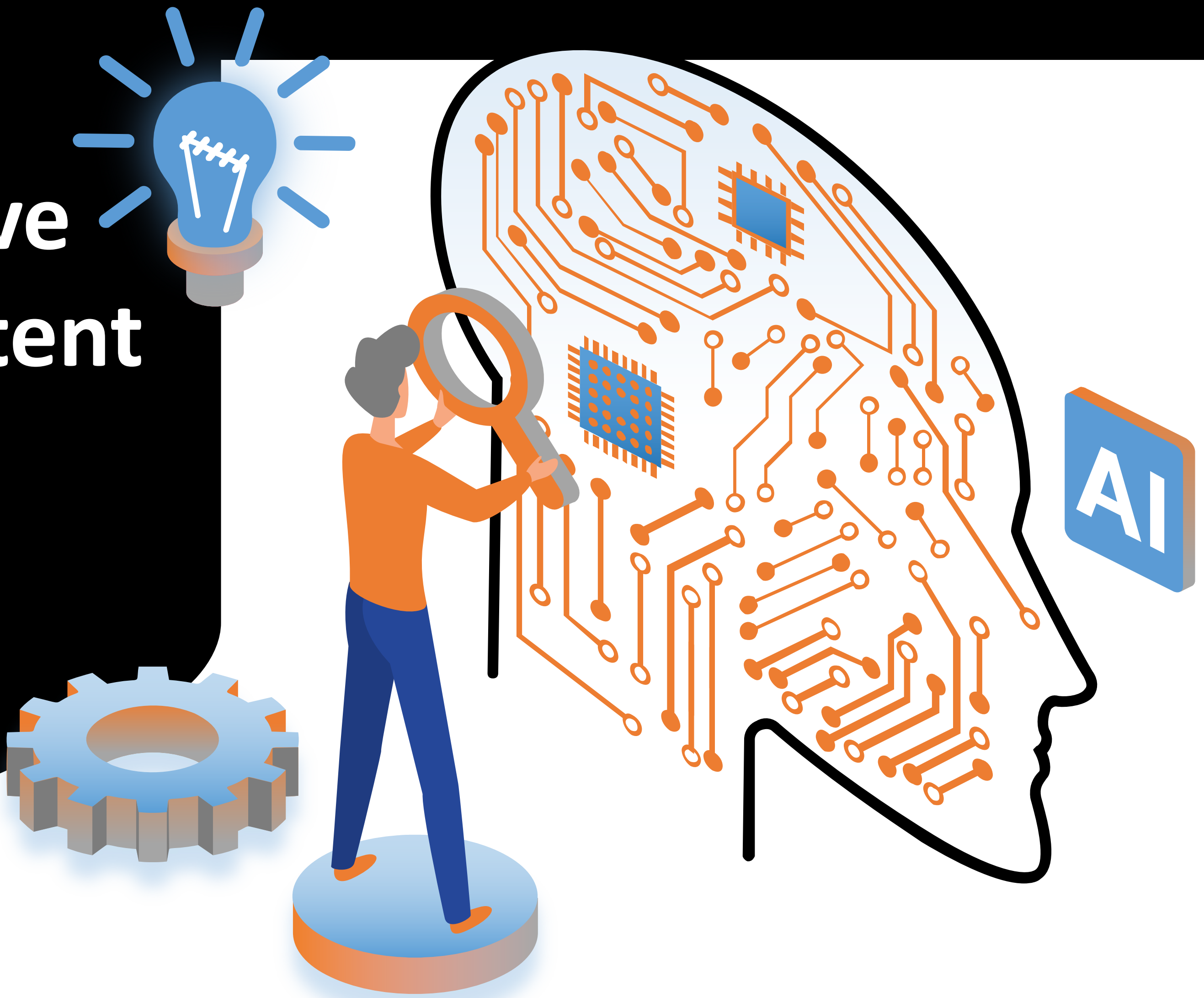
Harnessing Generative Priors for Visual Content Restoration



Chen-Change Loy

Last update: 22 Jan 2024

2024 Winter School on Biometrics

Outline

- **Introduction**
  - Problem objective
  - Challenges
  - Architectures
  - Losses
  - Handling complex degradation
  - Metric
- **Prior for Face Restoration**
- **CodeFormer**

Introduction

Problem objective

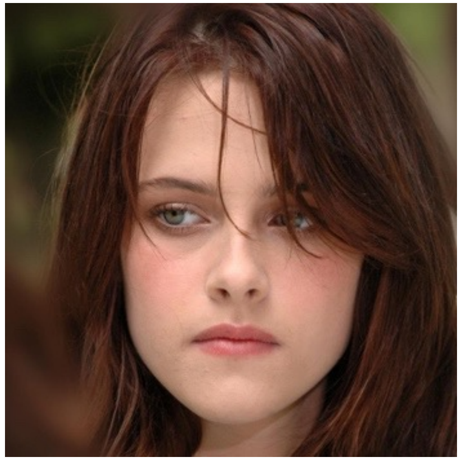
Recover the latent high-quality (HQ) faces $\mathbf{x}$ from its degraded low-quality (LQ) faces $\mathbf{y}$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$$

where $\mathbf{H}$ is a degradation matrix, $\mathbf{v}$ is additive noise

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \underbrace{\frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2}_{\text{fidelity term}} + \underbrace{\lambda\Phi(\mathbf{x})}_{\text{regularization term}}$$

LQ



HQ

Problem objective

Recover the latent high-quality (HQ) faces $\mathbf{x}$ from its degraded low-quality (LQ) faces $\mathbf{y}$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$$

where $\mathbf{H}$ is a degradation matrix, $\mathbf{v}$ is additive noise

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \underbrace{\frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2}_{\text{fidelity term}} + \underbrace{\lambda\Phi(\mathbf{x})}_{\text{regularization term}}$$

If we know the $\mathbf{H}$ and $\mathbf{v}$, then is a non-blind super-resolution. Otherwise it is a blind super-resolution (how to deal with this problem?).

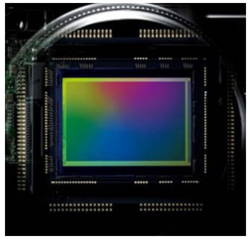
Challenges

Real-world degradations usually come from complicate processes, such as **imaging system of cameras**, **image editing**, and **Internet transmission**.

Take Photo

Camera Blur

(motion and defocus)

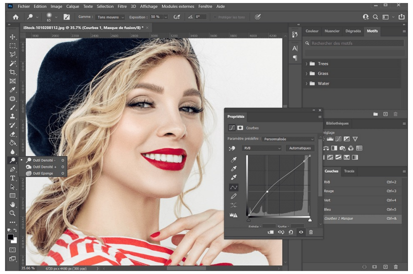


Sensor Noise

JPEG compression

Editing

Downsampling,
Artifacts



Social Media Sharing

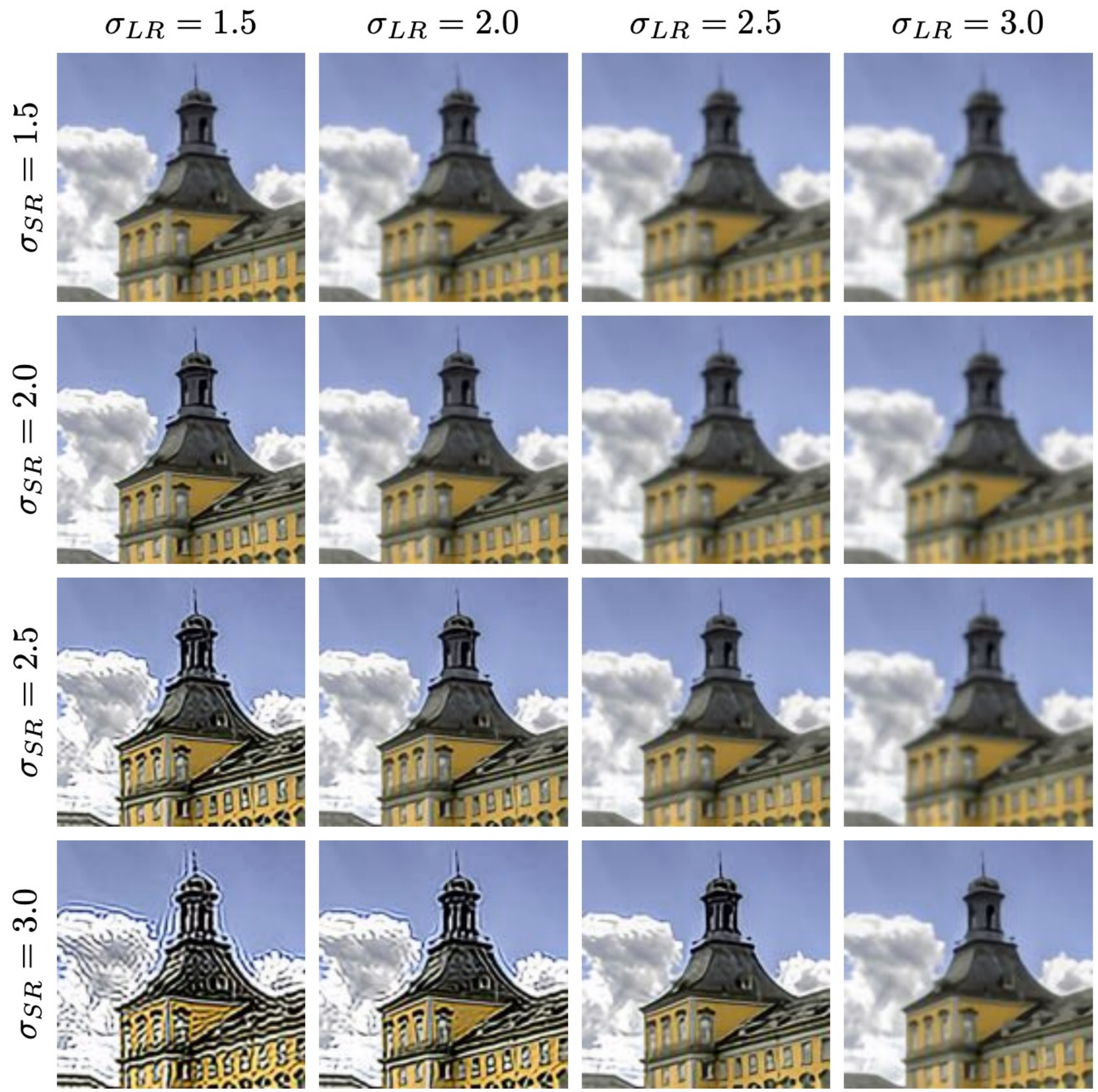
Compression,
Downsampling, Noises

…

Challenges

- Learning-based methods will suffer severe performance drop when the pre-defined degradation is different from the real one

- This phenomenon of kernel mismatch will introduce undesired artifacts to output images



SR sensitivity to the kernel mismatch.
σ_{LR} denotes the kernel used for downsampling and σ_{SR} denotes the kernel used for SR.

Figure credit: J. Gu et al., Blind Super-Resolution With Iterative Kernel Correction, CVPR 2019

Challenges

- Highly ill-posed problem - one LQ image corresponds to infinite number of HQ images

LQ

HQ

...

Challenges

- Vice versa - one HQ image corresponds to infinite number of LQ images

LQ

...

HQ

Architectures

- Convolutional neural networks
  - SRCNN
  - FSRCNN
  - VDSR
- Generative adversarial network
  - SRGAN
  - ESRGAN
- Transformers
  - SwinIR
  - Uformer
  - Restormer
- Diffusion models
  - StableSR

Losses

Mean squared error

- Minimizing the loss between the reconstructed images $F\ (\mathbf{Y};\ \Theta)$ and the corresponding ground truth high-resolution images X

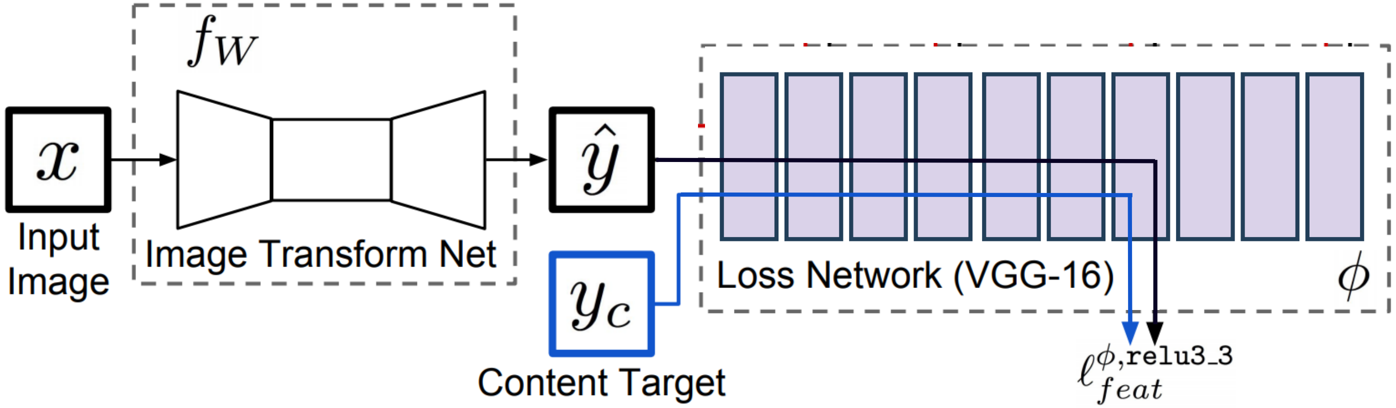
$$L(\Theta) = \frac{1}{n}\sum_{i=1}^{n} ||F(\mathbf{Y}_i;\Theta) - \mathbf{X}_i||^2$$

- The loss is minimized using stochastic gradient descent with the standard backpropagation

Losses

Perceptual loss

Encourages the output image to be perceptually similar to the target image, but does not force them to match exactly



The feature reconstruction loss is the (squared, normalized) Euclidean distance between feature representations

activations of the j-th layer of target image

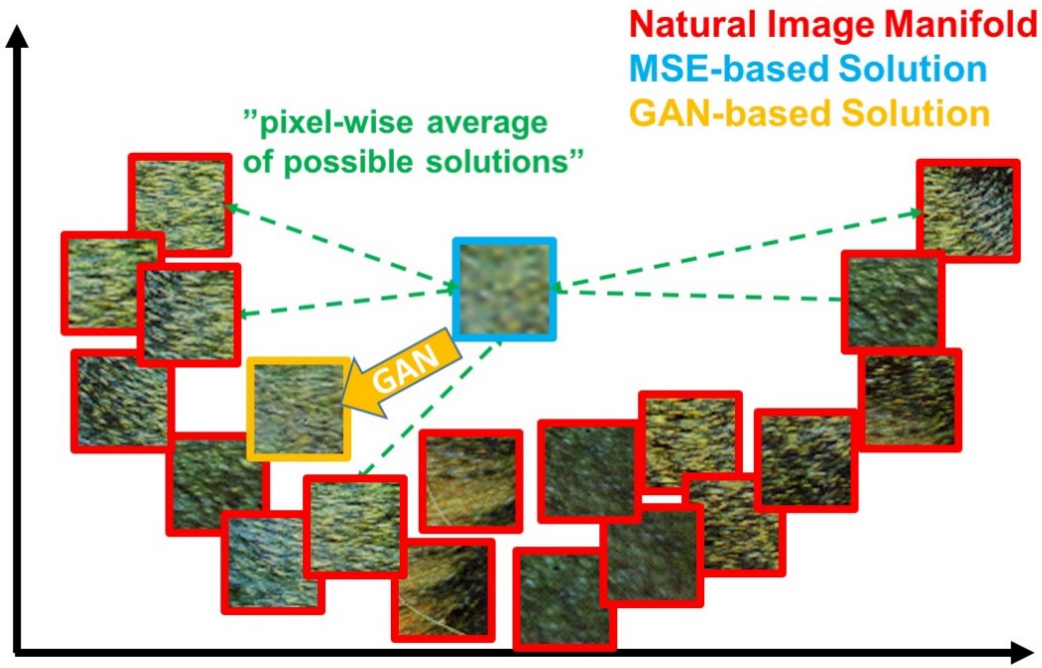
$$\ell_{feat}^{\phi,j}(\hat{y}, y) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y)\|_2^2$$

feature map of shape $C_j \times H_j \times W_j$

activations of the j-th layer of output image

Justin Johnson et al., Perceptual Losses for Real-Time Style Transfer and Super-Resolution, ECCV 2016

Losses



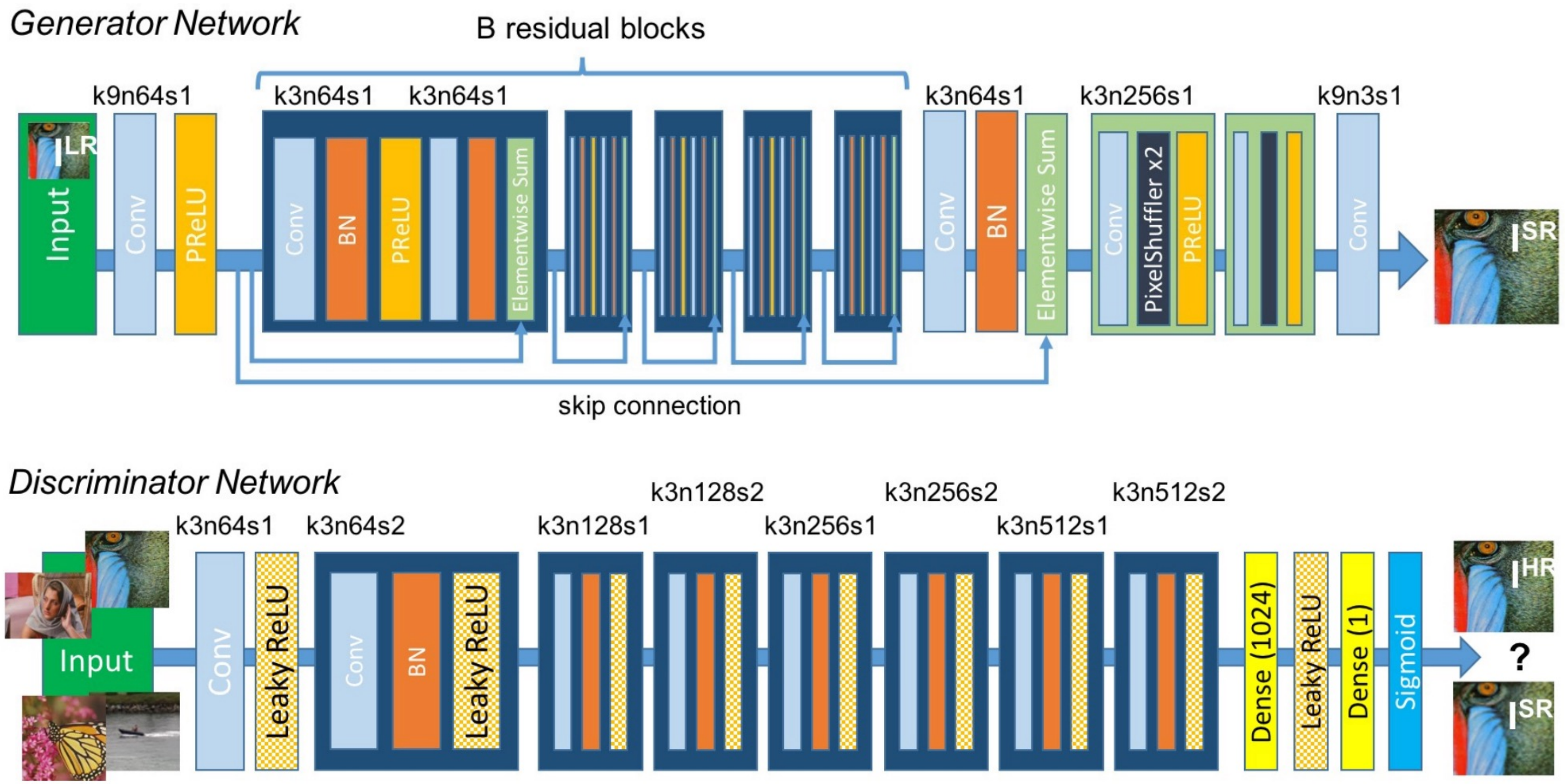
Adversarial loss

The MSE-based solution appears overly smooth due to the pixel-wise average of possible solutions in the pixel space

Generative Adversarial Network (GAN) drives the reconstruction towards the natural image manifold producing perceptually more convincing solutions

C. Ledig et al., [Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network](), CVPR 2017

Losses



$$\min_{\theta_G} \max_{\theta_D} \; \mathbb{E}_{I^{HR} \sim p_{\text{train}}(I^{HR})}[\log D_{\theta_D}(I^{HR})] +$$
$$\mathbb{E}_{I^{LR} \sim p_G(I^{LR})}[\log(1 - D_{\theta_D}(G_{\theta_G}(I^{LR}))]$$

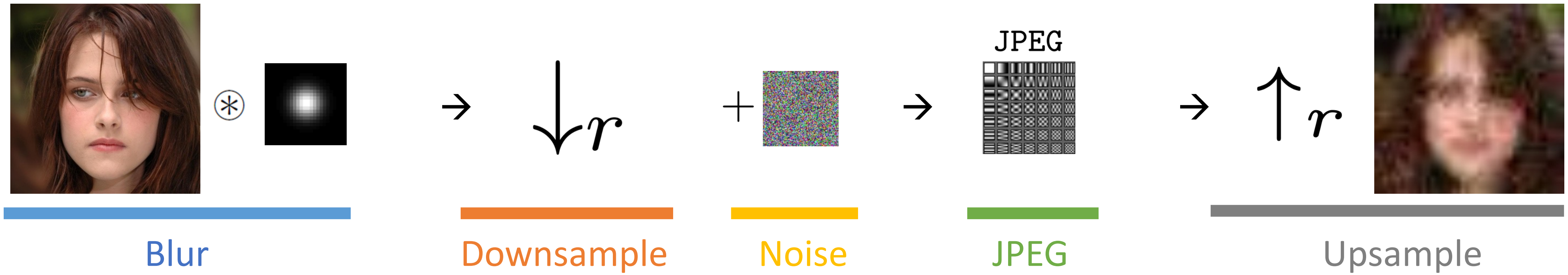
Losses

Input | MSE Loss | Perceptual Loss | Adversarial Loss | Ground Truth

Handling complex degradation

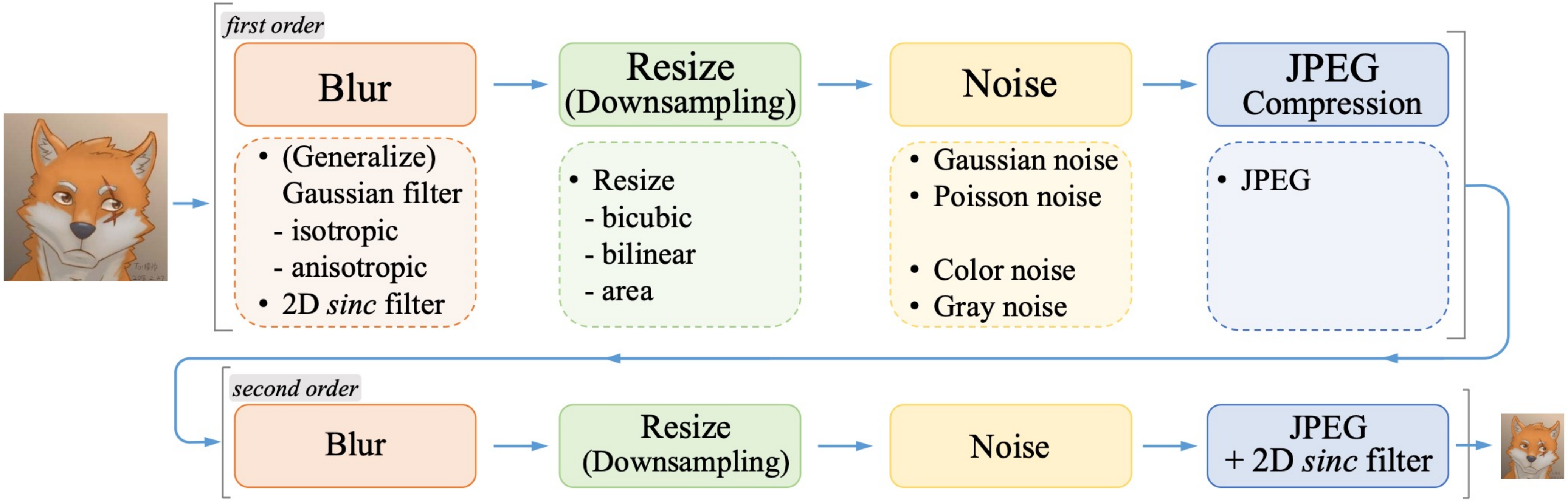
Degradation model

$$I_l = \{[(I_h \otimes k_\sigma)_{\downarrow_r} + n_\delta]_{\mathrm{JPEG}_q}\}_{\uparrow_r}$$



Handling complex degradation

Degradation model



Not a silver bullet - merely extends the solvable degradation boundary of previous blind SR methods through modifying the data synthesis process

Xintao Wang et al. [Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data](), AIM 2021

Metrics

Peak signal-to-noise ratio (**PSNR**) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$

$$= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

$$= 20 \cdot \log_{10}(MAX_I) - 10\log_{10}(MSE)$$

MAX_I = Maximum possible pixel value of the image. For 8 bits image, this is 255

Cons: Doesn't reflect human perception well

Metrics

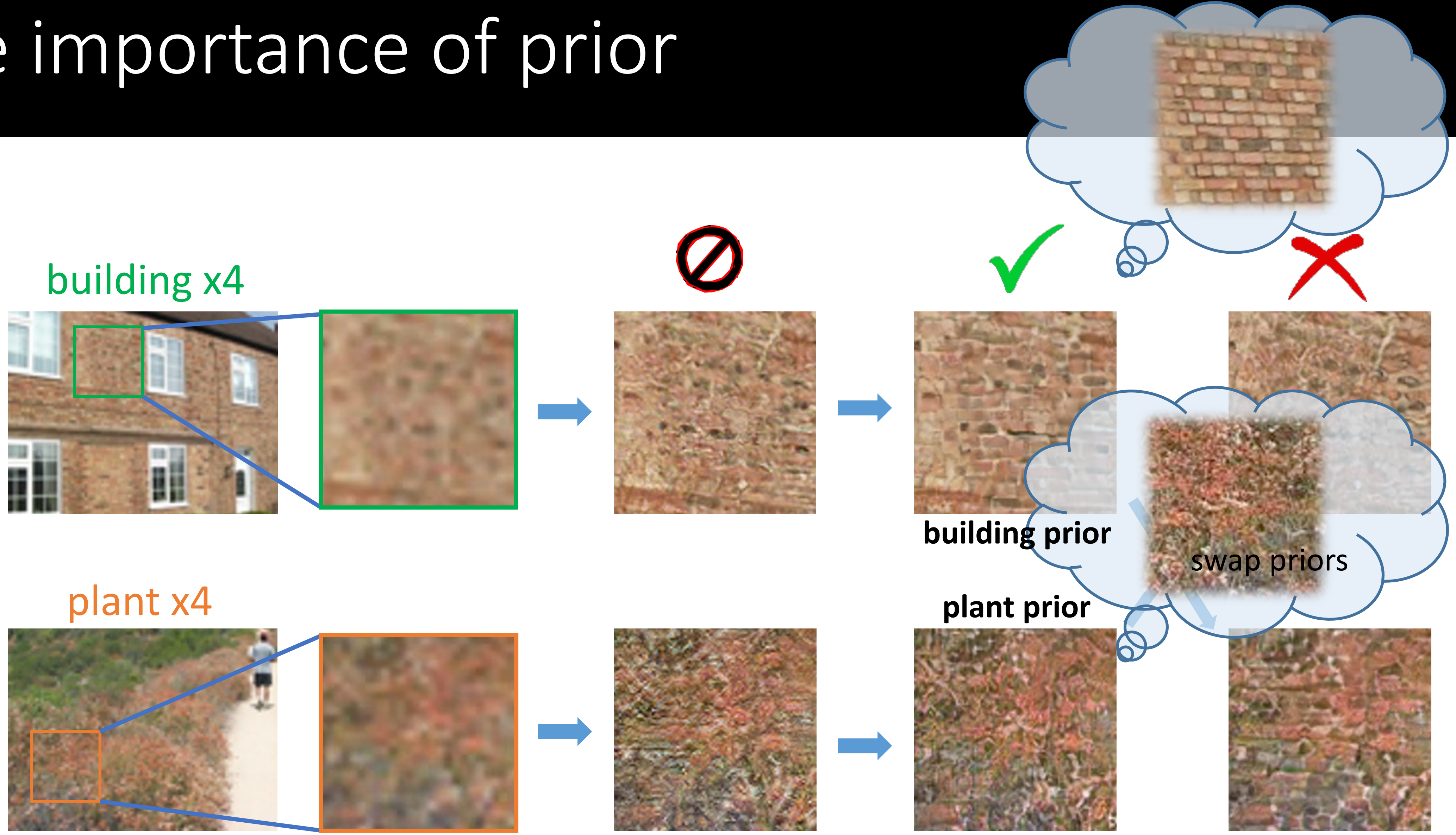
- Perceptual metric
  - LPIPS (Zhang et al., 2018a)
  - FID (Heusel et al., 2017)
  - CLIP-IQA (Wang et al., 2023)
  - MUSIQ (Ke et al., 2021)

Example:

Datasets	Metrics	RealSR	BSRGAN	DASR	Real-ESRGAN+	FeMaSR	LDM	SwinIR-GAN	IF_III	**StableSR**
DIV2K Valid	PSNR ↑	**24.62**	24.58	24.47	24.29	23.06	23.32	23.93	23.36	23.26
	SSIM ↑	0.5970	0.6269	0.6304	**0.6372**	0.5887	0.5762	0.6285	0.5636	0.5726
	LPIPS ↓	0.5276	0.3351	0.3543	**0.3112**	0.3126	0.3199	0.3160	0.4641	0.3114
	FID ↓	49.49	44.22	49.16	37.64	35.87	26.47	36.34	37.54	**24.44**
	CLIP-IQA ↑	0.3534	0.5246	0.5036	0.5276	0.5998	0.6245	0.5338	0.3980	**0.6771**
	MUSIQ ↑	28.57	61.19	55.19	61.05	60.83	62.27	60.22	43.71	**65.92**
RealSR	PSNR ↑	**27.30**	26.38	27.02	25.69	25.06	25.46	26.31	25.47	24.65
	SSIM ↑	0.7579	0.7651	0.7707	0.7614	0.7356	0.7145	**0.7729**	0.7067	0.7080
	LPIPS ↓	0.3570	0.2656	0.3134	0.2709	0.2937	0.3159	**0.2539**	0.3462	0.3002
	CLIP-IQA ↑	0.3687	0.5114	0.3198	0.4495	0.5406	0.5688	0.4360	0.3482	**0.6234**
	MUSIQ ↑	38.26	63.28	41.21	60.36	59.06	58.90	58.70	41.71	**65.88**
DRealSR	PSNR ↑	**30.19**	28.70	29.75	28.62	26.87	27.88	28.50	28.66	28.03
	SSIM ↑	0.8148	0.8028	**0.8262**	0.8052	0.7569	0.7448	0.8043	0.7860	0.7536
	LPIPS ↓	0.3938	0.2858	0.3099	0.2818	0.3157	0.3379	**0.2743**	0.3853	0.3284
	CLIP-IQA ↑	0.3744	0.5091	0.3813	0.4515	0.5634	0.5756	0.4447	0.2925	**0.6357**
	MUSIQ ↑	26.93	57.16	42.41	54.26	53.71	53.72	52.74	30.71	**58.51**
DPED-iphone	CLIP-IQA ↑	0.4496	0.4021	0.2826	0.3389	**0.5306**	0.4482	0.3373	0.2962	0.4799
	MUSIQ ↑	45.60	45.89	32.68	42.42	49.95	44.23	43.30	37.49	**50.48**

Prior for Face Restoration

The importance of prior



X. Wang et al. Recovering Realistic Texture in Image Super-resolution by Deep Spatial Feature Transform, CVPR 2018

Existing priors for face restoration

- Geometric priors
  - Facial semantic map
  - Facial component heatmap
  - Facial 3D shape
  - …
- Reference priors
  - Similar faces
  - Facial component dictionaries
  - …
- Generative priors
  - Pre-trained face generator, e.g., StyleGAN2
  - …

Geometric prior

High-frequency prior indicates the location with high-frequency details

Steps:

1. For each training image, we compute the residual image between the HR and the bicubic interpolation of LR

2. Warp the residual map into the mean face template domain

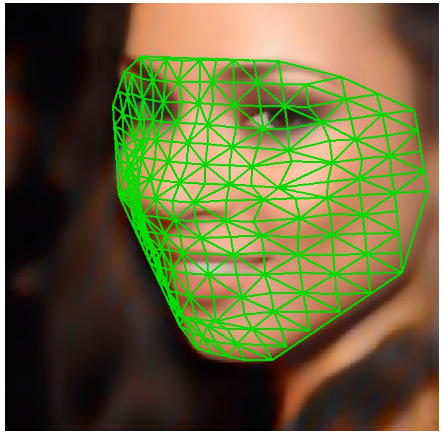
3. Average the magnitude of the warped residual maps over all training images

4. Cluster the preliminary high-frequency map into C continuous contours

5. Form a C-channel maps, with each channel carrying one contour

Geometric prior

Geometric prior



Dense correspondence field

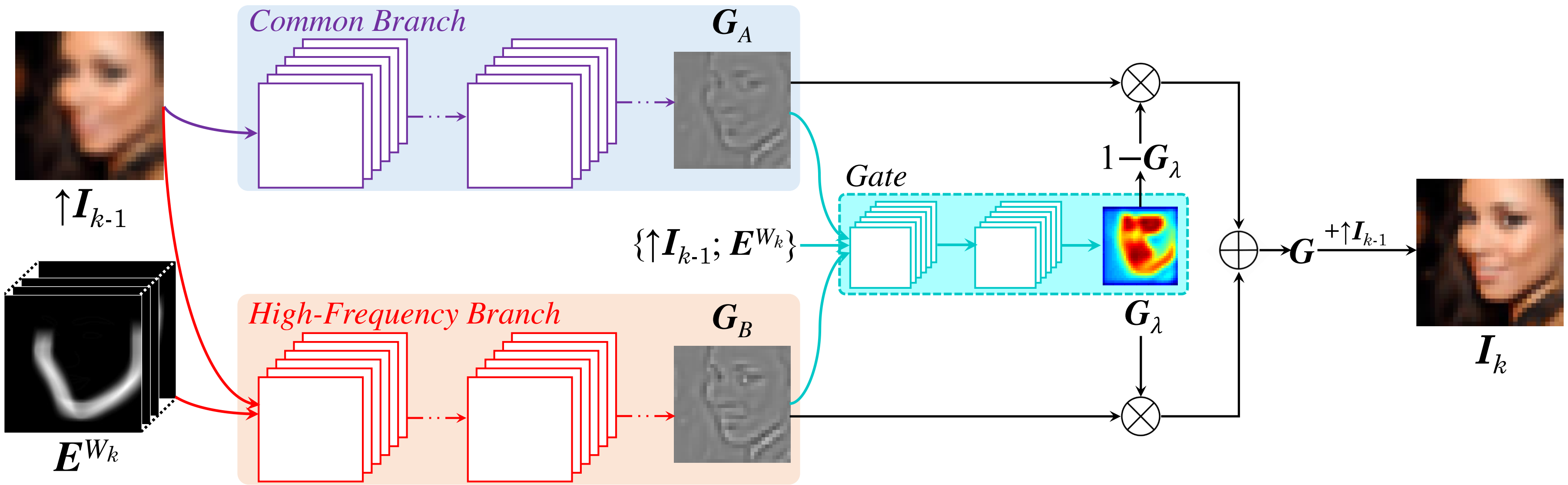
+

Face prior

Warping

Geometric prior

Face restoration conditioned on prior



(a) Bicubic (b) Common (c) High-Freq. (d) **CBN** (e) Original

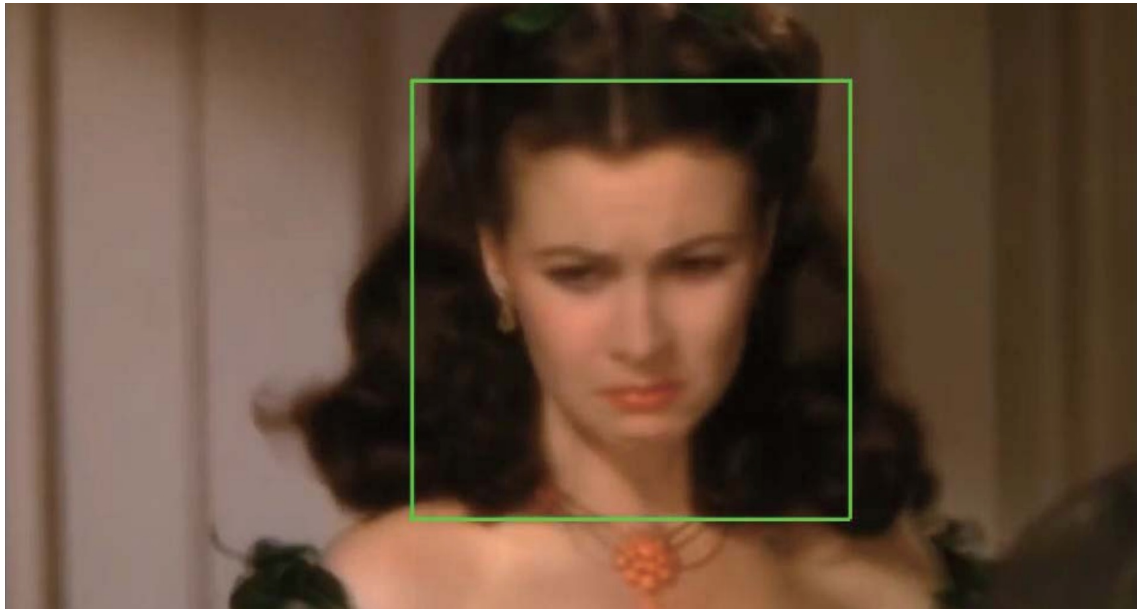
S. Zhu, S. Liu, C. C. Loy, X. Tang, Deep Cascaded Bi-Network for Face Hallucination, ECCV 2016

Existing priors for face restoration

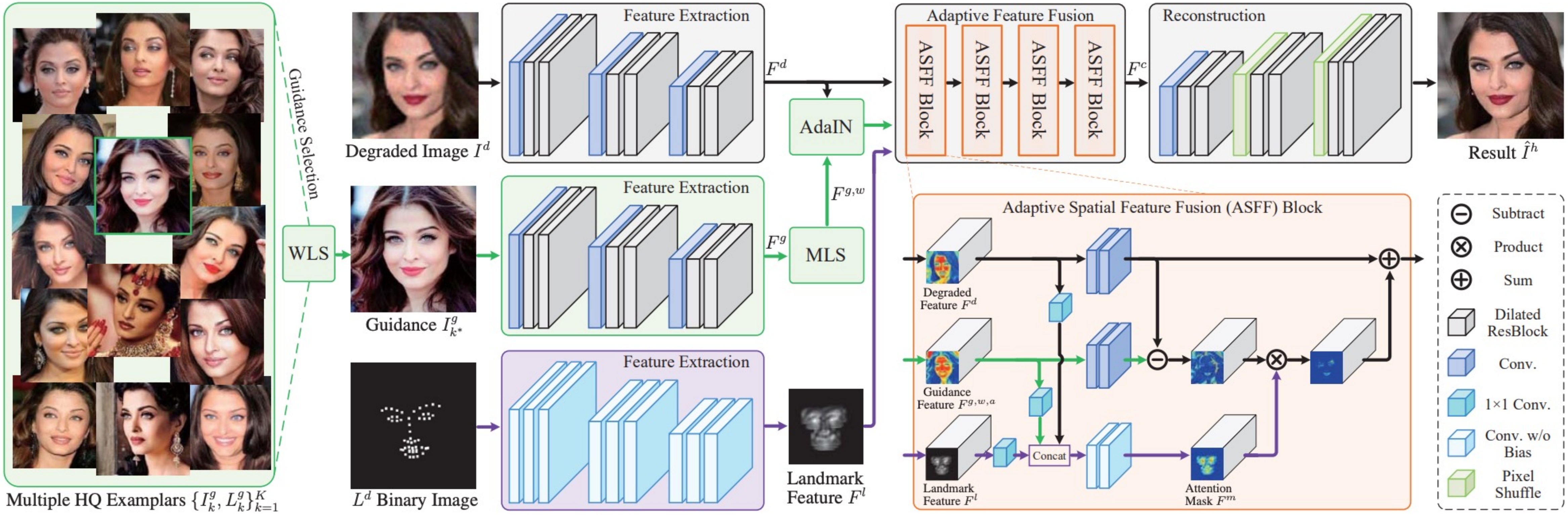
- Geometric priors
  - Facial semantic map
  - Facial component heatmap
  - Facial 3D shape
  - …
- Reference priors
  - Similar faces
  - Facial component dictionaries
  - …
- Generative priors
  - Pre-trained face generator, e.g., StyleGAN2
  - …

Reference prior

Face restoration conditioned on exemplars



Reference prior



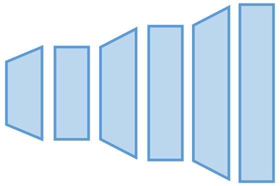
X. Li et al., Enhanced Blind Face Restoration with Multi-Exemplar Images and Adaptive Spatial Feature Fusion, CVPR 2020

Existing priors for face restoration

- Geometric priors
  - Facial semantic map
  - Facial component heatmap
  - Facial 3D shape
  - …
- Reference priors
  - Similar faces
  - Facial component dictionaries
  - …
- Generative priors
  - Pre-trained face generator, e.g., StyleGAN2
  - …

Generative prior

$\mathbf{z} \sim \mathcal{N}(0, I)$



Latent space

Generator

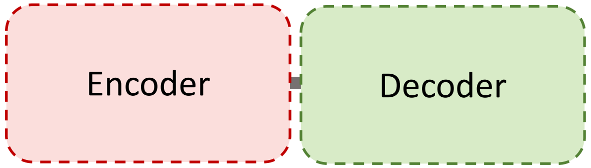
Can we leverage a GAN trained on large-scale natural images for richer priors?

GAN is a good approximator for natural image manifold.

Generative prior

Using GAN as latent bank

Encoder-Decoder Structure



A common architecture

It is typically trained from scratch using a combined objective function consisting of a fidelity term and an adversarial loss

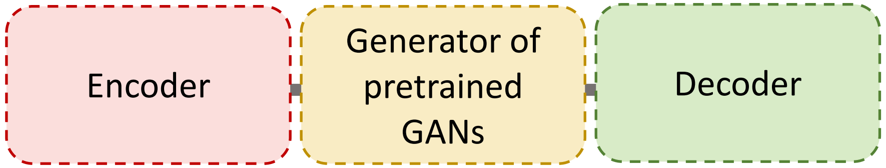
The generator is responsible for both capturing the natural image characteristics and maintaining the fidelity to the ground-truth.

This inevitably limit its capability of approximating the natural image manifold.

Generative prior

Using GAN as latent bank

Encoder-Bank-Decoder Structure



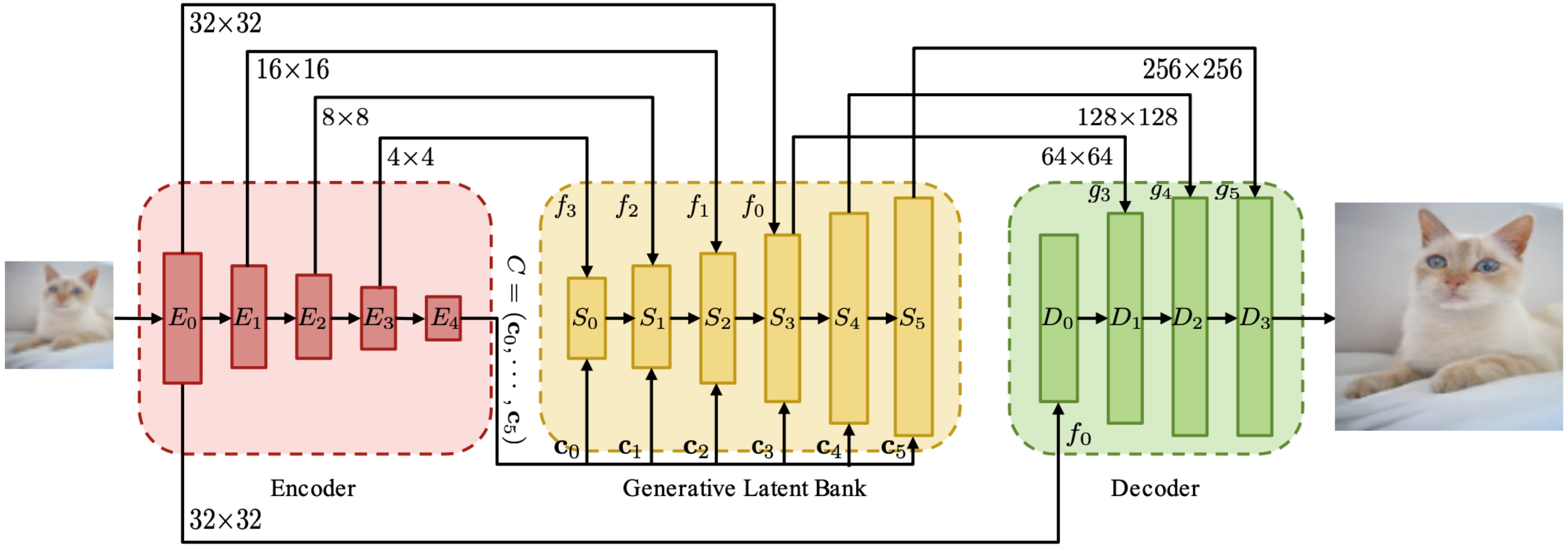
Lifts the burden of learning both fidelity and texture generation simultaneously

Does not involve image-specific optimization at runtime

Needs a single forward pass to perform image restoration

Inspired by the classic notion of dictionary but exploit GAN as a more effective way for storing priors

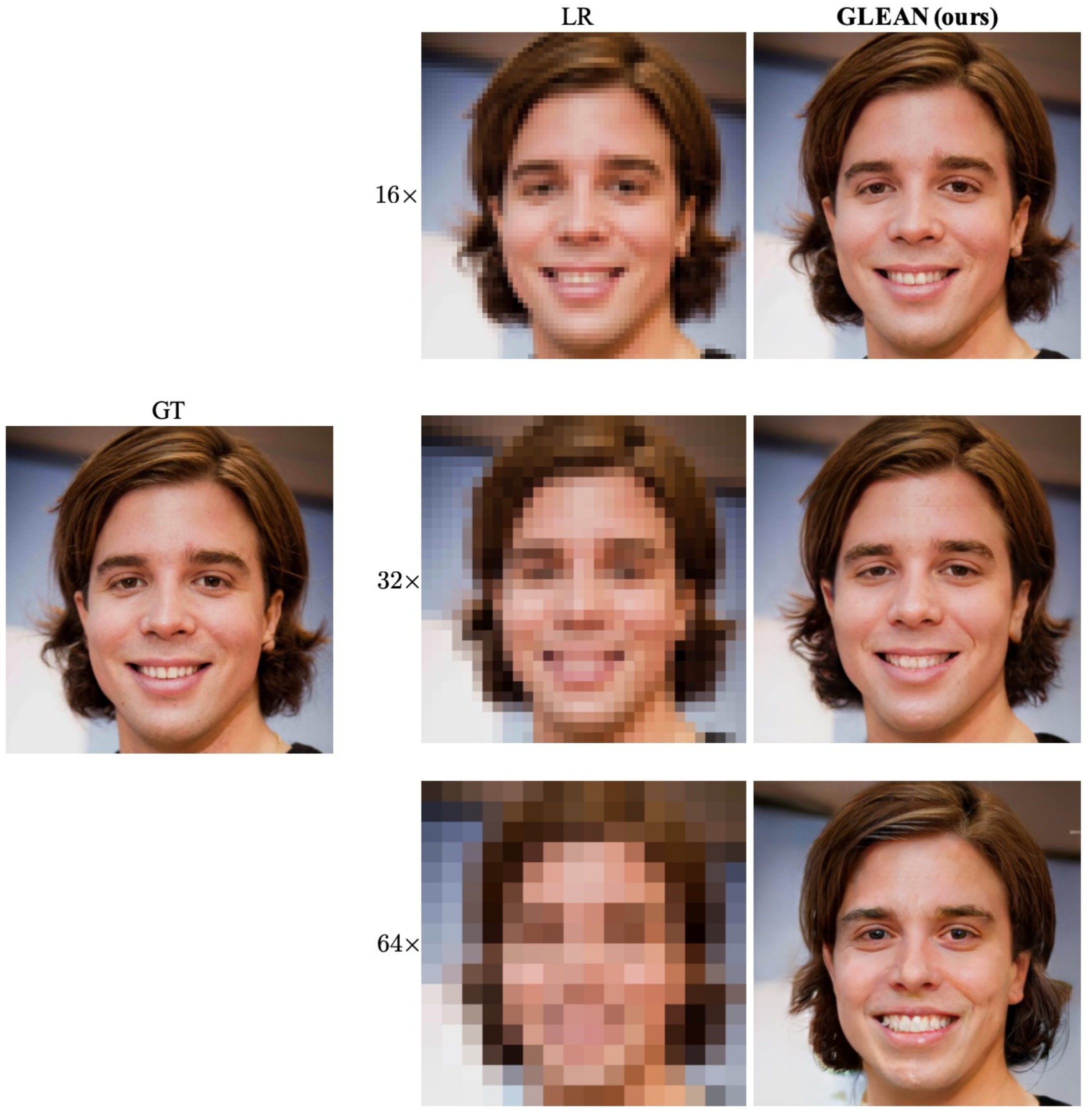
Generative prior

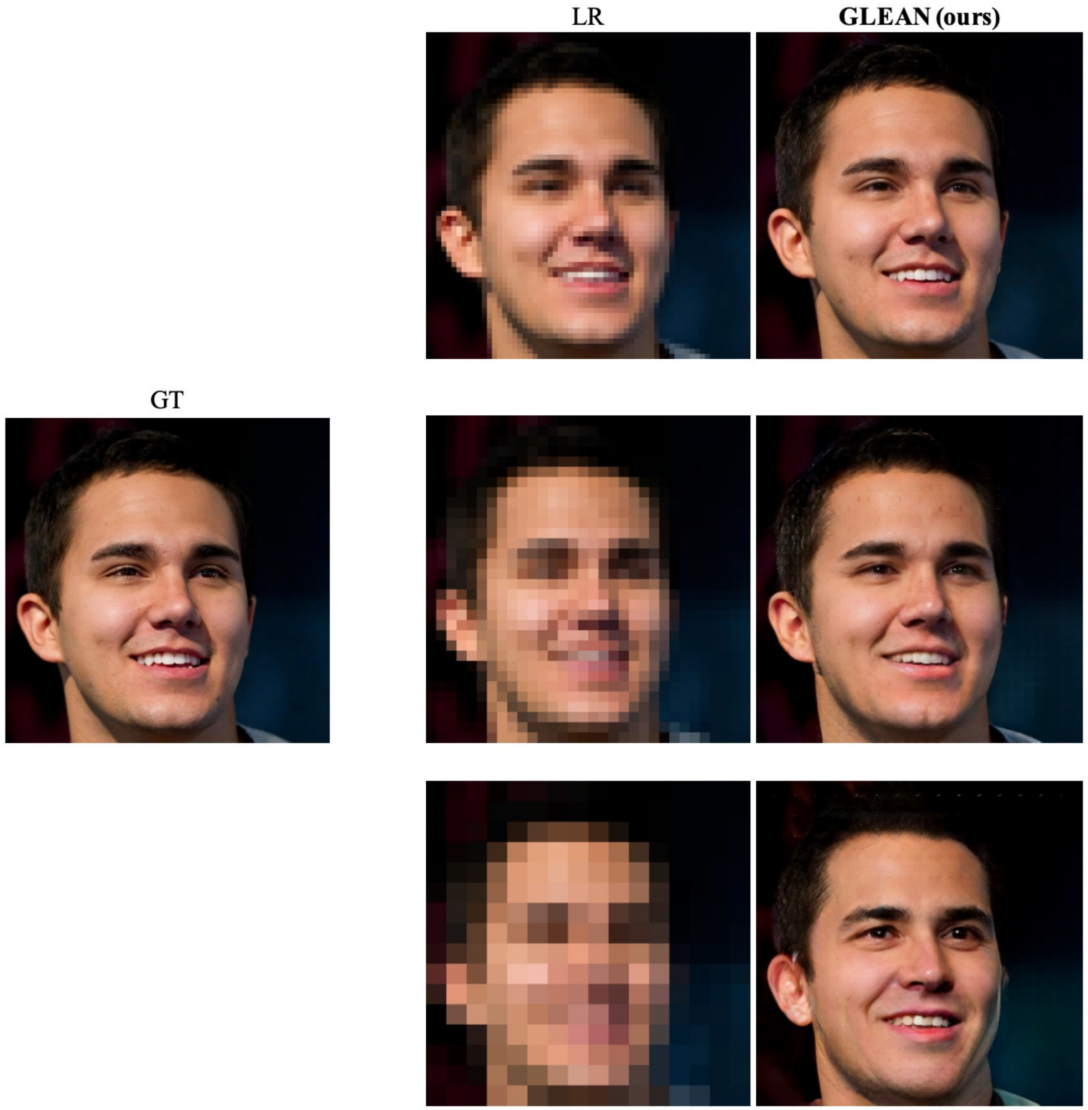


Condition the bank by passing both the latent vectors and multi-resolution convolutional features from the encoder to achieve high-fidelity results. Symmetrically, multi-resolution cues need to be passed from the bank to the decoder.

K. C. K. Chan, X. Wang, X. Xu, J. Gu, C. C. Loy, GLEAN: Generative Latent Bank for Image Super-Resolution and Beyond, TPAMI 2022

Generative prior





Generative prior

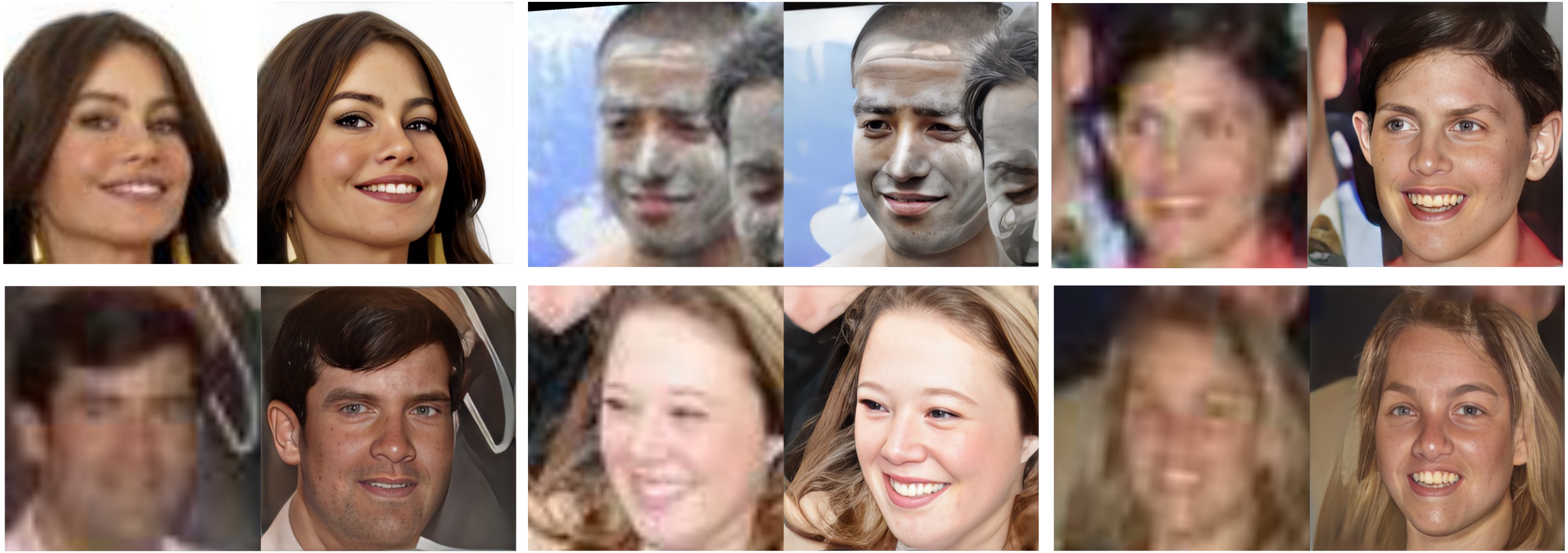
484x484

242x242

121x121

60x60

Generative prior



Generative prior

Generative prior

CodeFormer

Old photo enhancement

Old Photo

CodeFormer

Old photo enhancement

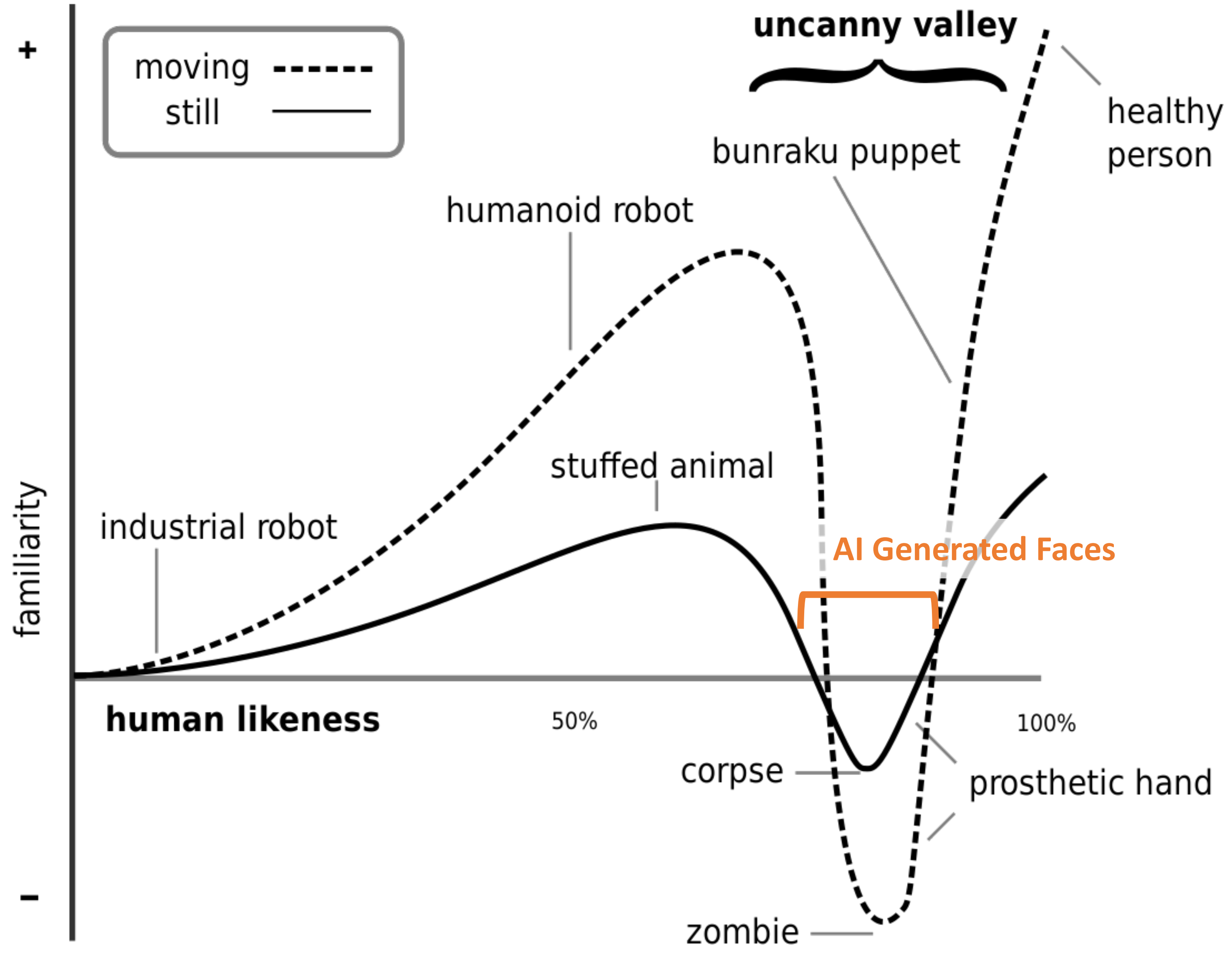
Old Photo

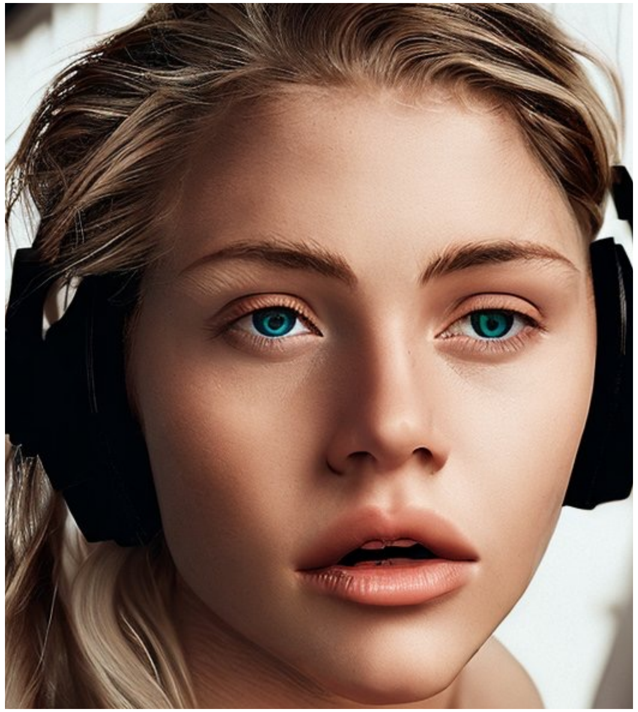
CodeFormer

Old photo enhancement

Old Photo

CodeFormer


uncanny valley
moving
still
+
-
healthy person
bunraku puppet
humanoid robot
stuffed animal
industrial robot
familiarity
AI Generated Faces
human likeness
50%
100%
corpse
prosthetic hand
zombie

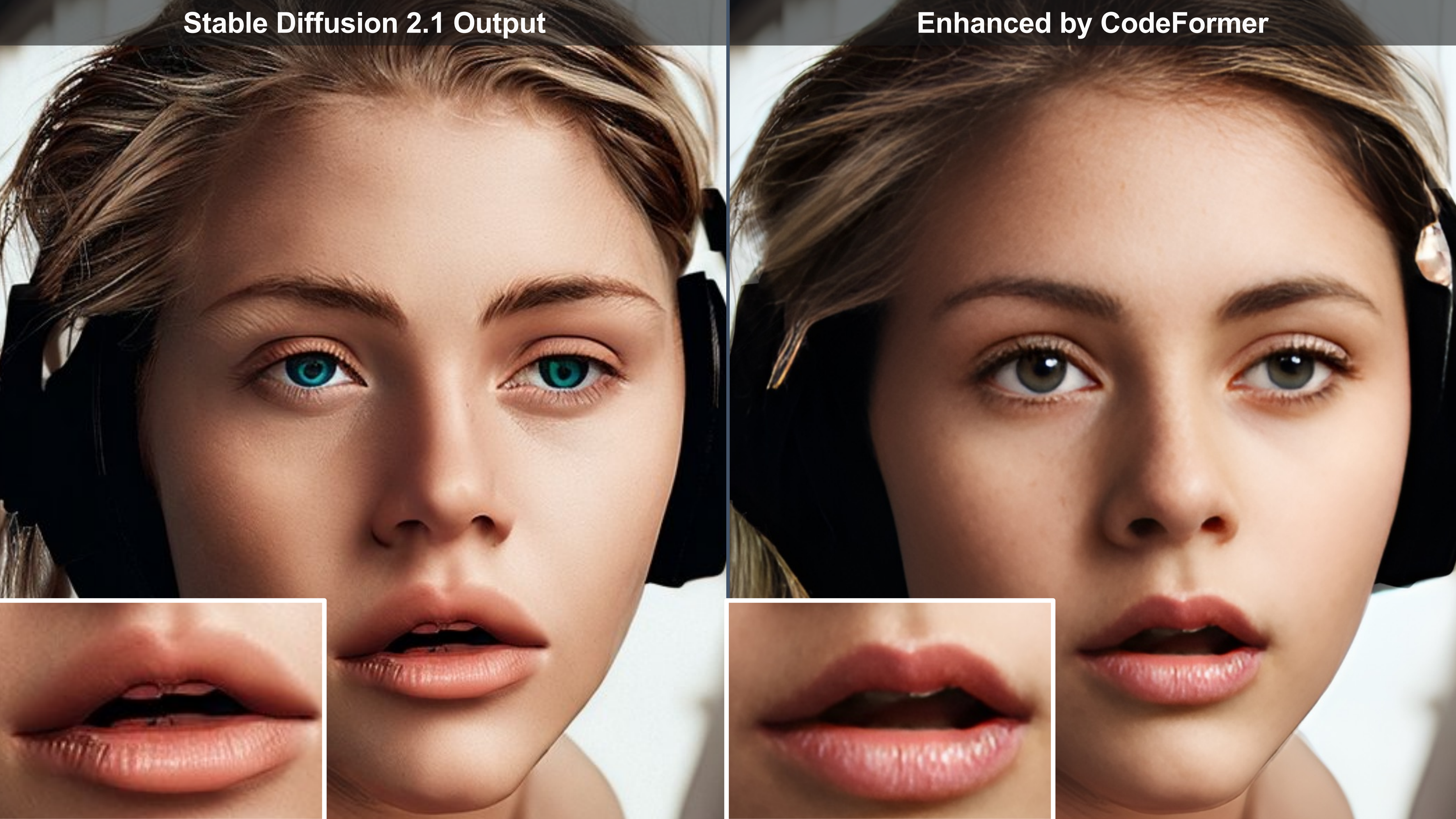


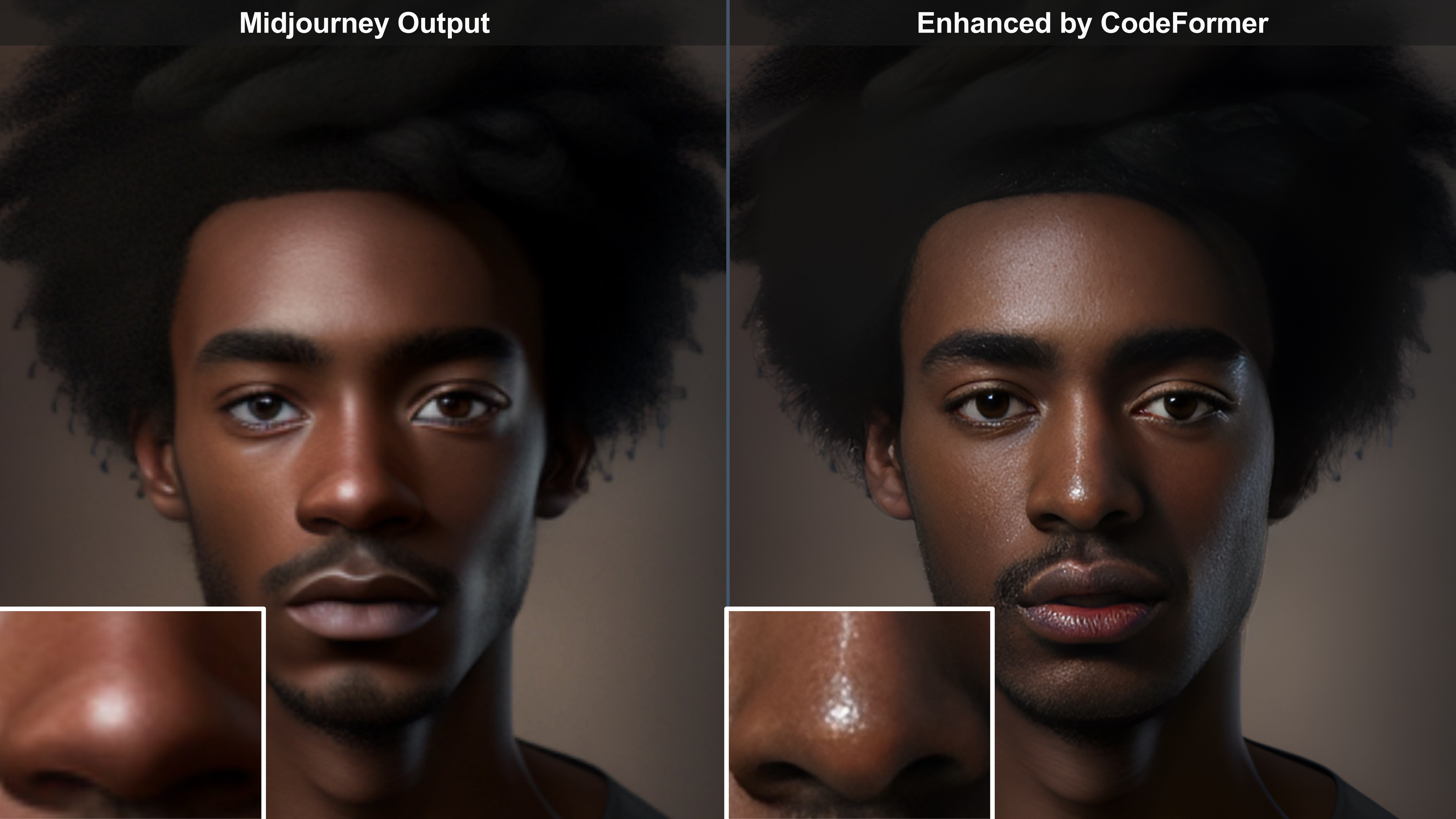

Stable Diffusion 2.1 Output
Enhanced by CodeFormer

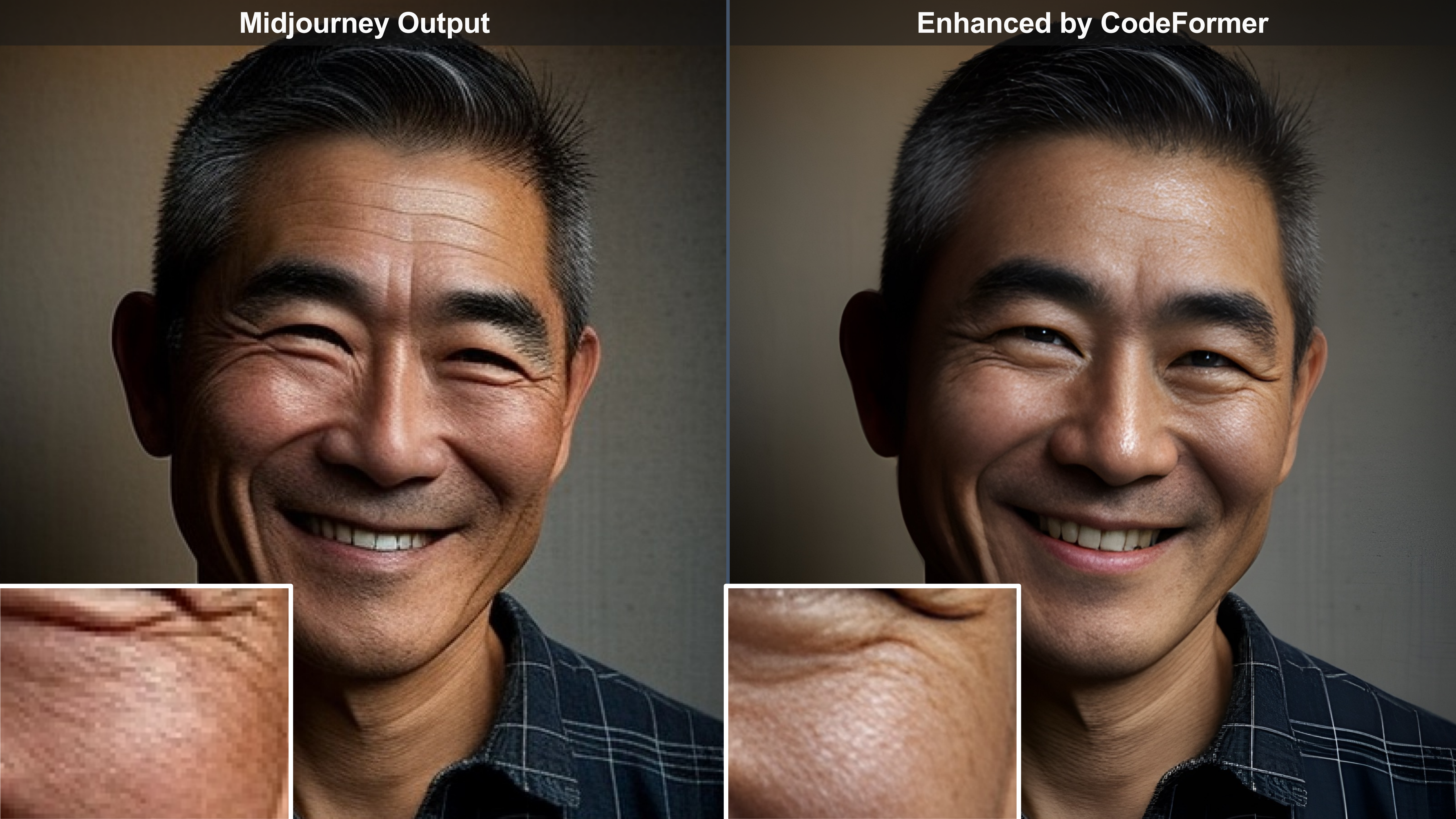

Stable Diffusion 2.1 Output
Enhanced by CodeFormer

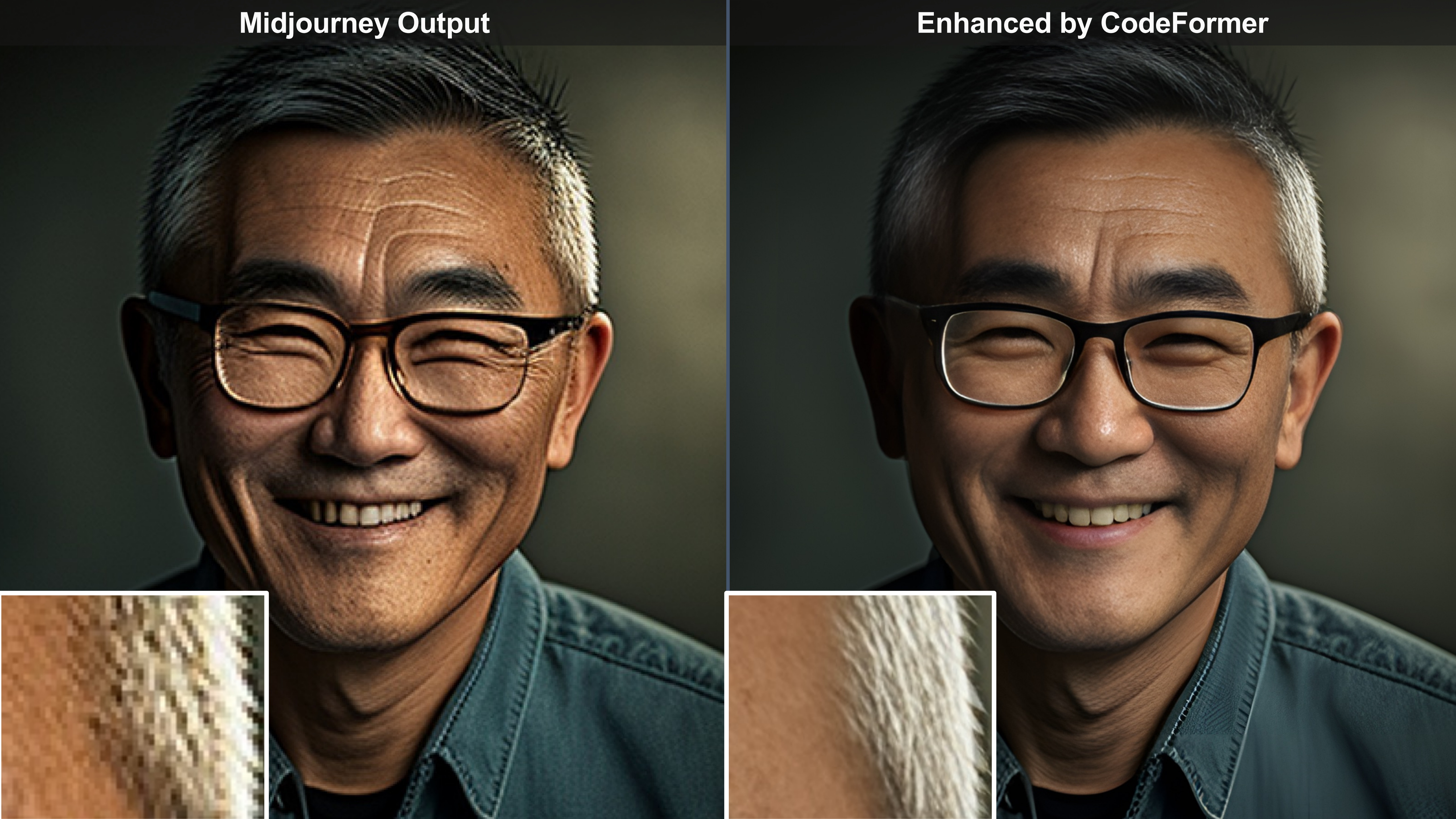
Stable Diffusion 2.1 Output

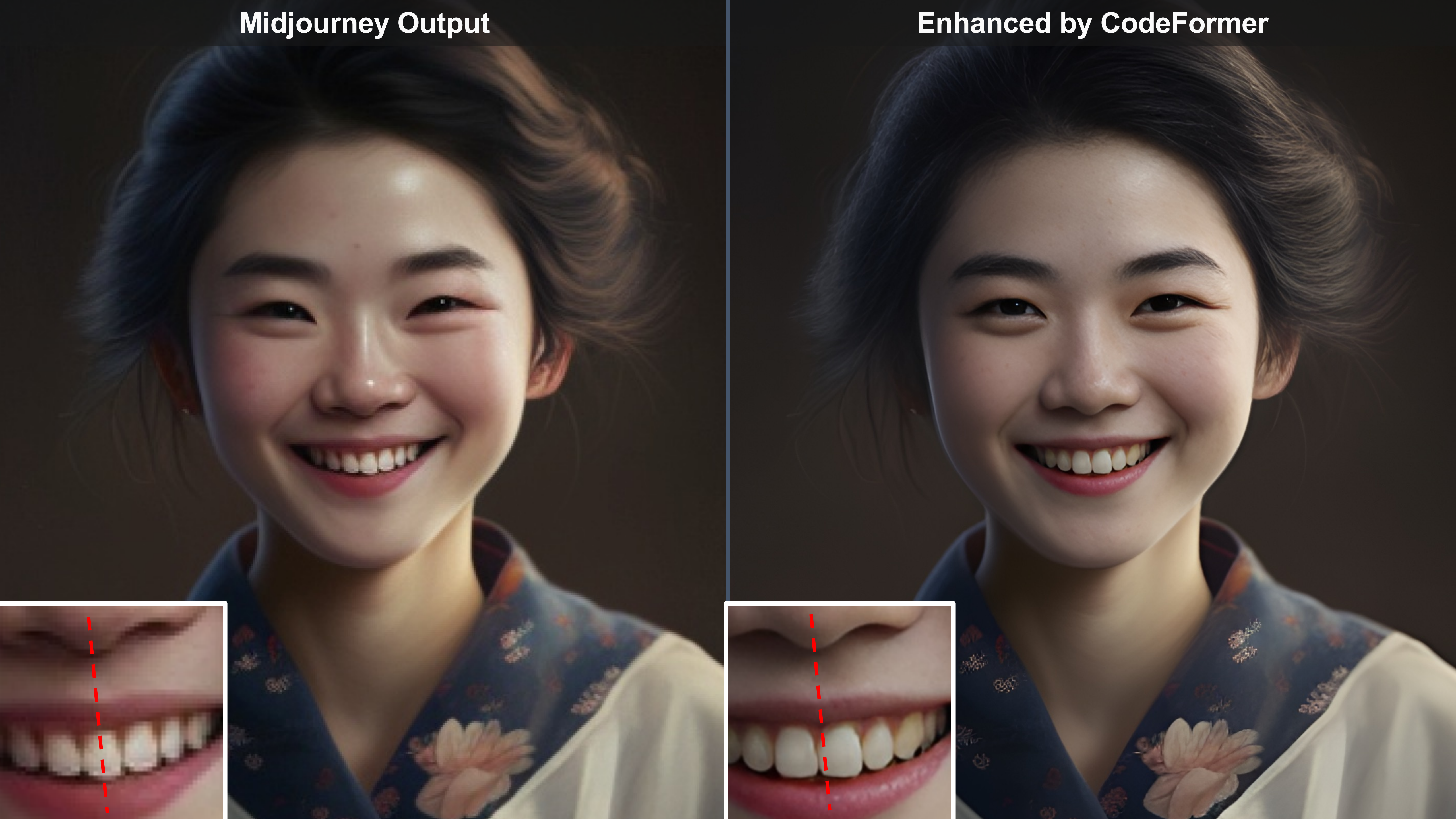
Enhanced by CodeFormer


Stable Diffusion 2.1 Output
Enhanced by CodeFormer


Midjourney Output
Enhanced by CodeFormer

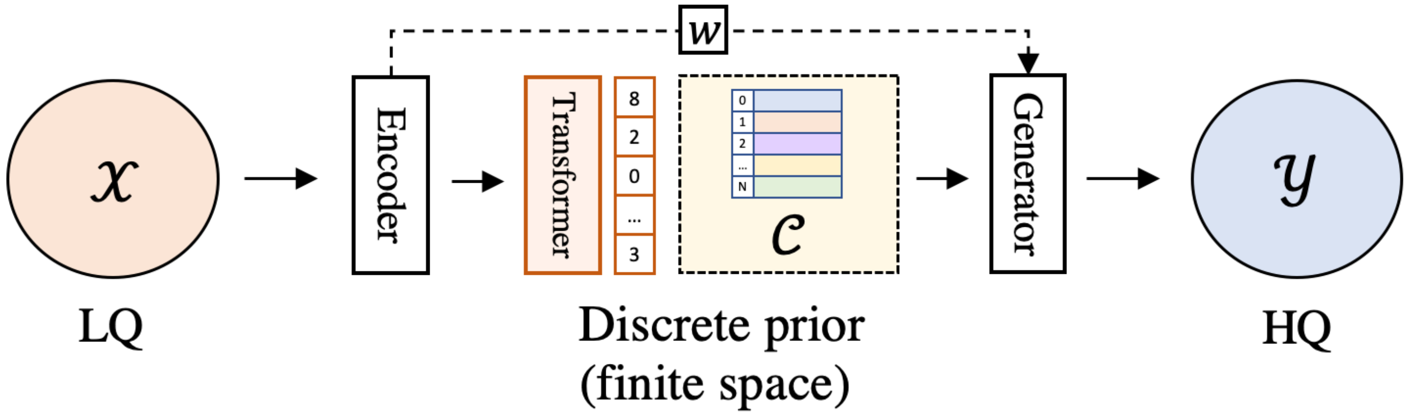

Midjourney Output
Enhanced by CodeFormer


Midjourney Output
Enhanced by CodeFormer


Midjourney Output
Enhanced by CodeFormer

Over 50,700,000 World-wide Access

CodeFormer



Learn discrete codebook prior in a small proxy space to reduce the uncertainty and ambiguity of restoration mapping by, while providing rich visual atoms for generating high-quality faces.

Cast blind face restoration as a code prediction task

A Transformer-based prediction network to model the global composition and context of the low-quality faces for code prediction

Enable the discovery of natural faces that closely approximate the target faces even when the inputs are severely degraded

VAE

The latent vector is a combination of the mean and standard deviation of the output of the convolutions.

This latent vector can be used to generate random images

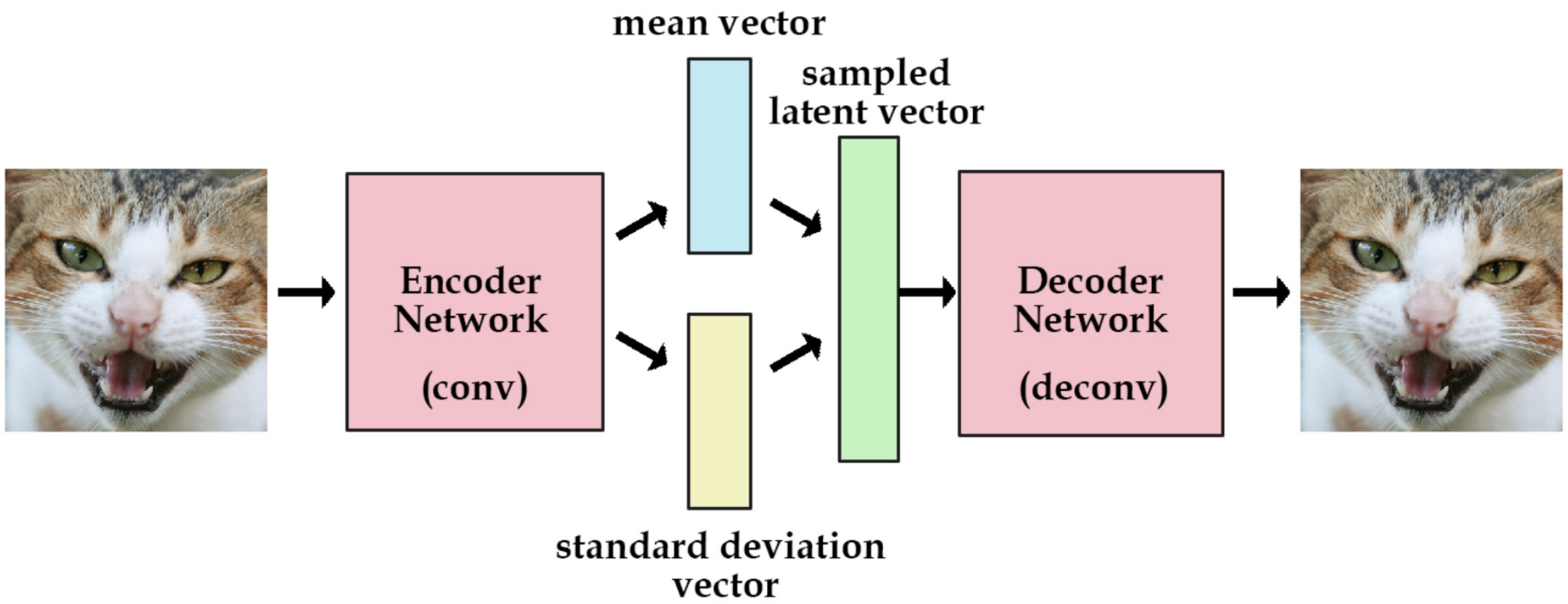
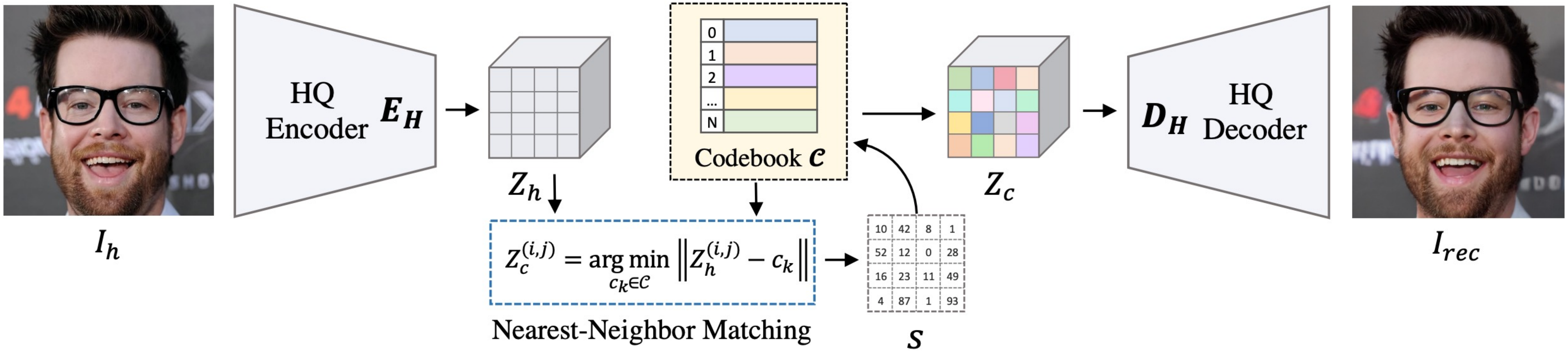


Illustration of a VAE
(Source: http://kvfrans.com/content/images/2016/08/vae.jpg)

VQVAE

VQ-VAE is a type of variational autoencoder that uses vector quantisation to obtain a discrete latent representation. It differs from VAEs in two key ways: the encoder network outputs discrete, rather than continuous, codes; and the prior is learnt rather than static (the posteriors and priors in VAEs are assumed normally distributed with diagonal covariance).



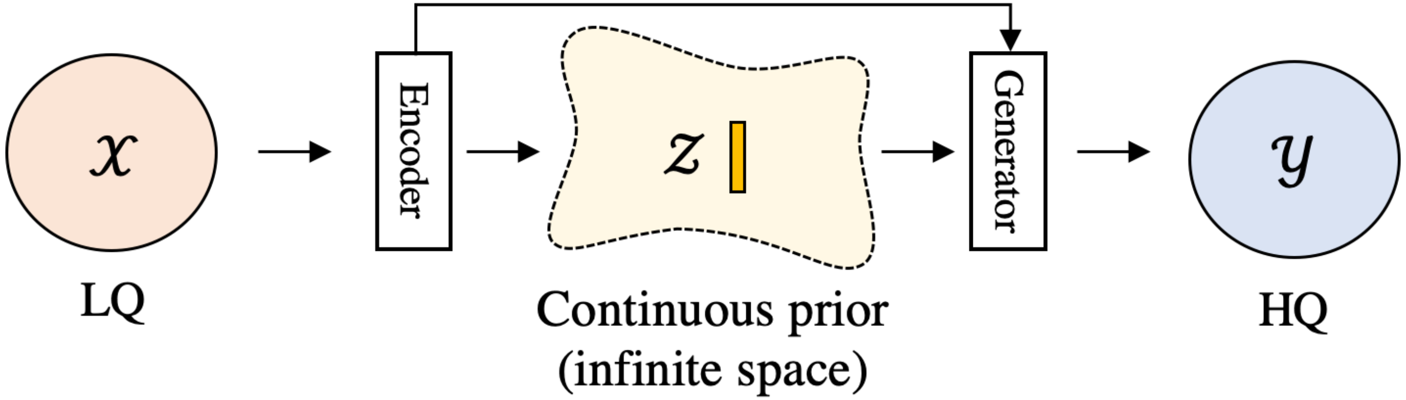
[VQGAN] *Esser et al.,* Taming Transformers for High-Resolution Image Synthesis, CVPR 2021

[VQVAE] *Oord et al.,* Neural Discrete Representation Learning, NeurIPS 2017

Continuous prior *v.s.* discrete prior

StyleGAN-based frameworks

To enhance the fidelity, skip connections between encoder and decoder are usually required



Input

PULSE
(continuous, w/o connection)

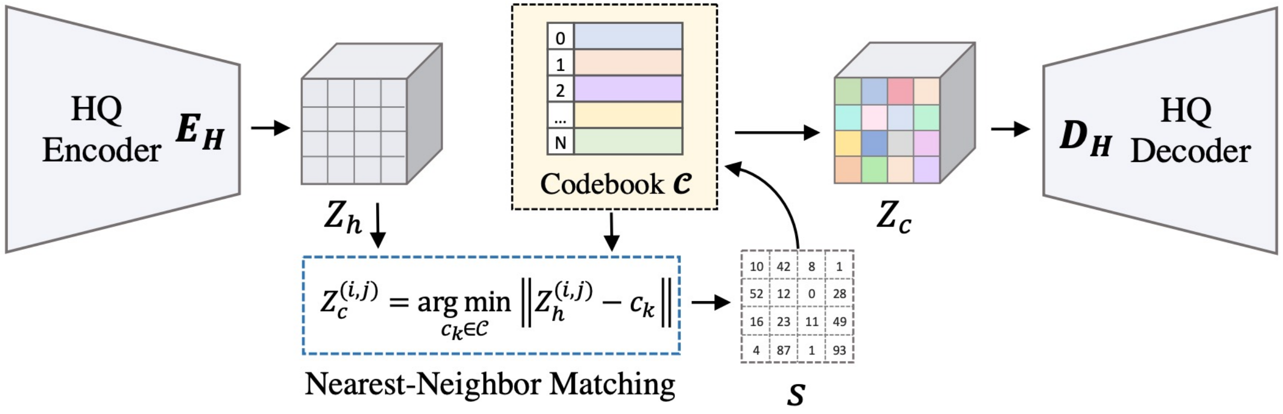
GFP-GAN
(continuous, w/ connection)

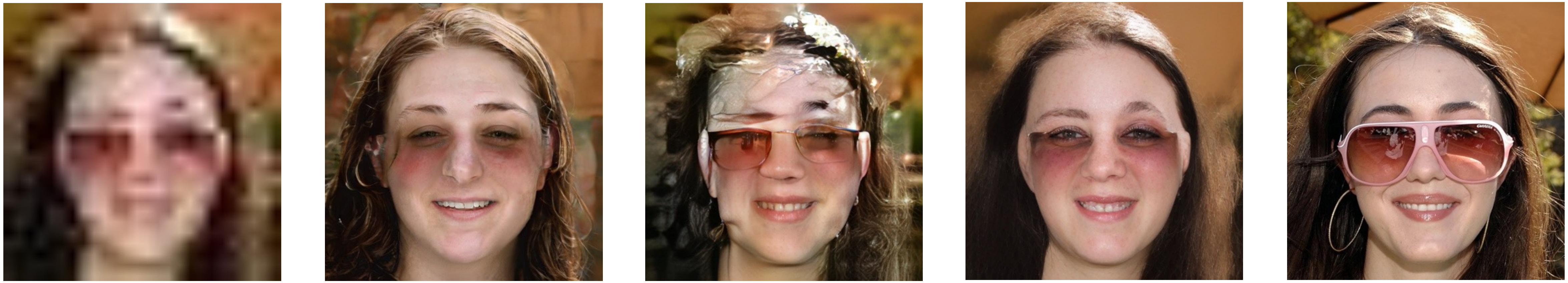
Ground Truth

Continuous prior *v.s.* discrete prior

VQGAN frameworks

Nearest-neighbour matching is problematic given low-res input





Input

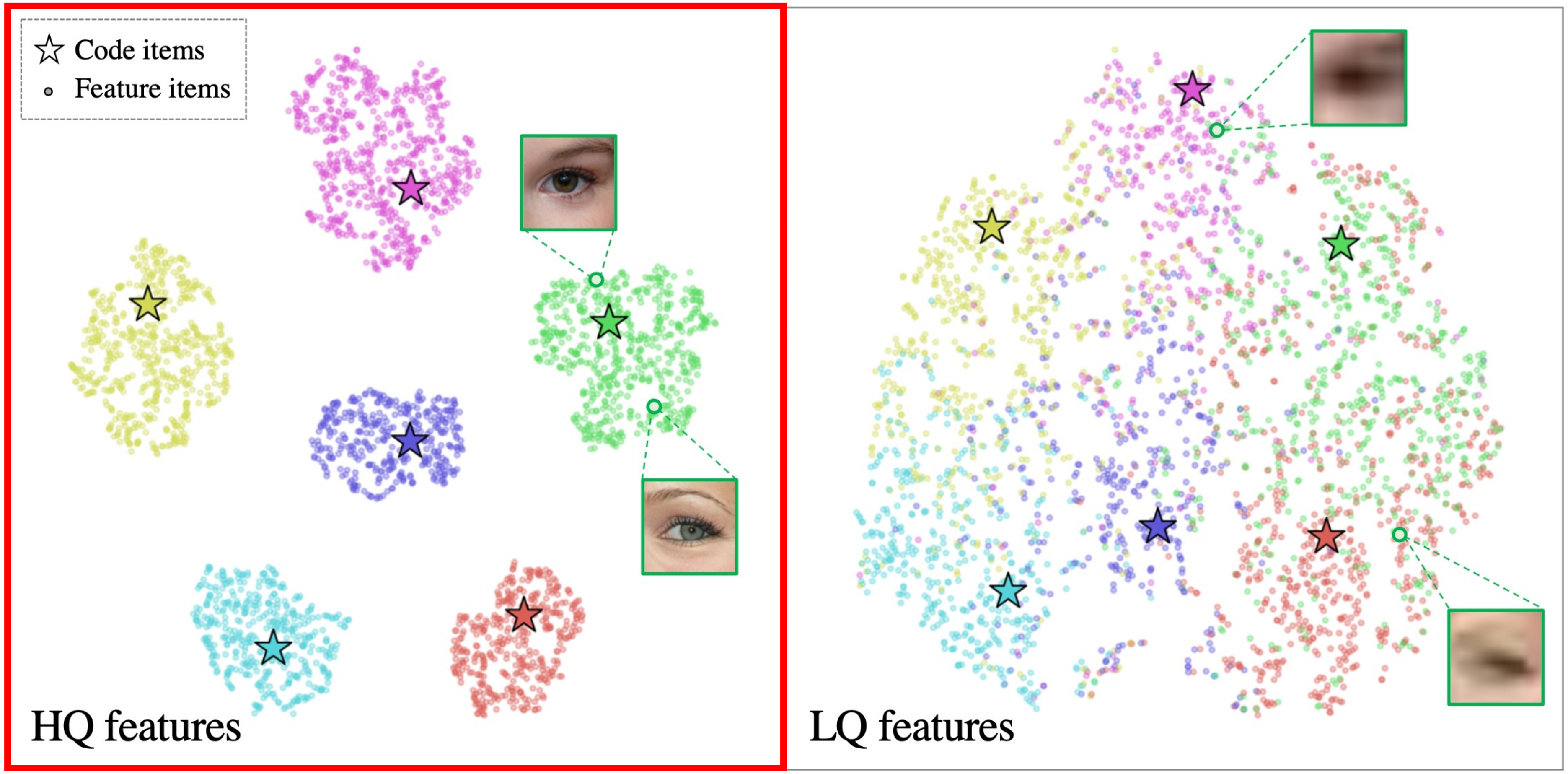
PULSE (continuous, w/o connection)

GFP-GAN (continuous, w/ connection)

Nearest Neighbor (discrete, w/o connection)

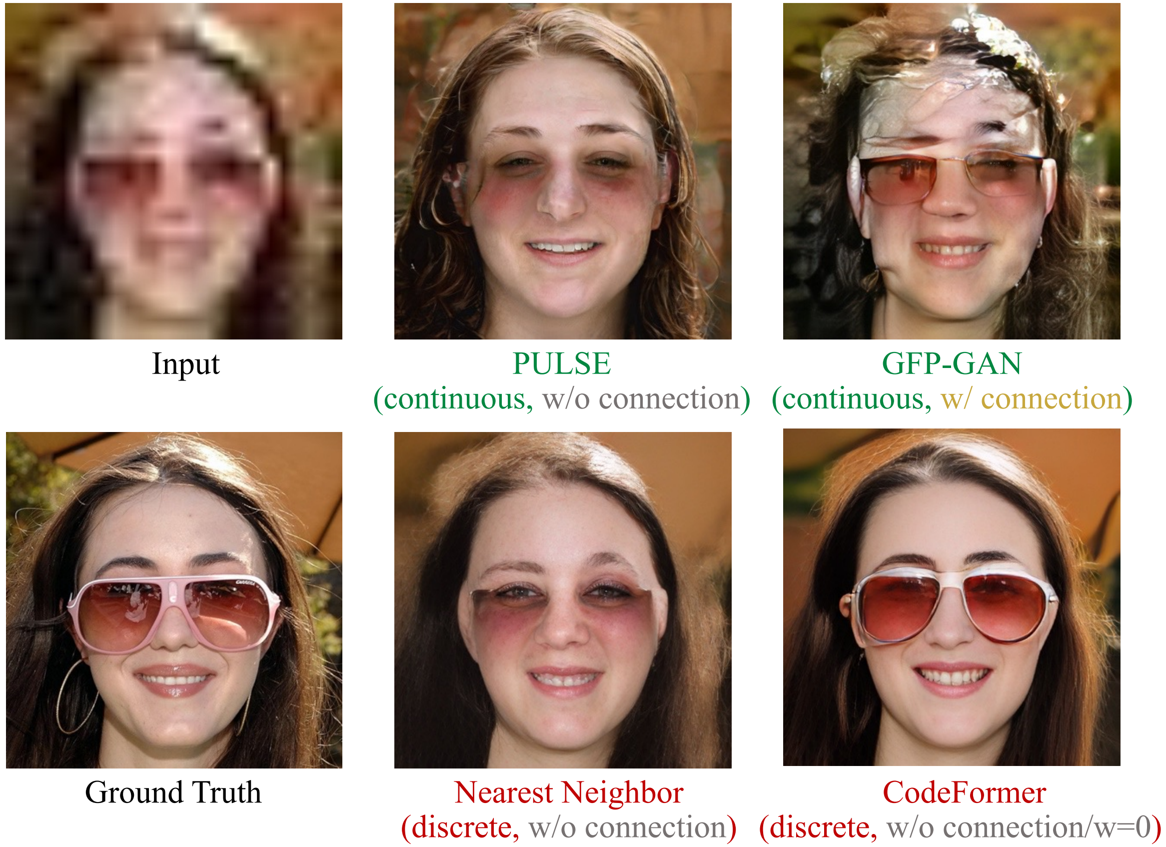
Ground Truth

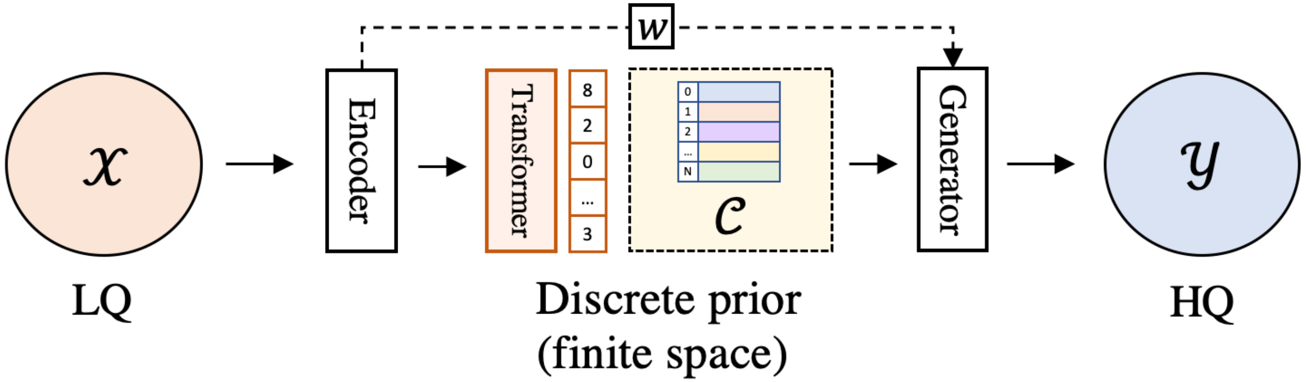
Codebook lookup



(b) Distributions of HQ (left) / LQ (right) features and the codebook items

Continuous prior *v.s.* discrete prior





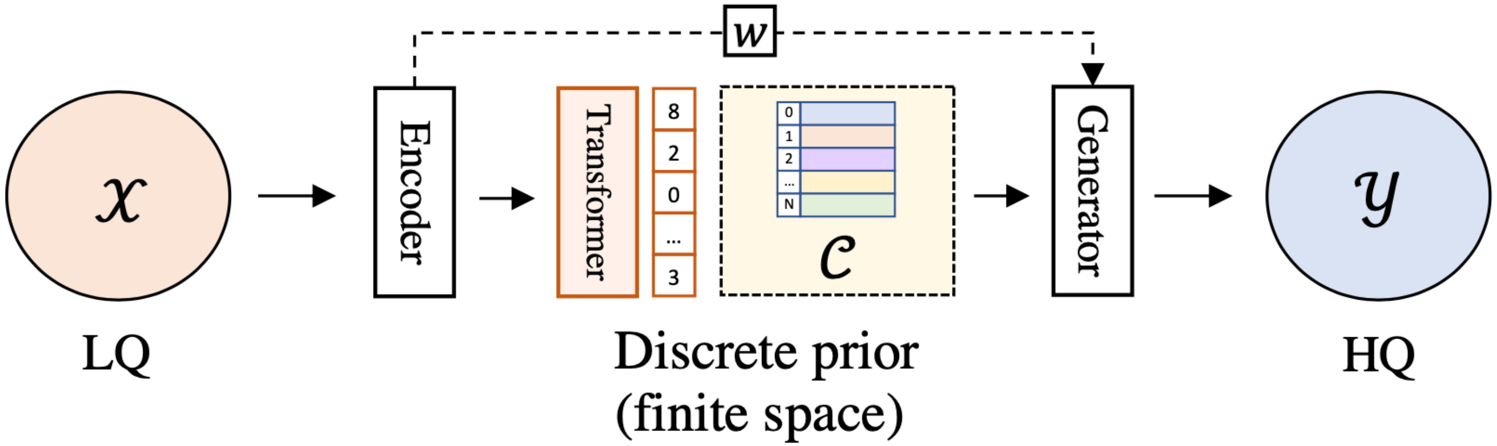
Global modeling for remedying the local information loss in LQ images

Nearest Neighbor v.s. CodeFormer

Real Input Nearest Neighbor CodeFormer

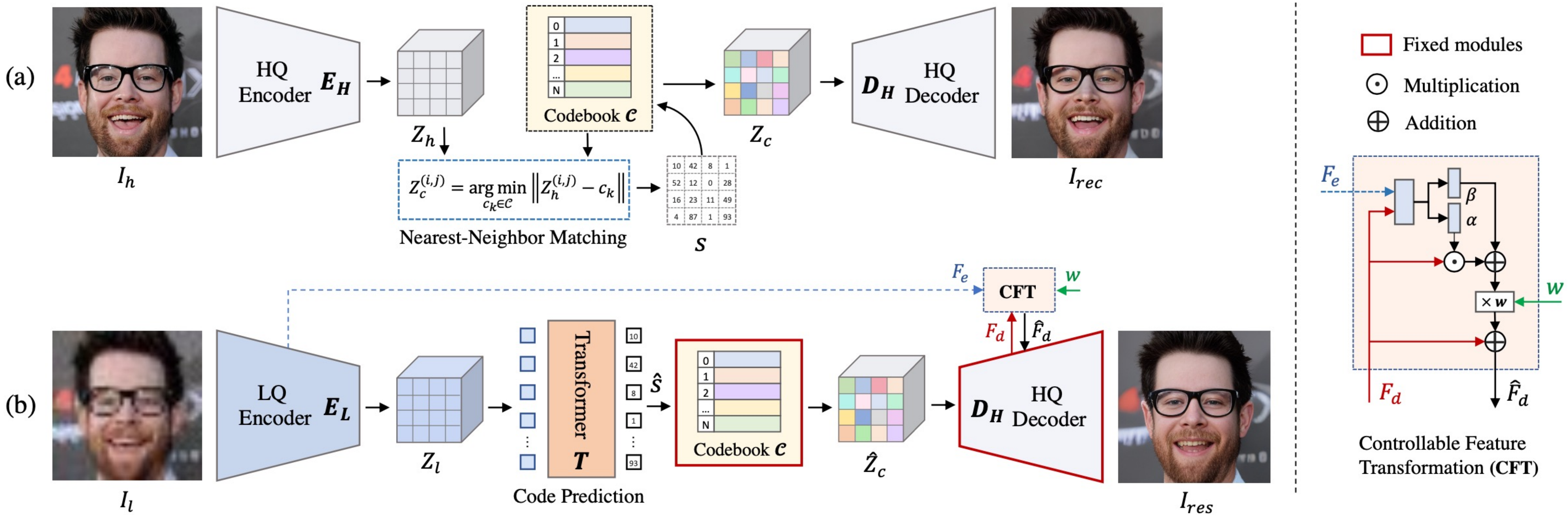
Controllability

A. LQ-HQ mapping
B. Details
C. Identity √

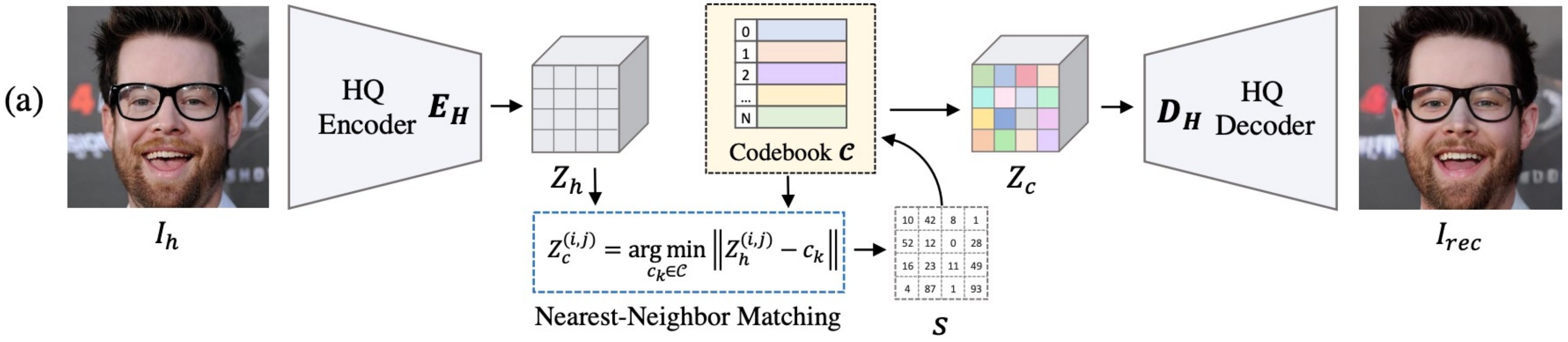


Framework of CodeFormer

It contains three training stages



Stage I: Codebook Learning (VQGAN)



As shown in Fig. 2(a), the HQ face image $I_h \in \mathbb{R}^{H \times W \times 3}$ is first embeded as a compressed feature $Z_h \in \mathbb{R}^{m \times n \times d}$ by an encoder E_H. Following VQVAE [35] and VQGAN [11], we replace each "pixel" in Z_h with the nearest item in the learnable codebook $\mathcal{C} = \{c_k \in \mathbb{R}^d\}_{k=0}^{N}$ to obtain the quantized feature $Z_c \in \mathbb{R}^{m \times n \times d}$ and the corresponding code token sequence $s \in \{0, \cdots, N-1\}^{m \cdot n}$:

$$Z_c^{(i,j)} = \arg\min_{c_k \in \mathcal{C}} \|Z_h^{(i,j)} - c_k\|_2; \quad s^{(i,j)} = \arg\min_{k} \|Z_h^{(i,j)} - c_k\|_2. \tag{1}$$

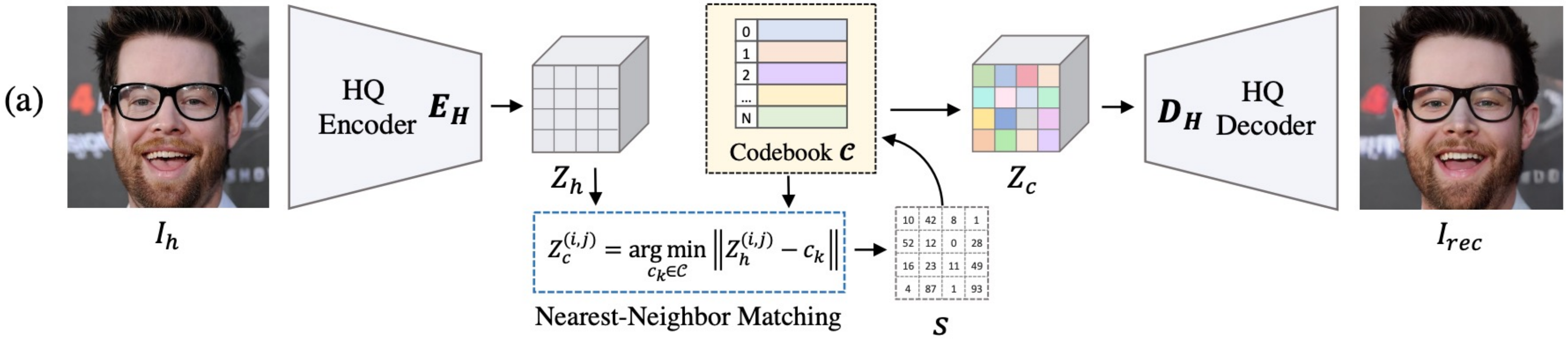
Straight-through gradient estimator

This argmin operation is a bit concerning, since it is non-differentiable with respect to the encoder.

But in practice everything seems to work fine if you just pass the decoder gradient directly through this operation to the encoder (i.e. set its gradient to 1 wrt the encoder and the quantized codebook vector; and to 0 wrt all other codebook vectors)

Bengio et al. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation, 2013
https://hassanaskary.medium.com/intuitive-explanation-of-straight-through-estimators-with-pytorch-implementation-71d99d25d9d0

Stage I: Codebook Learning (VQGAN)

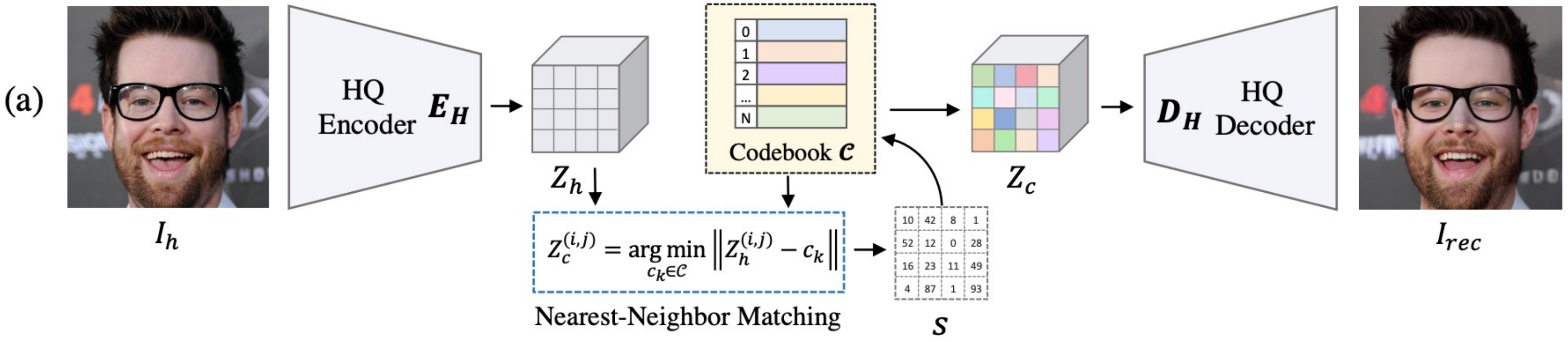


$$\mathcal{L}_1 = \|I_h - I_{rec}\|_1; \quad \mathcal{L}_{per} = \|\Phi(I_h) - \Phi(I_{rec})\|_2^2; \quad \mathcal{L}_{adv} = [\log D(I_h) + \log(1 - D(I_{rec}))]$$

$$\mathcal{L}_{code}^{feat} = \|\mathrm{sg}(Z_h) - Z_c\|_2^2 + \beta \|Z_h - \mathrm{sg}(Z_c)\|_2^2$$

- Image-level losses are underconstrained when updating the codebook items, we adopt intermediate code-level loss
- A **bi-directional problem** here: learning codebook vectors that align to the encoder outputs and learning encoder outputs that align to a codebook vector.

Stage I: Codebook Learning (VQGAN)



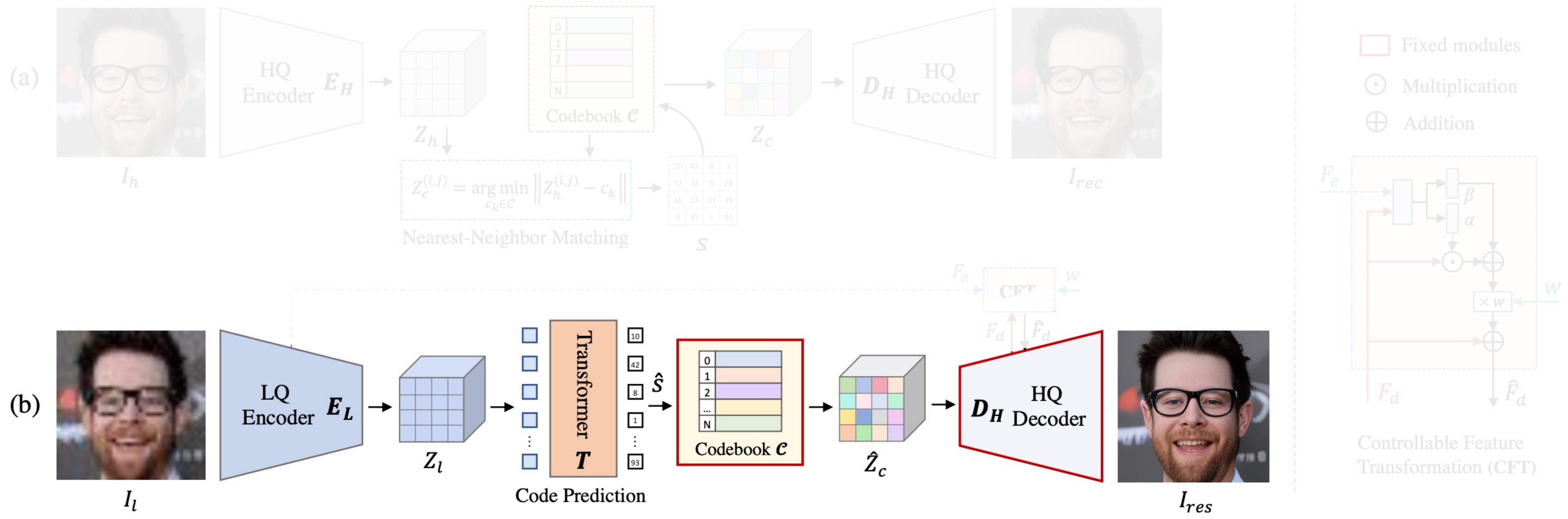
$$\mathcal{L}_1 = \|I_h - I_{rec}\|_1; \quad \mathcal{L}_{per} = \|\Phi(I_h) - \Phi(I_{rec})\|_2^2; \quad \mathcal{L}_{adv} = [\log D(I_h) + \log(1 - D(I_{rec}))]$$

$$\mathcal{L}_{code}^{feat} = \underline{\|\mathrm{sg}(Z_h) - Z_c\|_2^2} + \underline{\beta\|Z_h - \mathrm{sg}(Z_c)\|_2^2}$$

- **Two terms**:
  - Codebook alignment loss, whose goal is to get the chosen codebook vector as close to the encoder output as possible. There is a stop gradient operator on the encoder output because this term is only intended to update the codebook.
  - Codebook commitment loss, it is meant to solve the inverse problem of getting the encoder output to commit as much as possible to its closest codebook vector

sg stands for the stopgradient operator that is defined as identity at forward computation time and has zero partial derivatives, thus effectively constraining its operand to be a non-updated constant

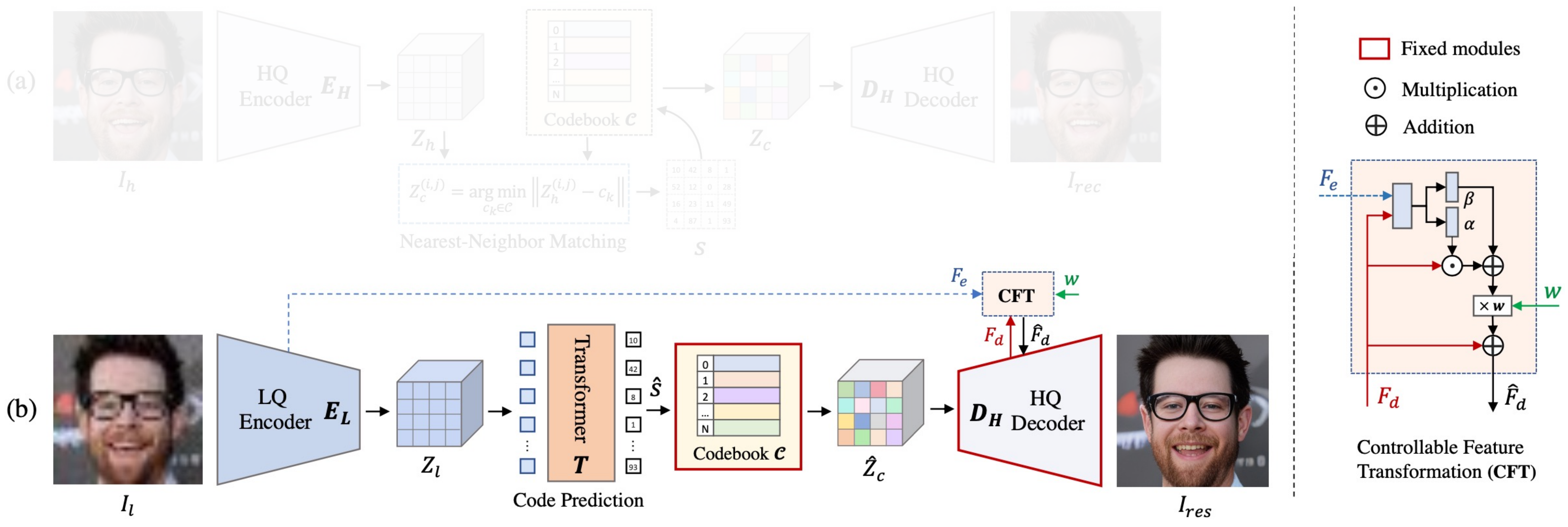
Stage II: Codebook Lookup Transformer



$$\mathcal{L}_{code}^{token} = \sum_{i=0}^{mn-1} -s_i \log(\hat{s}_i); \quad \mathcal{L}_{code}^{feat'} = \|Z_l - \mathrm{sg}(Z_c)\|_2^2$$

- Cross-entropy loss for code token prediction supervision
- L2 loss to force the LQ feature Z_l to approach the quantized feature Z_c from codebook

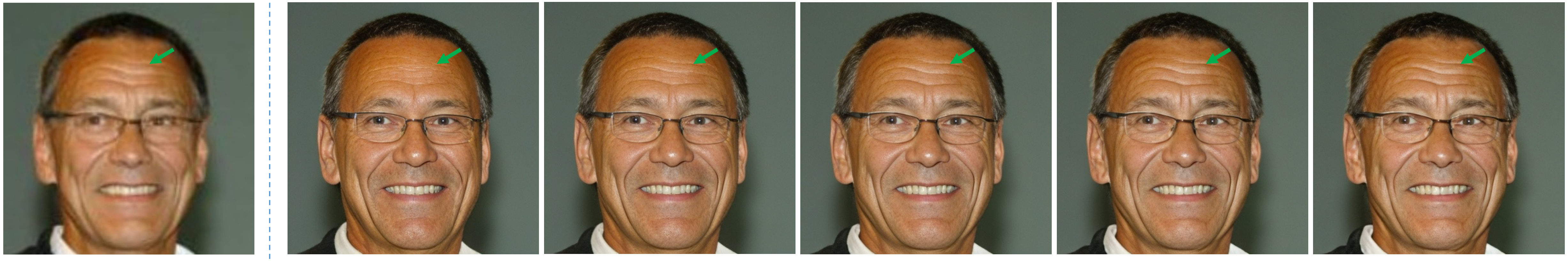
Stage III: Controllable Feature Transformation



$$\hat{F}_d = F_d + (\alpha \odot F_d + \beta) \times w; \quad \alpha, \beta = \mathcal{P}_\theta(c(F_d, F_e))$$

Stage III: Controllable Feature Transformation

higher quality ⟷ **higher fidelity**



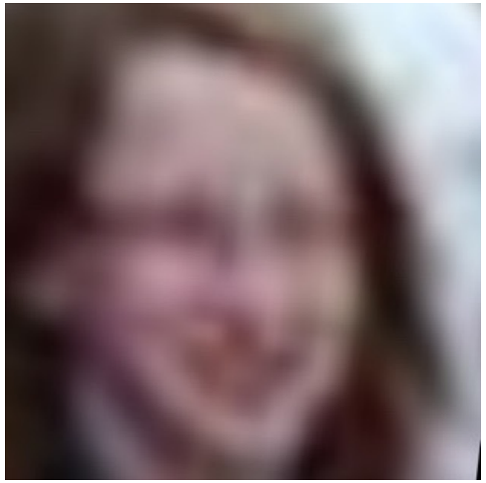
Input | $w = 0$ | $w = 0.25$ | $w = 0.5$ | $w = 0.75$ | $w = 1$

Continuous Transitions between Image **Quality** and **Fidelity** via **Controllable Feature Transformation Module**

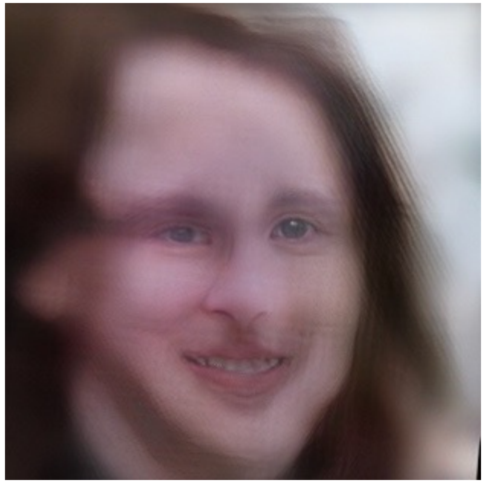
Evaluation on blind face restoration

Real Input DFDNet GFP-GAN GPEN CodeFormer (Ours)

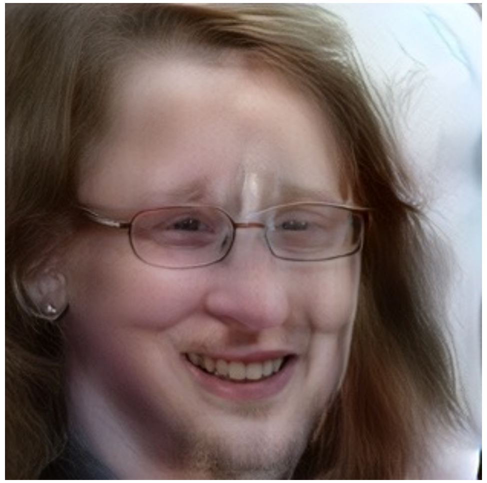
Evaluation on blind face restoration



Real Input



DFDNet



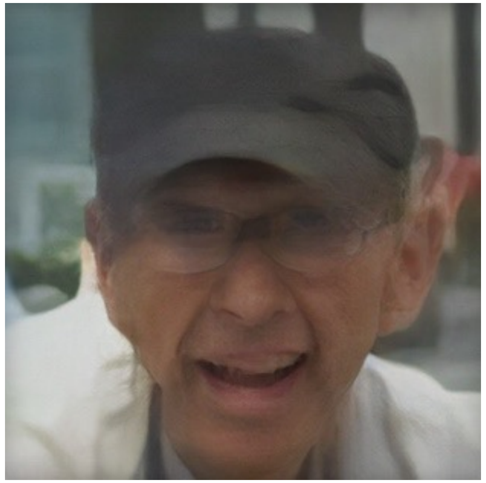
GFP-GAN

GPEN

CodeFormer (Ours)

Evaluation on blind face restoration

Real Input



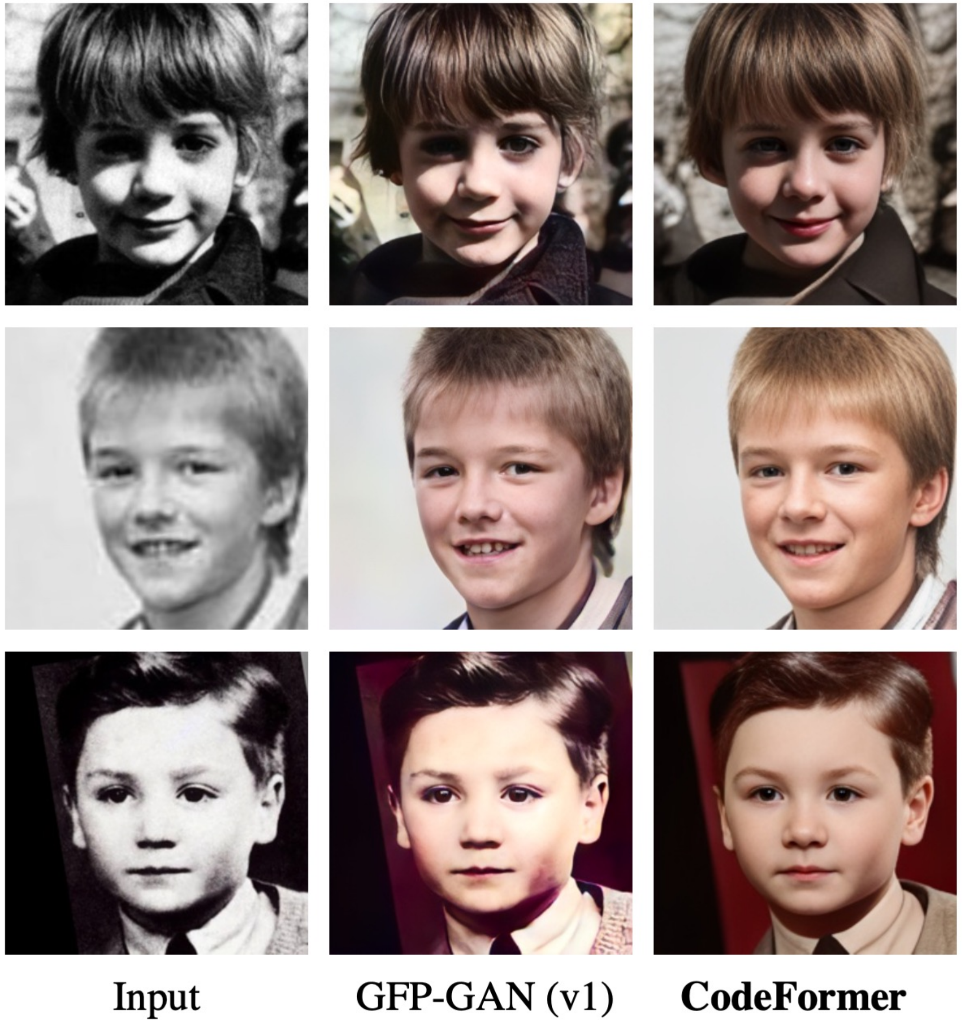
DFDNet

GFP-GAN

GPEN

CodeFormer (Ours)

Face color enhancement



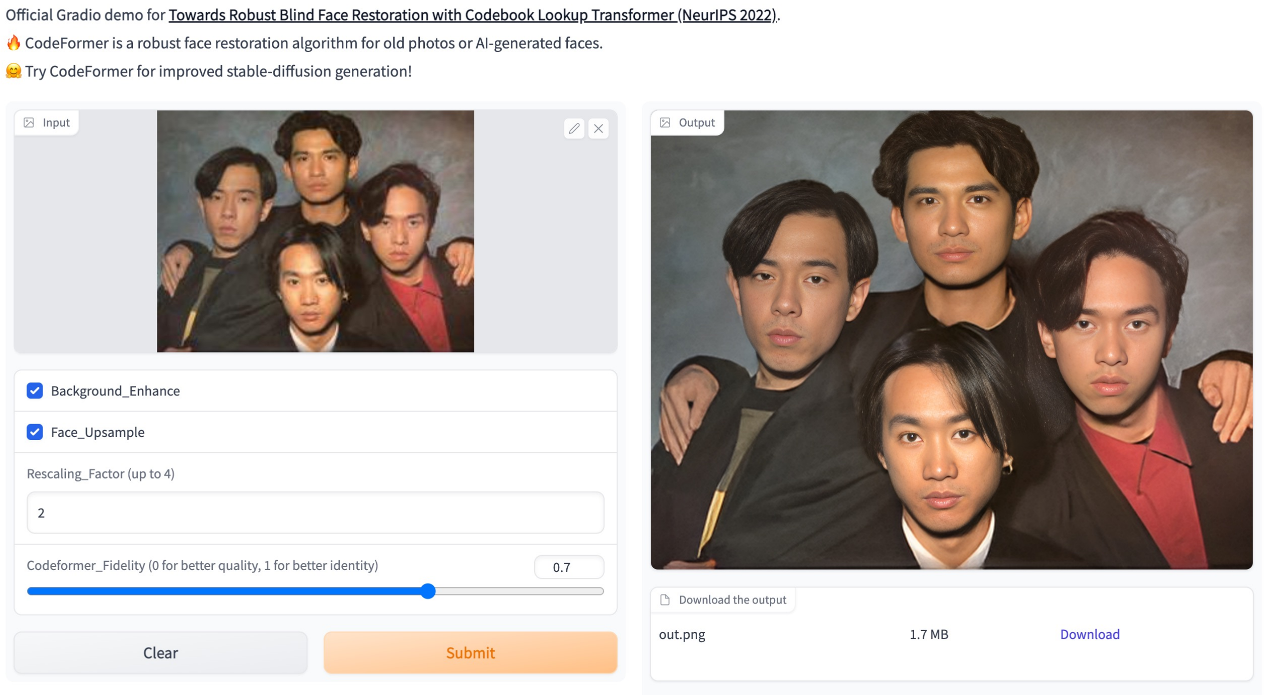
Face inpainting

Masked Input | CTSDG | GPEN | **CodeFormer** | GT

Face inpainting (extremely large mask)

Masked Input (extremely large mask) | CTSDG | GPEN | **CodeFormer**

Code and demo



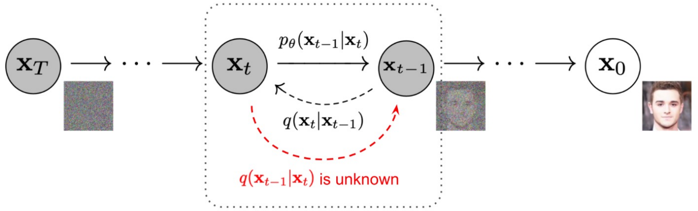
https://github.com/sczhou/CodeFormer

🤗 https://huggingface.co/spaces/sczhou/CodeFormer

Discussions

- Next generation of generative priors

  StyleGAN2 -> VQGAN -> Diffusion Model

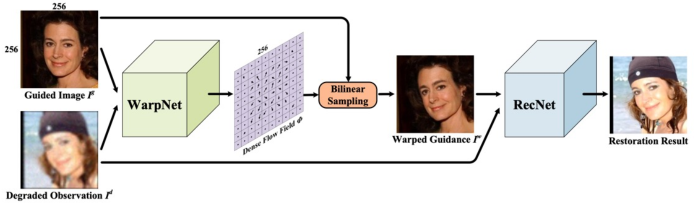


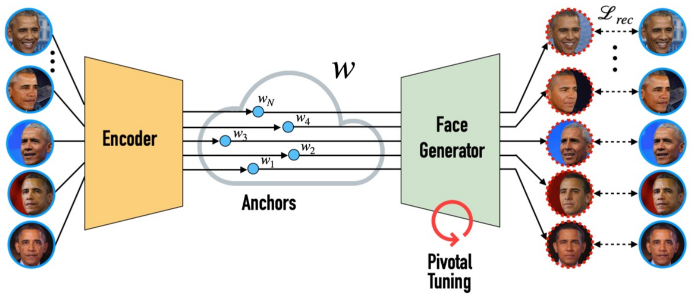
- Identity inconsistency issue

  Training Setting; Network Structure;

  Reference-based model (e.g., Li et al);

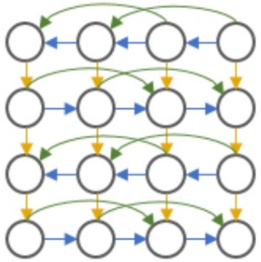
  Personalized model (e.g., MyStyle)



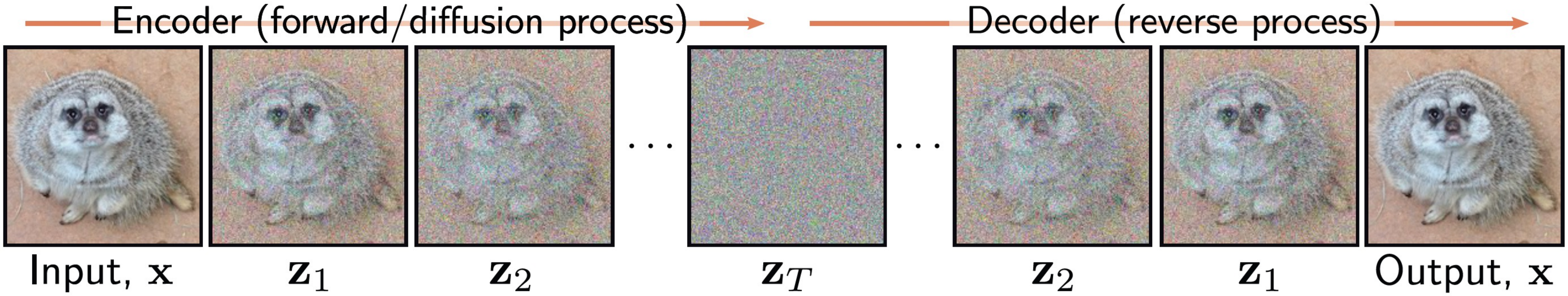


- Video face restoration

  Recurrent networks (e.g., BasicVSR series)



More Generic Prior from Diffusion Models?



It is unclear how restoration can be achieved via diffusion model

- Diffusion model is stochastic! How to keep the prior and maintain fidelity?
- Diffusion model hasn't seen relevant degradations! How to handle complex degradations?
- Diffusion model is slow! How to improve inference efficiency?

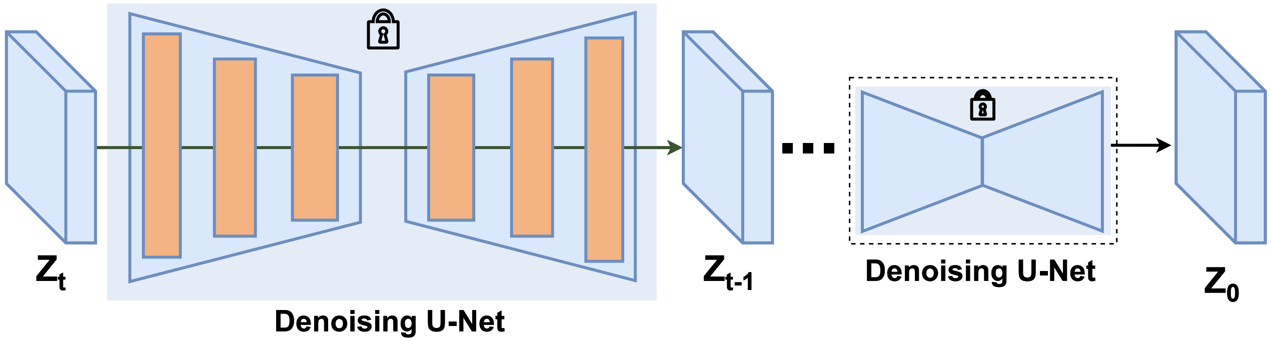
Image Credit: Simon J.D. Prince, [Understanding Deep Learning](), 2023

StableSR | Framework

Keeping the prior and fidelity

- Frozen stable diffusion model as a backbone

- Minimal alterations to prevent disrupting the prior



Jianyi Wang et al. Exploiting Diffusion Prior for Real-World Image Super-Resolution. arXiv May 2023

StableSR | Framework

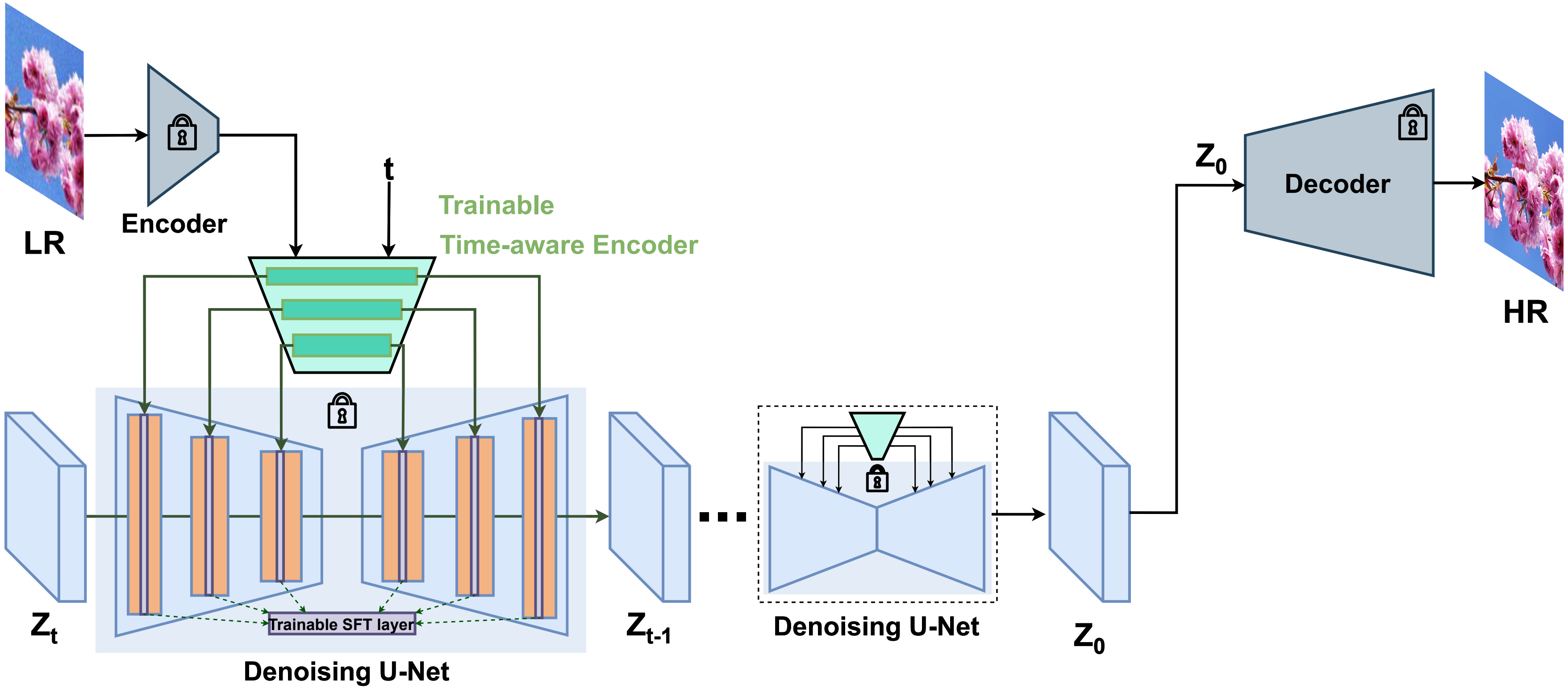
Keeping the prior and fidelity

- Train only the time-aware encoder and spatial feature transformation layer

$$\boldsymbol{\alpha}^n, \boldsymbol{\beta}^n = \mathcal{M}_\theta^n(\mathbf{F}^n)$$

$$\hat{\mathbf{F}}^n_{\text{dif}} = (1+\boldsymbol{\alpha}^n) \odot \mathbf{F}^n_{\text{dif}} + \boldsymbol{\beta}^n$$

- Adaptively adjust the condition strength derived from the LR feature through *t*

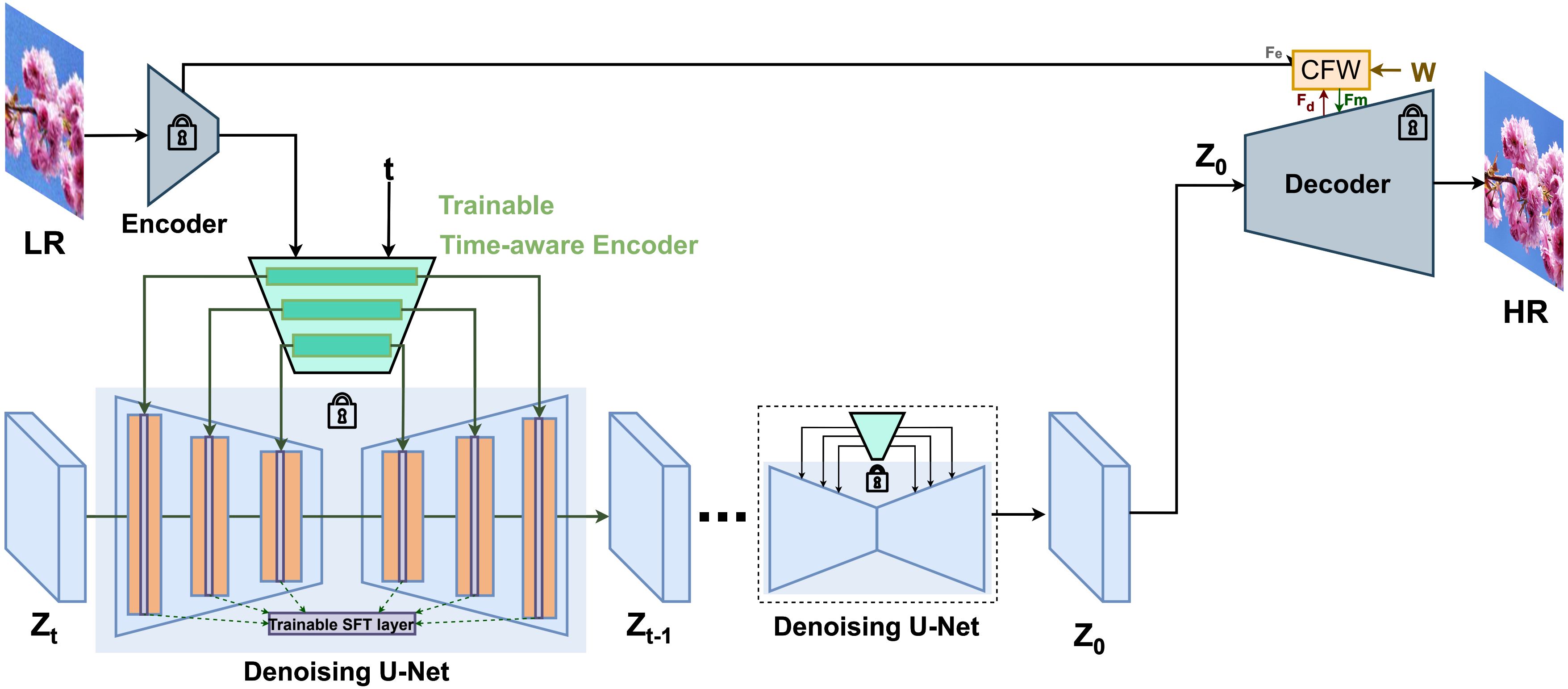


Xintao Wang et al. Recovering Realistic Texture in Image Super-resolution by Deep Spatial Feature Transform. CVPR 2018

StableSR | Fidelity-Realism Trade-off

Keeping the prior and fidelity

- Add a controllable skip connection to benefit from structural guidance from the LR image, enhancing fidelity
- Control the modulation strength through **w**
- A larger **w** allows stronger structural guidance



Shangchen Zhou et al. Towards Robust Blind Face Restoration with Codebook Lookup Transformer. NeurIPS 2022

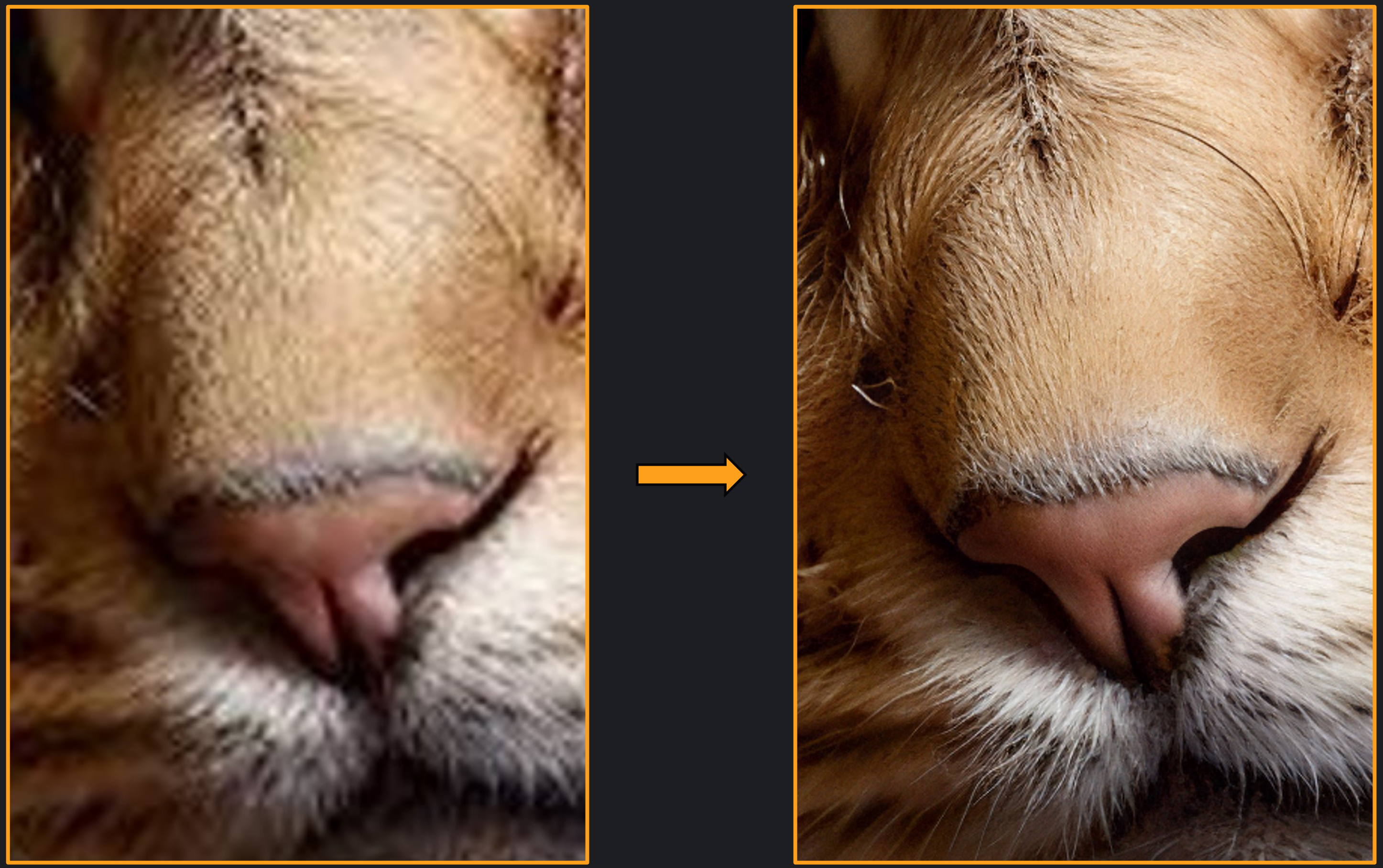

Zoomed LR (4x)
StableSR
Real-world SR


Zoomed LR (4x)
StableSR
Real-world SR


Zoomed LR (4x)
StableSR
Old Photo SR


Zoomed LR (4x)
StableSR
Old Photo SR


Zoomed LR (4x)
StableSR
AIGC SR (1024x1536 to 4096x6144)



Problems to solve

- Extending diffusion prior to video restoration
- Recovering natural scene with the right semantics is hard
- A neat way to deal with different resolutions
- Diffusion model is still slow