

# Efficient Scale-Invariant Generator with Column-Row Entangled Pixel Synthesis



Thuan Hoang Nguyen\*



Thanh Van Le\*



Anh Tran

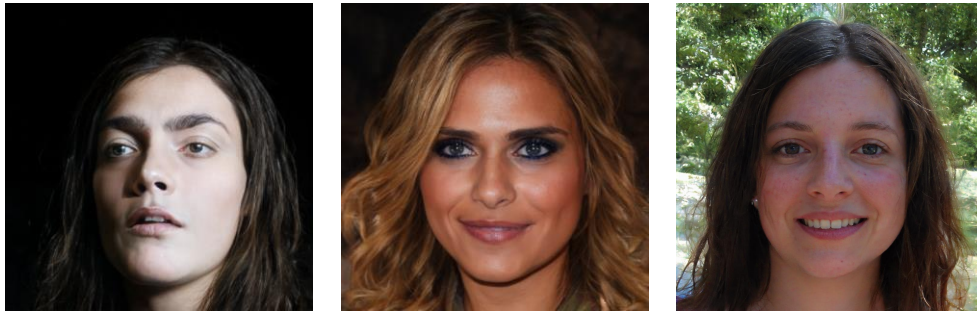
\*Equal contribution

Poster: THU-PM-172

# CREPS Preview

- Traditional image generators are trained for a **fixed** resolution
- Cannot generate images at higher resolution

**256 x 256**

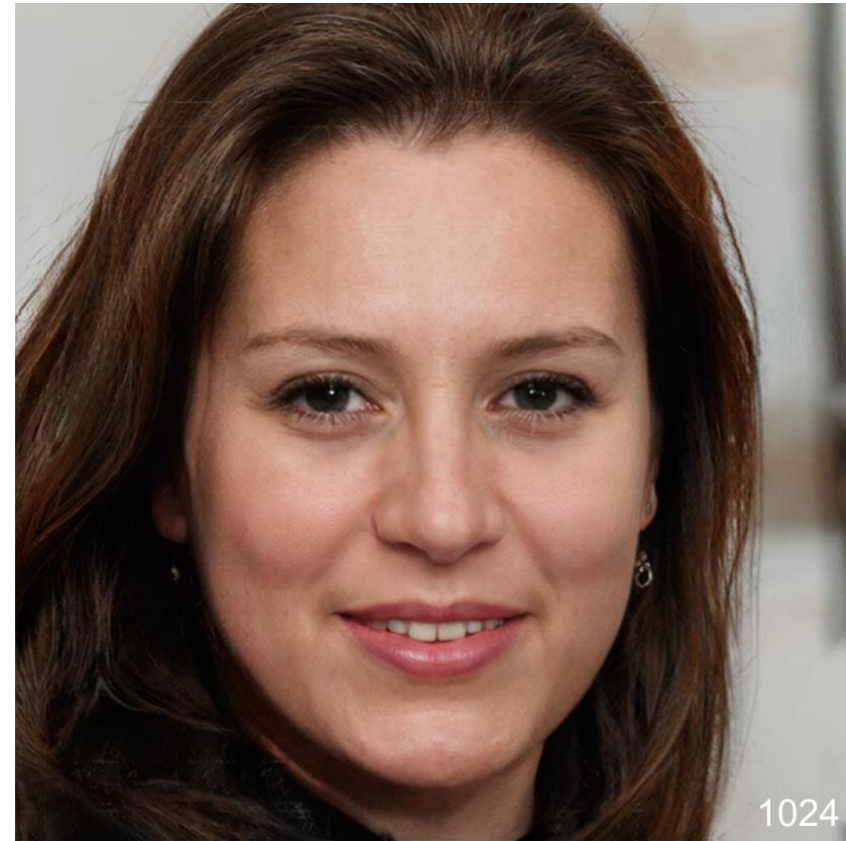
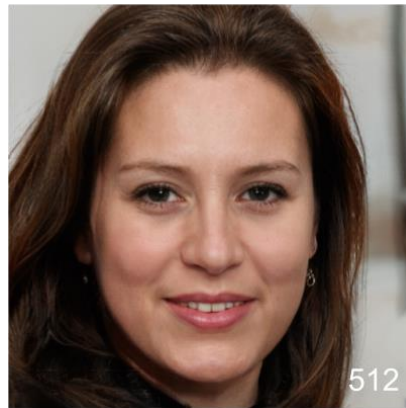
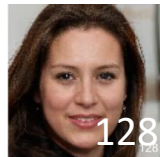


**1024 x 1024**



# CREPS Preview

- We focus on the task of any-scale synthesis
- Once trained, the model can generate images at arbitrary scales.



# CREPS Preview

- This task has been gaining attention recently
- However, previous works are either:

**Inconsistent**



**Slow**



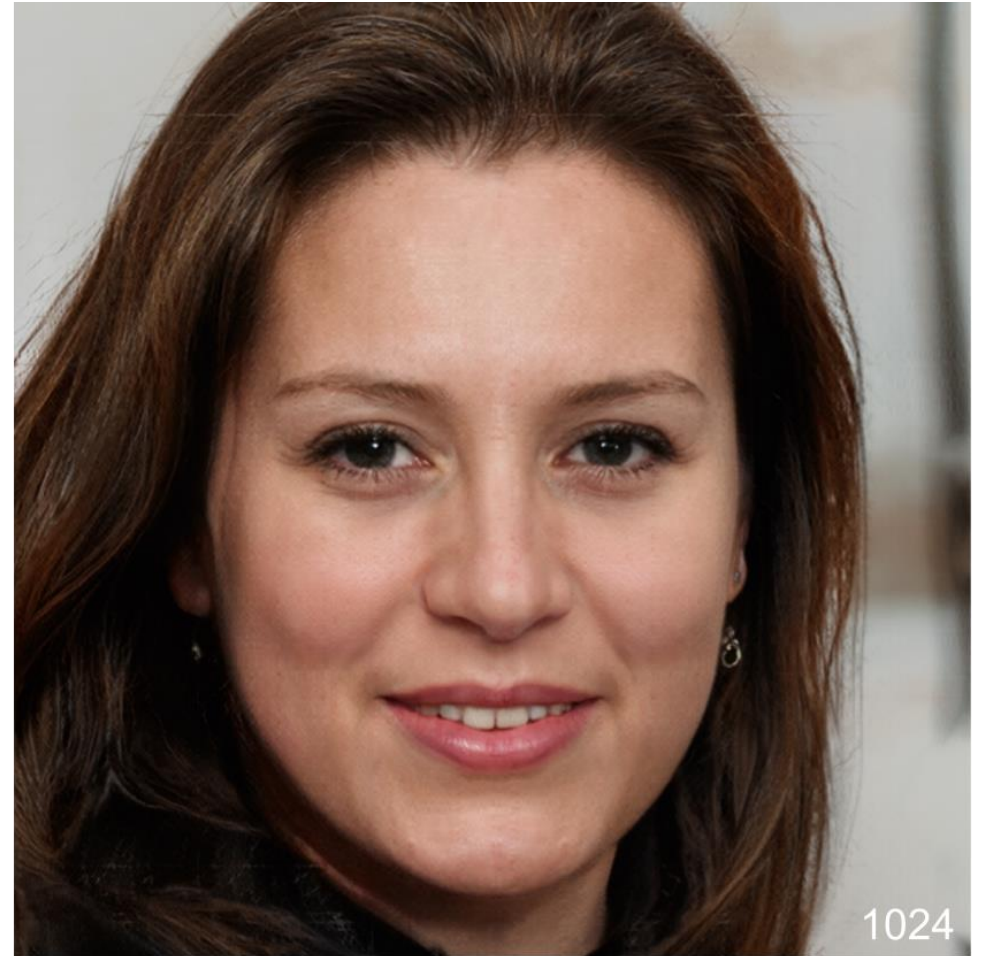
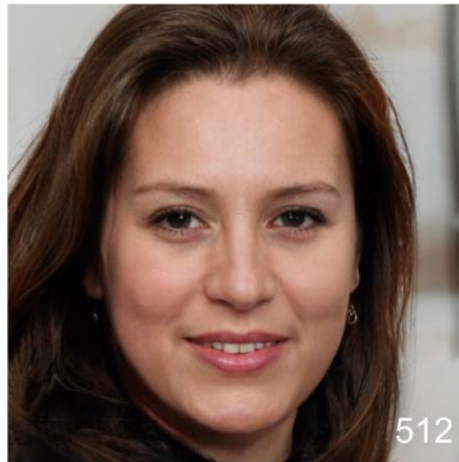
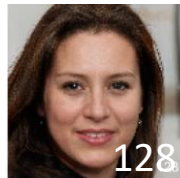
# CREPS Preview

- Whereas, CREPS is
  - **Efficient** as previous generators like StyleGAN2.
  - **Scale-invariant** regardless the output resolution.
  - Able to **generalize to unseen** geometric transformations.



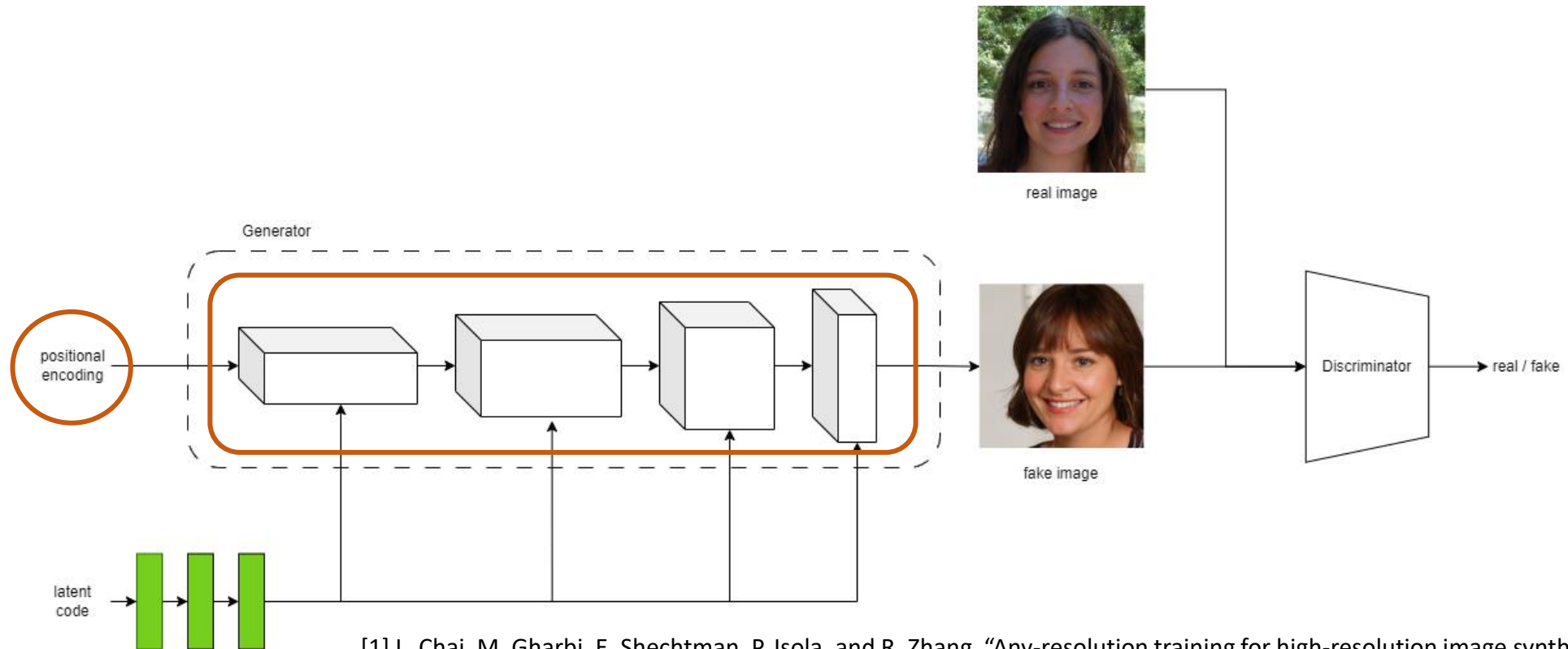
# Any-scale Image Synthesis

Same latent input  $z$



# Previous works (1)

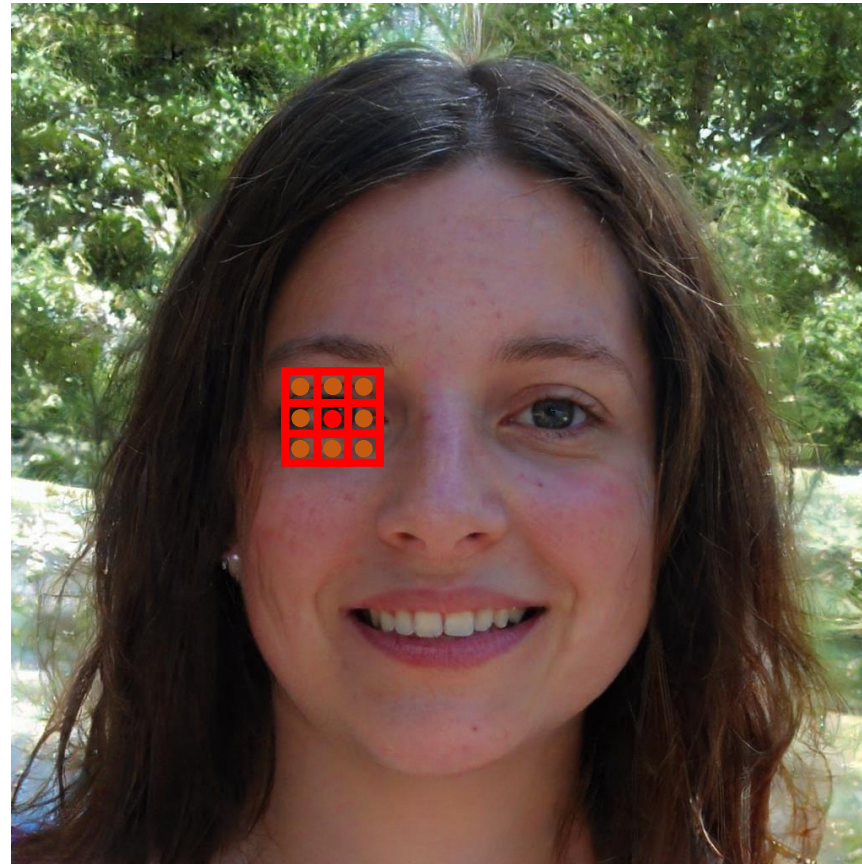
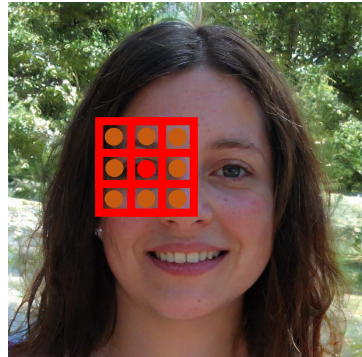
- AnyresGAN [1] and ScaleParty [2]
  - Traditional “fixed-size” generators & introduce non-trivial re-design



[1] L. Chai, M. Gharbi, E. Shechtman, P. Isola, and R. Zhang. “Any-resolution training for high-resolution image synthesis”. In *ECCV 2022*  
 [2] E. Ntavelis, M. Shahbazi, I. Kastanis, R. Timofte, M. Danelljan, and L. V. Gool. “Arbitrary-Scale Image Synthesis”. In *CVPR 2022*.

# Previous works (1)

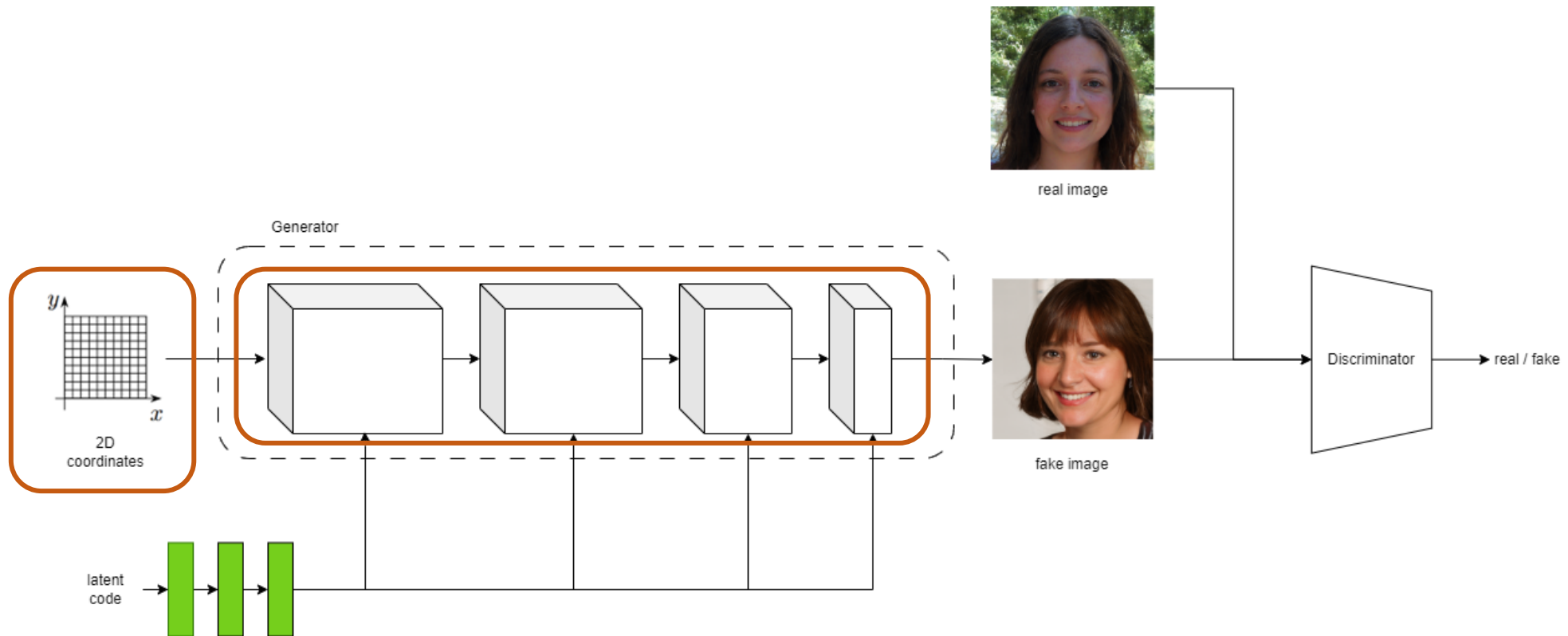
- AnyresGAN [1] and ScaleParty [2]
  - Rely on scale-inconsistent operators (upsampling, spatial conv...)





# Previous works (2)

- INR-GAN [3] and CIPS [4]
  - Utilize **spatial-invariance** of implicit neural representation.



# Previous works (2)

- INR-GAN [3] and CIPS [4]
  - Slow and much more memory-consuming other GAN counterparts

Resolution	Batch size	Memory Usage			Running time		
		StyleGAN2	CIPS	Ours	StyleGAN2	CIPS	Ours
256 × 256	1	1.5GB	3.3GB	2.3GB	0.04s	0.06s	0.03s
	4	2.5GB	10.2GB	5.2GB	0.05s	0.23s	0.06s
512 × 512	1	1.7GB	10.4GB	4.5GB	0.04s	0.21s	0.05s
	4	3.4GB	OOM	14.6GB	0.06s	OOM	0.16s

# Column-Row Entangled Pixel Synthesis

Therefore, in this work, we present Column-Row Entangled Pixel Synthesis (CREPS) with three criteria in mind:

Efficiency

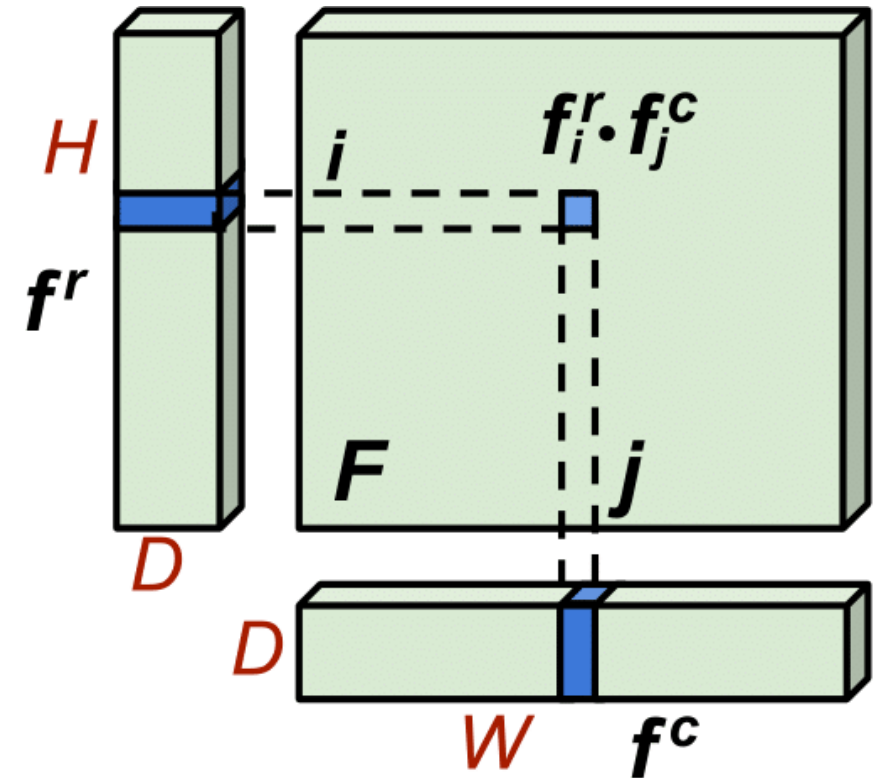
Spatial-Consistency

High-quality

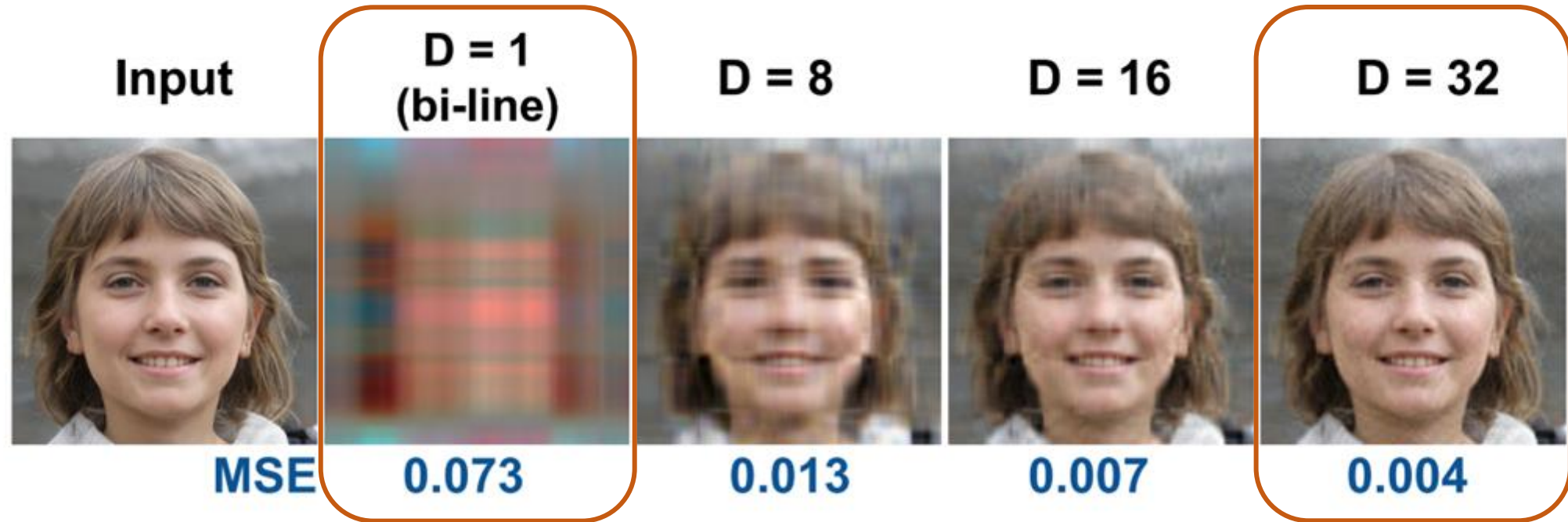
# Bi-line Representation

Decompose a **2D** feature into **row and column embeddings**:

- Add “**thickness**”  $D$  to increase representation power
- Compose back to 2D feature via **dot product**



# Bi-line Representation

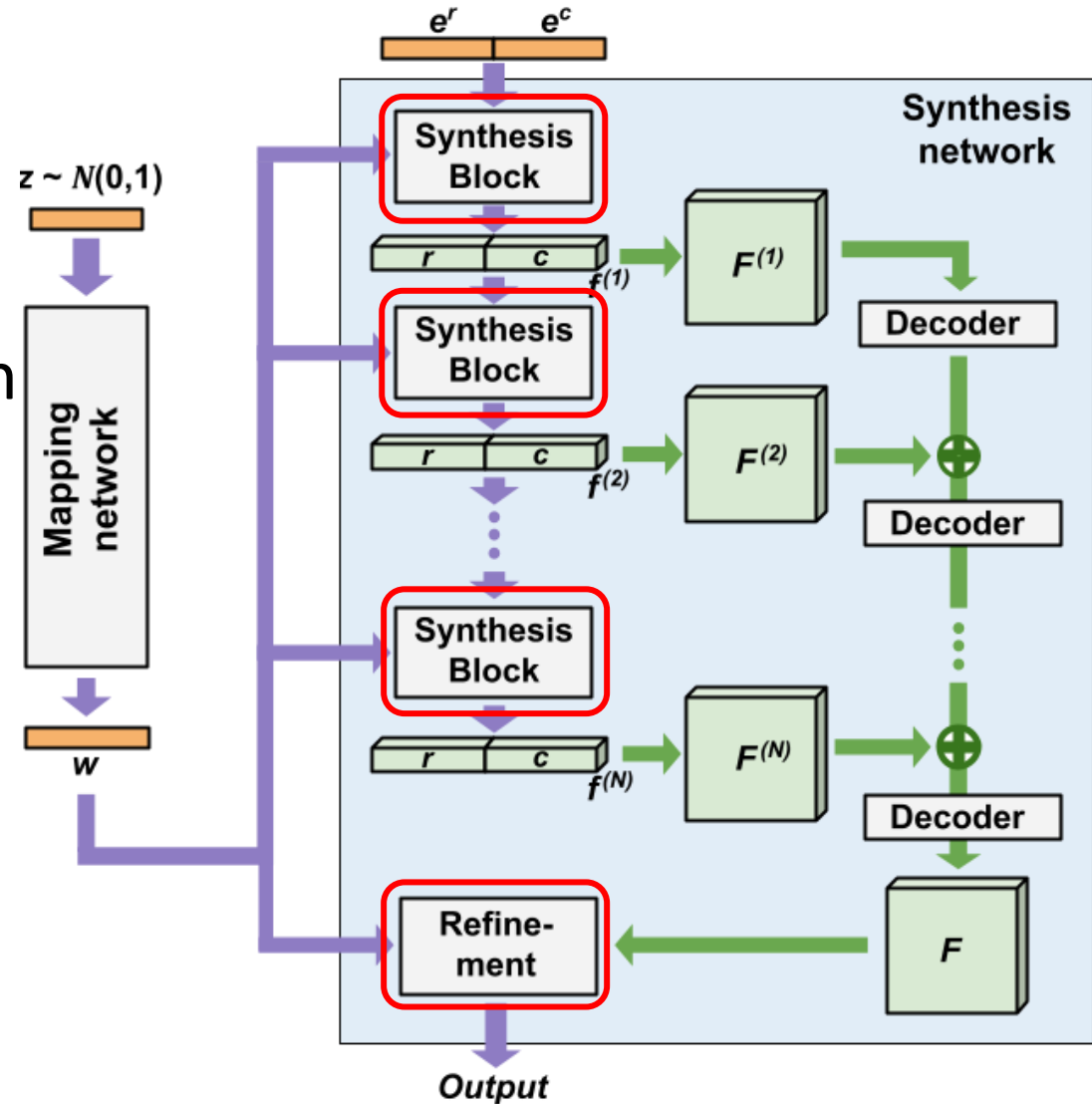


Fitting an input image to the thick bi-line representation in the image space.

# CREPS Architecture

On the right is our architecture based on StyleGAN2, with three notable features:

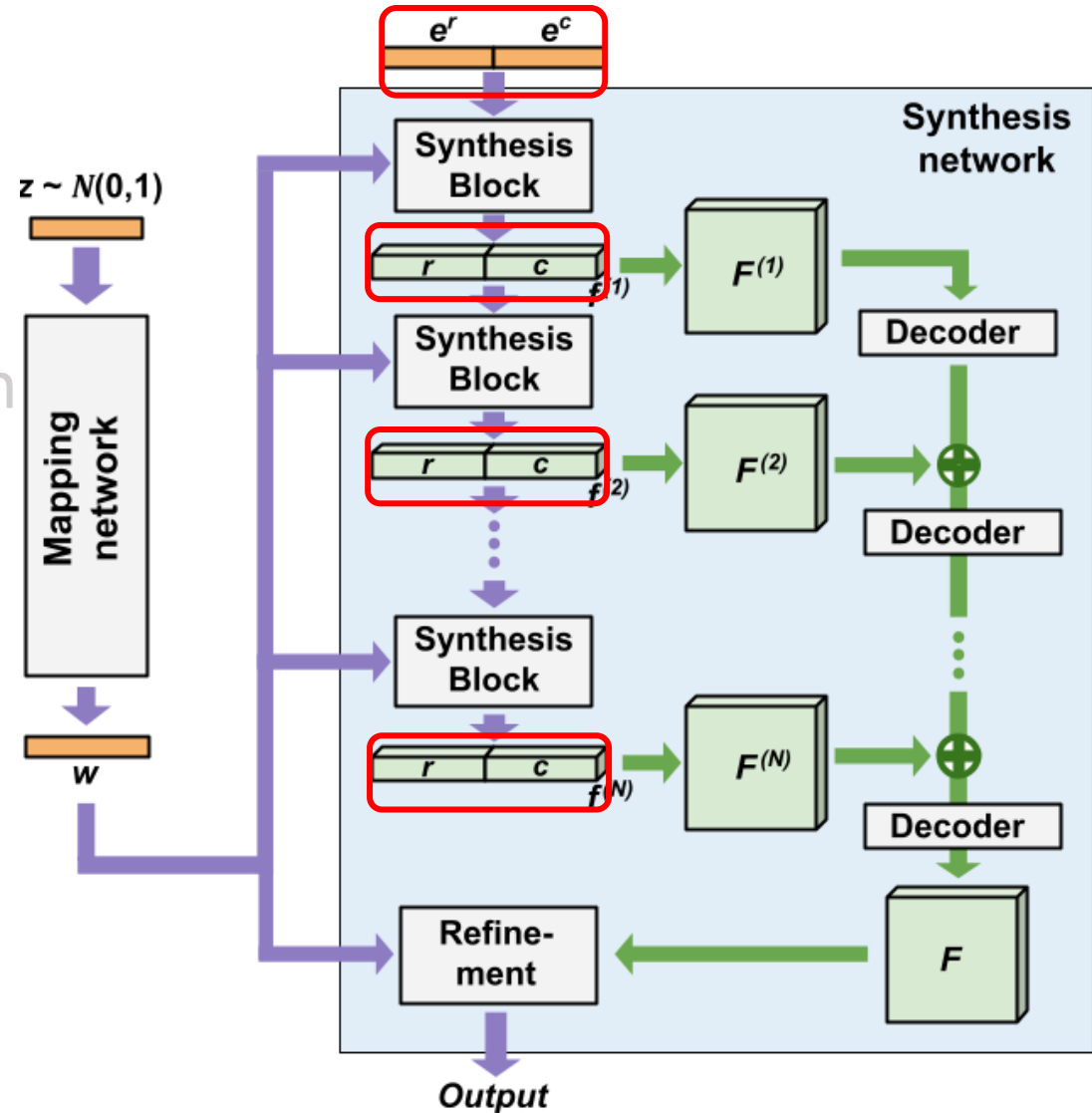
- Upsampling and 3x3 convolution removal



# CREPS Architecture

On the right is our architecture based on StyleGAN2, with three notable features:

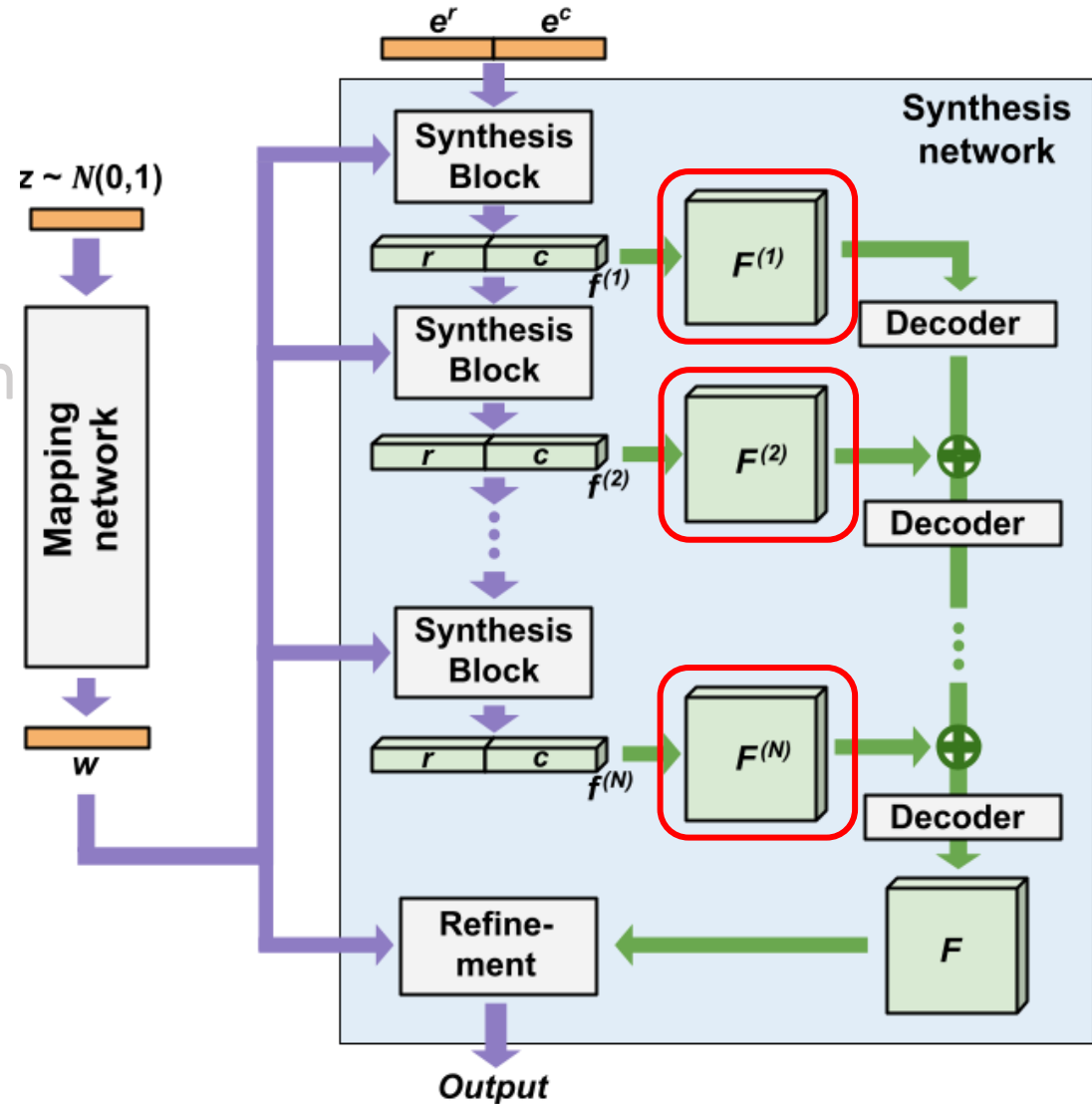
- Upsampling and 3x3 convolution removal
- Bi-line decomposition



# CREPS Architecture

On the right is our architecture based on StyleGAN2, with three notable features:

- Upsampling and 3x3 convolution removal
- Bi-line decomposition

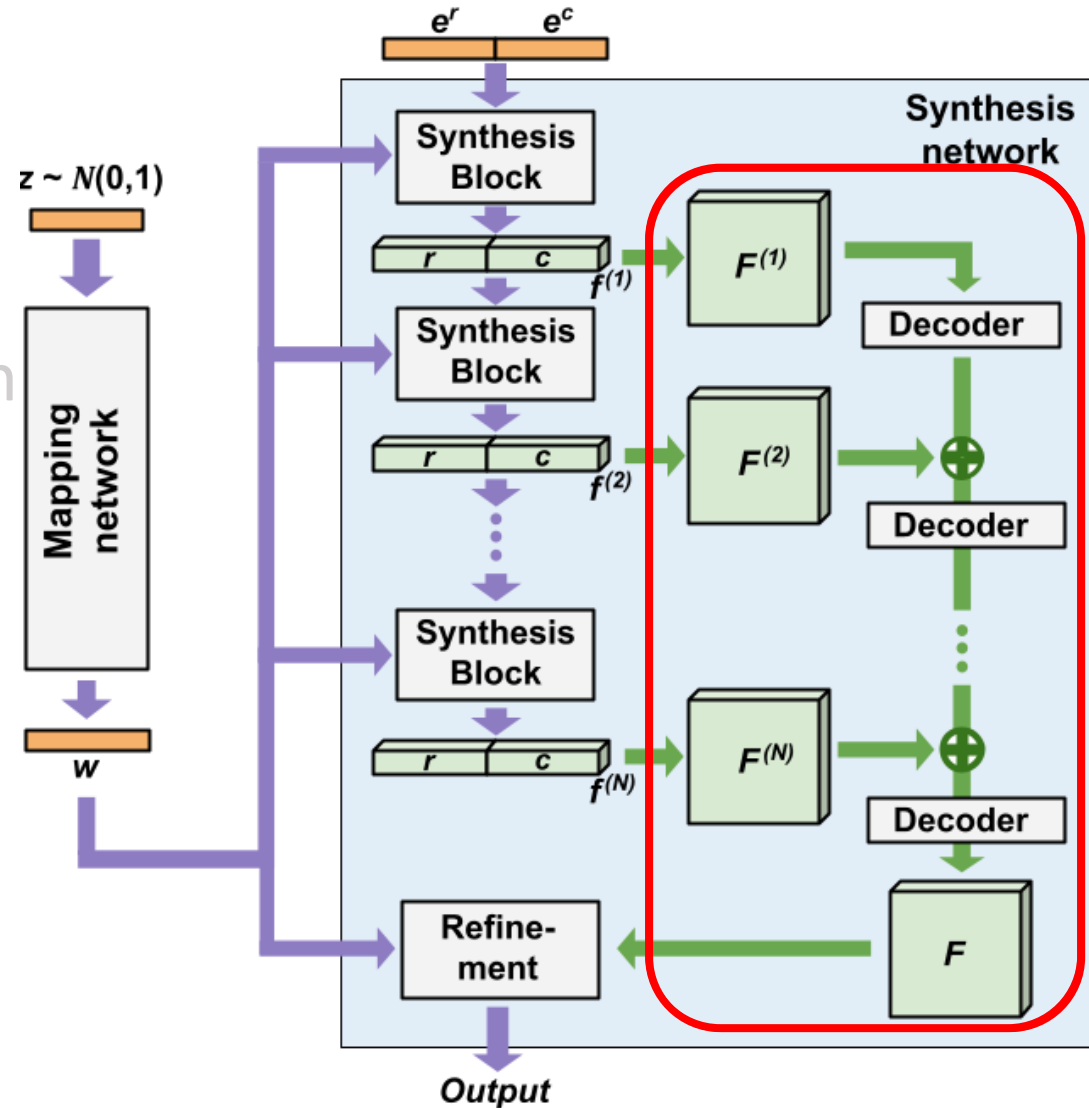




# CREPS Architecture

On the right is our architecture based on StyleGAN2, with three notable features:

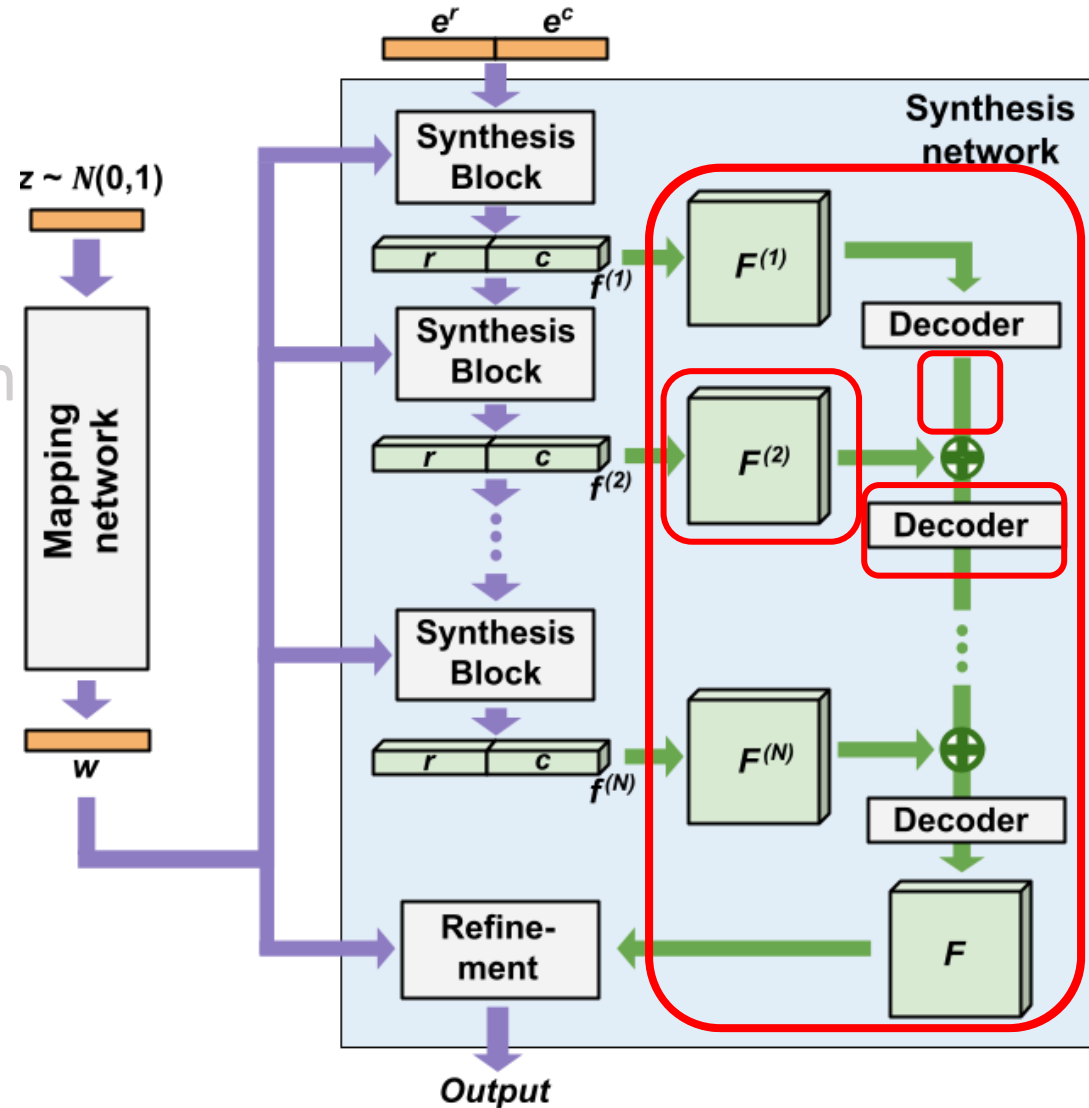
- Upsampling and 3x3 convolution removal
- Bi-line decomposition
- Layer-wise feature composition



# CREPS Architecture

On the right is our architecture based on StyleGAN2, with three notable features:

- Upsampling and 3x3 convolution removal
- Bi-line decomposition
- Layer-wise feature composition



# Experiments

## Image Generation

Generator	FFHQ-512	FFHQ-1024	LSUN Church - 256	Metfaces-1024	Scenery-256
StyleGAN2	<b>3.41</b>	<b>2.84</b>	<u>3.86</u>	<b>18.22</b>	<b>6.40</b>
CIPS	6.18	10.07	<b>2.92</b>	OOM	8.49
Ours	<u>4.43</u>	<u>4.09</u>	5.50	<u>20.52</u>	<u>7.21</u>

# Experiments

## Image Generation

**FFHQ**



**Flickr-Scenery**



**Metfaces**

**LSUN-Church**

# Experiments

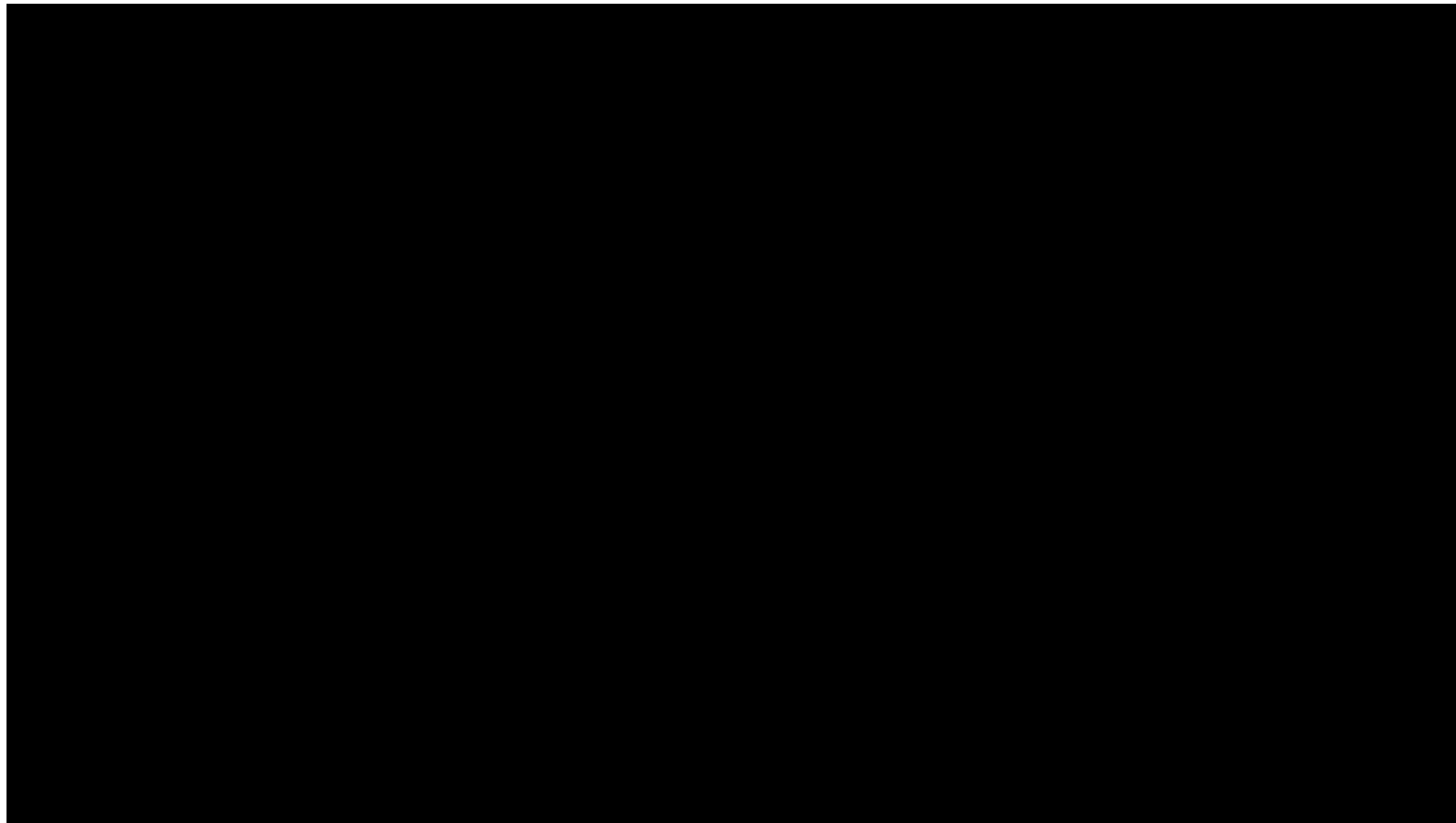
## Scale consistency

- Compare two sets:
  1. 256x256 images
  2. 512x512 images + down-sampled to 256x256

	PSNR↑	SSIM↑	LPIPS↓
AnyresGAN	24.19	0.73	0.07
ScaleParty	24.50	0.70	0.08
CIPS	<u>33.33</u>	<u>0.93</u>	<u>0.05</u>
Ours	<b>34.65</b>	<b>0.96</b>	<b>0.01</b>

# Experiments

## Scale consistency



# Experiments

## Geometric transformation

**Translate**



**Zoom**



**Rotate**



**Distort**



# Experiments

## Geometric transformation

**Translate**



**Zoom**



**Rotate**



**Distort**





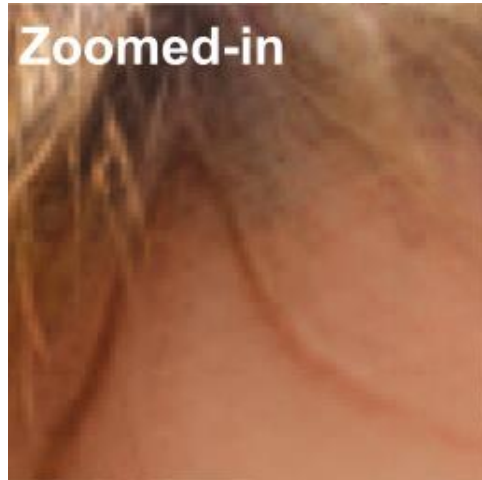
# Experiments



- Generate an image of size  $3687 \times 3687$  in a single run
- Generate an image at **any size** with **patch-based** approach

# Limitation

There are common artifacts occurring in our samples:



# THANK YOU



**Project Page**

[thuanz123.github.io/creps/](https://thuanz123.github.io/creps/)



**GitHub**

[github.com/VinAIResearch/CREPS/](https://github.com/VinAIResearch/CREPS/)