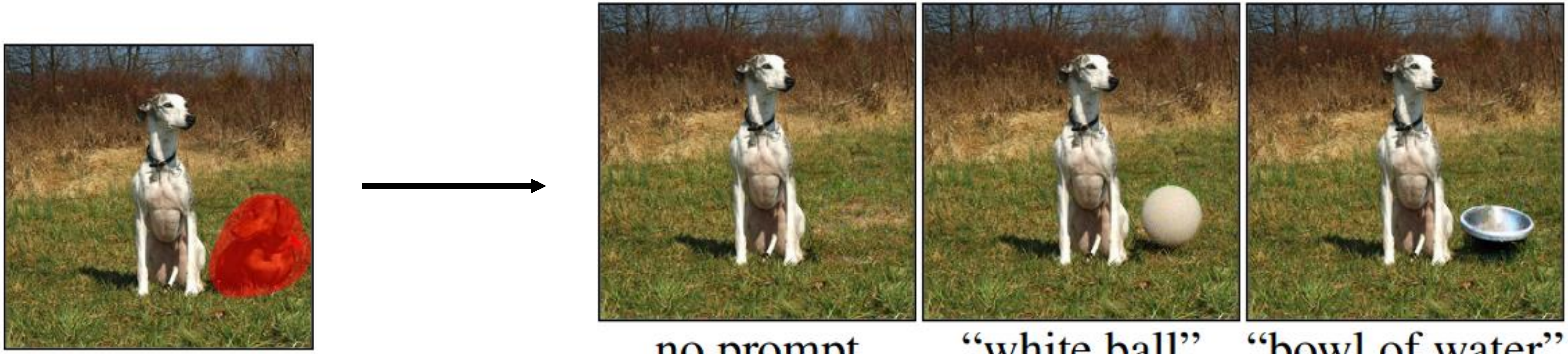
undesired global changes

[Brooks et al, InstructPix2Pix, CVPR 2023]

Example of Interaction-based: Masked-Inpainting



[Lugmayr et al., RePaint, CVPR 2022]



input+mask

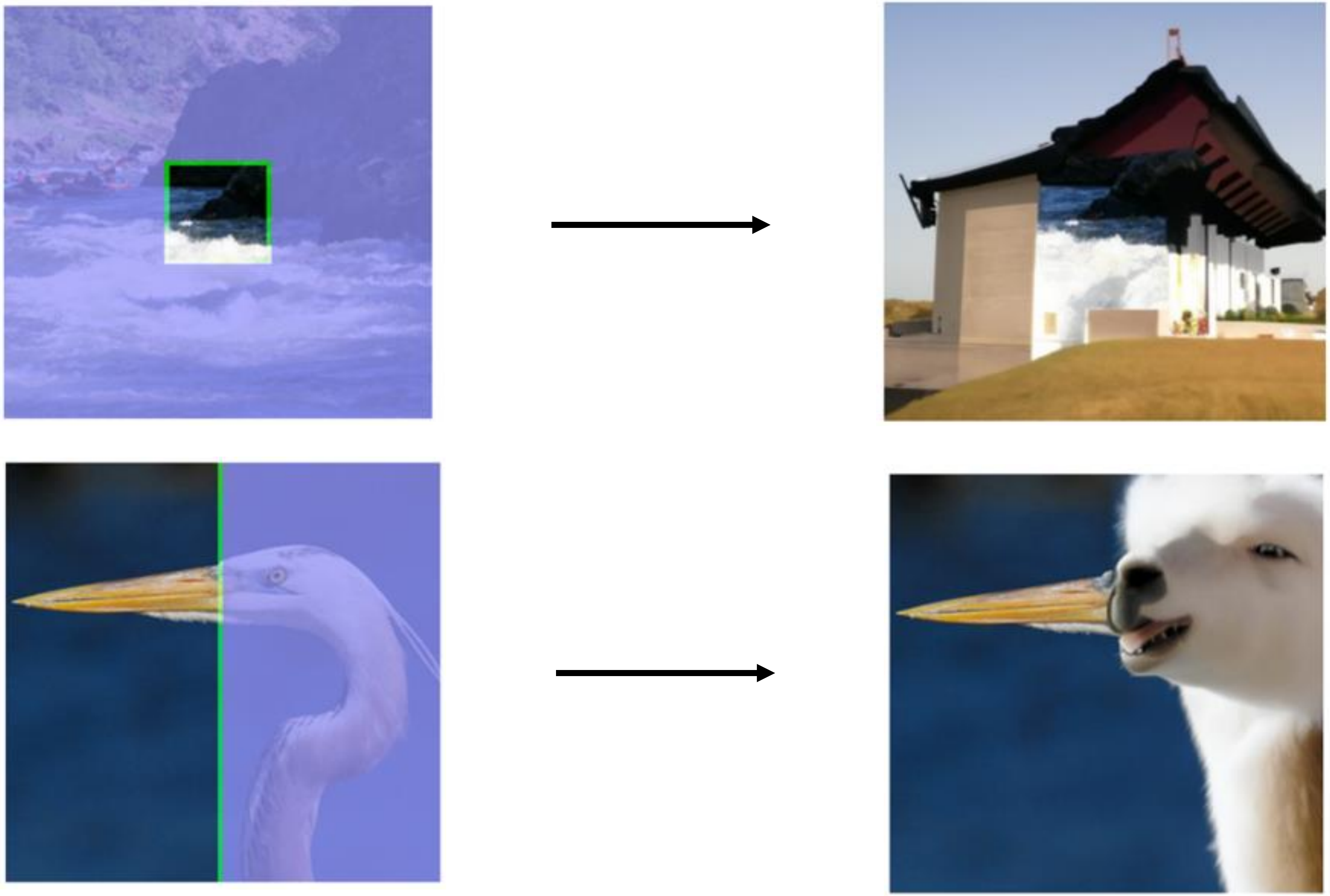
no prompt

“white ball”

“bowl of water”

[Avrahami et al., Blended Diffusion, CVPR 2022]

Example of Interaction-based: Masked-Inpainting



Fails if the mask is too large

[Andreas Lugmayr et al., RePaint, CVPR 2022]