Repurposing GANs for One-shot Semantic Part Segmentation

VISTEC Thailand

CVPR2021(oral)

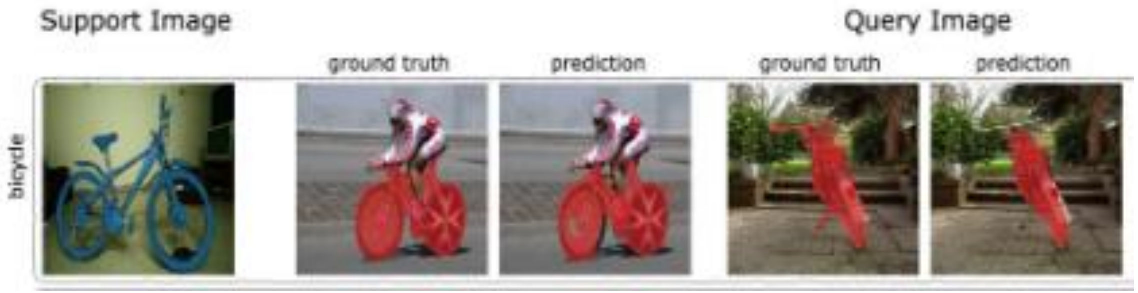
A minimal implementation https://github.com/bryandlee/repurpose-gan/

semantic part segmentation

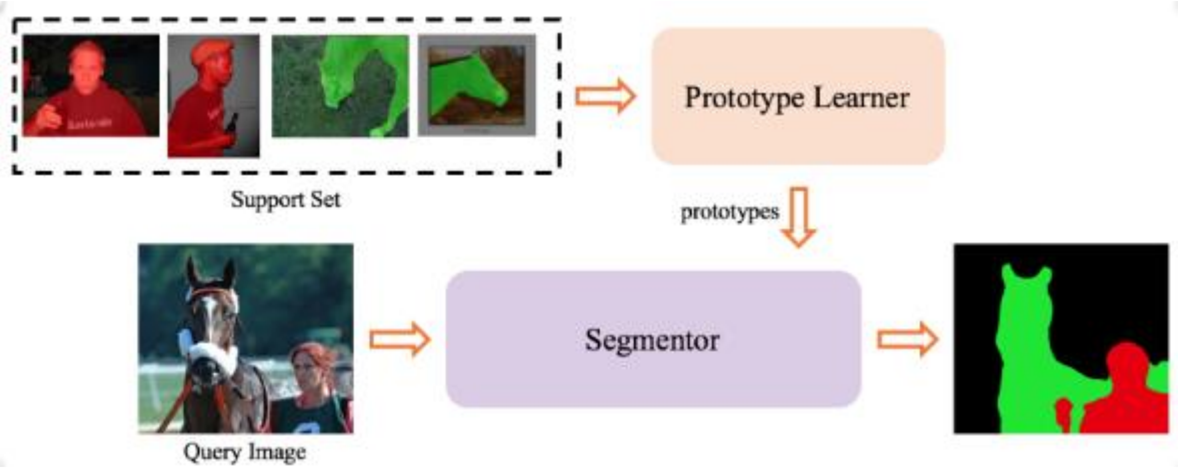
segment an unseen object from the same class.(eg. hand face leg ,belongs to human).

an *n-way* *per-pixel* classification problem

1-way

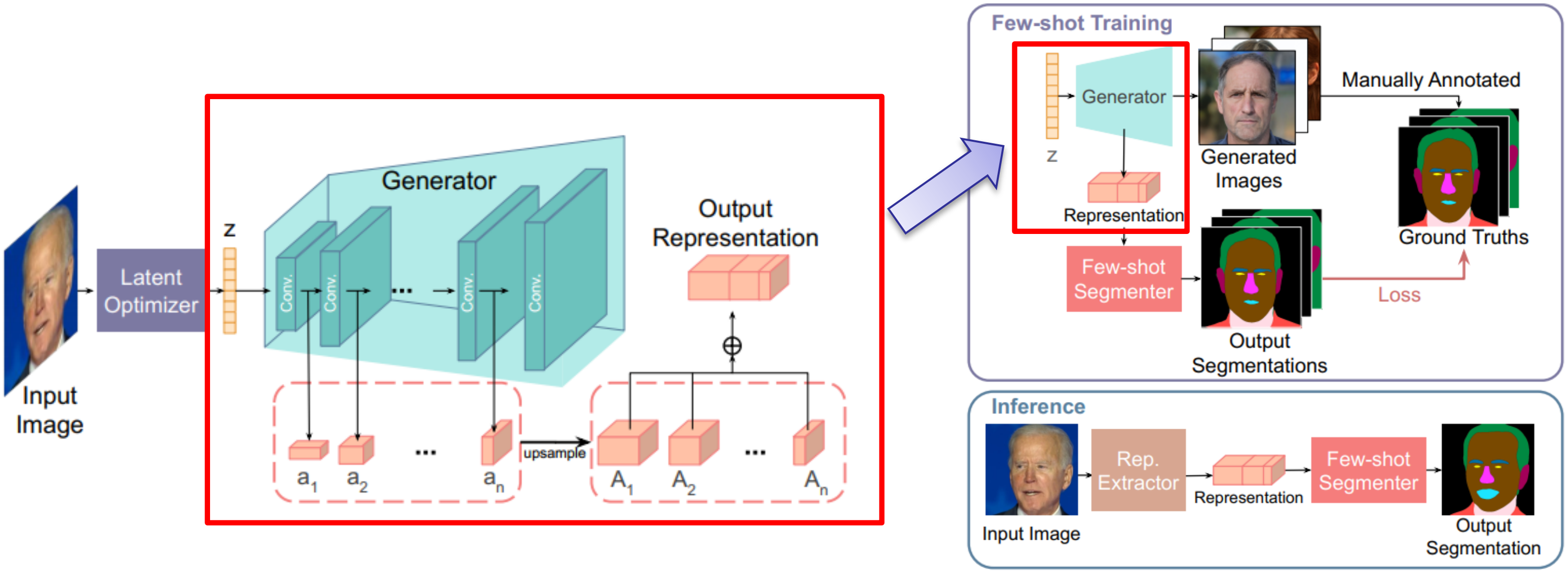


2-way



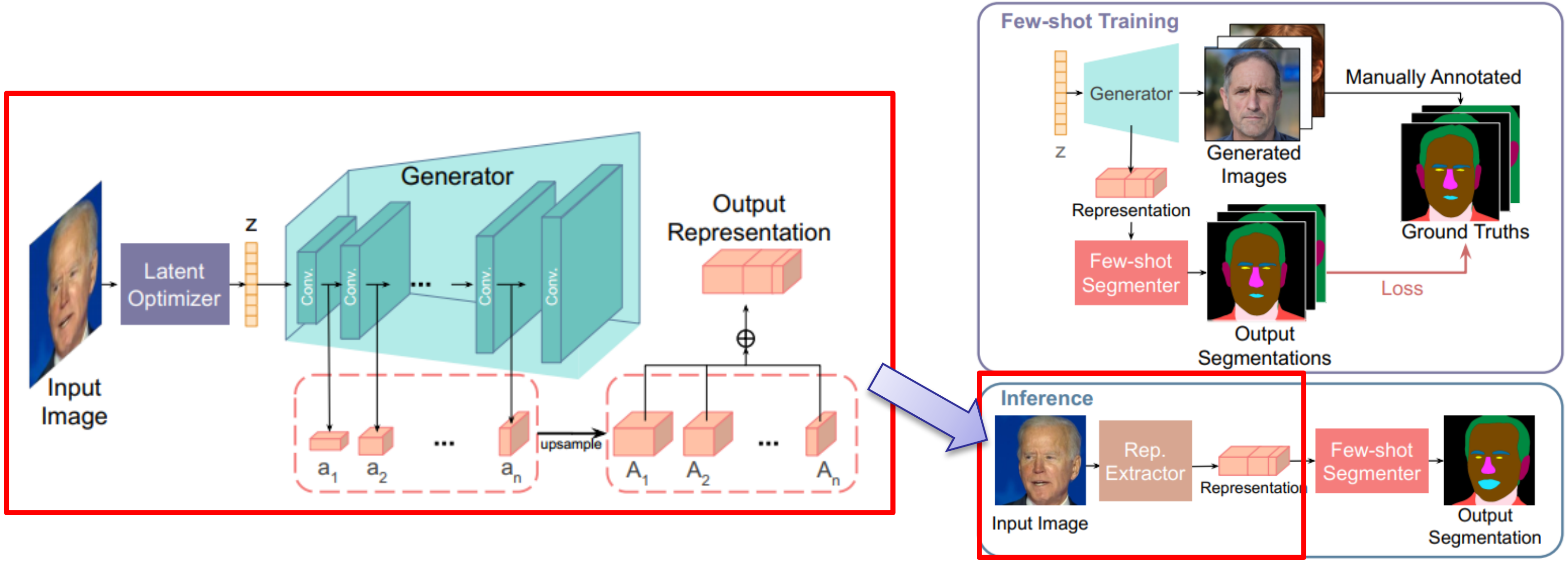
leverage a trained GAN to extract a pixel-wise representation from the input image and use it as feature vectors for a segmentation network.

1. Representation Extraction from GANs
2. Segmentation with Extracted Representation
3. Extension: Auto-shot Segmentation Network

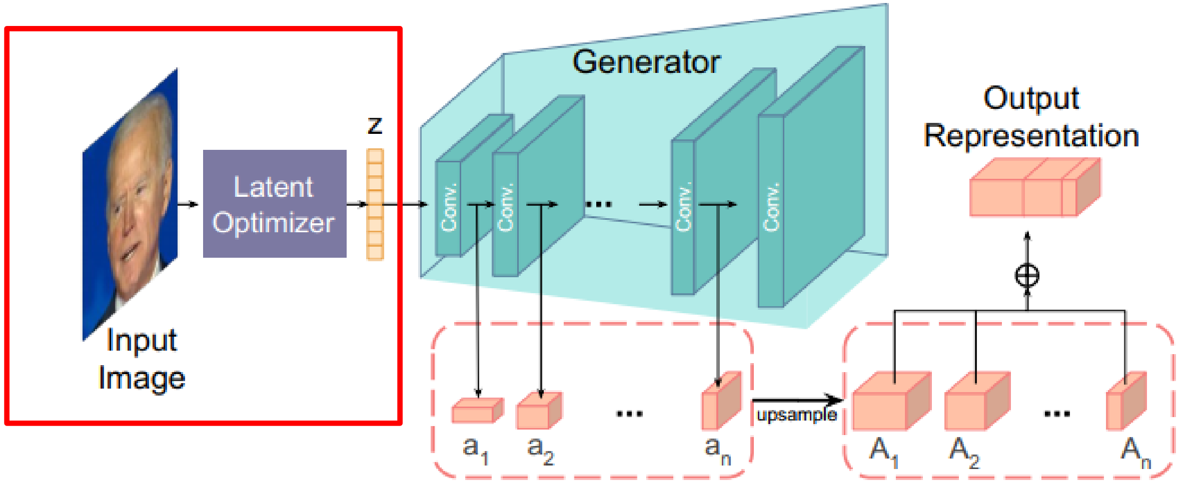


leverage a trained GAN to extract a pixel-wise representation from the input image and use it as feature vectors for a segmentation network.

1. Representation Extraction from GANs
2. Segmentation with Extracted Representation
3. Extension: Auto-shot Segmentation Network

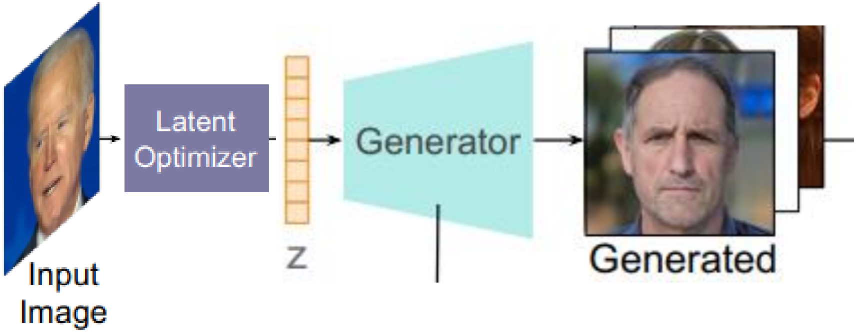


Representation Extraction from GANs

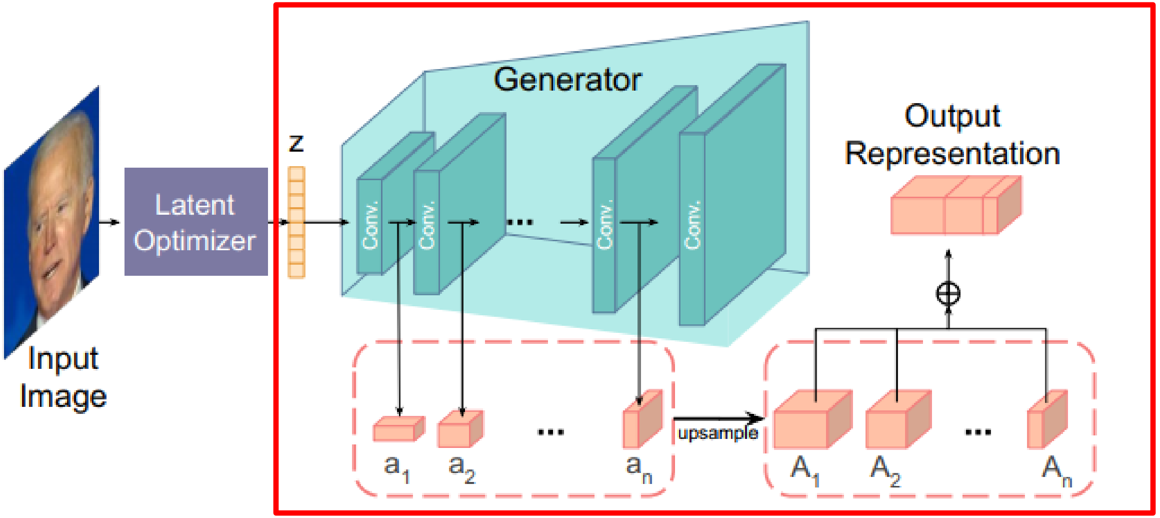


Projection of images to latent space (**StyleGAN**2)

Manipulating a given image in the latent feature space requires finding a matching latent code w.



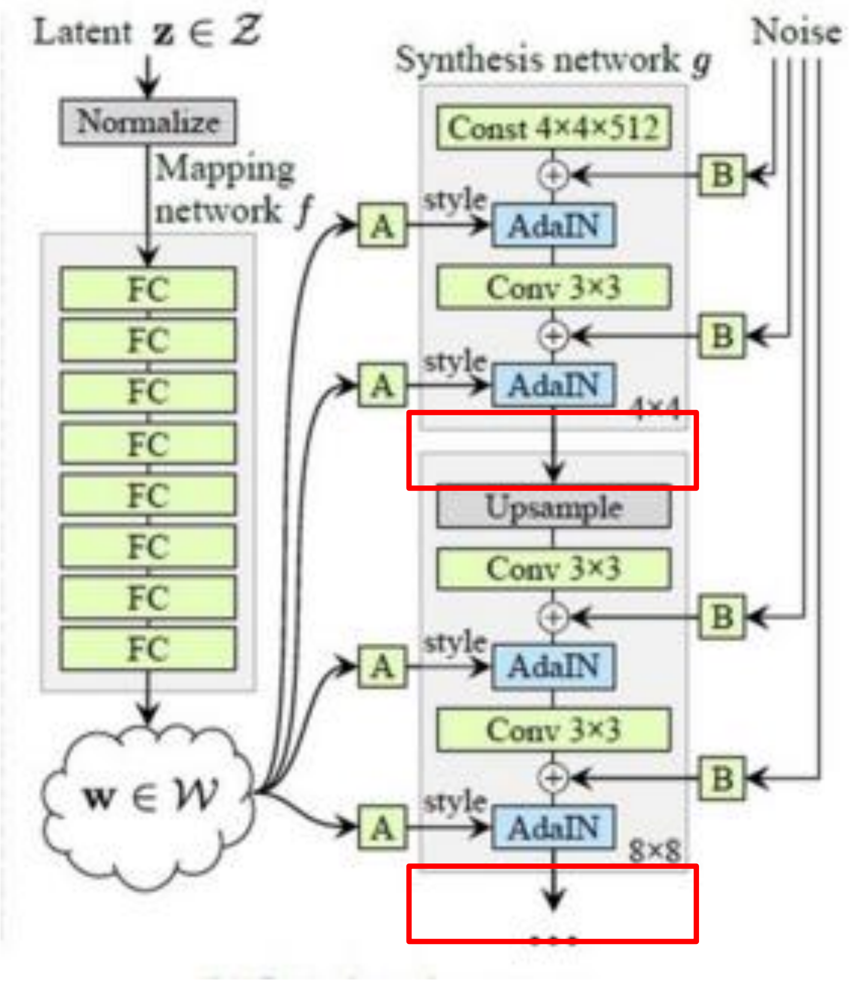
Representation Extraction from GANs

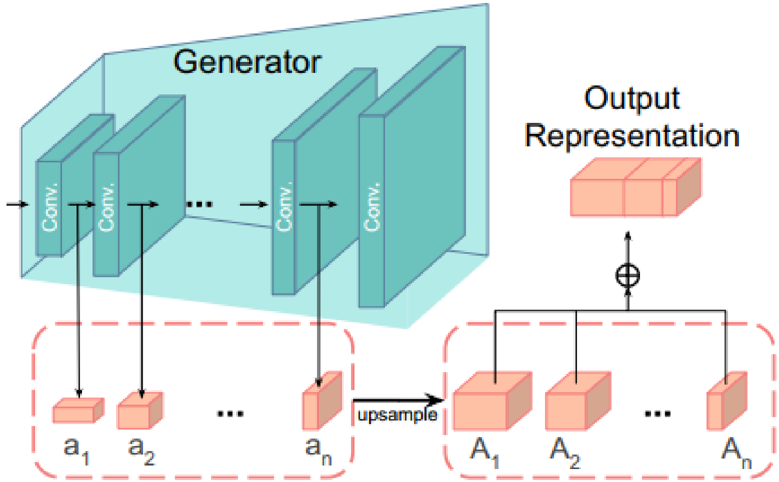


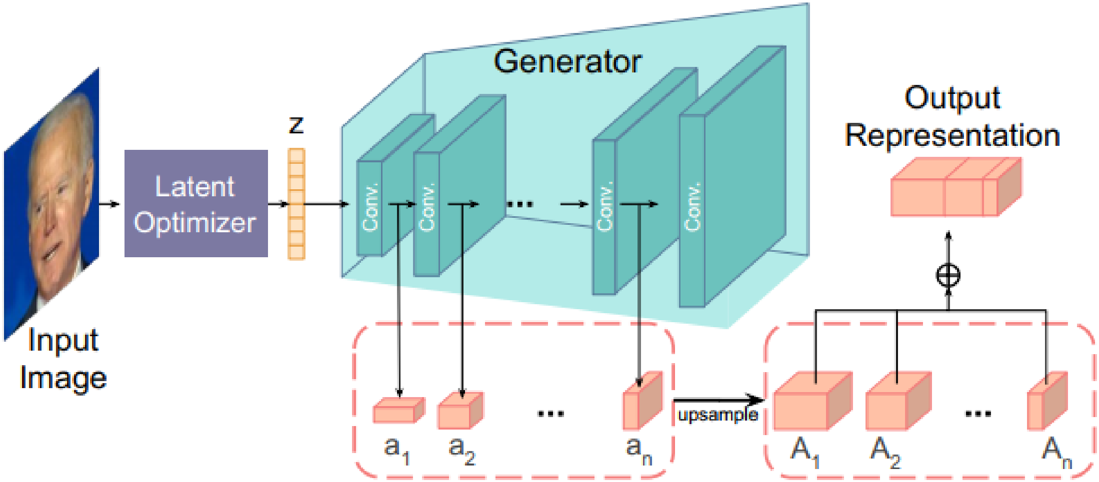
$$(h_i, w_i, c_i)$$

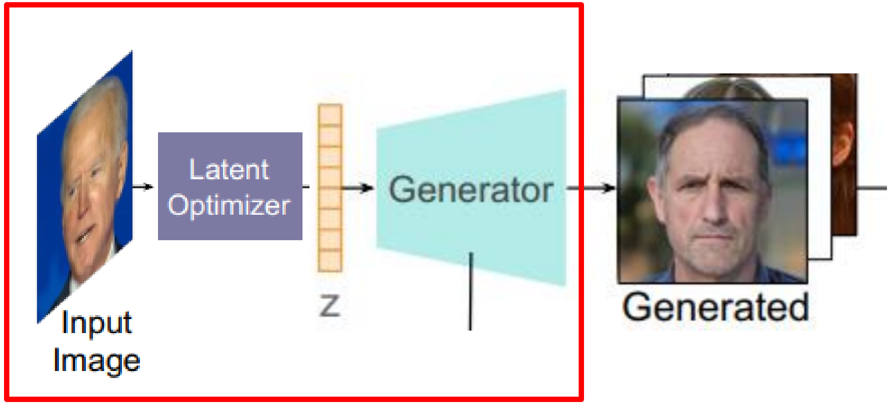
$$F = \mathbb{U}(a_1) \oplus_c \mathbb{U}(a_2) \oplus_c \ldots \oplus_c \mathbb{U}(a_n)$$

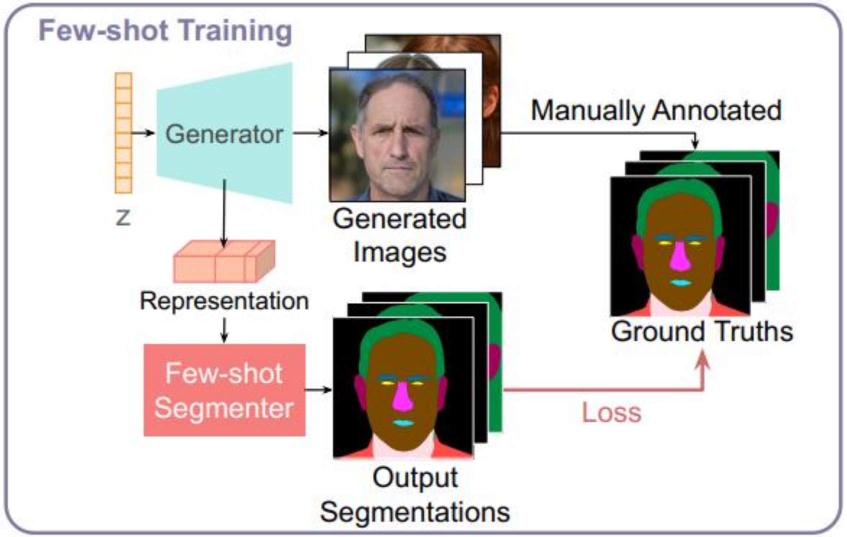
This process maps each 3-dimensional RGB pixel to a *C*-dimensional feature vector.

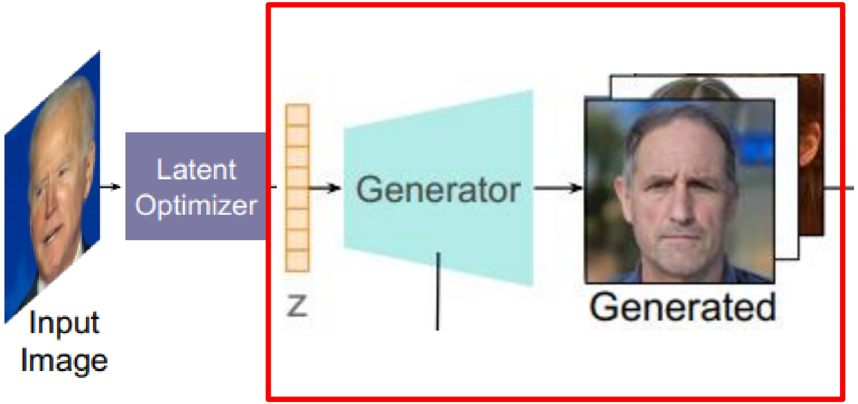

Latent $\mathbf{z} \in \mathcal{Z}$
Normalize
Mapping network f
FC
FC
FC
FC
FC
FC
FC
FC
$\mathbf{w} \in \mathcal{W}$
Synthesis network g
Noise
Const 4×4×512
A
style
AdaIN
Conv 3×3
B
4×4
Upsample
8×8


Generator
Conv.
Conv.
Conv.
Conv.
...
Output Representation
a_1
a_2
a_n
upsample
A_1
A_2
A_n
$\oplus$

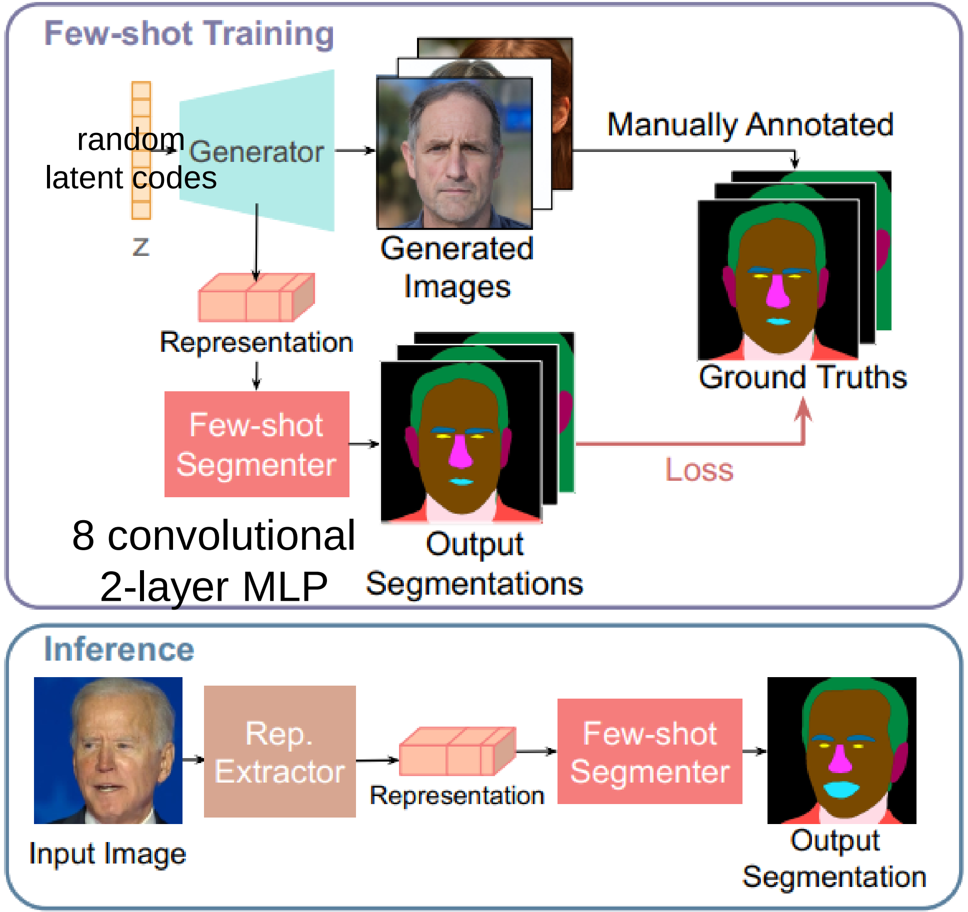

Generator
Output
Representation
z
Latent
Optimizer
Conv.
Conv.
Conv.
Conv.
Input
Image
upsample
a_1
a_2
a_n
A_1
A_2
A_n


Latent
Optimizer
Generator
z
Input
Image
Generated


Few-shot Training
Generator
z
Generated
Images
Manually Annotated
Representation
Few-shot
Segmenter
Ground Truths
Loss
Output
Segmentations


Input
Image
Latent
Optimizer
Generator
z
Generated

Segmentation with Extracted Representation



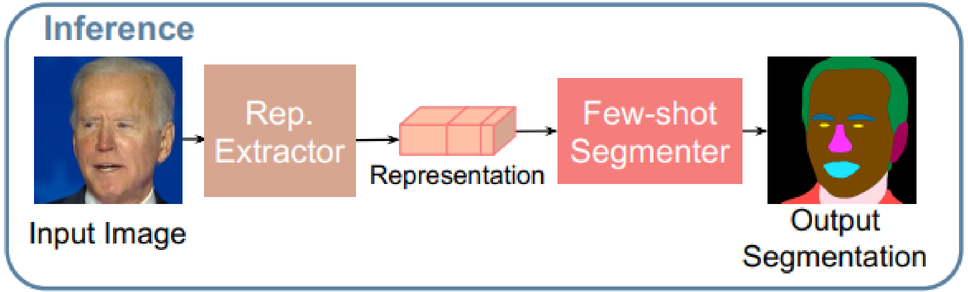
1. Generate images using trained GAN
2. manually annotate images
3. train few-shot segmenter

Extension: Auto-shot Segmentation Network

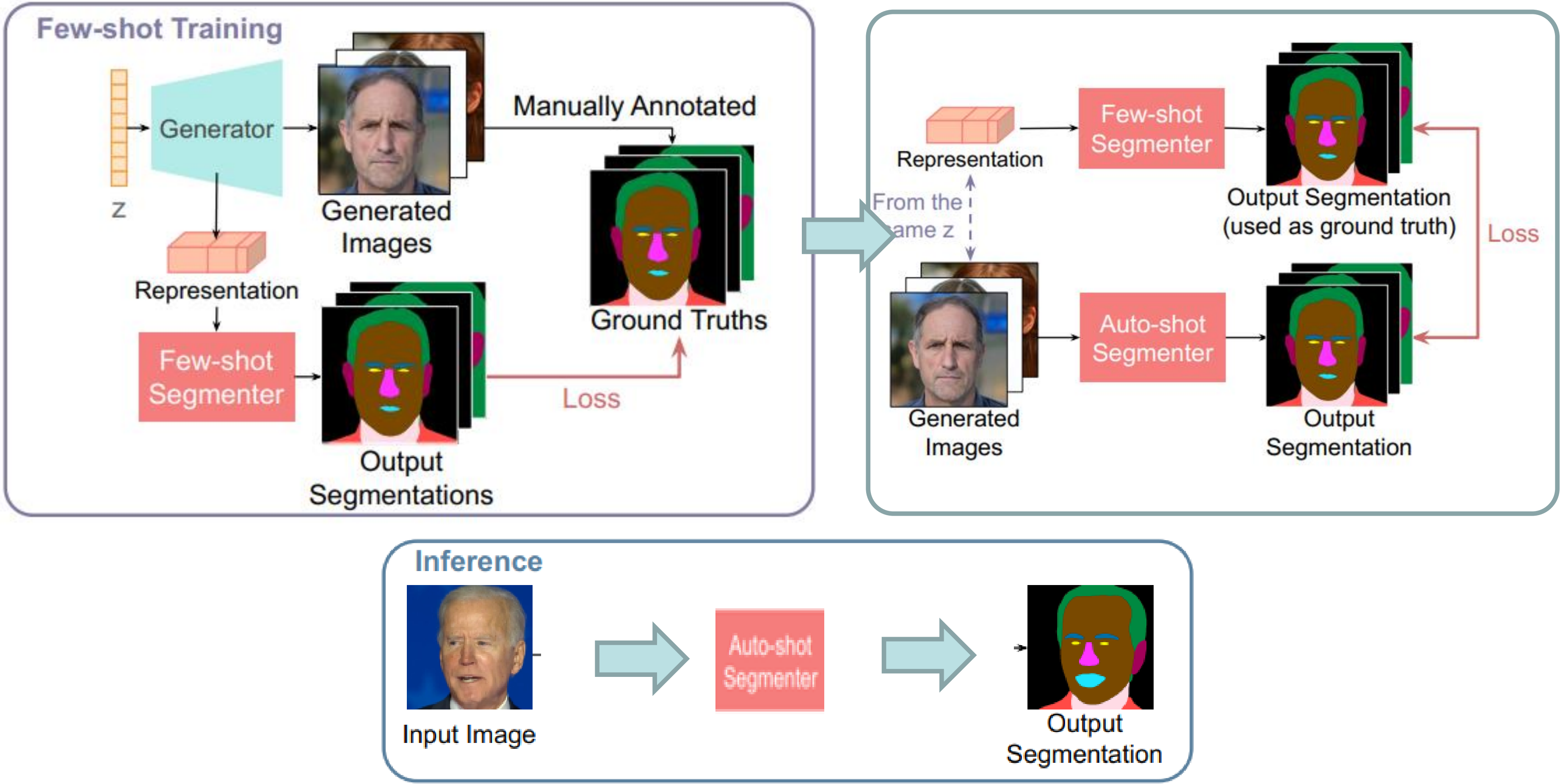
Computing pixel-wise feature vectors using a GAN have a number of restrictions.

First, the test image needs to lie close to the image distribution modeled by the GAN; otherwise, the latent optimization may fail to reproduce the test image, leading to poor feature vectors.

Second, relying on a GAN to generate feature vectors through the latent optimization process is expensive and time-consuming



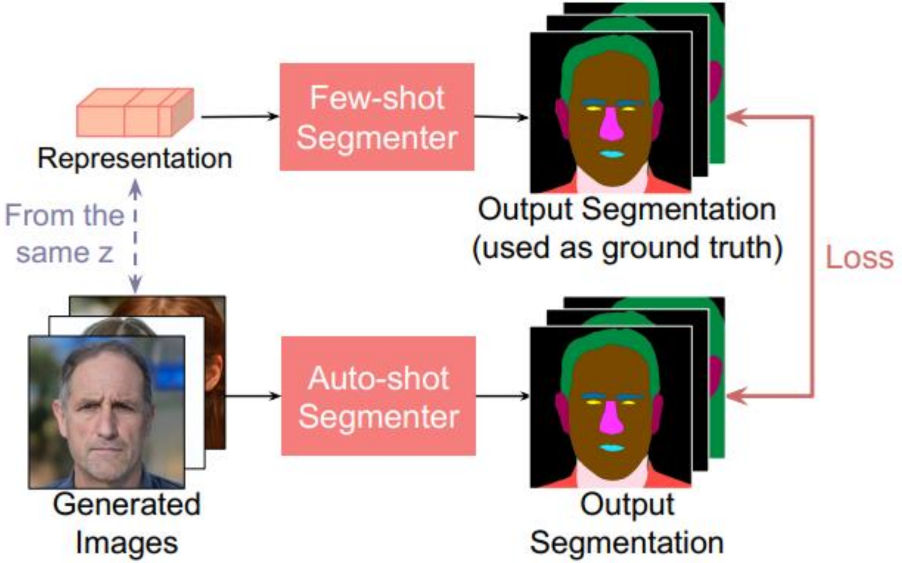
Extension: Auto-shot Segmentation Network



Extension: Auto-shot Segmentation Network

synthesize a large set of images form paired training data.

For the auto-shot segmenter, we use 5,000 images generated from each GAN trained on eachobject class and the predicted annotations from the few-shot segmenter.



Experiments

Table 2: IOU scores of our 10-shot vs auto-shot segmenters on 10-class face segmentation. The auto-shot segmenter is trained with a dataset generated by the 10-shot segmenter. Both techniques have similar performance, which demonstrates the effectiveness of the dataset generation and auto-shot training process.

Network	Weighted IOU	Eyes	Mouth	Nose	Face	Clothes	Hair	Eyebrows	Ears	Neck	BG
10-shot segmenter	85.2	74.0	84.6	82.9	90.0	23.6	79.2	63.1	27.0	73.6	84.2
Auto-shot segmenter	84.5	75.4	86.5	84.6	90.0	15.5	84.0	68.2	37.3	72.8	84.7

Annotation One-shot Results

Segmentation Network	Shots	3-class	10-class
CNN	1	71.7	77.9
	5	82.1	83.9
	10	**83.5**	**85.2**
MLP	1	75.3	74.1
	5	77.8	79.6
	10	77.2	77.2

Experiments

Table 2: IOU scores of our 10-shot vs auto-shot segmenters on 10-class face segmentation. The auto-shot segmenter is trained with a dataset generated by the 10-shot segmenter. Both techniques have similar performance, which demonstrates the effectiveness of the dataset generation and auto-shot training process.

Network	Weighted IOU	Eyes	Mouth	Nose	Face	Clothes	Hair	Eyebrows	Ears	Neck	BG
10-shot segmenter	85.2	74.0	84.6	82.9	90.0	23.6	79.2	63.1	27.0	73.6	84.2
Auto-shot segmenter	84.5	75.4	86.5	84.6	90.0	15.5	84.0	68.2	37.3	72.8	84.7

Annotation One-shot Results

Table 3: Per-class IOU scores on 3-class human face segmentation.

	Weighted IOU	Eyes	Mouth	Nose
1-shot	71.7	57.8	71.1	76.0
5-shot	82.1	73.6	84.0	82.1
10-shot	83.5	75.9	85.3	82.7

Table 4: IOU scores on PASCAL-Parts car segmentation.

Model	Body	Plate	Light	Wheel	Window	BG	Average
CNN[48]	73.4	41.7	42.2	66.3	61	67.4	58.7
CNN+CRF[48]	75.4	35.8	36.1	64.3	61.8	68.7	57
Ours (Auto-shot)	75.5	17.8	29.3	57.2	62.4	70.7	52.2
OMPS[62]	86.3	50.5	55.1	75.5	65.2	-	66.5
Ours (Auto-shot) w/o bg	76.4	17.5	29.3	52.5	64.1	-	47.9

Table 5: IOU scores on PASCAL-Parts horse segmentation. "-" indicates no available result.

Model	Head	Neck	Torso	Neck+Torso	Legs	Tail	BG
Shape+Appearance[53]	47.2	-	-	66.7	38.2	-	-
CNN+CRF[48]	55.0	34.2	52.4	-	46.8	37.2	76.0
Ours (Auto-shot)	50.1	-	-	70.5	49.6	19.9	81.6

Experiments

To train the few-shot segmenter, we useface images and annotated segmentation masks from CelebAMask-HQ

For horse and car, we use images generated by pretrained StyleGAN2s and manually annotate them ourselves.