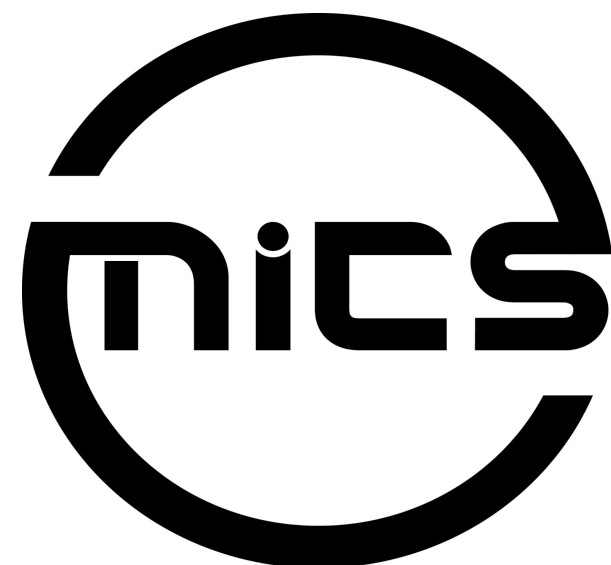


Recent topics in vision and language Deep Learning course

Maria Vakalopoulou
Mathematics and Informatics (MICS)
CentraleSupélec, University Paris-Saclay



What we have seen up to now

- Introduction to Deep Learning
- Deep Learning and Optimization
- Deep Learning for Computer Vision
- Deep Learning and Generative Models
- Deep Learning and Self Supervised Models
- Deep Learning and Reinforcement Learning
- Deep Learning and Natural Language Processing

What we will see today



Part 1:
Introduction to
Vision &
Language

Part 2:
Manipulation
of latent space
in GANS

Part 3:
Flamingo

Slides adapted from: [Vicky Kalogeiton, Andrej Karpathy, Justin Johnson, Jerry Zhu, Al Summer, Ishan Misra, Christian Rupprecht, Dim P. Papadopoulos, Samuel Albanie]

Multimodal Learning

- Multimodal learning refers to the process of **learning representations from different types of modalities** using the same model.
- Different modalities are characterized by different statistical properties
- Machine learning: input modalities include images, text, audio, etc
- Today: **only images and language** (text)

Part 1: Outline

- Taxonomy
- BERT-like architectures
- Pre-training and fine-tuning
- VL Generative models
- VL with contrastive learning

Vision-Language tasks

Generation

- Vision Question Answering (VQA)
- Visual Captioning (VC)
- Visual Commonsense Reasoning (VCR)
- Visual Generation (VG)

Classification

- Multimodal Affective Computing (MAC)
- Natural Language for Visual Reasoning (NLVR)

Retrieval

- Visual Retrieval (VR)
- Vision-Language Navigation (VLN)
- Multimodal Machine Translation (MMT)

Vision-language Generation tasks

- Visual Question Answering (**VQA**)
 - refers to the process of providing an answer to a question given a visual input (image or video)
- Visual Captioning (**VC**)
 - generates descriptions for a given visual input
- Visual Commonsense Reasoning (**VCR**)
 - infers common-sense information and cognitive understanding given a visual input
- Visual Generation (**VG**)
 - generates visual output from a textual input

Vision-language Classification tasks

- Multimodal Affective Computing (**MAC**)
 - interprets visual affective activity from visual and textual input. In a way, it can be seen as multimodal sentiment analysis.
- Natural Language for Visual Reasoning (**NLVR**)
 - determines if a statement regarding a visual input is correct or not

Vision-language Retrieval tasks

- Visual Retrieval (**VR**)
 - retrieves images based only on a textual description
- Vision-Language Navigation (**VLN**)
 - is the task of an agent navigating through a space based on textual instructions
- Multimodal Machine Translation (**MMT**)
 - involves translating a description from one language to another with additional visual information

Part 1: Outline

- Taxonomy
- **BERT-like architectures**
- Pre-training and fine-tuning
- VL Generative models
- VL with contrastive learning

BERT-like architectures

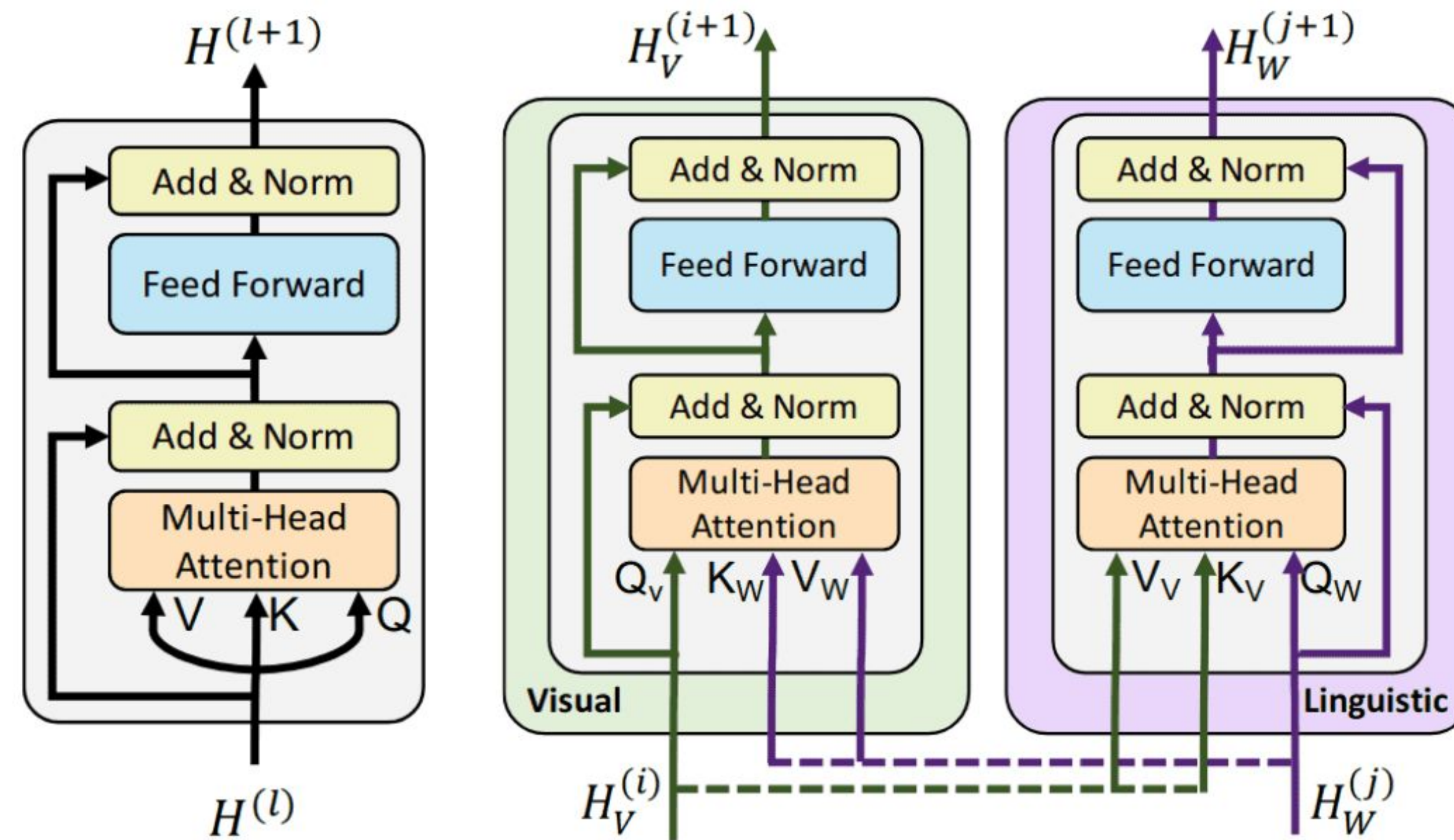
- **Idea:** process language and images at the same time with a transformer-like architecture
- Two categories:
 - Two-stream models
 - process text and images using two separate modules, e.g., ViLBERT, LXMERT
 - Single-stream models
 - Encode both modalities within the same module, e.g. VisualBERT, VL-BERT, UNITER

Two-stream models

- Trained on image-text pairs
 - Text: encoded with the standard transformer process using tokenization and positional embeddings. It is then processed by the self-attention modules of the transformer.
 - Images are decomposed into non-overlapping patches projected in a vector, as in vision transformer's patch embeddings.

Two-stream models- ViLBERT

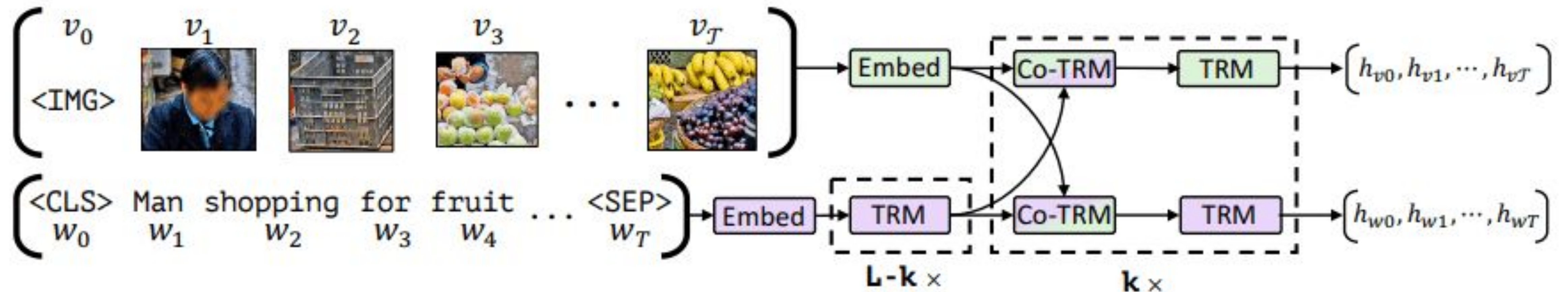
- “Co-attention module”:
 - Joint representation of images and text
 - Calculates importance scores based on both images and text embeddings



Standard encoder transformer block VS **co-attention** transformer layer

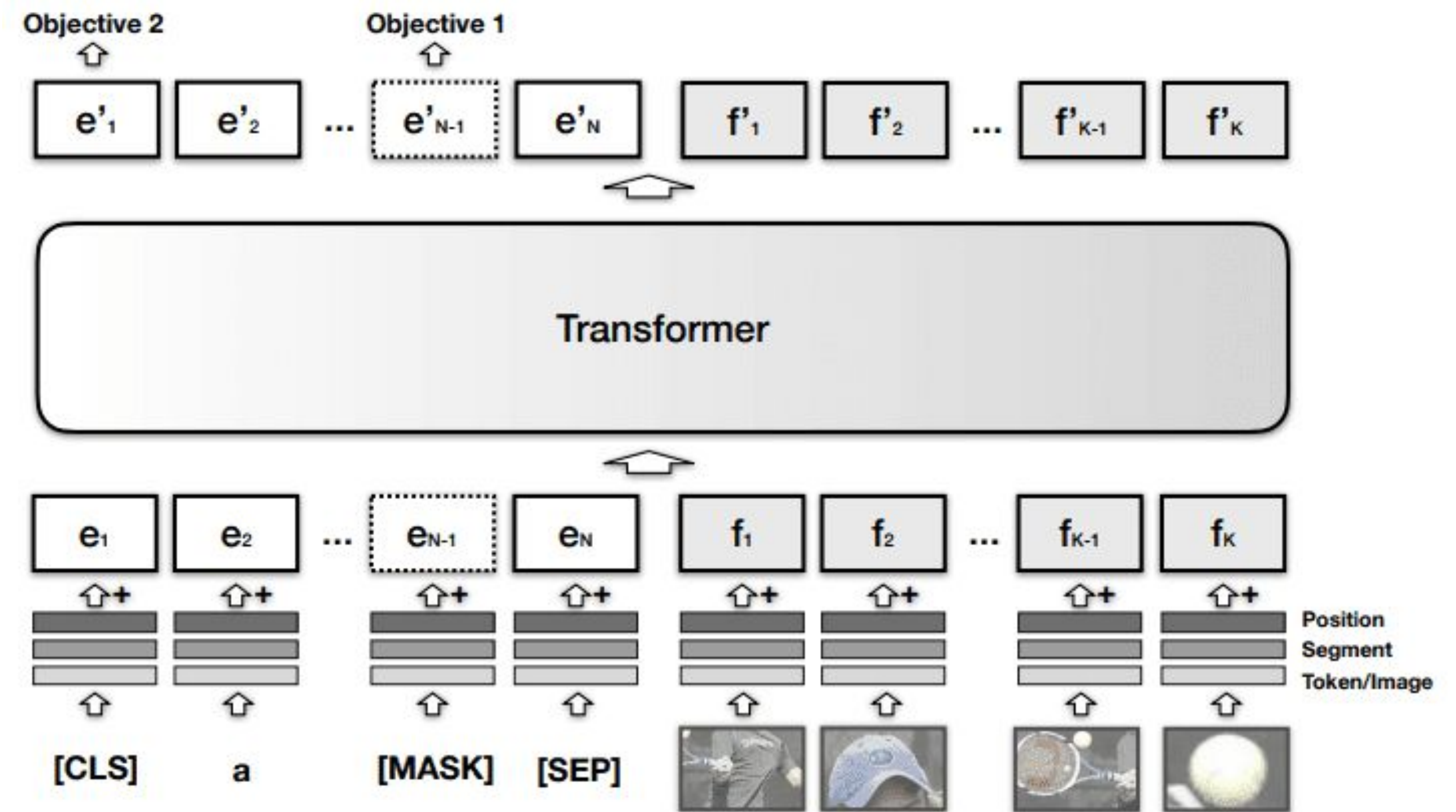
Two-stream models- ViLBERT

- **Text** stream (purple)
 - the weights are set by pretraining the model on standard text corpus
- **Image** stream (green): Faster R-CNN
- **Training**: dataset of image-text pairs
- **Objective**: understand the relationship between text and images



Single-stream models- VisualBERT

- VisualBERT: combines image + language with transformer
- Uses self-attention to discover alignments between them
- Add visual embedding to BERT:
 - A visual feature representation of the region produced by a CNN
 - A segment embedding that distinguishes image from text embeddings
 - A positional embedding to align regions with words if provided in the input



Part 1: Outline

- Taxonomy
- BERT-like architectures
- **Pre-training and fine-tuning**
- VL Generative models
- VL with contrastive learning

Pretraining

- **Masked Language Modeling**

- Used when the transformer is trained only on text
- Certain tokens of the input are being masked at random. Training: predict the masked tokens (words)

- **Next Sequence Prediction**

- Only text as input
- Evaluates if a sentence is an appropriate continuation of the input sentence
- Using false and correct sentences as training data + capture long-term dependencies

- **Masked Region Modeling**

- Masks image regions (similar to masked language modeling)
- Training: predict features of masked region

Pretraining

- **Image-Text Matching**
 - Forces model predict if a sentence is appropriate for a specific image
- **Word-Region Alignment**
 - Correlations between image region and words
- **Masked Region Classification**
 - Predicts object class for each masked region
- **Masked region Feature Regression**
 - learns to regress the masked image region to its visual features

VL modeling

- **Unsupervised VL Pre-training**

- Pre-training without paired image-text data but with a single modality
- During fine-tuning, the model is fully-supervised

- **Multi-task Learning**

- Joint learning across multiple tasks + transfer the learnings from one task to another

- **Contrastive Learning**

- Learn visual-semantic embeddings in a self-supervised way
- Idea: learn such an embedding space in which similar pairs stay close to each other while dissimilar ones are far apart

- **Zero-shot Learning**

- Generalize at inference time on samples from unseen classes

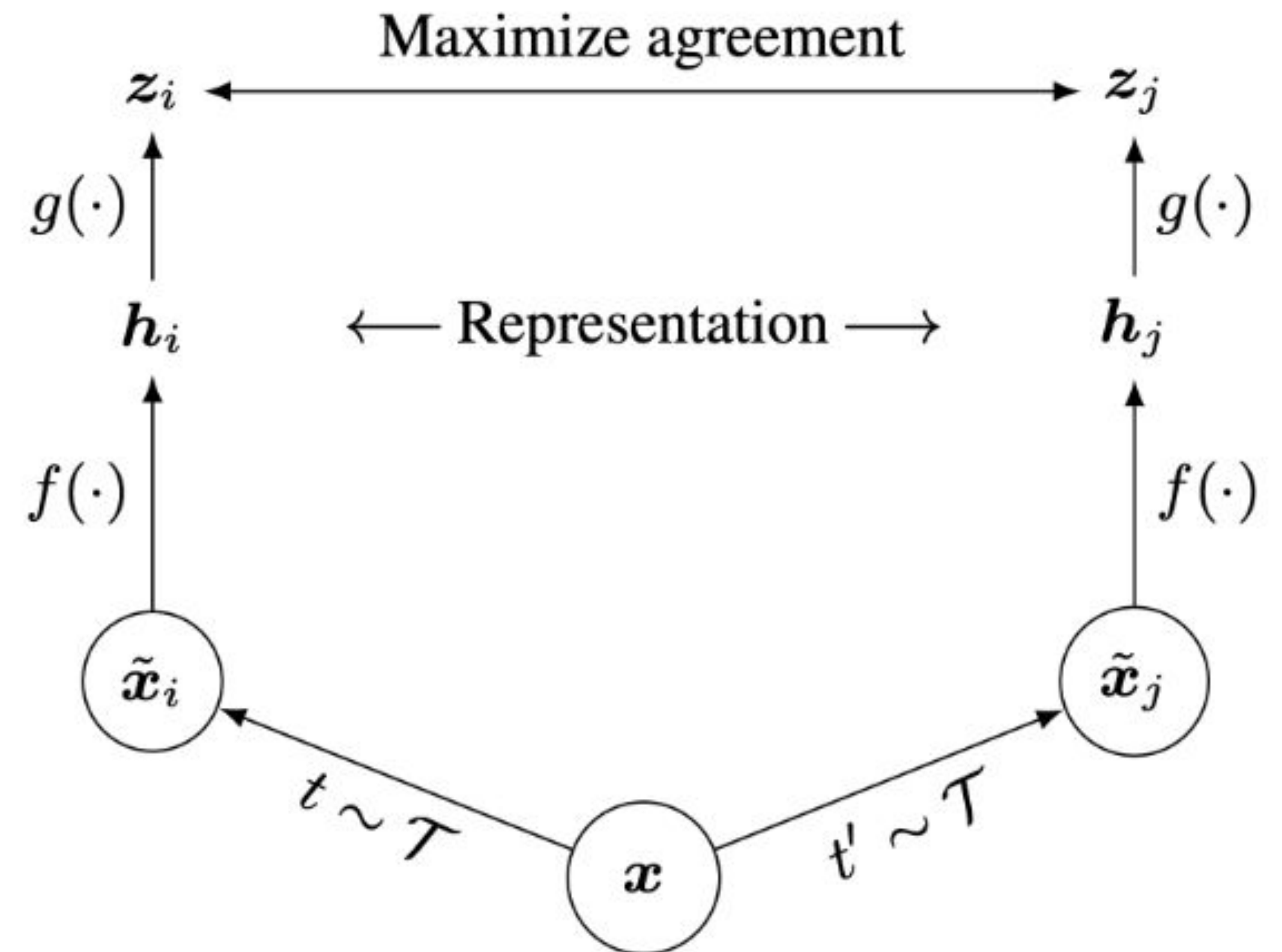
2020 - Contrastive Learning

Chen, Ting, et al. "A simple framework for contrastive learning of visual representations." ICML 2020.

Objective Function

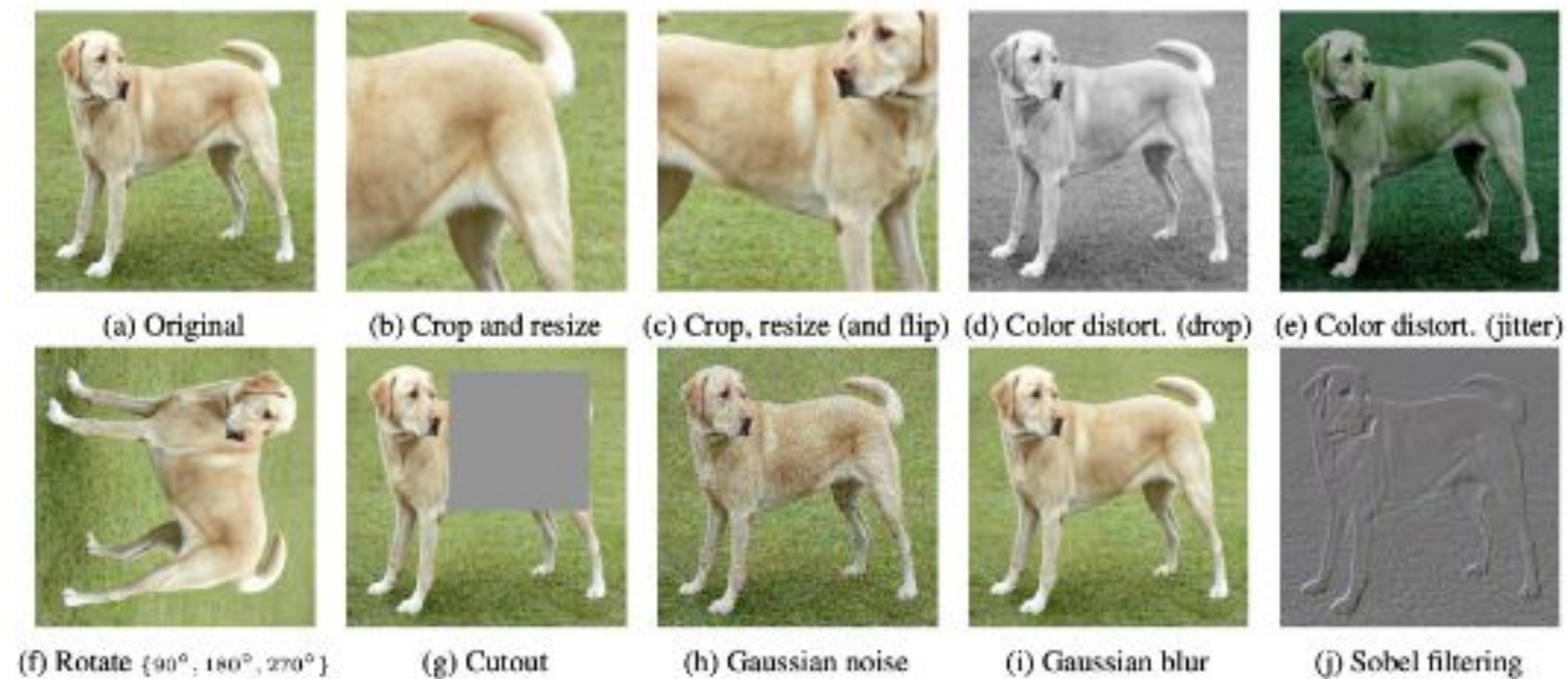
- Take two augmentations
- Maximize similarity

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}$$



SimCLR

- SimCLR proposes extensive quantitative study on the augmentations that it can be used.



Methods	Color distortion strength					AutoAug
	1/8	1/4	1/2	1	1 (+Blur)	
SimCLR	59.6	61.0	62.6	63.2	64.5	61.1
Supervised	77.0	76.7	76.5	75.7	75.4	77.1

Analysis of data augmentations

- What pair of augmentations would be most important?



Careful tweaking the temperature

- Loss function has a temperature parameter

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j) / \tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k) / \tau)}$$

ℓ_2 norm?	τ	Entropy	Contrastive acc.	Top 1
Yes	0.05	1.0	90.5	59.7
	0.1	4.5	87.8	64.4
	0.5	8.2	68.2	60.7
	1	8.3	59.1	58.0
No	10	0.5	91.7	57.2
	100	0.5	92.1	57.0

Conclusion

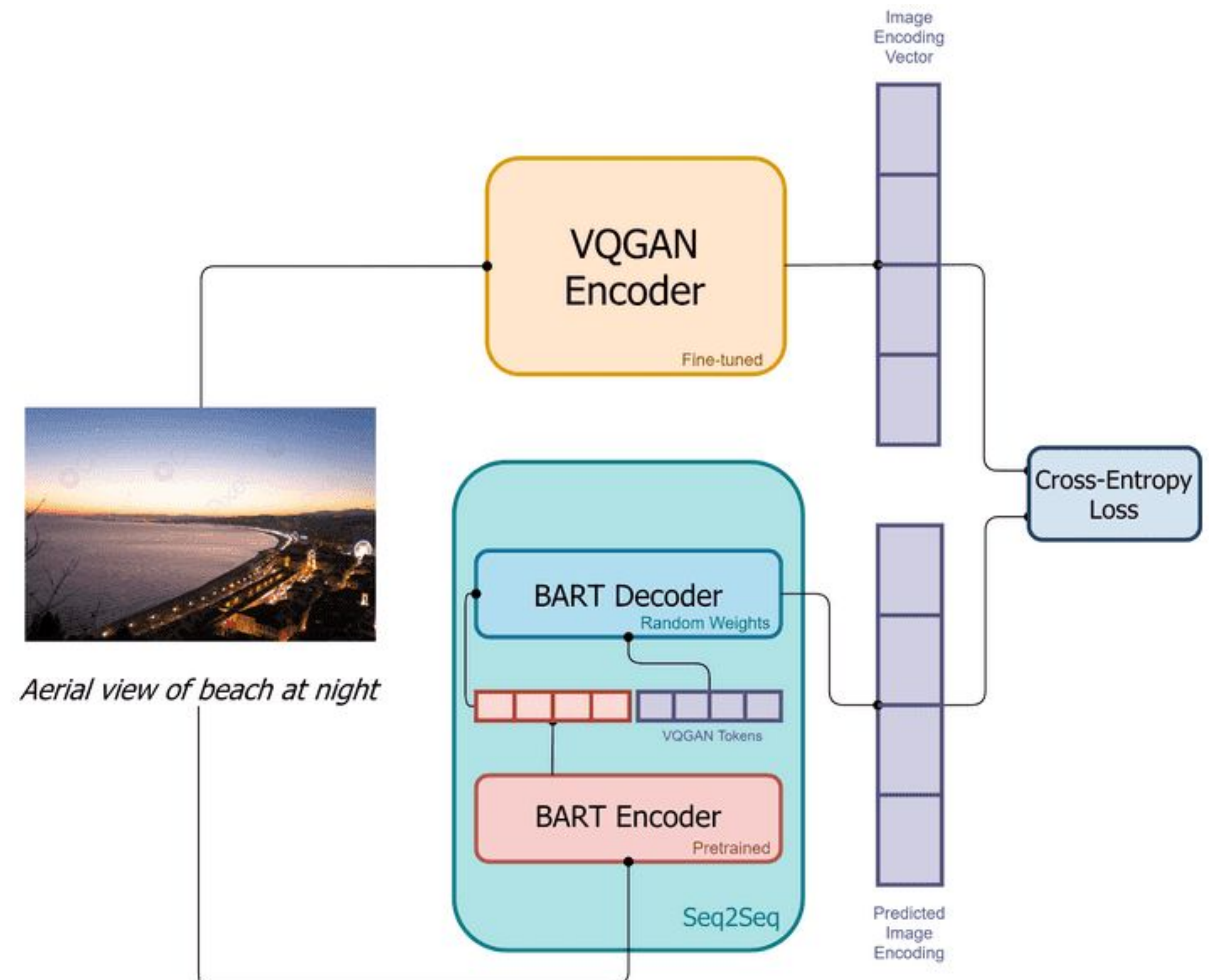
- Outstanding performance with ResNet-50 (69.3% top-1)
- Strong scaling results with ResNet-50x4 (76.5% top-1)
- Impressive grid search over many parameters
- Better head parameters
- Temperature
- Data augmentations
- Longer training and large batch size

Part 1: Outline

- Taxonomy
- BERT-like architectures
- Pre-training and fine-tuning
- **VL Generative models**
- VL with contrastive learning

DALL-E

- Discrete variational autoencoder (VAE): maps images to image tokens.
- dVAE uses a discrete latent space compared to a typical VAE.
- Text: tokenized with byte-pair encoding
- Image and text tokens concatenated, processed as a single data stream



DALL-E

TEXT PROMPT an armchair in the shape of an avocado. an armchair imitating an avocado.

AI-GENERATED
IMAGES



<https://openai.com/blog/dall-e/>

Imagen by Google AI



A photo of a Corgi dog riding a bike in Times Square. It is wearing sunglasses and a beach hat.

<https://imagen.research.google/>

Imagen by Google AI



<https://imagen.research.google/>

Part 1: Outline

- Taxonomy
- BERT-like architectures
- Pre-training and fine-tuning
- VL Generative models
- **VL with contrastive learning**

CLIP

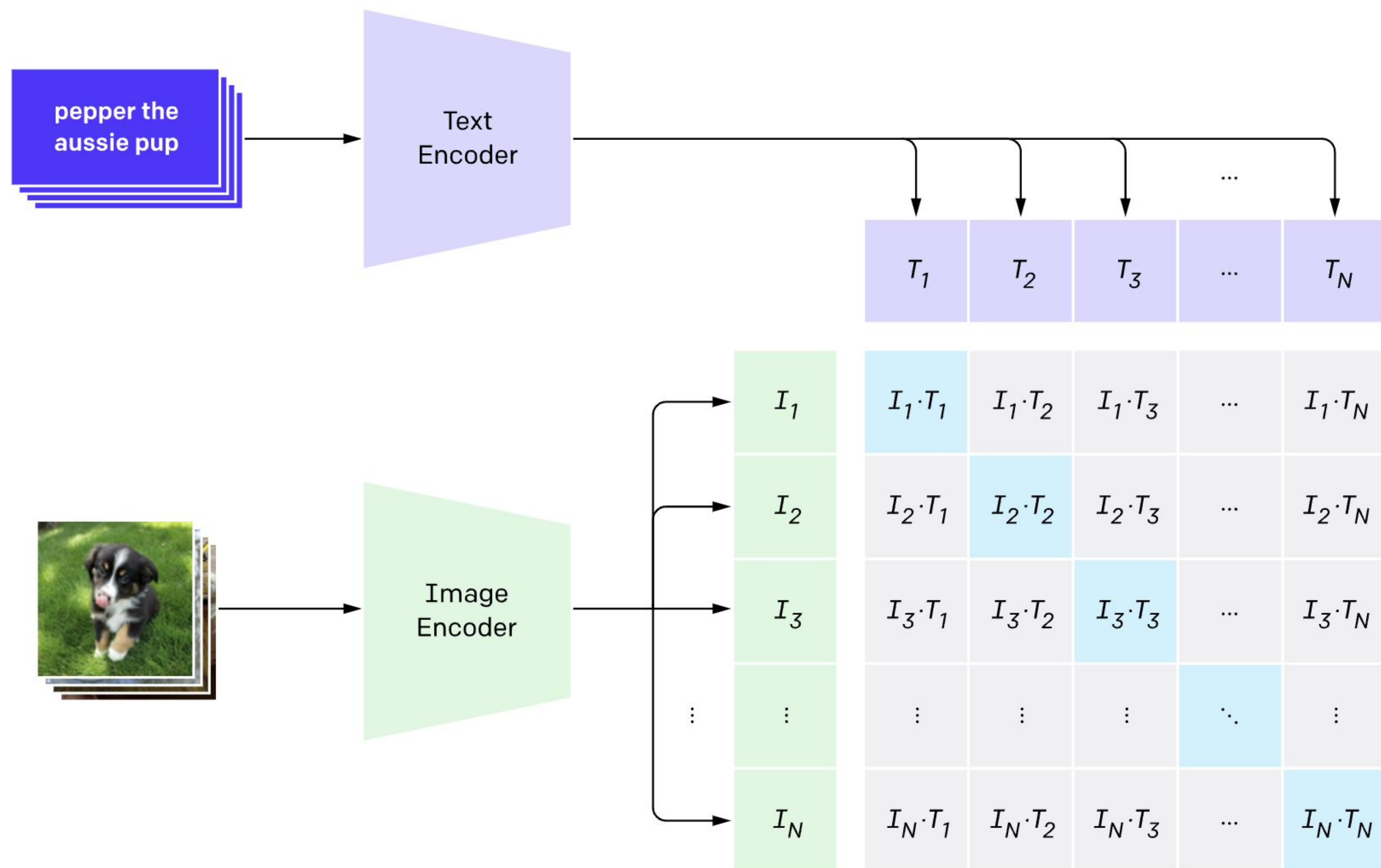
- Natural Language for Visual Reasoning (**NLVR**)
- Classify an image to a specific label based on its context
 - Label is a phrase or a sentence describing the image
- CLIP is a zero-shot classifier, i.e., can be used to previously unseen labels
- Trained on a highly-diversified, huge dataset: 400 million! images with textual descriptions
- Images, Text: transformer

CLIP

- The training's **objective**: “connect” image representations with text representations
 - Discover which text vector is more “appropriate” for a given image vector contrastive learning
 - Instead of bringing together views of the same image, we are pulling together the positive image and text “views”, while pulling apart texts that do not correspond to the correct image (negatives)
 - fully supervised, i.e. labeled pairs are required
- Training: assign high similarity for fitting image-text pairs and low similarity for unfitting ones and downstream tasks

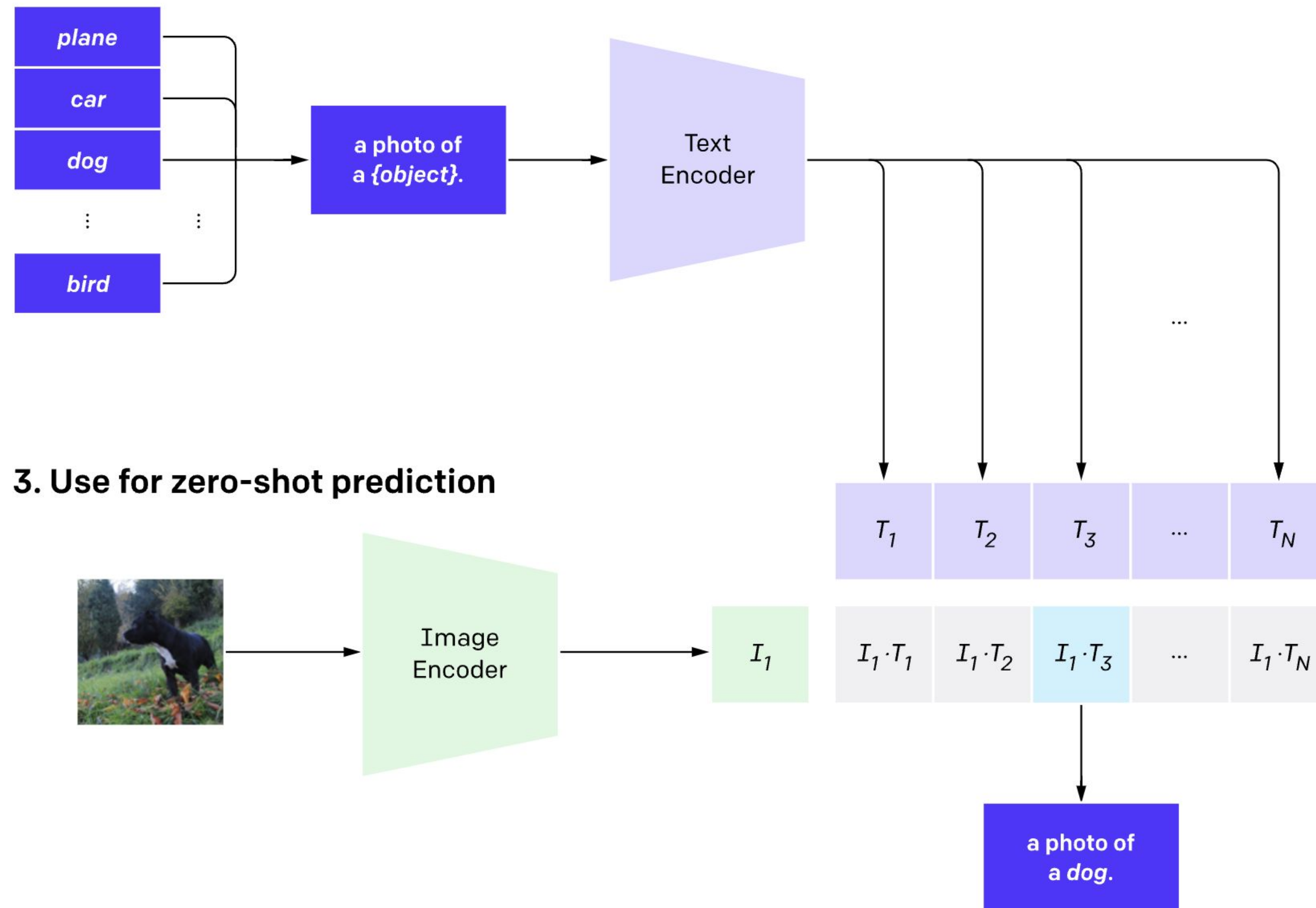
CLIP

1. Contrastive pre-training

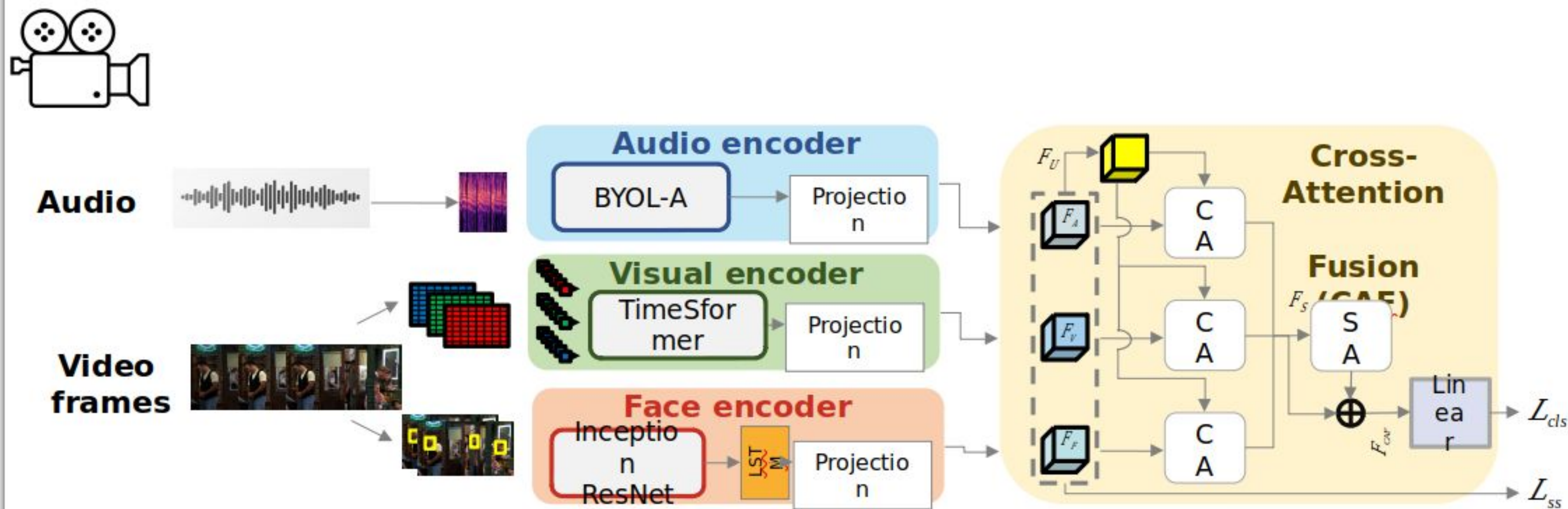


CLIP

2. Create dataset classifier from label text



Similar models for more modalities



What we will see today

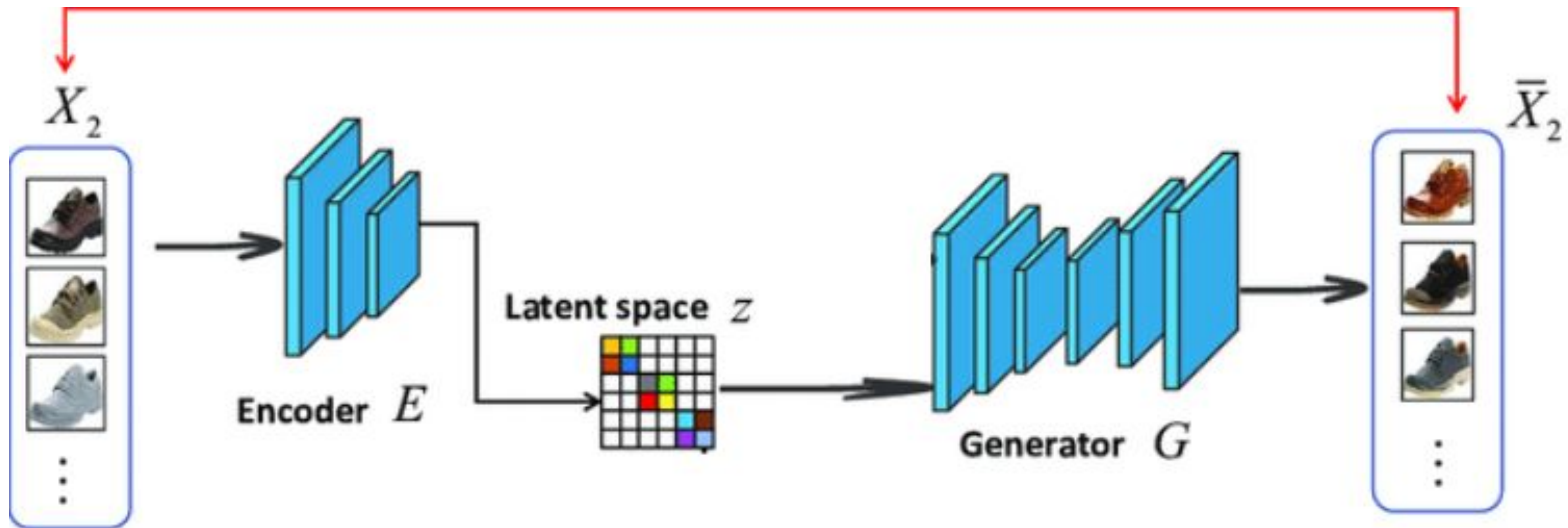


Slides adapted from: [Vicky Kalogeiton, Andrej Karpathy, Justin Johnson, Jerry Zhu, Al Summer, Ishan Misra, Christian Rupprecht, Dim P. Papadopoulos, Samuel Albanie]

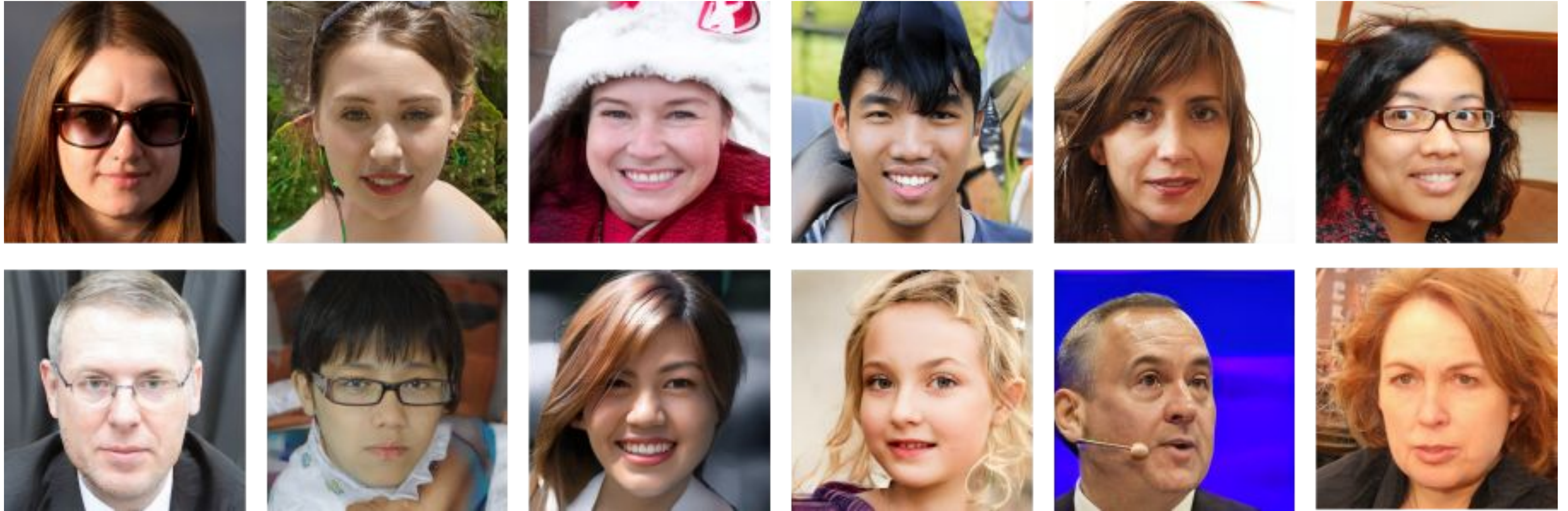
Part 2: Outline

- Interpolation in latent space
- Reconstructing real images
- Learn and apply latent directions
- CLIP + StyleGAN

Latent space of a GAN



Generate random images



<https://github.com/NVLabs/stylegan2-ada>

```
python generate.py --outdir=out --trunc=1 --seeds=666-700 --  
network=https://nvlabs-fi-cdn.nvidia.com/stylegan2-ada-pytorch/pretrained/  
ffhq.pkl
```


Latent space of a GAN



z_1

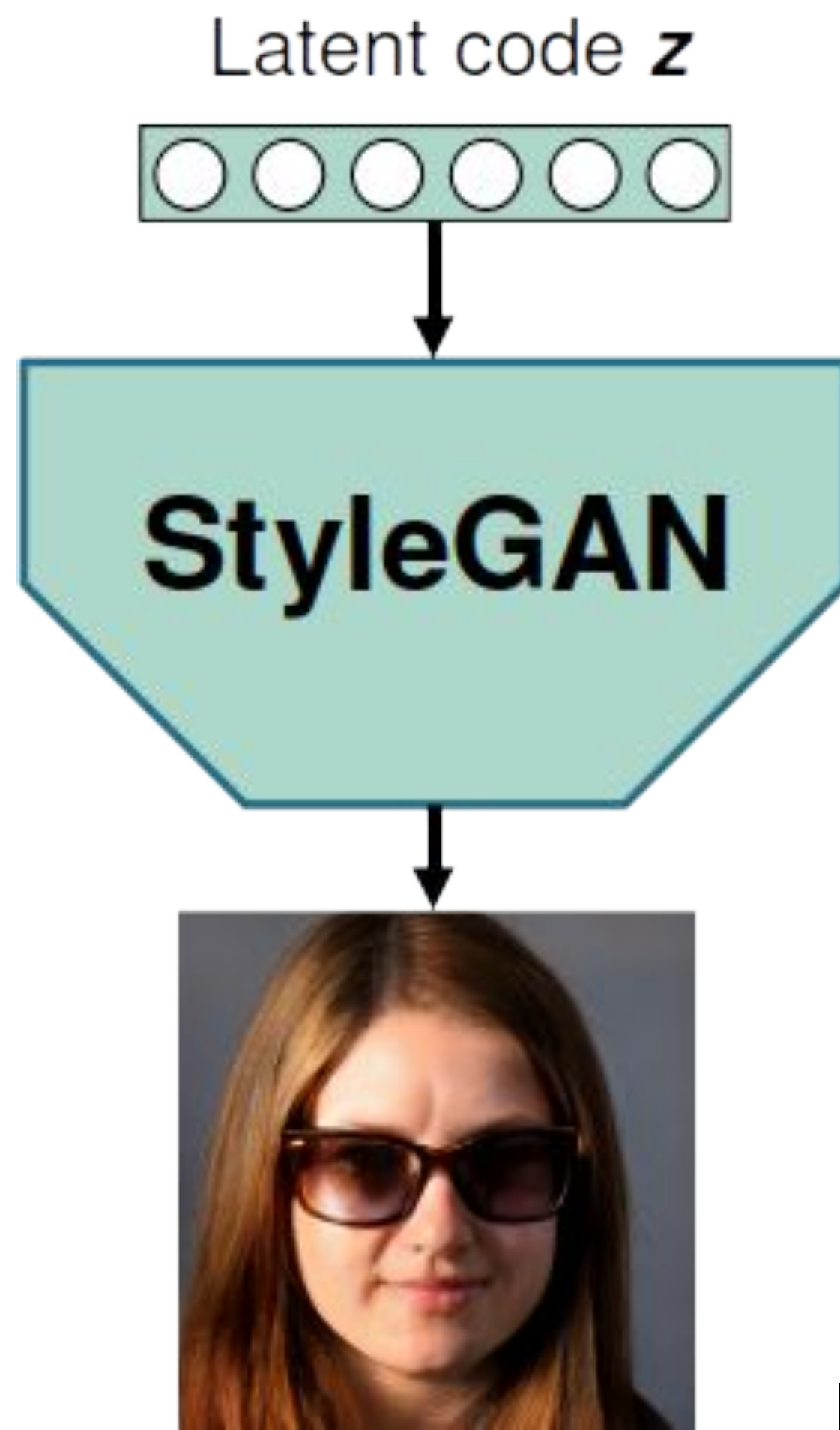
$a*z_1 + (1-a)*z_2$

z_2

<https://github.com/NVLabs/stylegan2-ada>

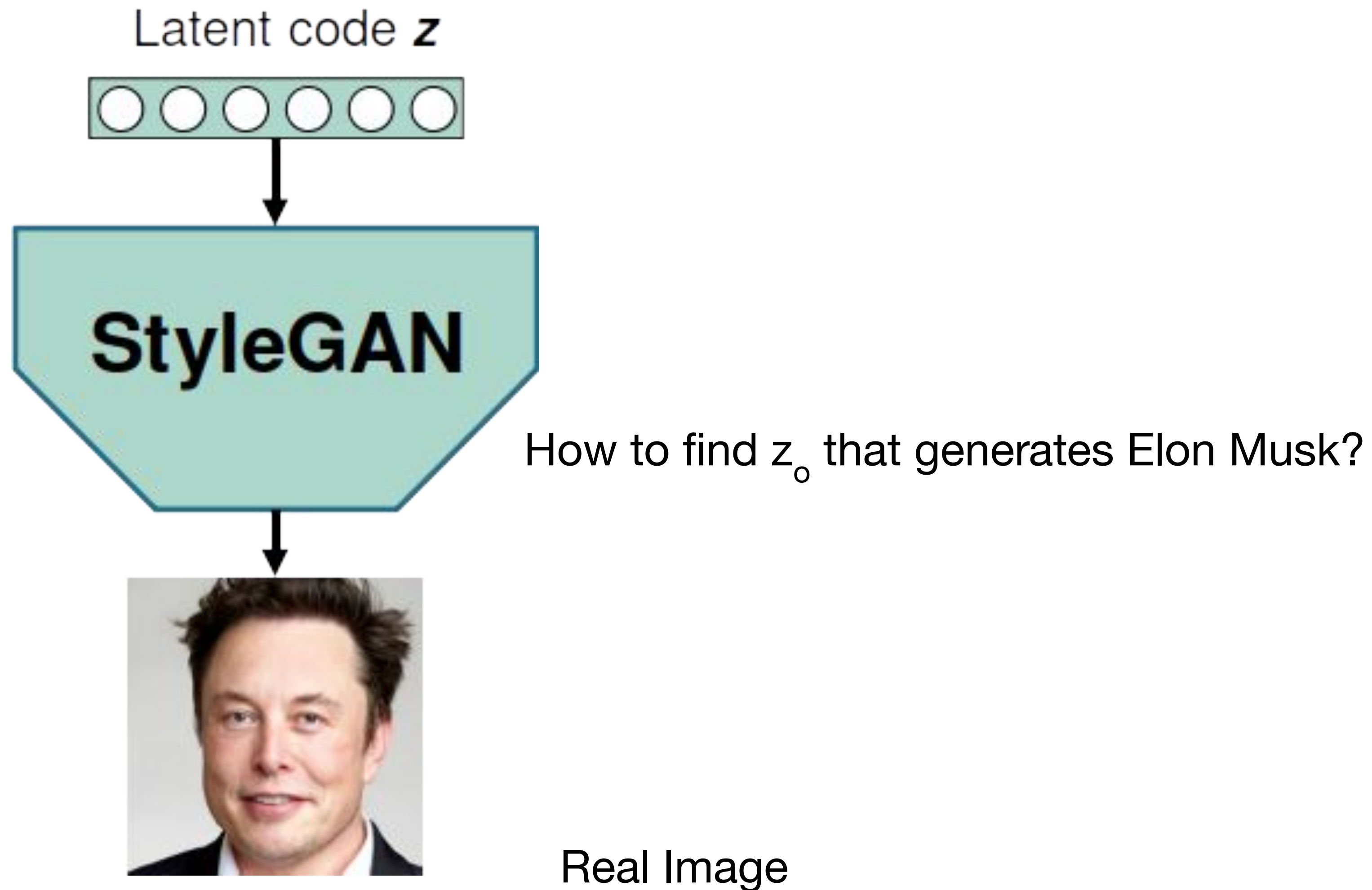
```
python generate.py --outdir=out --trunc=1 --seeds=666-700 --  
network=https://nvlabs-fi-cdn.nvidia.com/stylegan2-ada-pytorch/pretrained/  
ffhq.pkl
```


Real images: GAN inversion

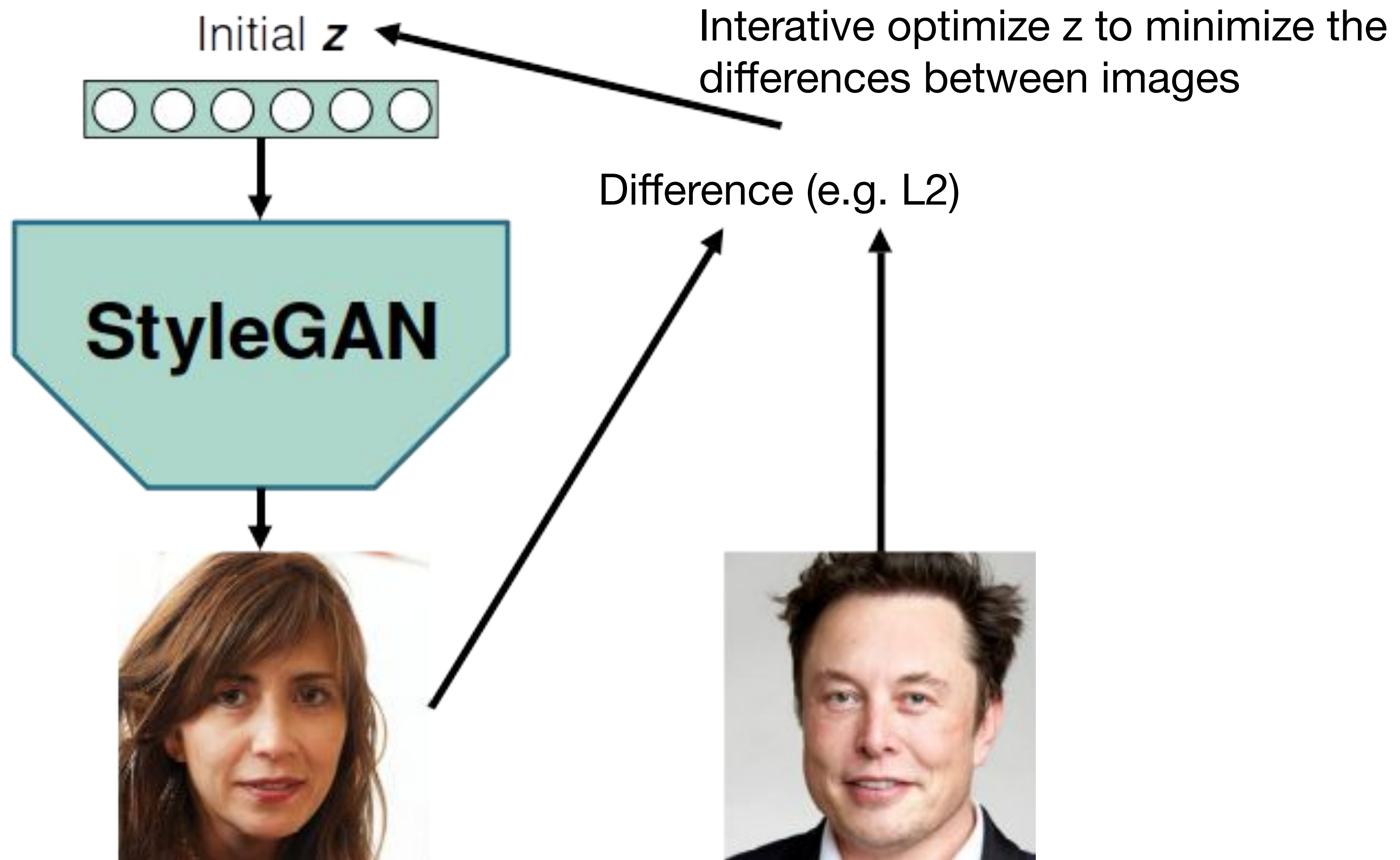


Fake generated image

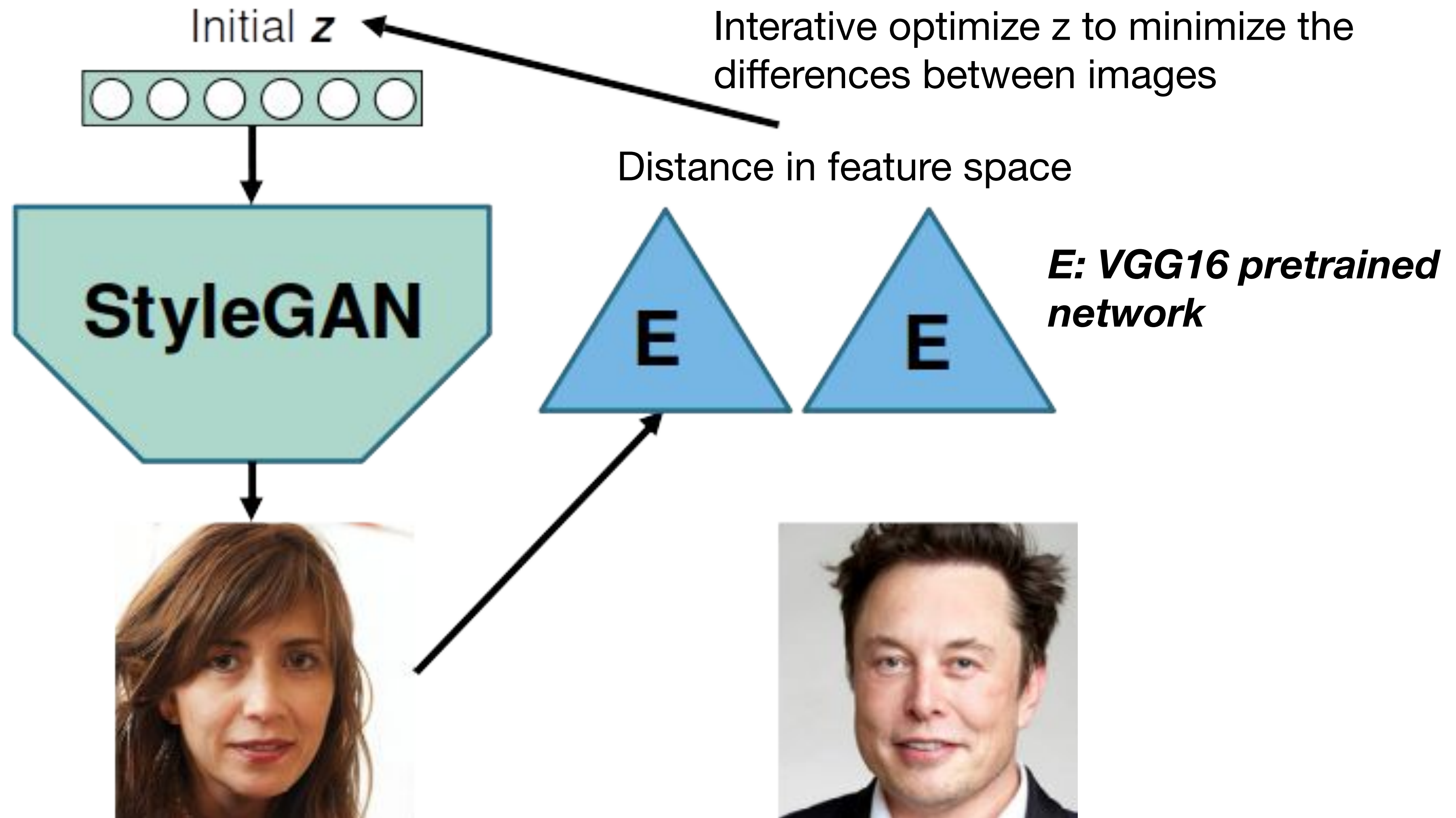
Real images: GAN inversion



Real images: GAN inversion



Real images: GAN inversion

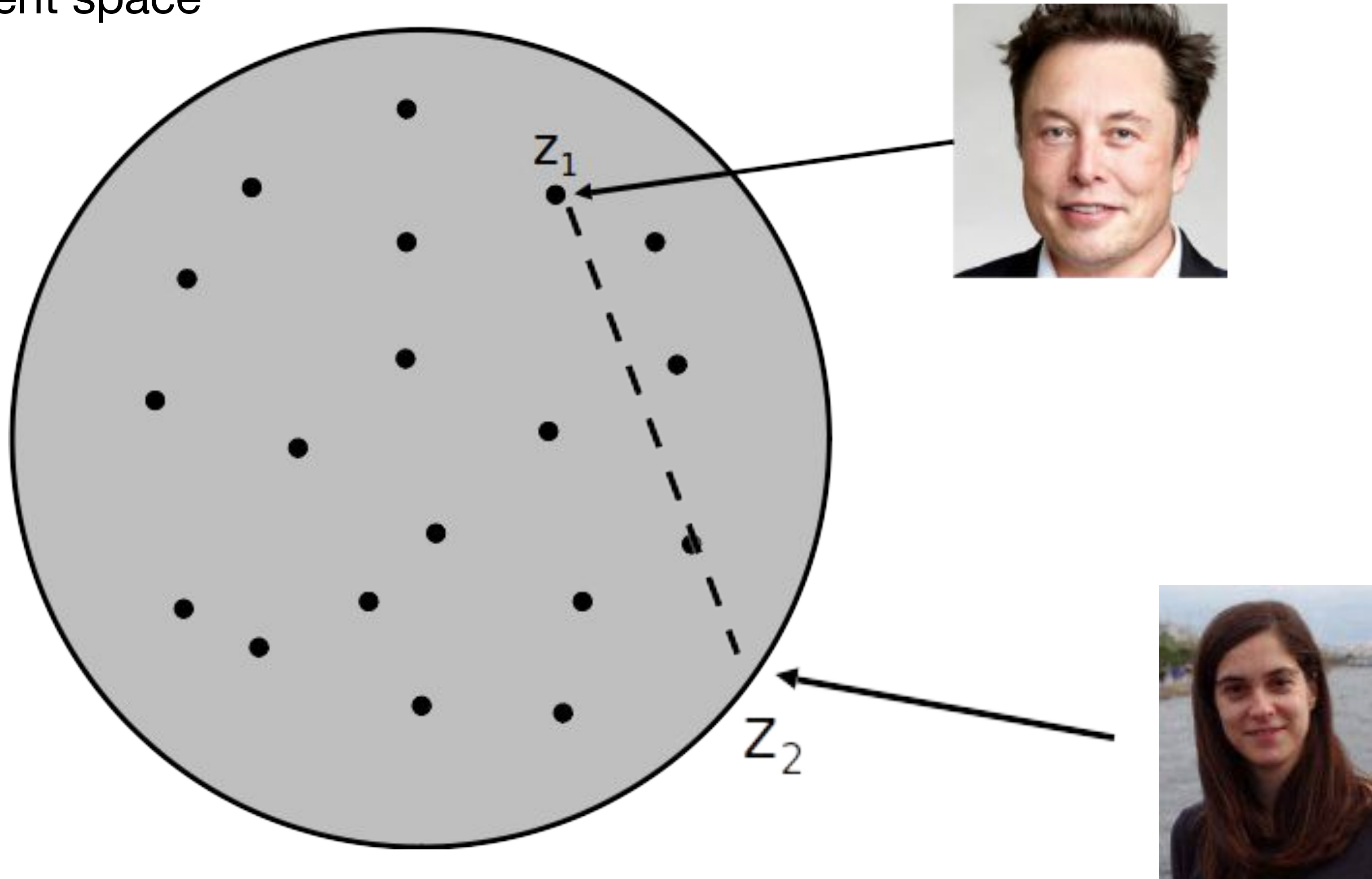


Real images: GAN inversion



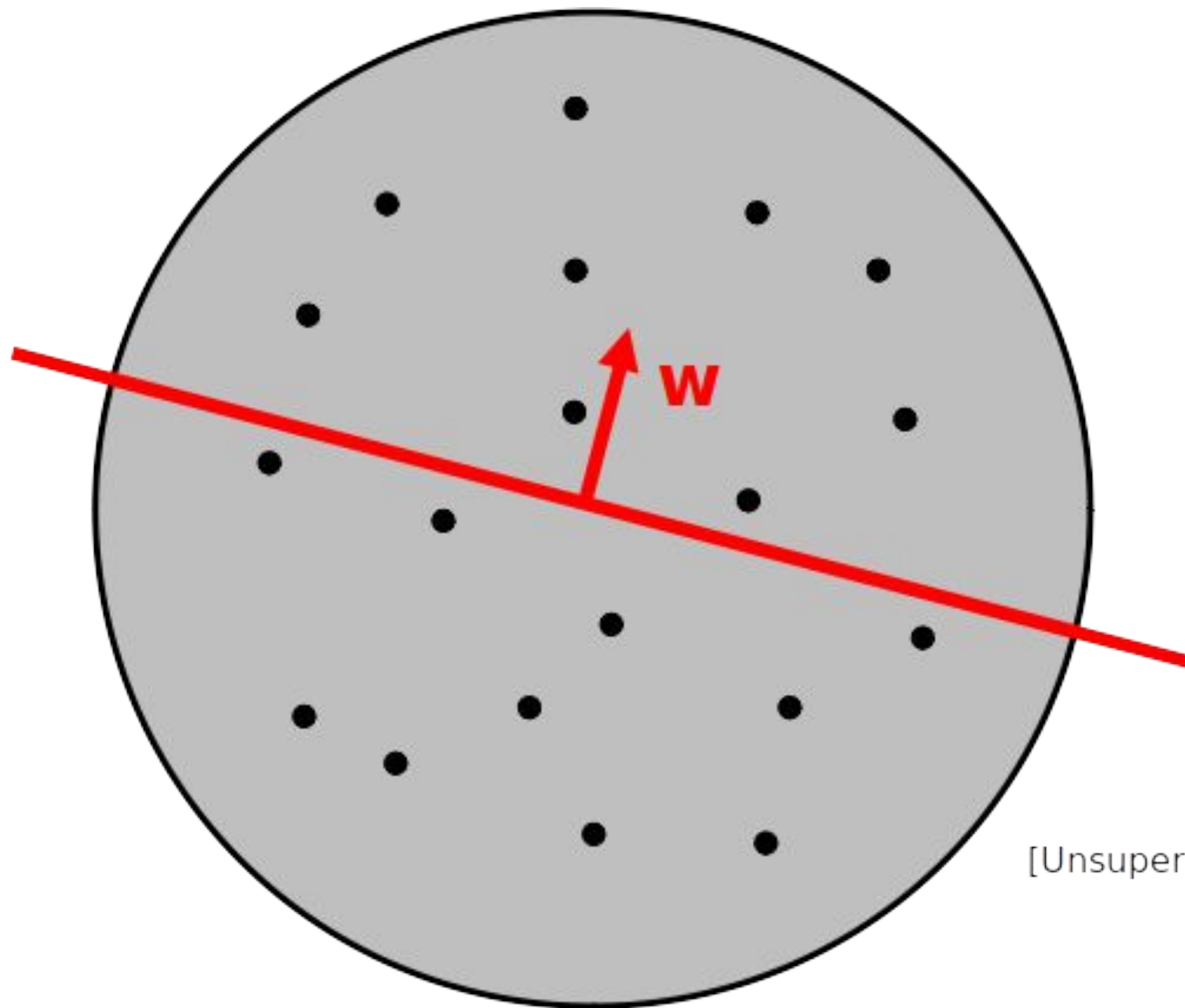
Latent directions

Latent space



Latent directions

Latent space

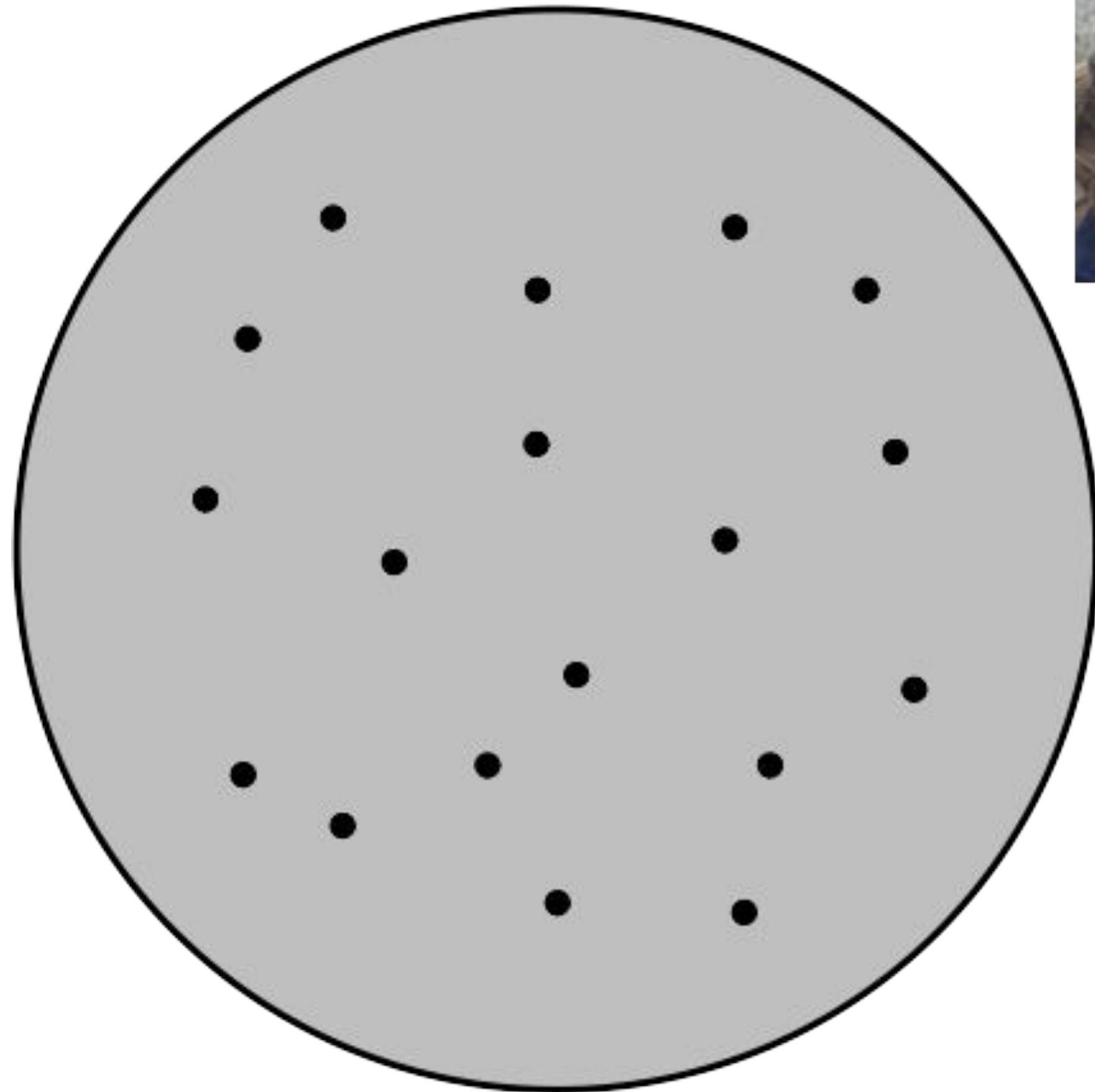


- Learn meaningful latent directions
- Either supervised or unsupervised
- Apply these directions in any image
- Example of directions in faces:
 - Aging
 - Smiling
 - Hair or skin color
 - Gender

[Unsupervised: Voynov and Babenko. "Unsupervised discovery of interpretable directions in the gan latent space." In ICML 2020]

Supervised Learning of Latent directions

Latent space



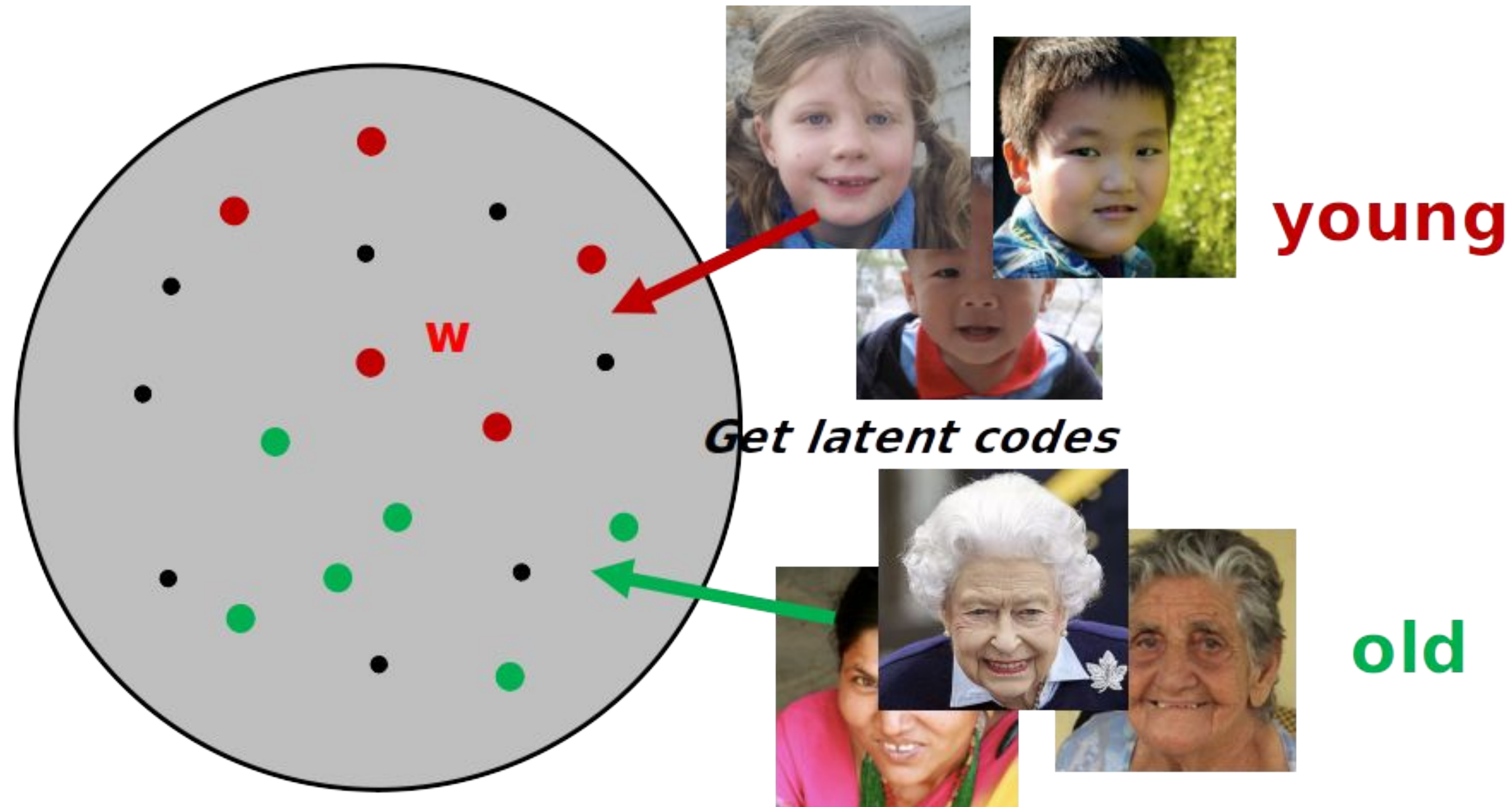
young



old

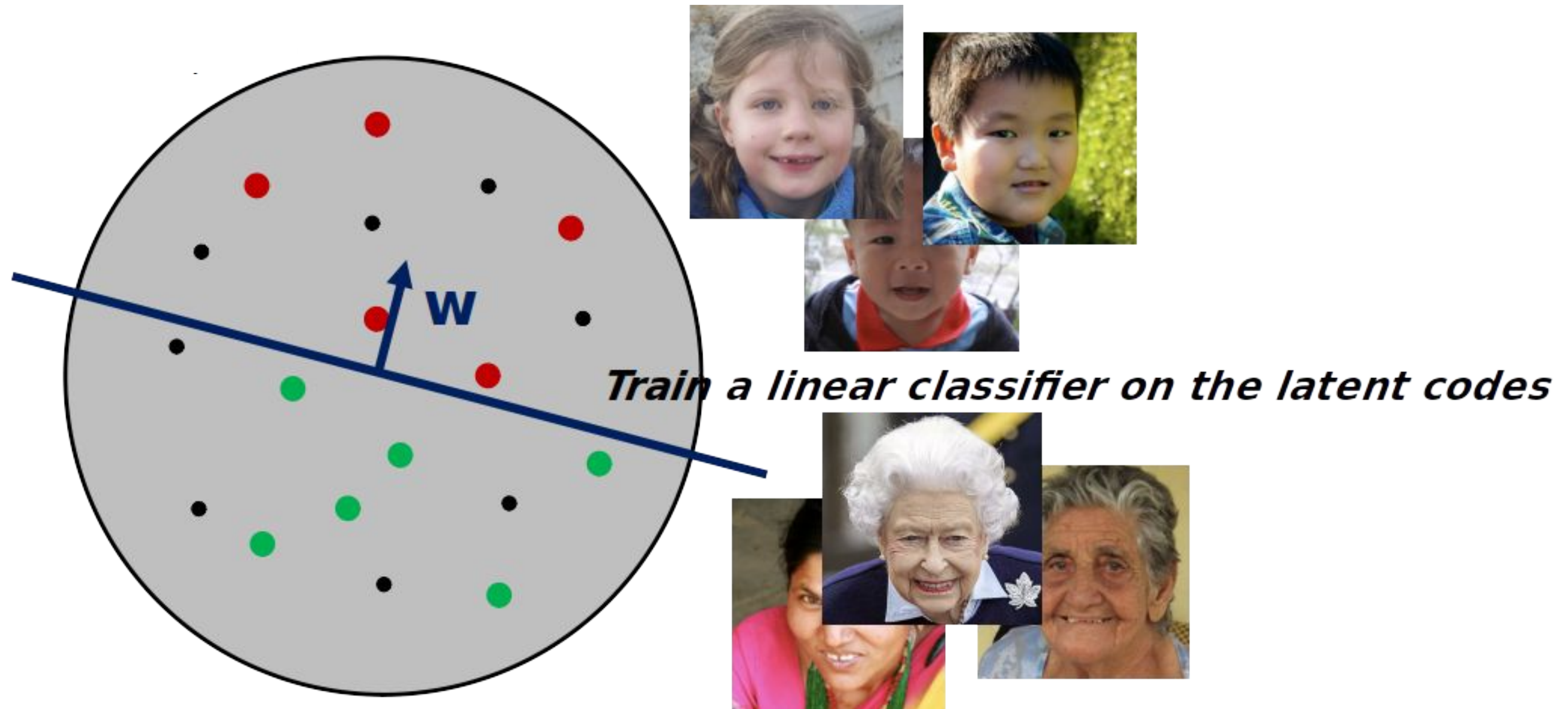
Supervised Learning of Latent directions

Latent space



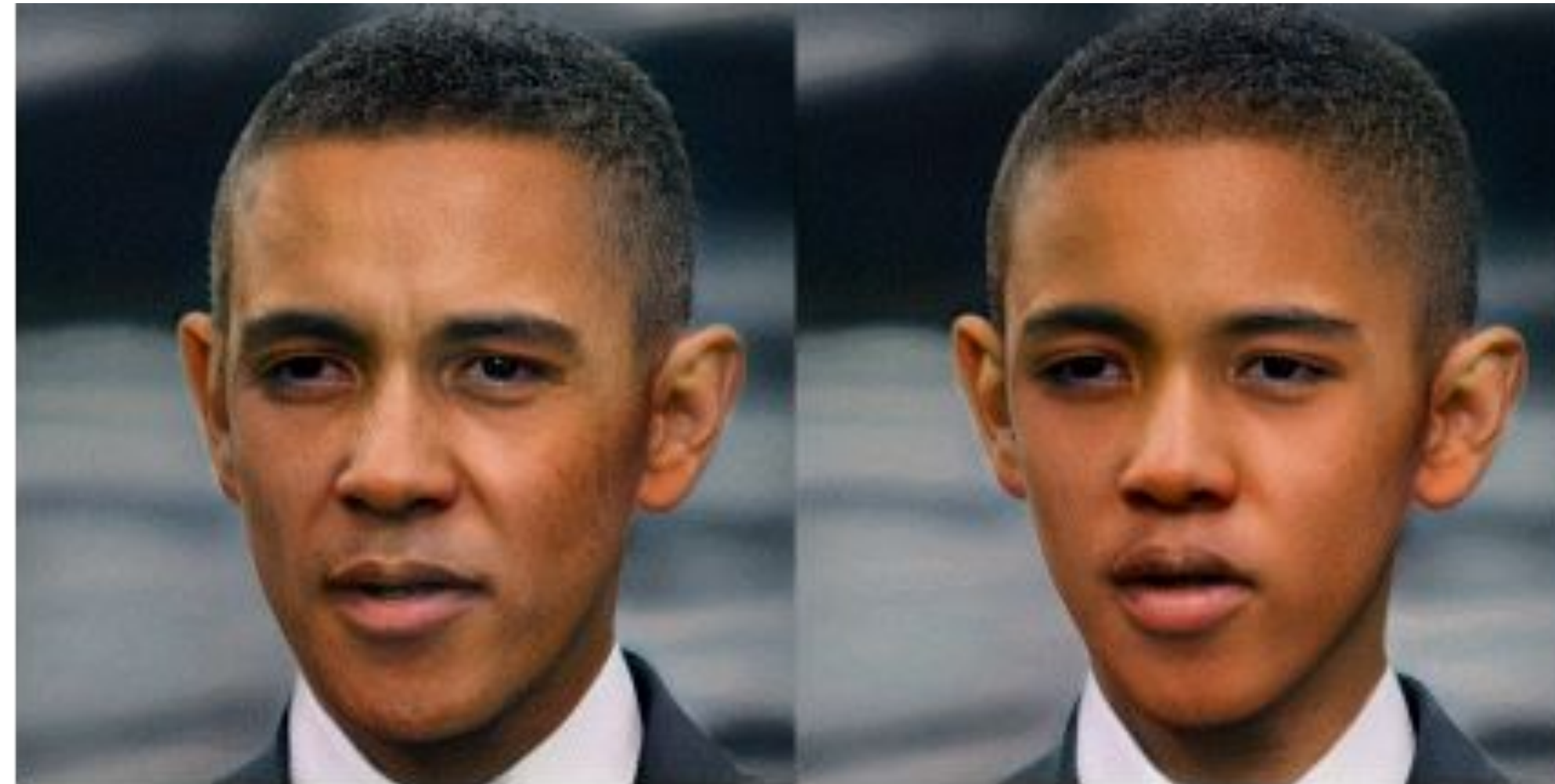
Supervised Learning of Latent directions

Latent space

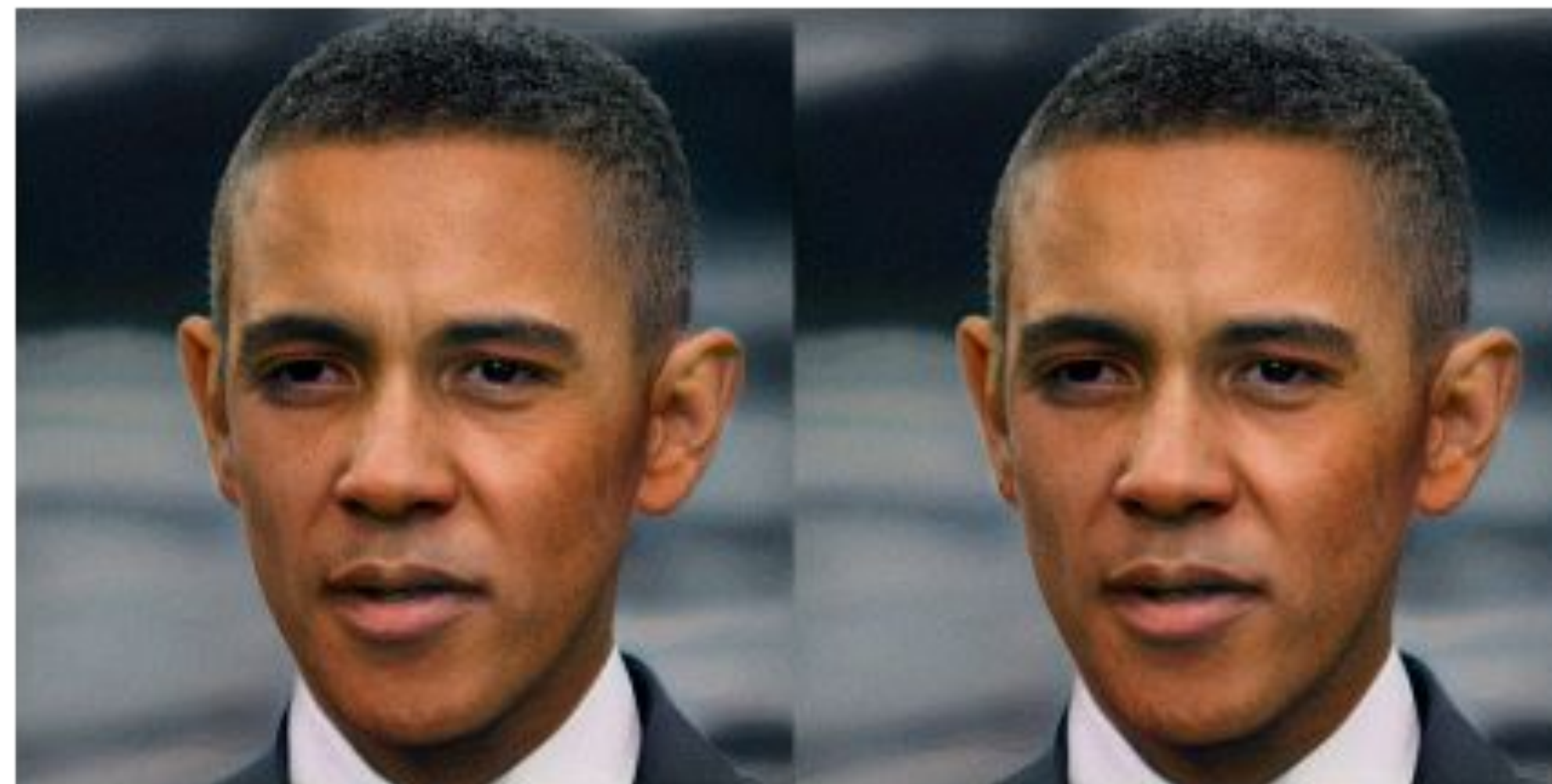


Supervised Learning of Latent directions

aging



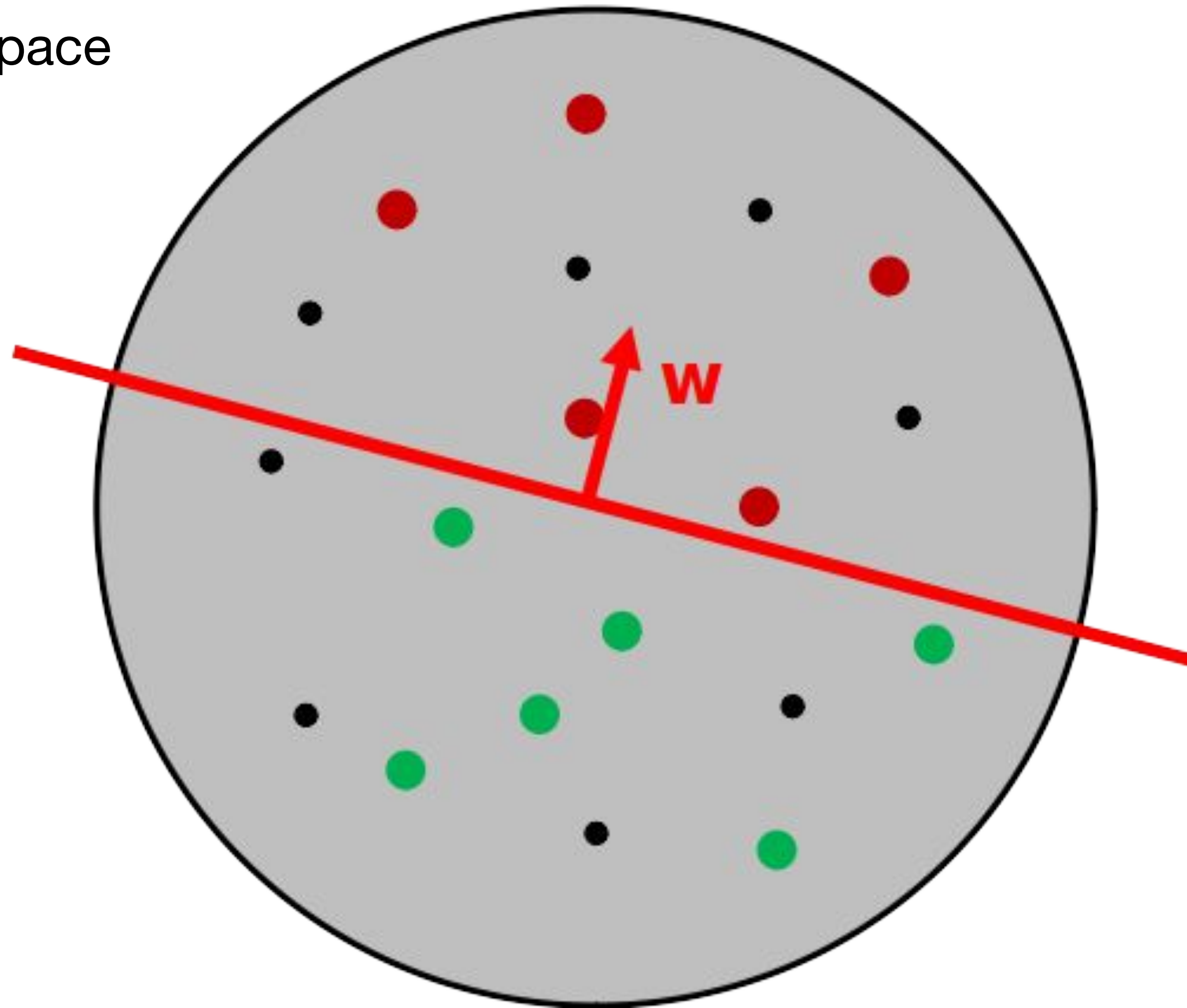
smiling



$$z_{\text{new}} = z + w * m$$

Supervised Learning of Latent directions

Latent space



$$z_{\text{new}} = z + w * m$$

m: magnitude

Image generation from text

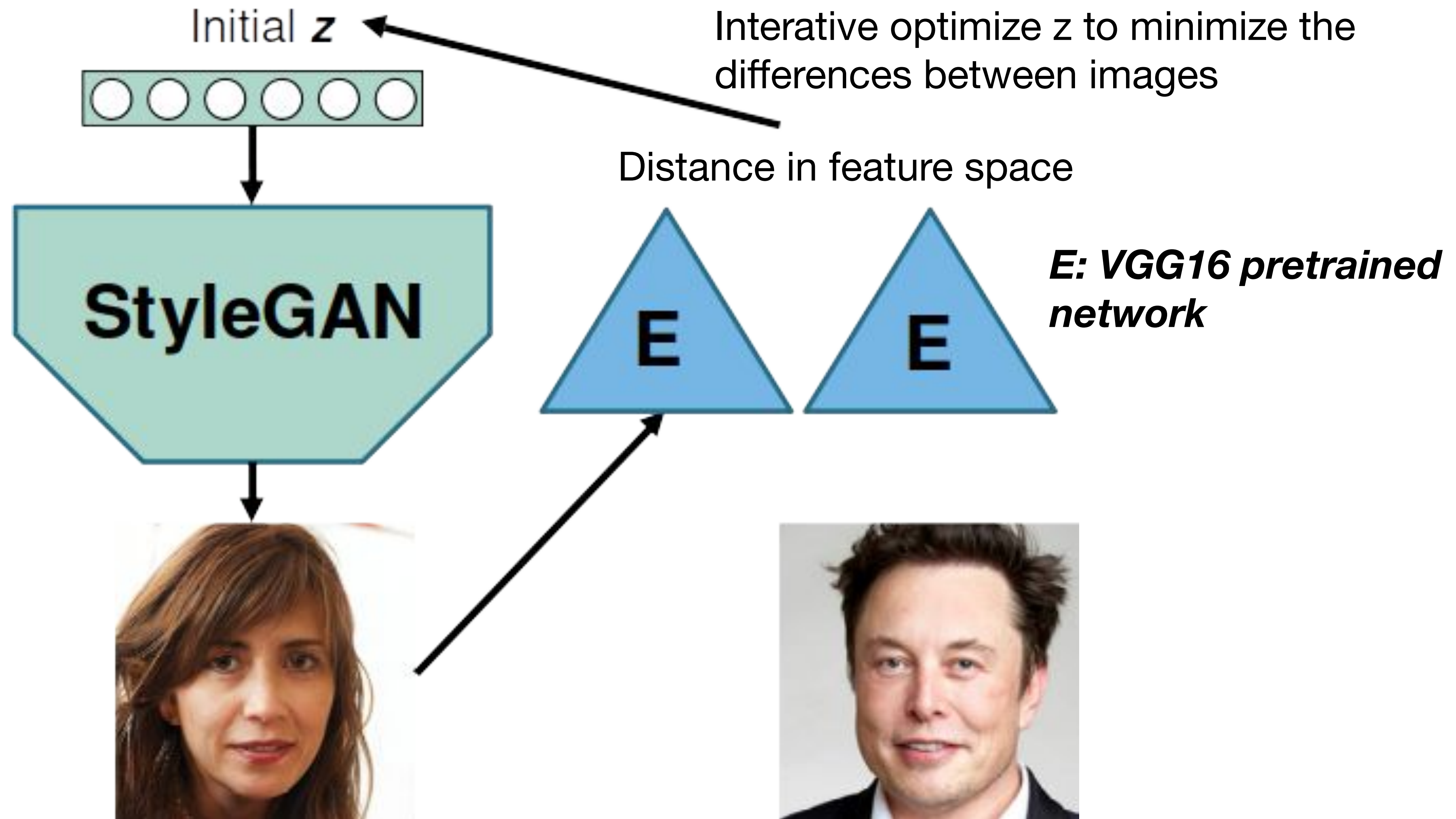


Image generation from text

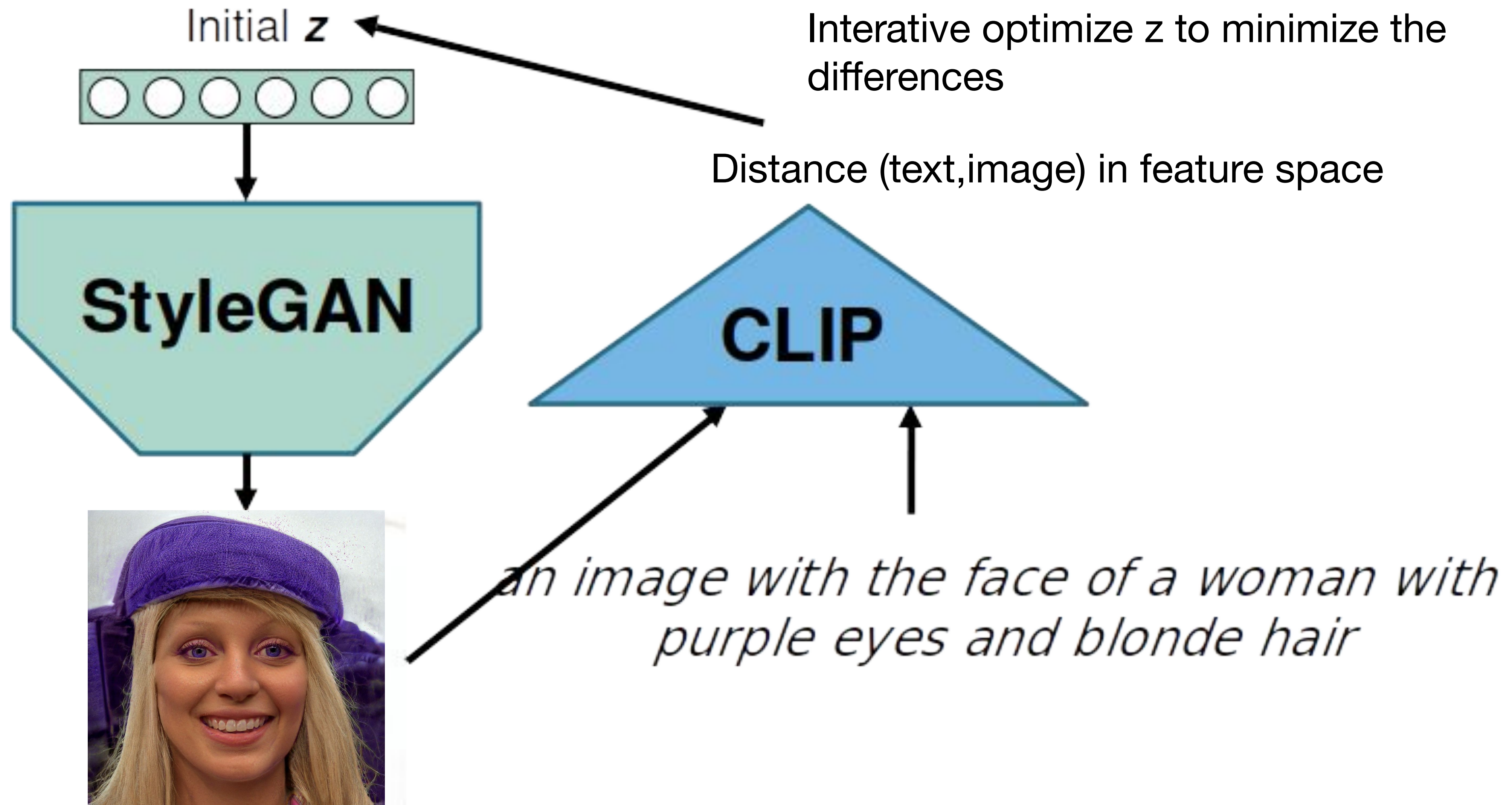
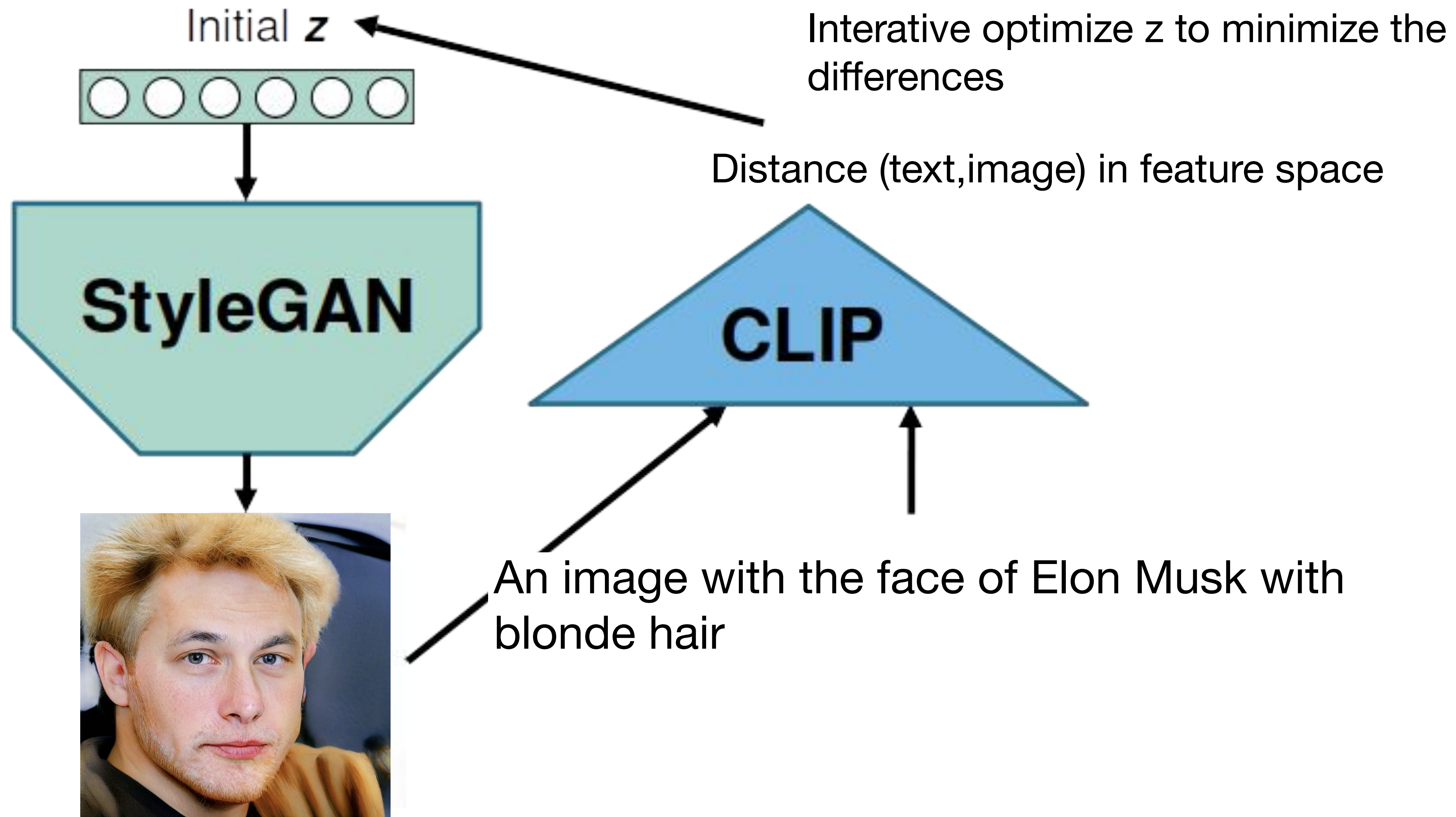
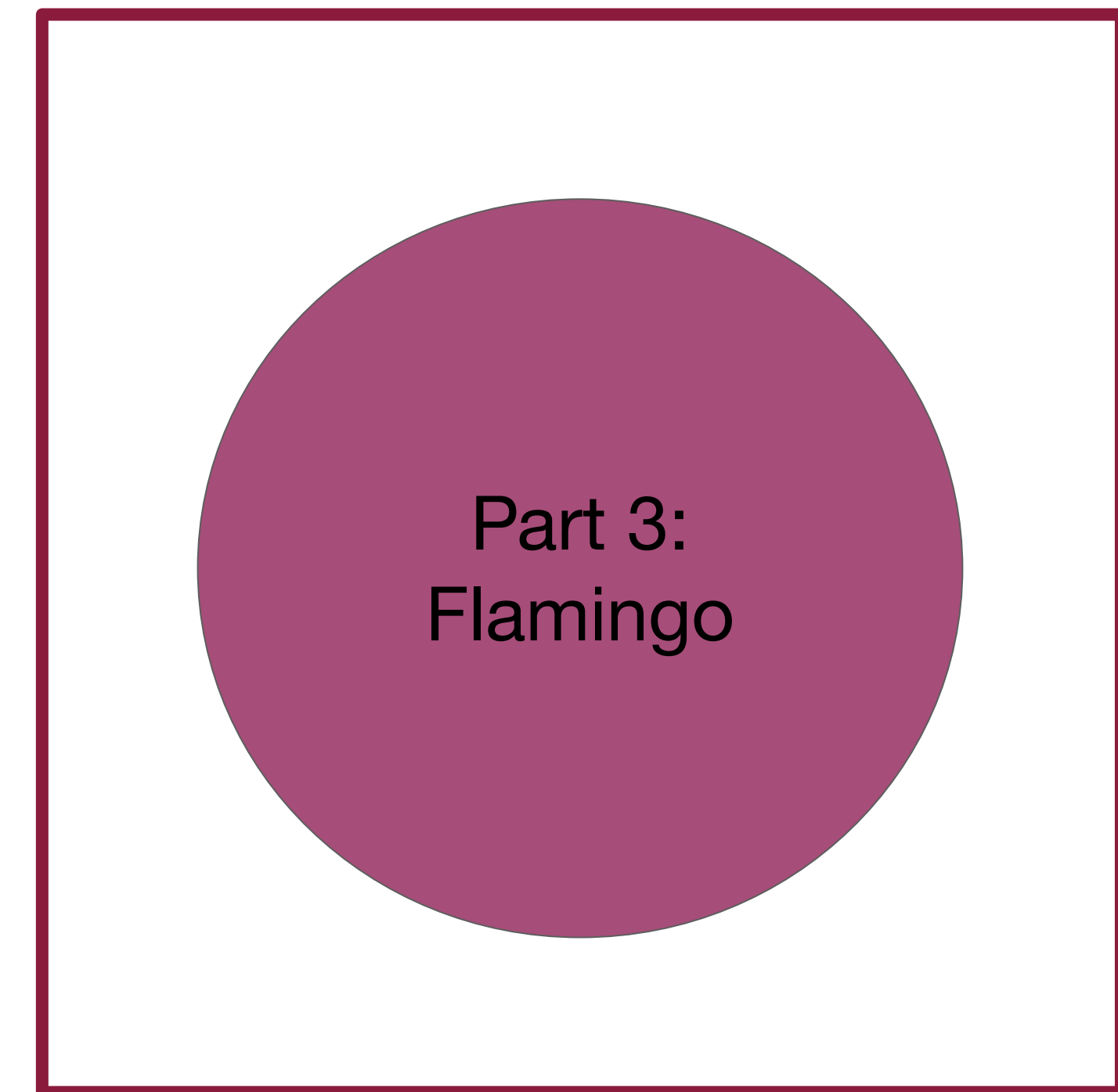
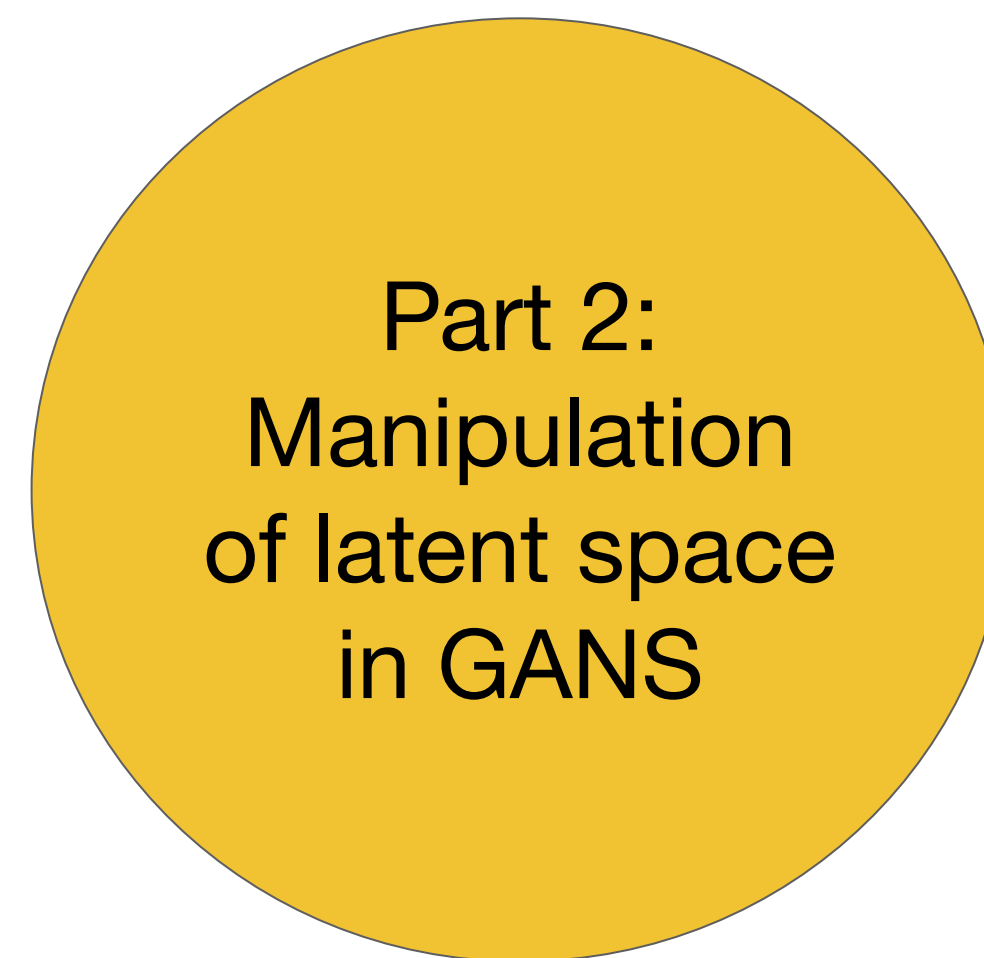
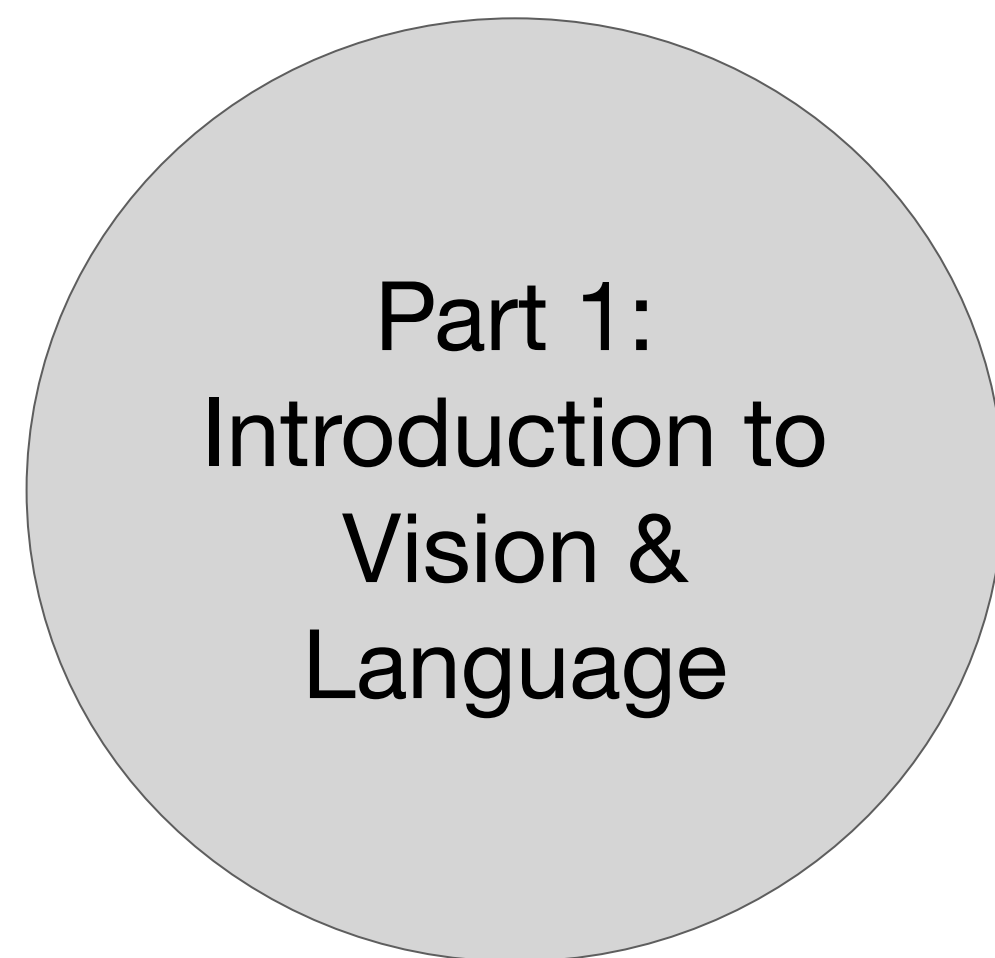


Image generation from text



What we will see today



Slides adapted from: [Vicky Kalogeiton, Andrej Karpathy, Justin Johnson, Jerry Zhu, Al Summer, Ishan Misra, Christian Rupprecht, Dim P. Papadopoulos, Samuel Albanie]

Part 3: Outline

- Motivation
- Challenges for multimodal generative modelling
- Flamingo Model
- Results

Motivation: The few-shot dream

- Aspect of intelligence: ability to quickly learn a task given short instruction
 - Fast acquisition of categories in children (Markman et al., 1989)
 - Model learning environment to make better use of data (Griffiths et al. 2019)

Motivation: The few-shot dream

- Aspect of intelligence: ability to quickly learn a task given short instruction
 - Fast acquisition of categories in children (Markman et al., 1989)
 - Model learning environment to make better use of data (Griffiths et al. 2019)
- We'd like **multimodal systems** (vision and language) that achieve this property
- **Dominant computer vision paradigm:**

Large-scale pretraining

+

Task-specific fine-tuning

Motivation: The few-shot dream

- Aspect of intelligence: ability to quickly learn a task given short instruction
 - Fast acquisition of categories in children (Markman et al., 1989)
 - Model learning environment to make better use of data (Griffiths et al. 2019)
- We'd like **multimodal systems** (vision and language) that achieve this property
- **Dominant computer vision paradigm:**

Large-scale pretraining

+

Task-specific fine-tuning

- But current fine-tuning approaches often require:
 - thousands of training samples
 - careful per-task hyperparameter tuning
 - significant computationally resource

Motivation: The few-shot dream

- **Can we train a multimodal model to work well in a “few-shot” regime?**
 - Fast acquisition of categories in children (Markman et al., 1983)
 - Model learning environment to make better use of data (Griffiths et al. 2019)
- We'd like **multimodal systems** (vision and language) that achieve this property
- **Dominant computer vision paradigm:**

Large-scale pretraining

 +

Task-specific fine-tuning
- But current fine-tuning approaches often require:
 - thousands of training samples
 - careful per-task hyperparameter tuning
 - significant computational resource

Motivation: Open-ended task abilities

- Multimodal models like CLIP have shown promising zero-shot performance, but they are **inflexible**:
 - they lack the ability to generate language
- Flexible models for visually-conditioned language generation like VL-T5 exist
- But these have not demonstrated strong few-shot performance

Motivation: Open-ended task abilities

- Inspiration from NLP: large language models like GPT-3 are flexible few-shot learners
- Given a few examples of a task as a prompt -> query input, the language model generates a continuation to produce a predicted output

Motivation: Open-ended task abilities

- Inspiration from NLP: large language models like GPT-3 are flexible few-shot learners
- Given a few examples of a task as a prompt -> query input, the language model generates a continuation to produce a predicted output
- A key factor of their success is large-scale pretraining
- In principle: image/video understanding tasks (e.g. classification, captioning, question answering) are **text prediction problems** with visual input conditioning

Motivation: Open-ended task abilities

Can we learn a model capable of open-ended multimodal tasks via pretraining?

FEW-SHOT LEARNERS

- Given a few examples of a task as a prompt + query input, the language model generates a continuation to produce a predicted output
- A key factor of their success is large-scale pretraining
- In principle: image/video understanding tasks (e.g. classification, captioning, question answering) are **text prediction problems** with visual input conditioning

Unifying strong unimodal models

- Training large language models is extremely **computationally expensive**
- We'd like to save computational resources by starting from a **pretrained** language model
- But a text-only model has no build-in way to **incorporate input** from other modalities.
- We want to enable this while **retaining the knowledge** of the original language model
- **Proposed approach**: interleave cross-attention layers with language-only self-attention layers (frozen)

Supporting images and videos

- **Goal:** enable both images and video inputs
- These are high-dimensional, so flattening to 1D sequences (as used in text-generation) is costly
- Exacerbated by quadratic cost of self-attention
- **Secondary goal:** would also like a unified treatment of images and video
- **Proposed approach:** Perceiver-based architecture with a fixed number of visual tokens

Heterogeneous training data

- Large models require vast training dataset.
 - Existing (image, text) datasets used by (e.g. used by CLIP and ALIGN) may not be **general** enough to reach GPT-3 style few-shot learning.
 - Large internet-based text-only datasets exist, but not for multimodal data.
 - **One scalable approach:** scrape web pages with interleaved images and text. But such images and text are often only weakly related
-
- **Proposed approach:** combine web scraping with existing paired (image, text) and (video, text) datasets

Flamingo model

- Visual language model that accepts interleaved inputs:

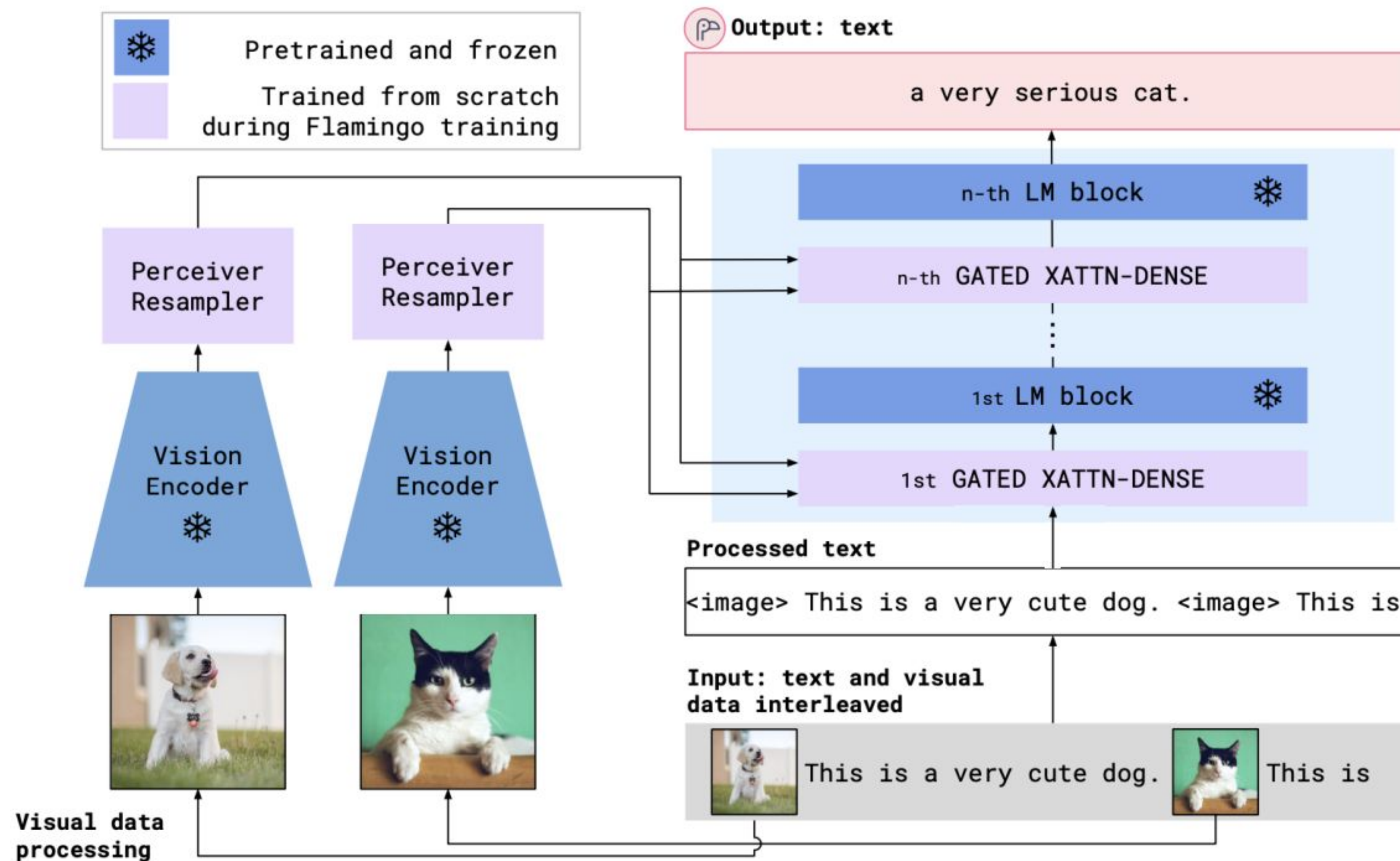


- This enables a broad range of tasks:
 - Open-ended tasks:
 - Visual question answering
 - Captioning
 - Close-ended tasks
 - Classification

Flamingo model

- **Goal 1:** leverage pretrained models to save compute
 - Vision: CLIP
 - Language: Chinchilla
- **Goal 2:** bridge pretrained models harmoniously
 - Perceiver resampler
 - Cross attention

Flamingo model



Vision encoder: pixels to features

- Vision encoder: F6 Normalizer-Free ResNet (NFNet) backbone
 - Pretrained as dual encoder using contrastive loss employed by CLIP

Vision encoder: pixels to features

- Vision encoder: F6 Normalizer-Free ResNet (NFNet) backbone
 - Pretrained as dual encoder using contrastive loss employed by CLIP
 - BERT is used for the text encoder (discarded after pretraining)
- Slight difference to CLIP: global average pooling is used to produce the vision embedding (rather than global attention pooling)
 - Input Resolution 288 x 288 pixels
 - Embedding 1376
- **Outputs 2D** spatial grid of features which is flattened to 1D
- **For videos:** frames are sampled at 1FPS (features are concatenated)

Vision encoder is frozen after pre-training

Pre-trained data

- Trained on a combination of two internal (image, text) datasets:
 - ALIGN (1.8 billion) noisy
 - LTIP (312 million) - cleaner, longer descriptions
- The manner of combination is important for the performance

Pre-trained data

- (Ablation study) small NFNet-F0 with BERT-Mini for different regimes:

Dataset	Combination strategy	ImageNet accuracy top-1	COCO					
			image-to-text			text-to-image		
			R@1	R@5	R@10	R@1	R@5	R@10
LTIP	None	40.8	38.6	66.4	76.4	31.1	57.4	68.4
ALIGN	None	35.2	32.2	58.9	70.6	23.7	47.7	59.4
LTIP + ALIGN	Accumulation	45.6	42.3	68.3	78.4	31.5	58.3	69.0
LTIP + ALIGN	Data merged	38.6	36.9	65.8	76.5	15.2	40.8	55.7
LTIP + ALIGN	Round-robin	41.2	40.1	66.7	77.6	29.2	55.1	66.6

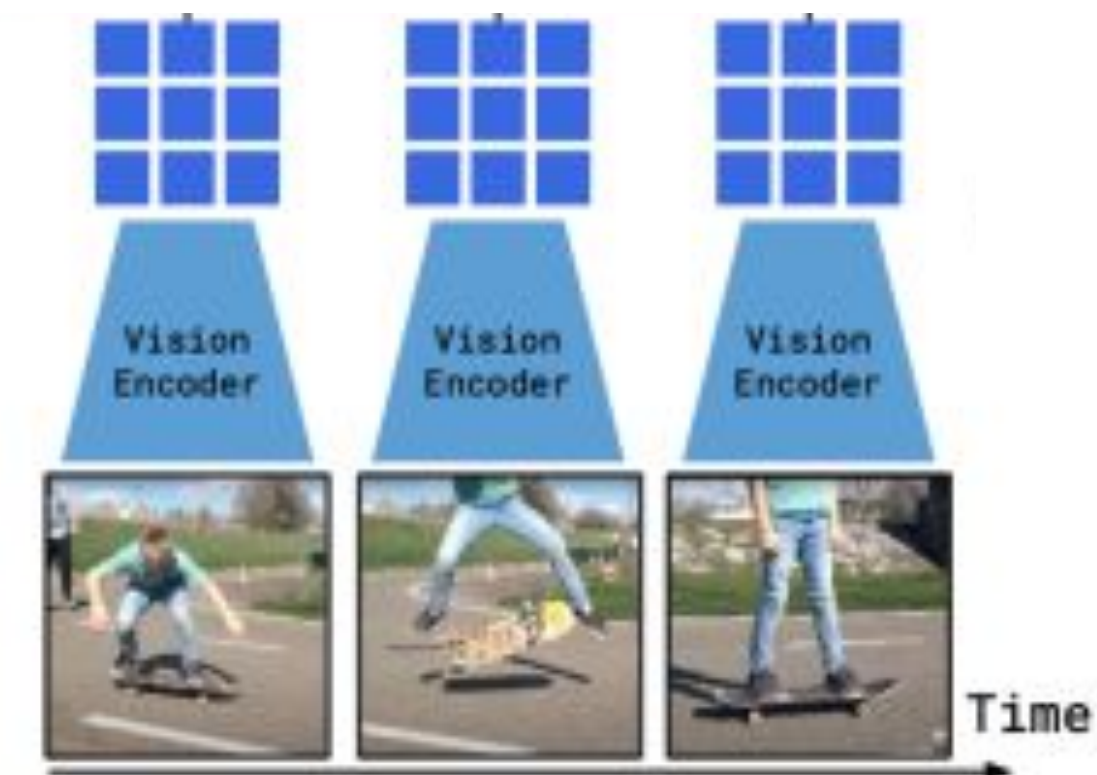
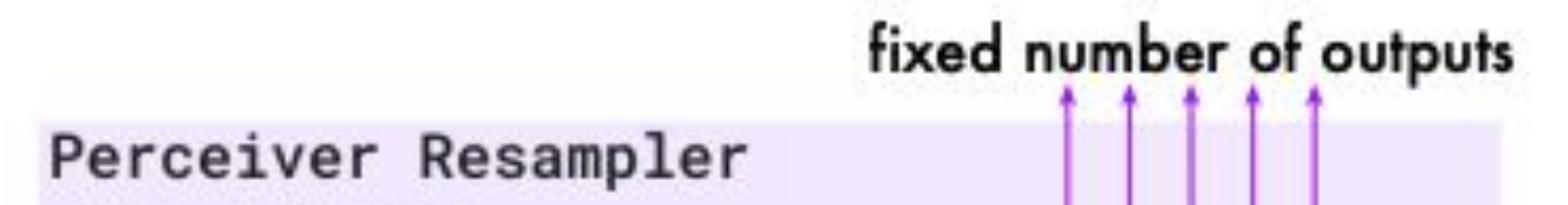
Accumulation: compute gradient on batch from each dataset, combine via weighted sum

Data merged: merge examples from each dataset into each batch

Round-robin: alternate batches from each dataset, update parameters each batch

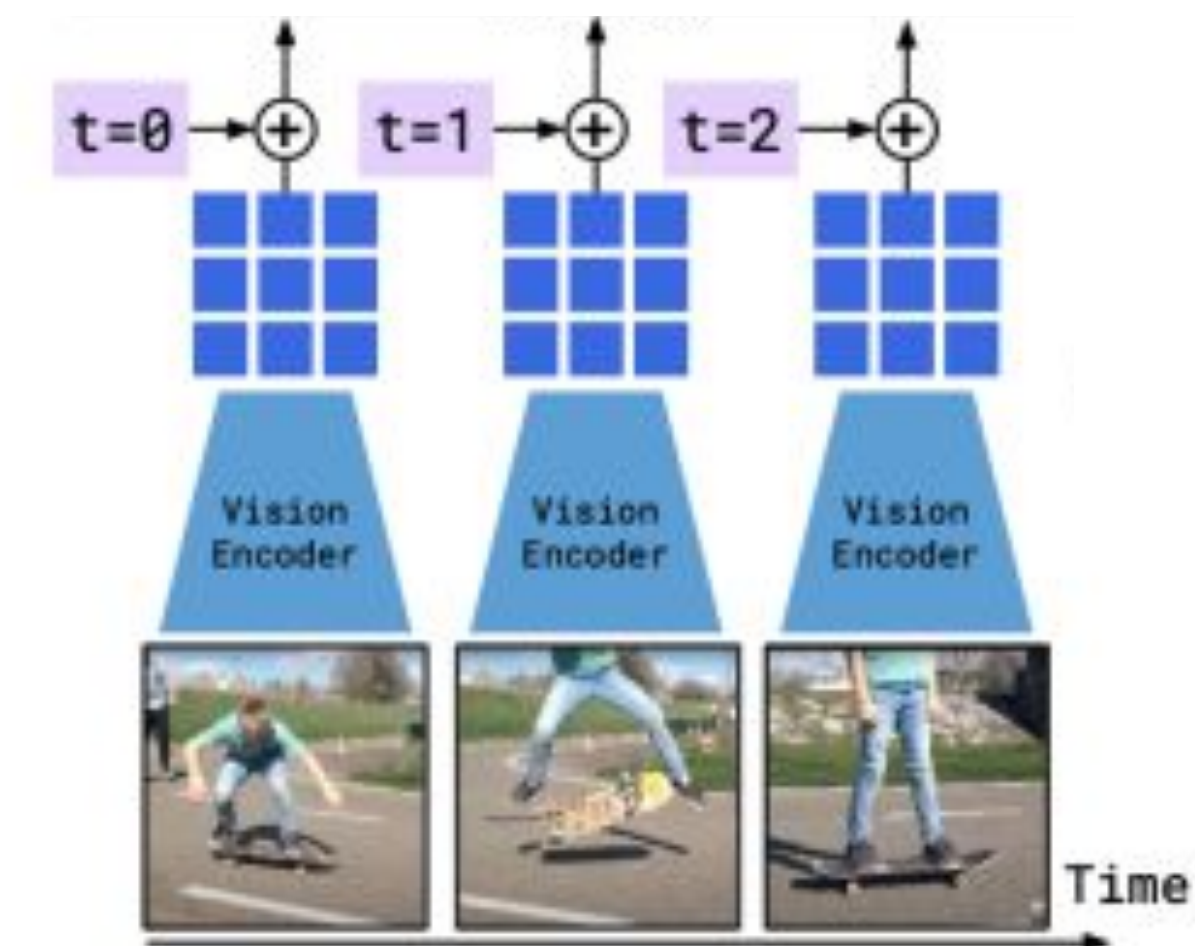
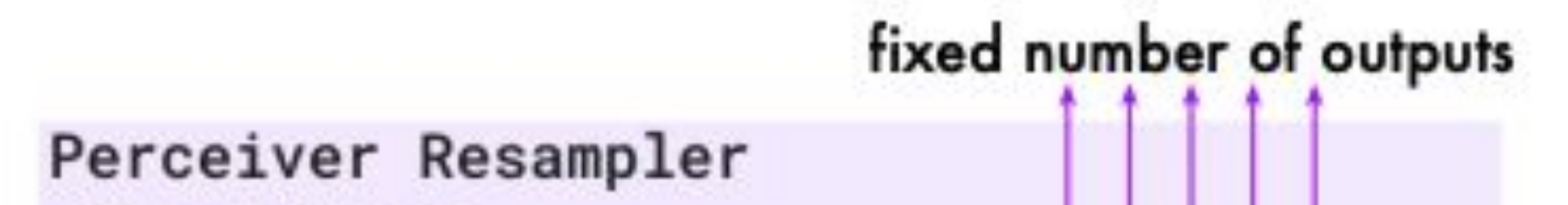
Perceiver resampler module

- A variable number of input frames are processed (for videos)
- The vision encoder thus produces a variable number of features
- It outputs a fixed number of visual tokens (64) to limit complexity



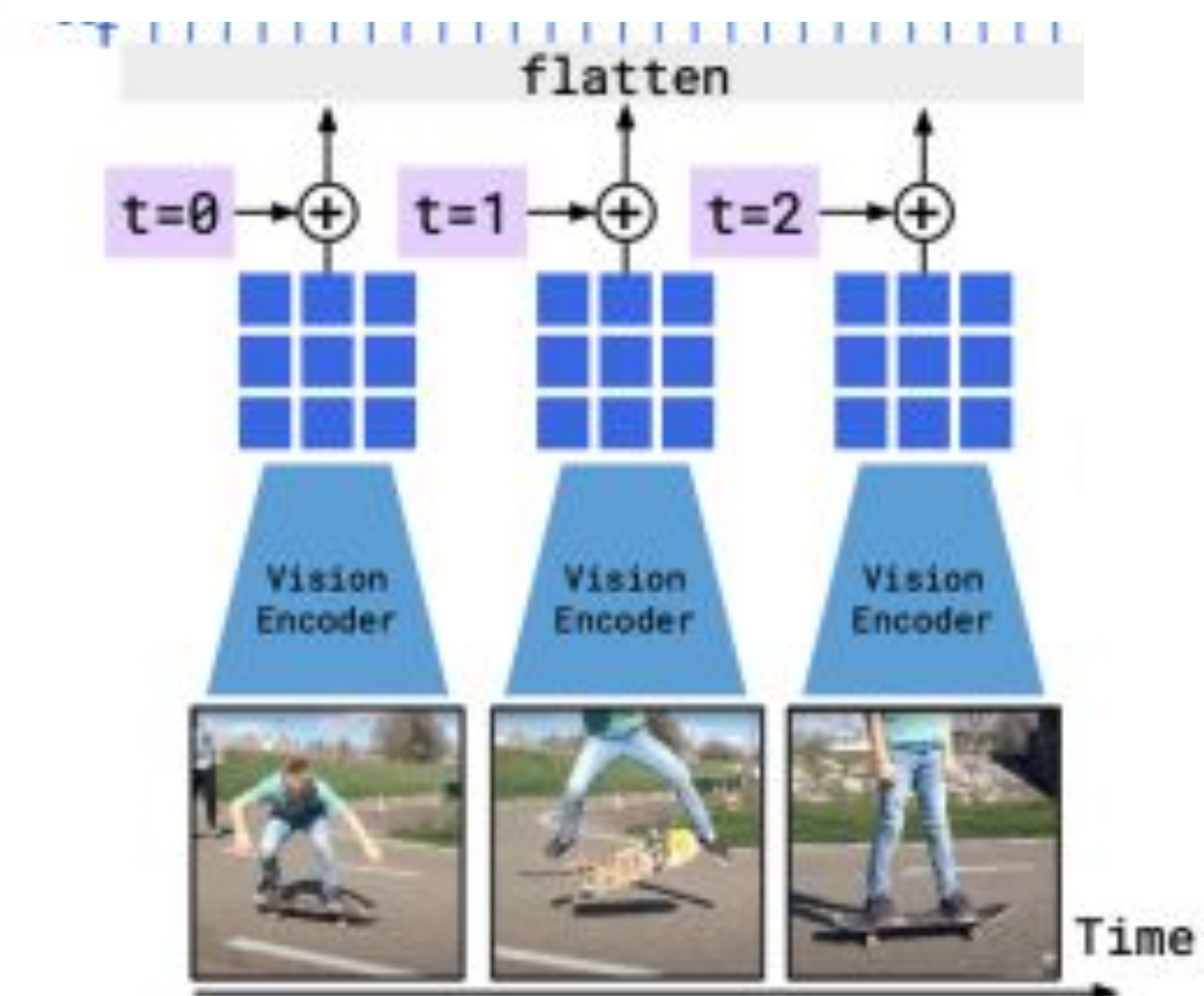
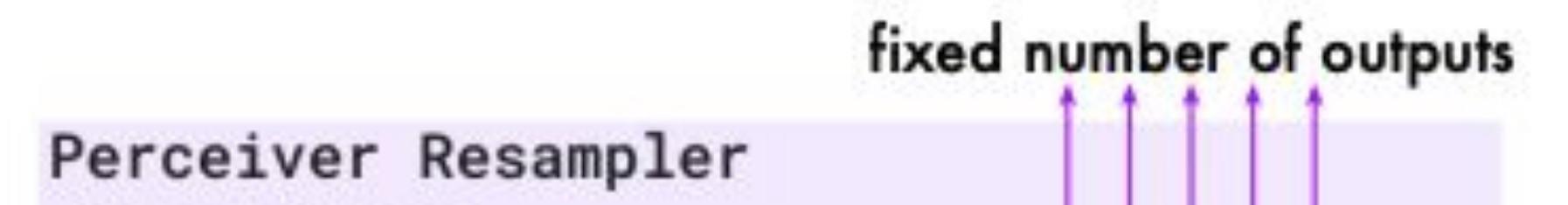
Perceiver resampler module

- A variable number of input frames are processed (for videos)
- The vision encoder thus produces a variable number of features
- It outputs a fixed number of visual tokens (64) to limit complexity
- Temporal encodings are added to visual inputs (spatial grid position encodings are not used, since they did not help)



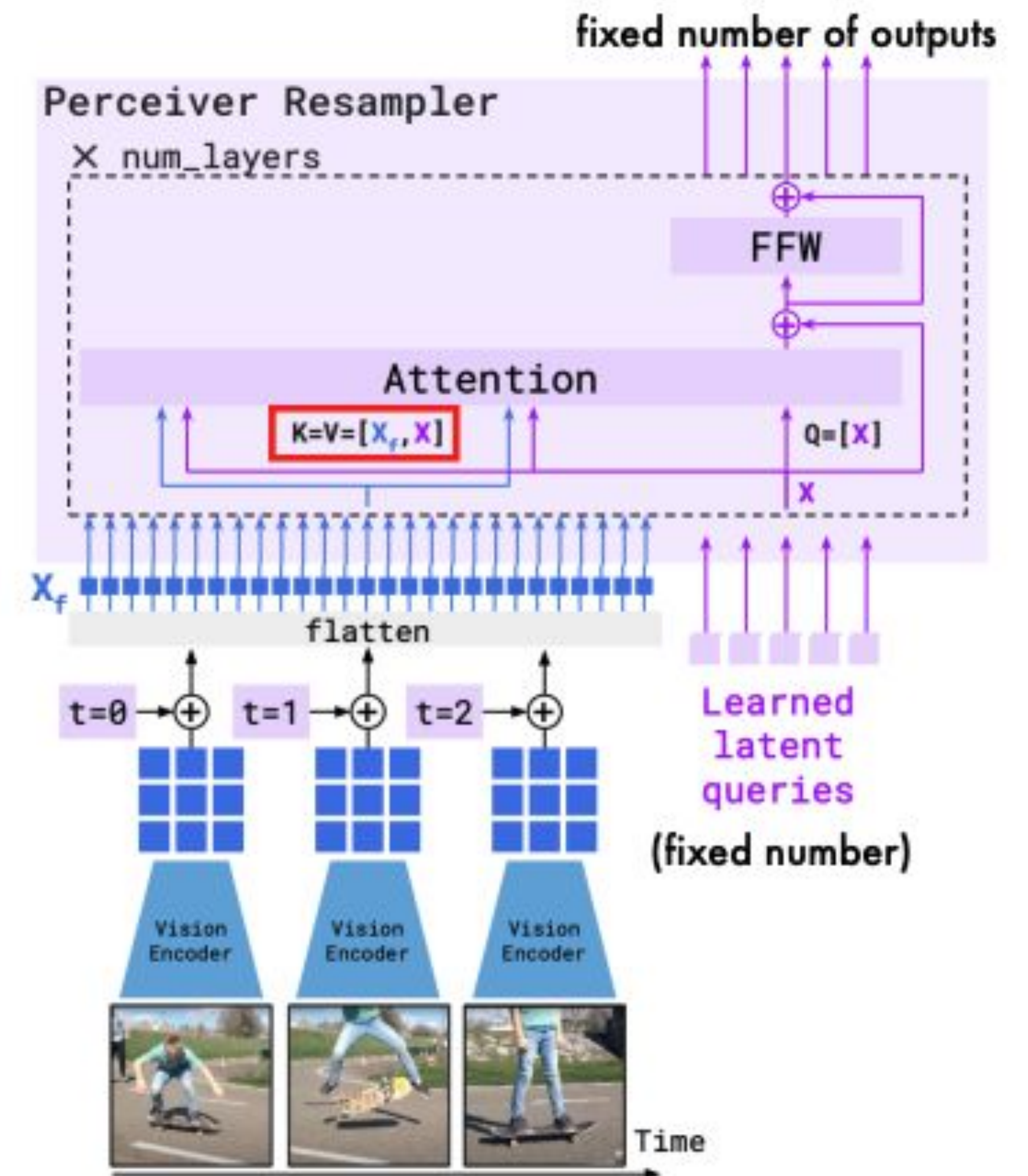
Perceiver resampler module

- A variable number of input frames are processed (for videos)
- The vision encoder thus produces a variable number of features
- It outputs a fixed number of visual tokens (64) to limit complexity
- Temporal encodings are added to visual inputs (spatial grid position encodings are not used, since they did not help)
- The results are then flattened to form a 1D sequence
- These are combined with a fixed set of learned latent queries (64)



Perceiver resampler module

- A variable number of input frames are processed (for videos)
- The vision encoder thus produces a variable number of features
- It outputs a fixed number of visual tokens (64) to limit complexity
- Temporal encodings are added to visual inputs (spatial grid position encodings are not used, since they did not help)
- The results are then flattened to form a 1D sequence
- These are combined with a fixed set of learned latent queries (64)
- Both are processed by attention and feed-forward layers
- **Note:** differently to DETR and Perceiver, keys and values for latent queries are concatenated to those from the visual embeddings

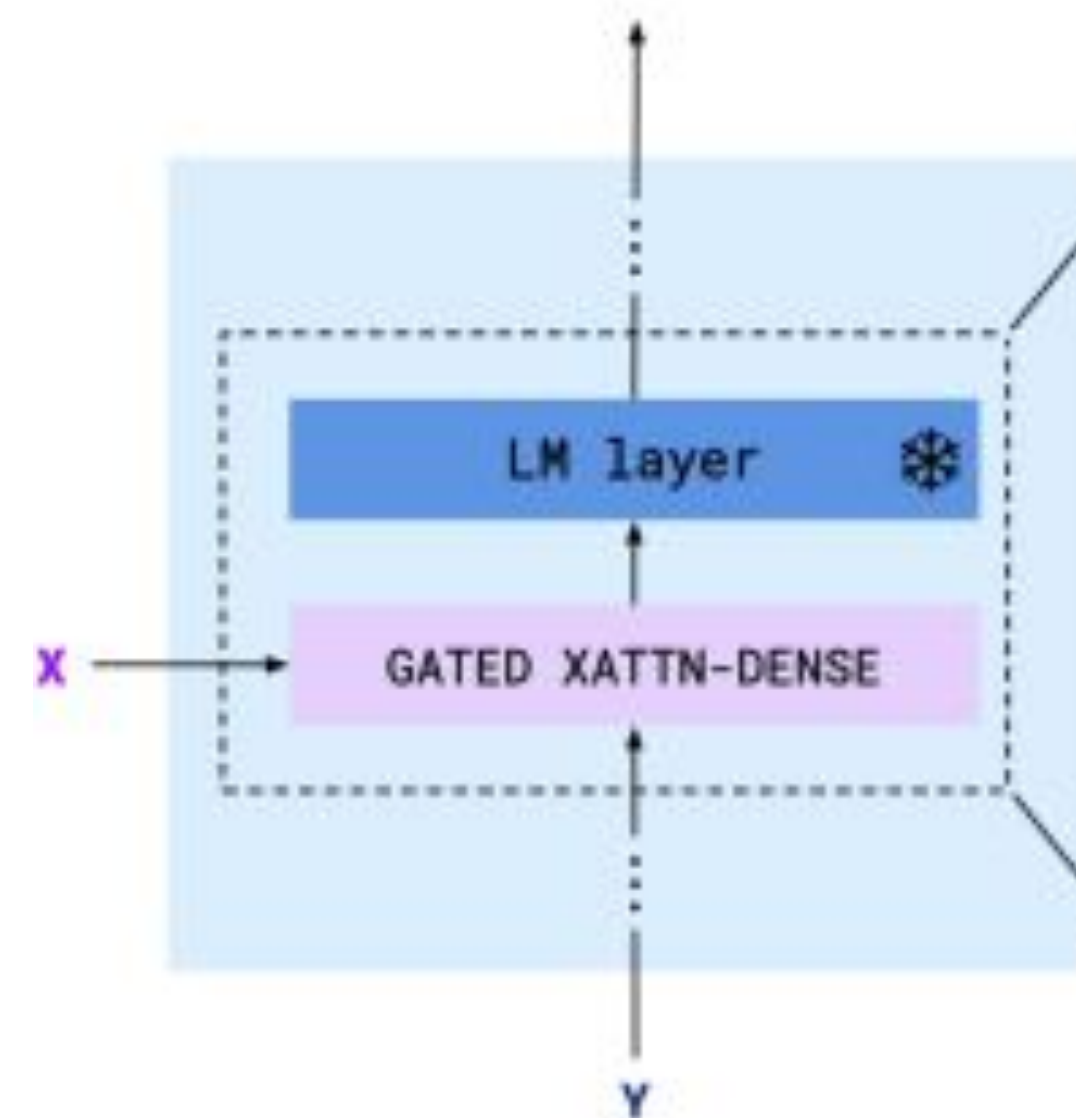


Conditioning the language model

- Language models: frozen Chinchillas (trained on MassiveText)

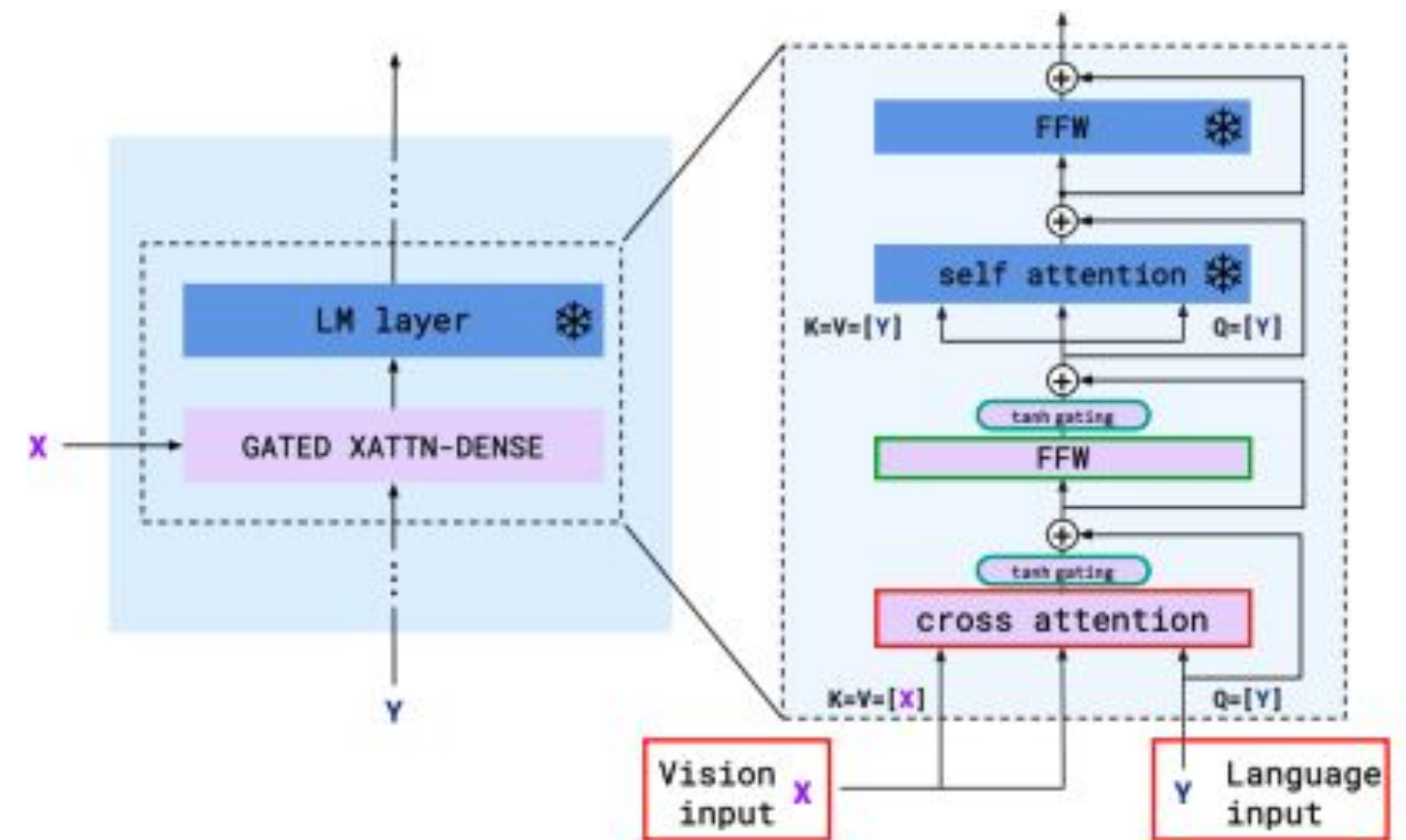
Conditioning the language model

- Language models: frozen Chinchillas (trained on MassiveText)
- Gated xattn dense blocks (trained from scratch) are inserted between layers






Conditioning the language model










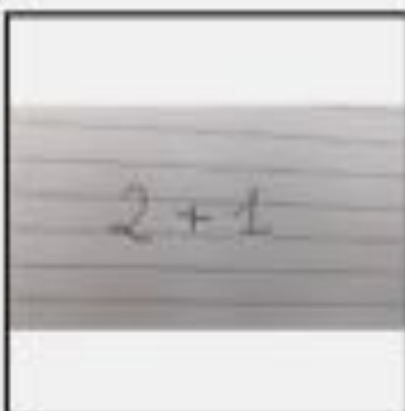
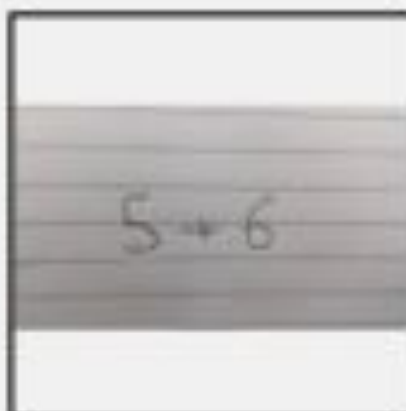
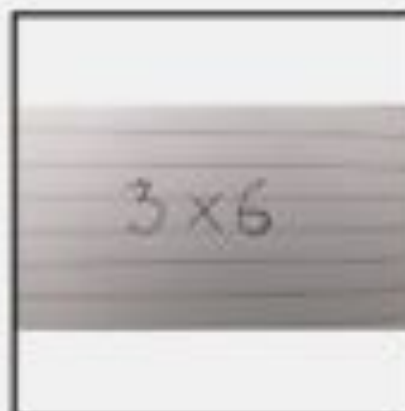
- Each block includes:
 - Cross attention
 - Feed-forward
- Layer norm is applied to all attention inputs and the feed-forward layers (GPT-2 style)
- Use tanh gates to preserve original language model behaviour at initialisation
- Each $\tanh(\alpha)$ gate controlled via a layer-specific learnable scalar α (initialised to zero)




Text completion of image and text prompts

Input Prompt	 <p>Question: What do you think the capacities of these are? Answer:</p>	 <p>Question: What is odd about this image? Explain why it is unusual. Answer:</p>	 <p>Question: What country is this? Why do you think so? Answer:</p>
Completion	<p>The floppy disk is 1.44MB and the CD is 700MB.</p>	<p>The image is odd because the elephant is in the back of the truck. It is unusual because elephants are not usually transported in the back of a truck.</p>	<p>It is Canada. I think so because the flag is the Canadian flag.</p>

Text completion of image and text prompts

Input Prompt					Completion	
	This is a chinchilla. They are mainly found in Chile.		This is a shiba. They are very popular in Japan.		This is	a flamingo. They are found in the Caribbean and South America.
	What is the title of this painting? Answer: The Hallucinogenic Toreador.		Where is this painting displayed? Answer: Louvres Museum, Paris.		What is the name of the city where this was painted? Answer:	Arles.
	Output: "Underground"		Output: "Congress"		Output:	"Soulomes"
	2+1=3		5+6=11			3x6=18

Dialogues



This is a dog. It's a white fluffy dog.

What season is this?

It looks like a summer day.

Why do you think so?

The grass is green and the dog is not wearing a sweater.

What breed is the dog?


It's a Samoyed.

Is the dog running?

No, it's sitting.

Can you describe the pose of its back legs?

The back legs are bent and the dog is sitting on its haunches.



Which city is this?

This is a street sign in New York City.

Which street corner is it?

It is the intersection of Madison Avenue and East 42nd Street.

Which subway station is nearby?

The closest subway station is Grand Central Station.

How about the nearest park?

The nearest park is Bryant Park.

There is another sign. What does it say?

It says "One way".

I would like you to read the color and then say the color it is written in. For example:

PURPLE

Color is "Purple" and it is written in red.

GREEN

Color is "Green" and it is written in blue.

YELLOW

Color is "Yellow" and it is written in green.

BLACK

Color is "Black" and it is written in yellow.

Well done! Do you know the name of the test these images come from?



I think it is called the Stroop test.

Can you explain how humans perform in this test?

Humans are slower when the color of the word and the color of the word are different.

How about you?

I am not affected by this difference.



What is the difference between these two images?

The first one is a triangle and the second one is a hexagon.

Which one has larger internal angles?

The hexagon has larger internal angles.

What color are they?

They are both red.

Videos

