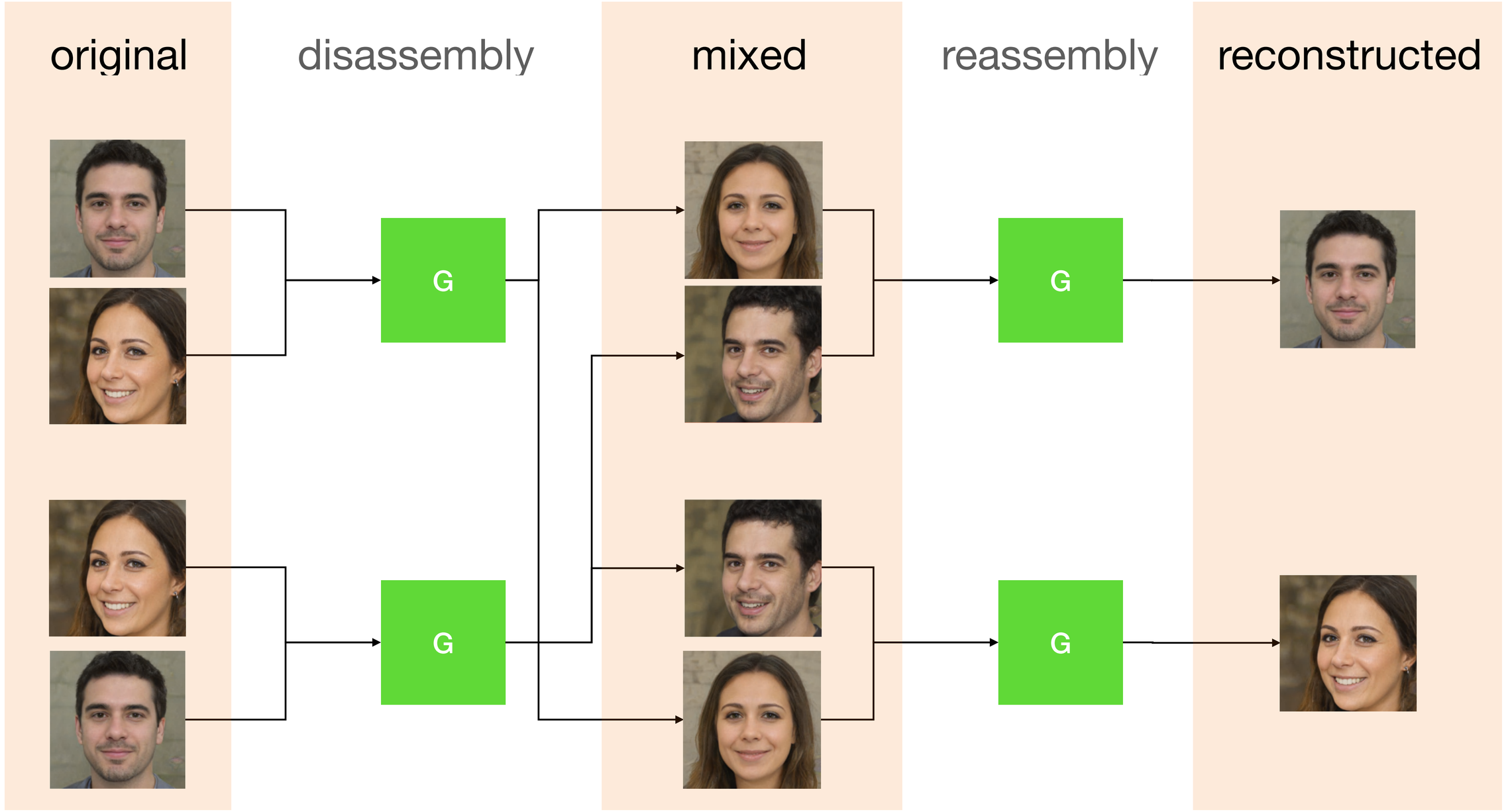
G2G — Current Progress

Disentanglement by Cross Training

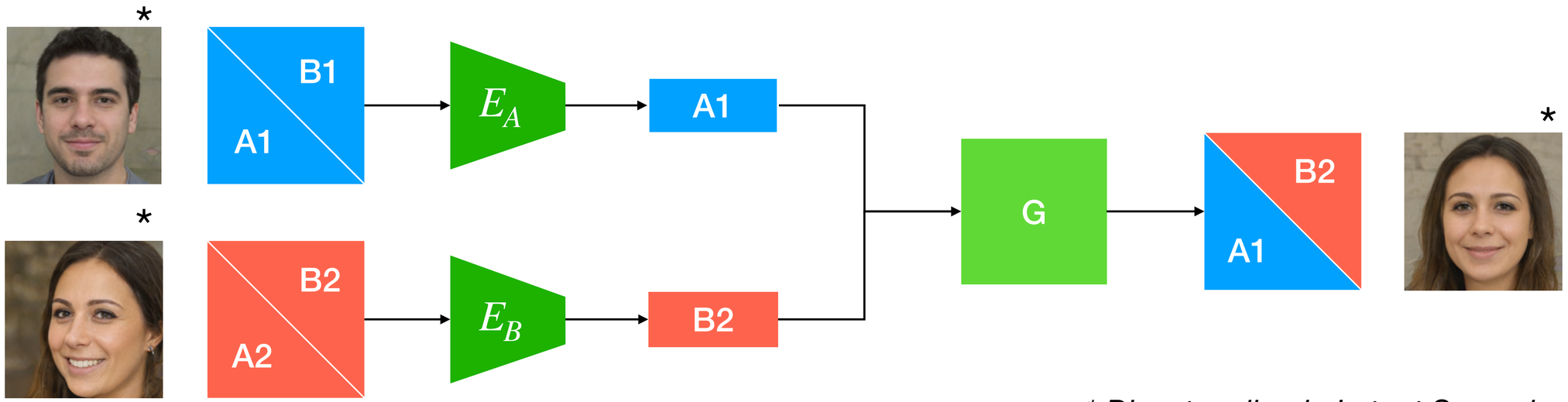
Alexander Koenig, Li Nguyen, Ron Mokady, Prof Daniel Cohen-Or



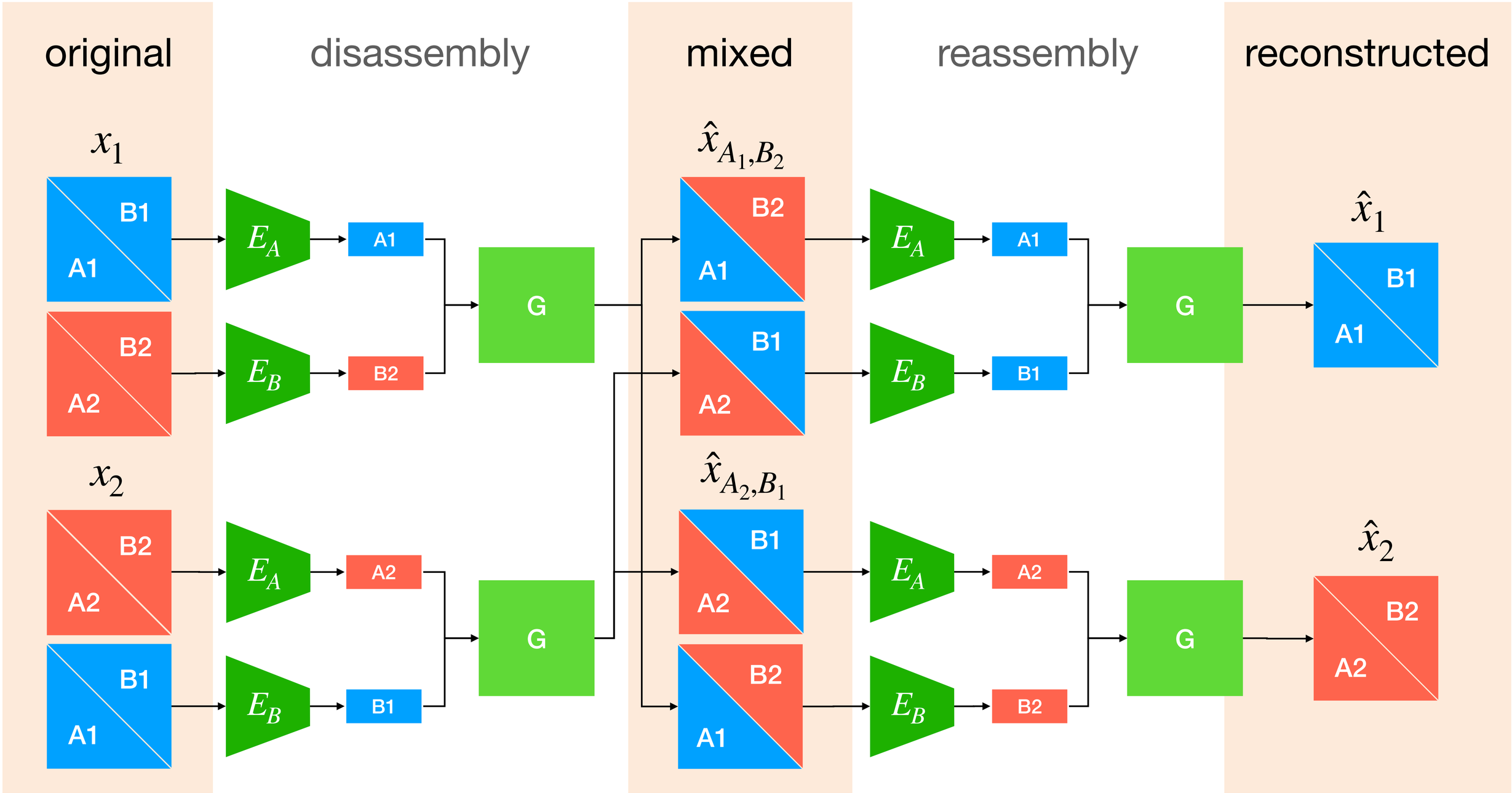
All Images: Disentangling in Latent Space by Harnessing a Pretrained Generator, 2020, Yotam Nitzan et al.

Adding Encoders

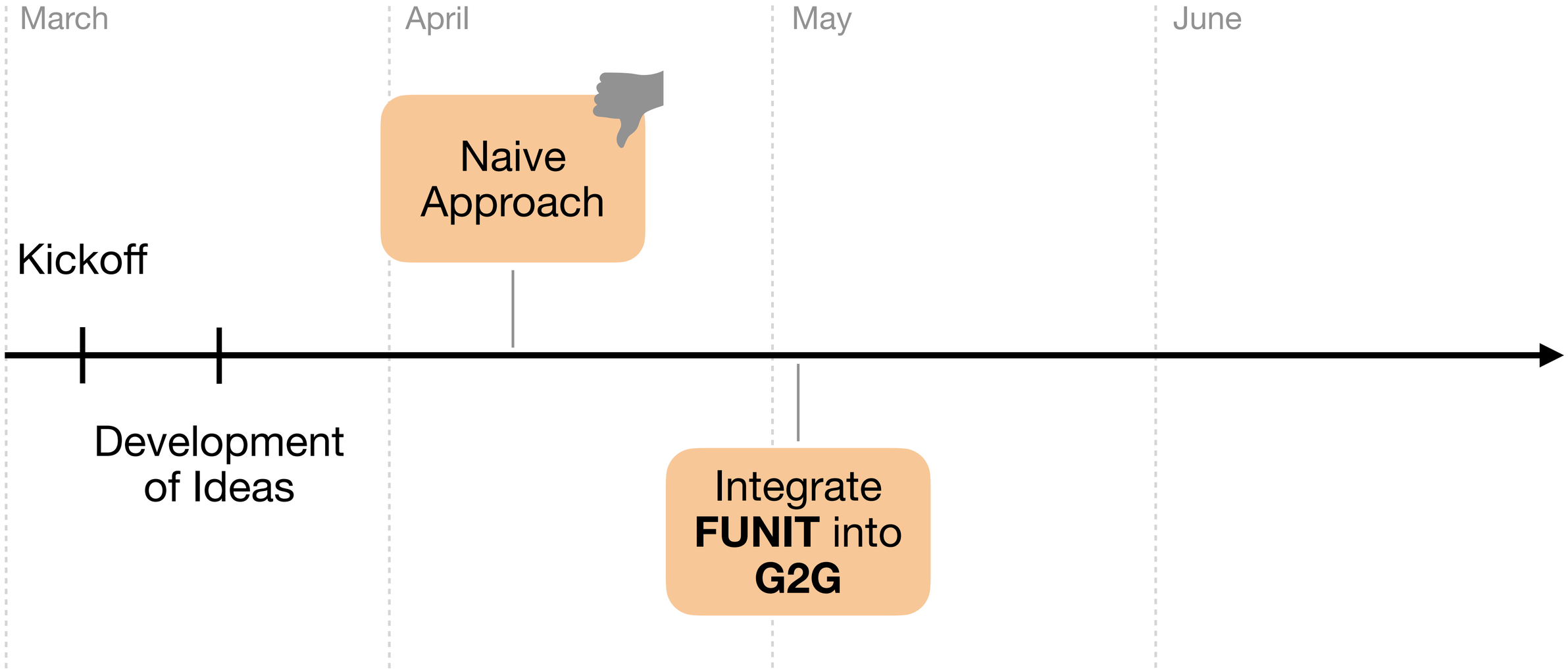
- Let **Encoder** E_A encode **rest**, e.g. pose, facial expression, background, lighting, …
- Let **Encoder** E_B encode a specific feature: **identity**
- The encoded features A and B should be **disentangled**
- **Generator G** can focus on learning to stitch the encoded features together to a new picture



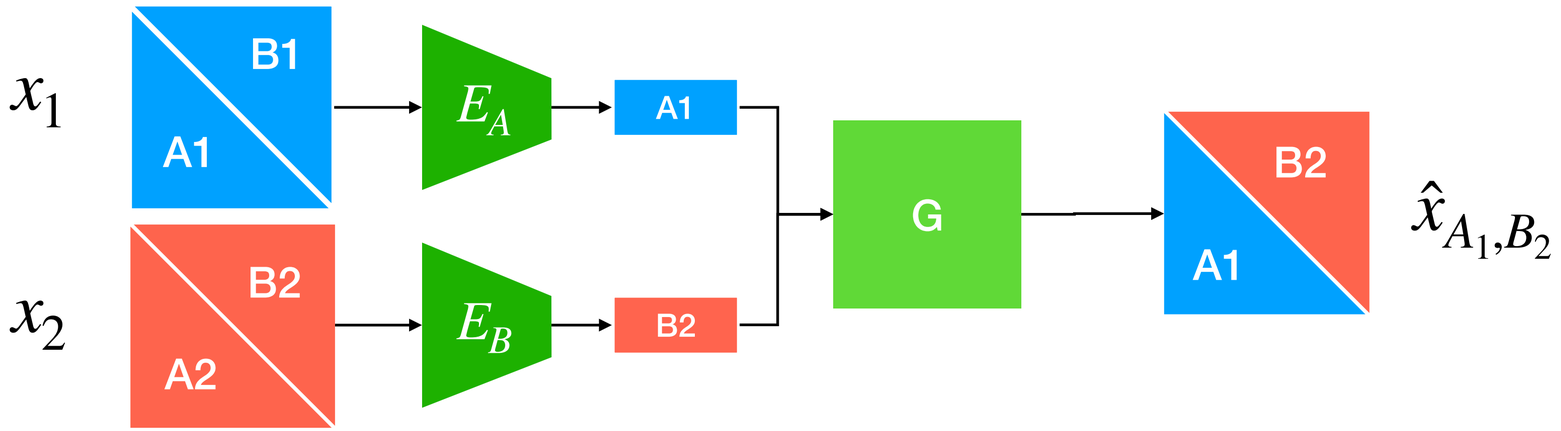
** Disentangling in Latent Space by Harnessing a Pretrained Generator, 2020, Yotam Nitzan et al.*


original
disassembly
mixed
reassembly
reconstructed
x_1
B1
A1
B2
A2
x_2
B2
A2
B1
A1
E_A
E_B
A1
B2
A2
B1
G
$\hat{x}_{A_1,B_2}$
$\hat{x}_{A_2,B_1}$
$\hat{x}_1$
$\hat{x}_2$

Timeline


March
April
May
June
Naive Approach
Kickoff
Development of Ideas
Integrate **FUNIT** into **G2G**

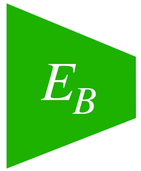
Using FUNIT Encoders and Decoder



FUNIT Content Encoder

Global Style of Image

FUNIT Decoder

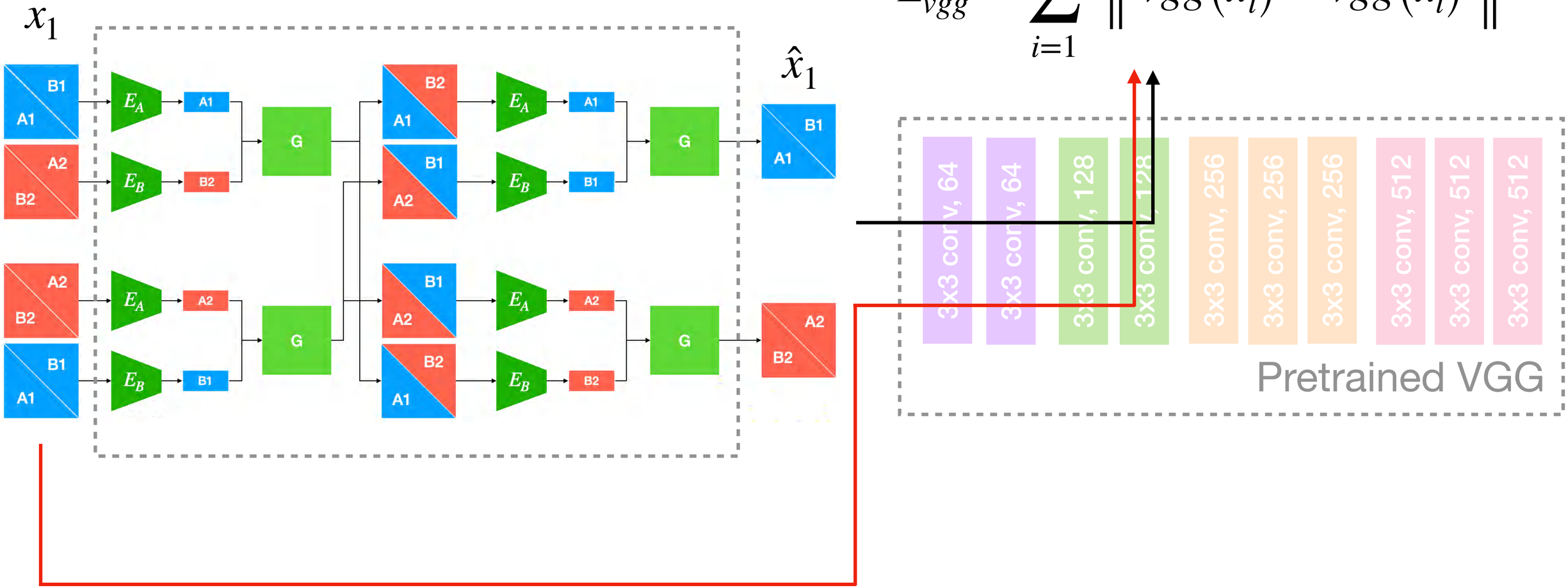


FUNIT Class Encoder

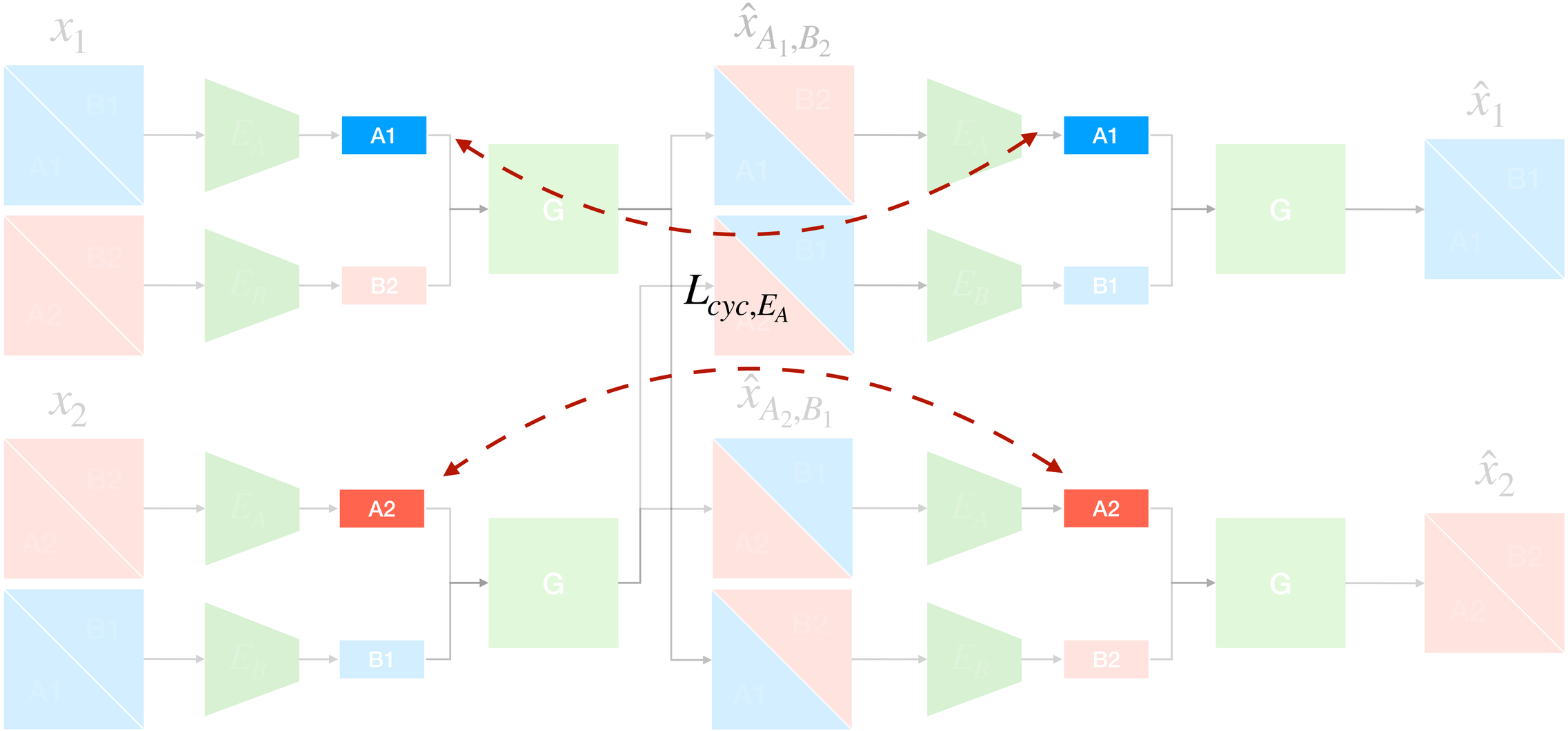
Identity of Image

Perceptual Loss

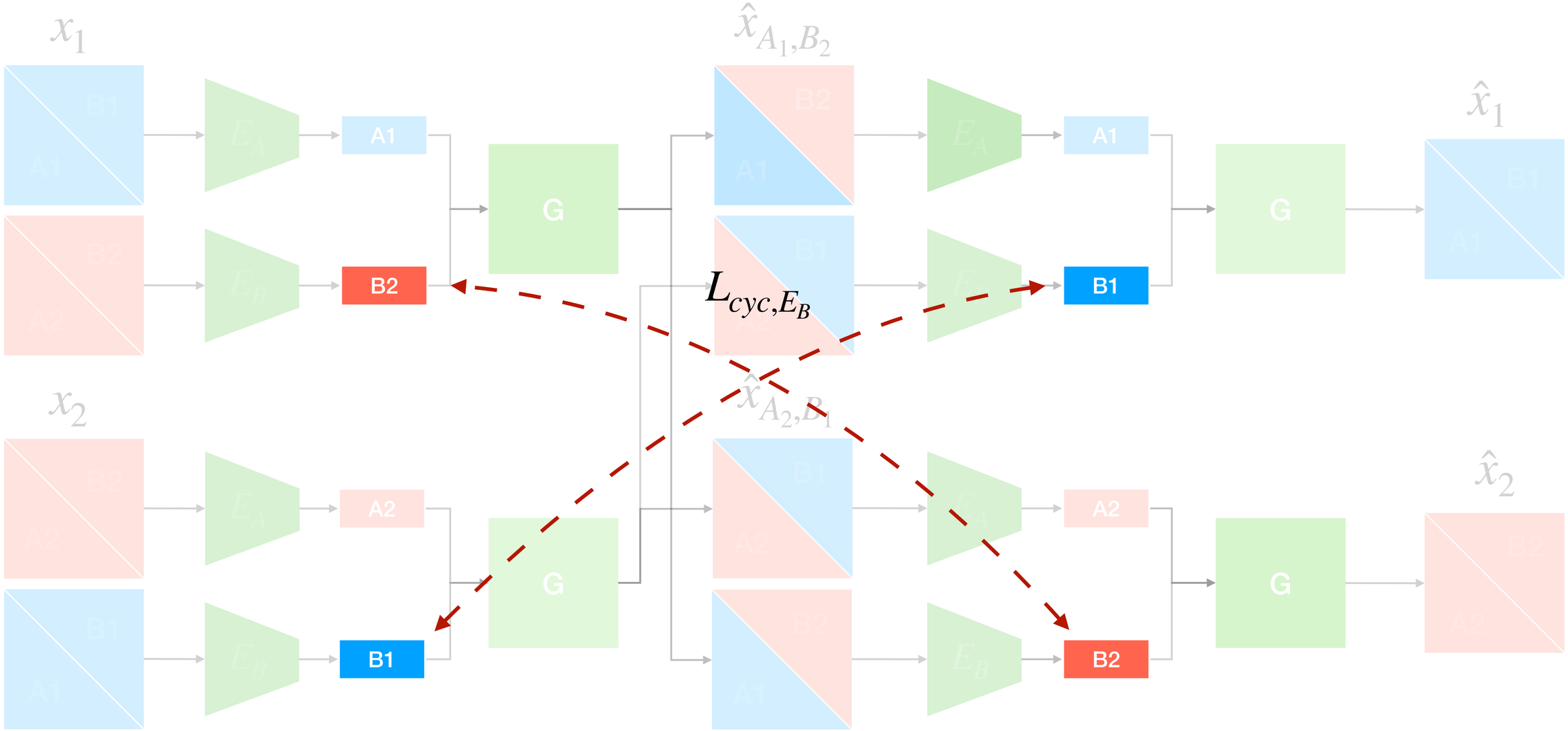
$$L_{vgg} = \sum_{i=1}^{n} \left\| vgg(\hat{x}_i) - vgg(x_i) \right\|^2$$



Cycle Consistency Loss A



Cycle Consistency Loss B



Overall Loss

$$L_{gen} = \alpha \cdot L_{vgg} + \gamma \cdot L_{cyc}$$

with $\alpha = 1.0$

= VGG Perceptual Loss

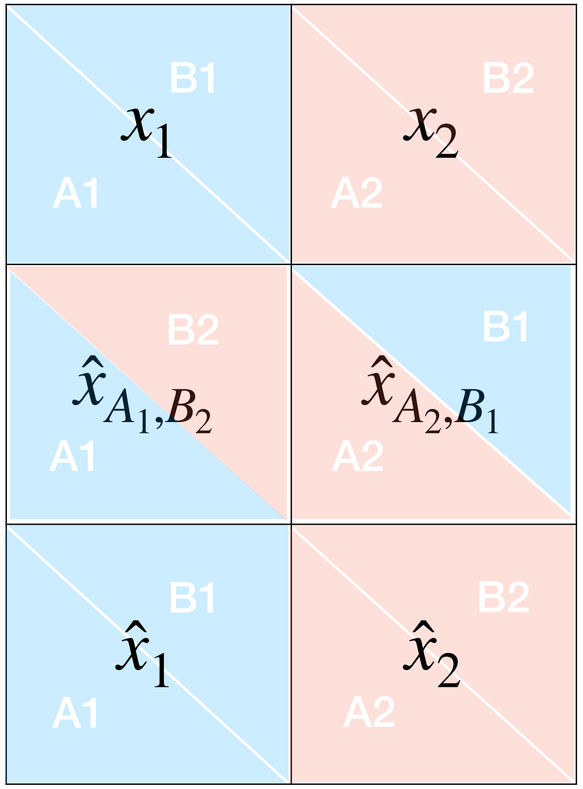
= Cycle Consistency Losses

Plot Layout

Original

Mixed

Reconstructed



*Goal **

* *Disentangling in Latent Space by Harnessing a Pretrained Generator, 2020, Yotam Nitzan et al.*

Results

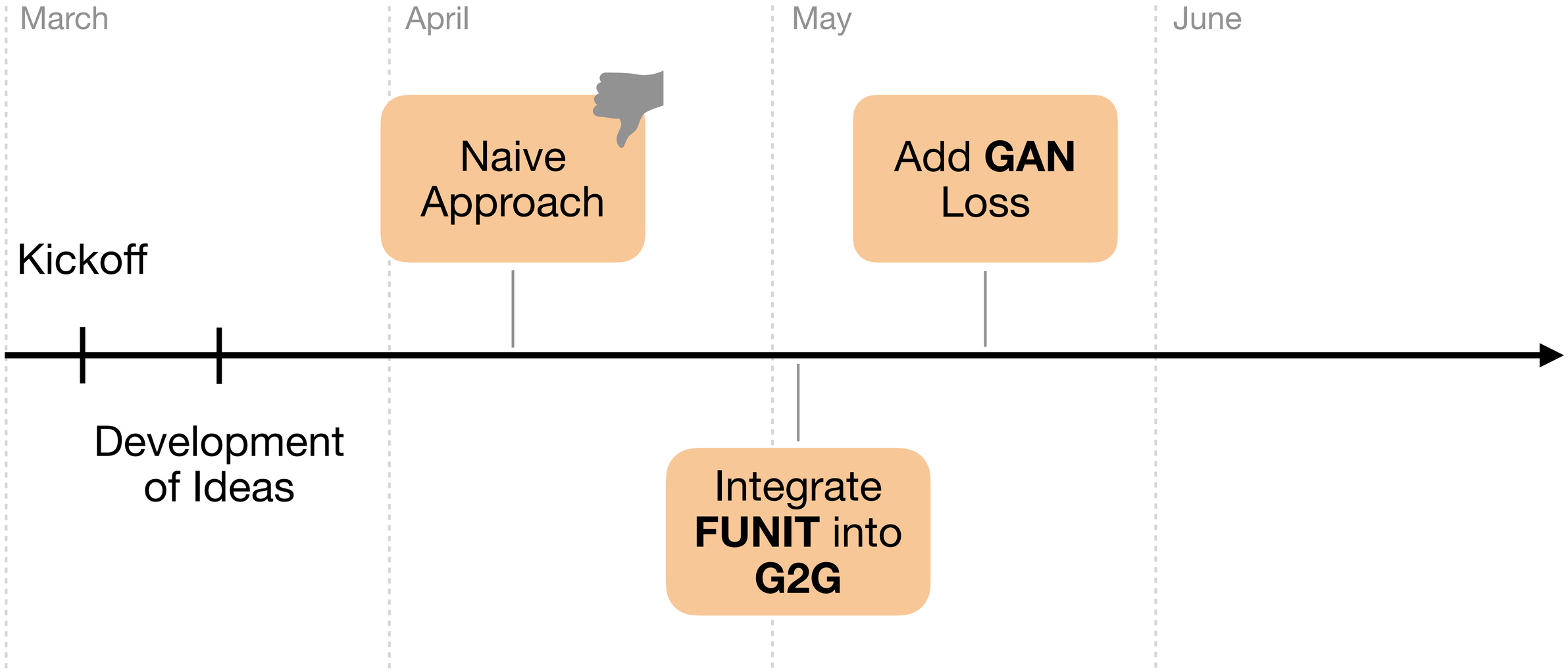
Original

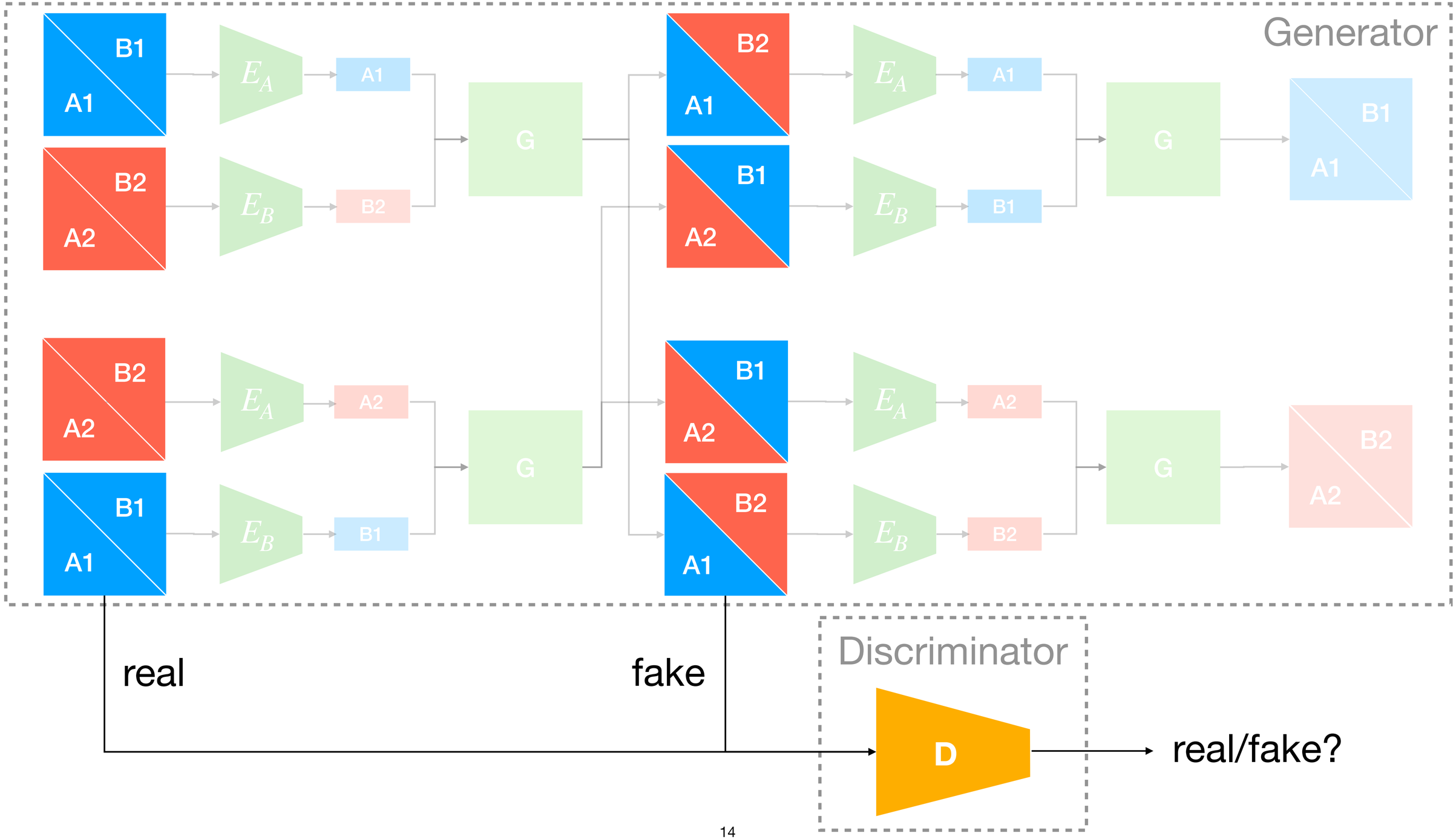
Mixed

Reconstructed

Epochs: 9, Batch Size: 16, $\gamma = 1.3$

Timeline


March
April
May
June
Kickoff
Development of Ideas
Naive Approach
Integrate **FUNIT** into **G2G**
Add **GAN** Loss


Generator
B1
A1
B2
A2
E_A
E_B
A1
B2
G
B2
A1
B1
A2
E_A
E_B
A1
B1
G
B1
A1
B2
A2
E_A
E_B
A2
B1
G
B2
A2
B1
A1
E_A
E_B
A2
B2
real
fake
Discriminator
D
real/fake?

Overall Losses

Generator

$$L_{gen} = \alpha \cdot L_{vgg} + \gamma \cdot L_{cyc} + \lambda_g \cdot L_{adv} \qquad \text{with } \alpha = 1.0$$

= VGG Perceptual Loss

= Generator GAN Loss

= Cycle Consistency Losses

Discriminator

$$L_{dis} = \zeta \cdot \left(\frac{L_{real} + L_{fake}}{2}\right) \qquad \text{with } \zeta = 0.2$$

Results

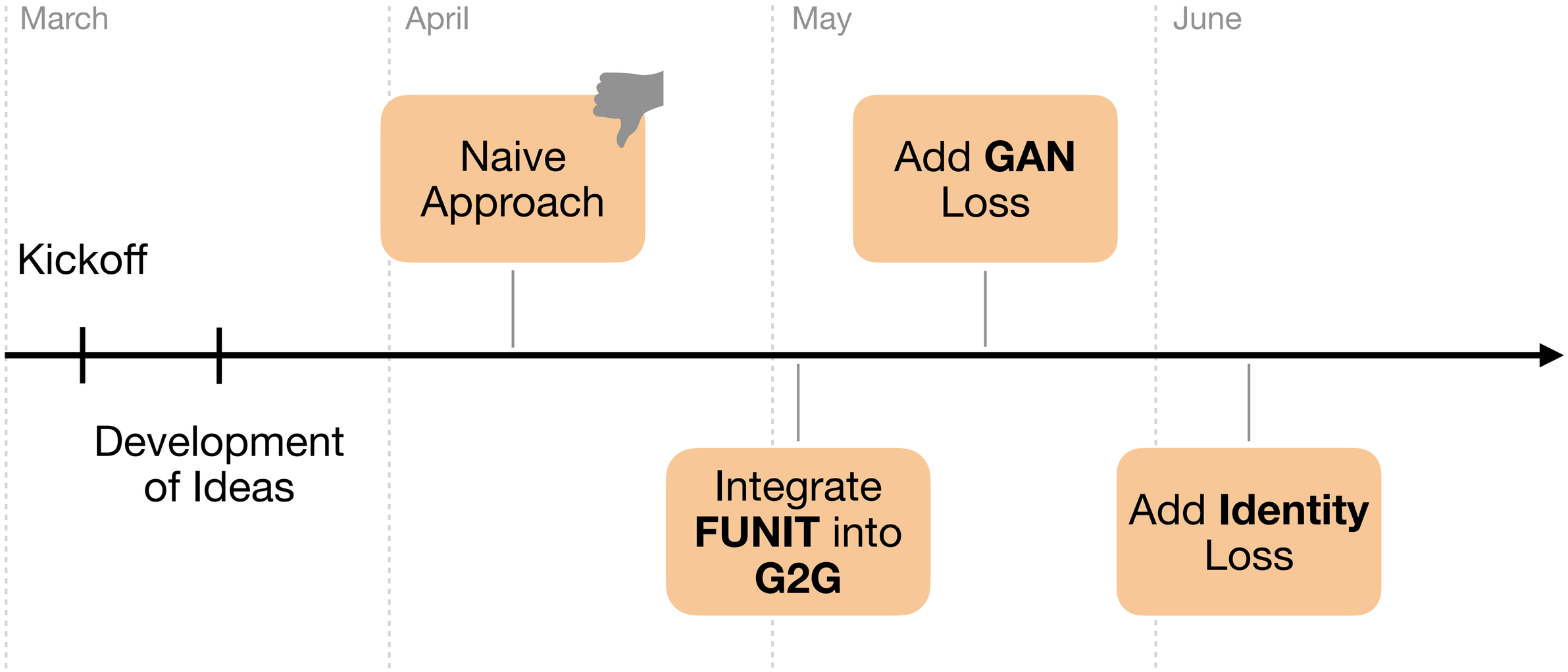
Original

Mixed

Reconstructed

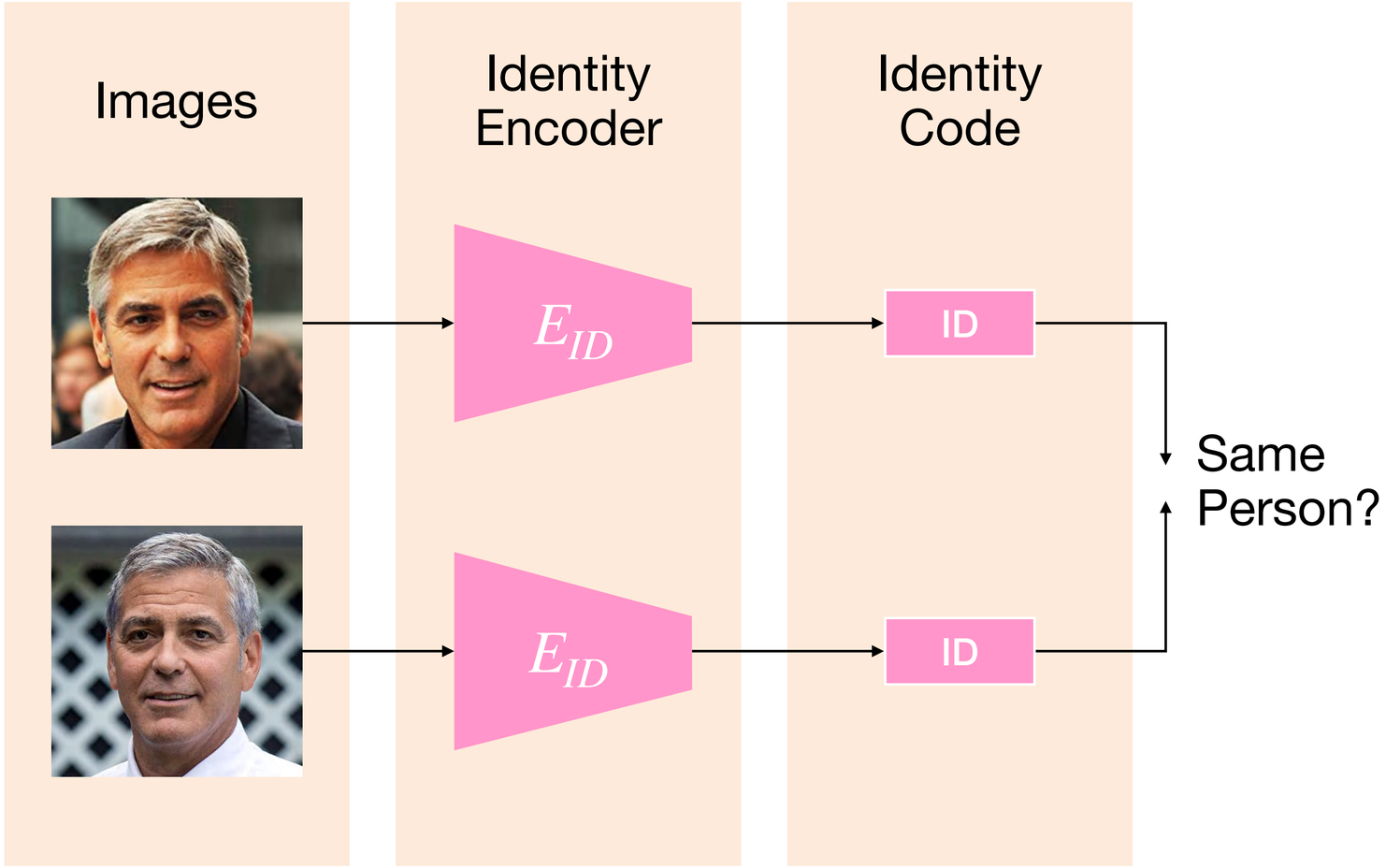
Epoch: 124, Batch Size: 4, $\gamma = 10.0$, $\lambda_g = 1.0$

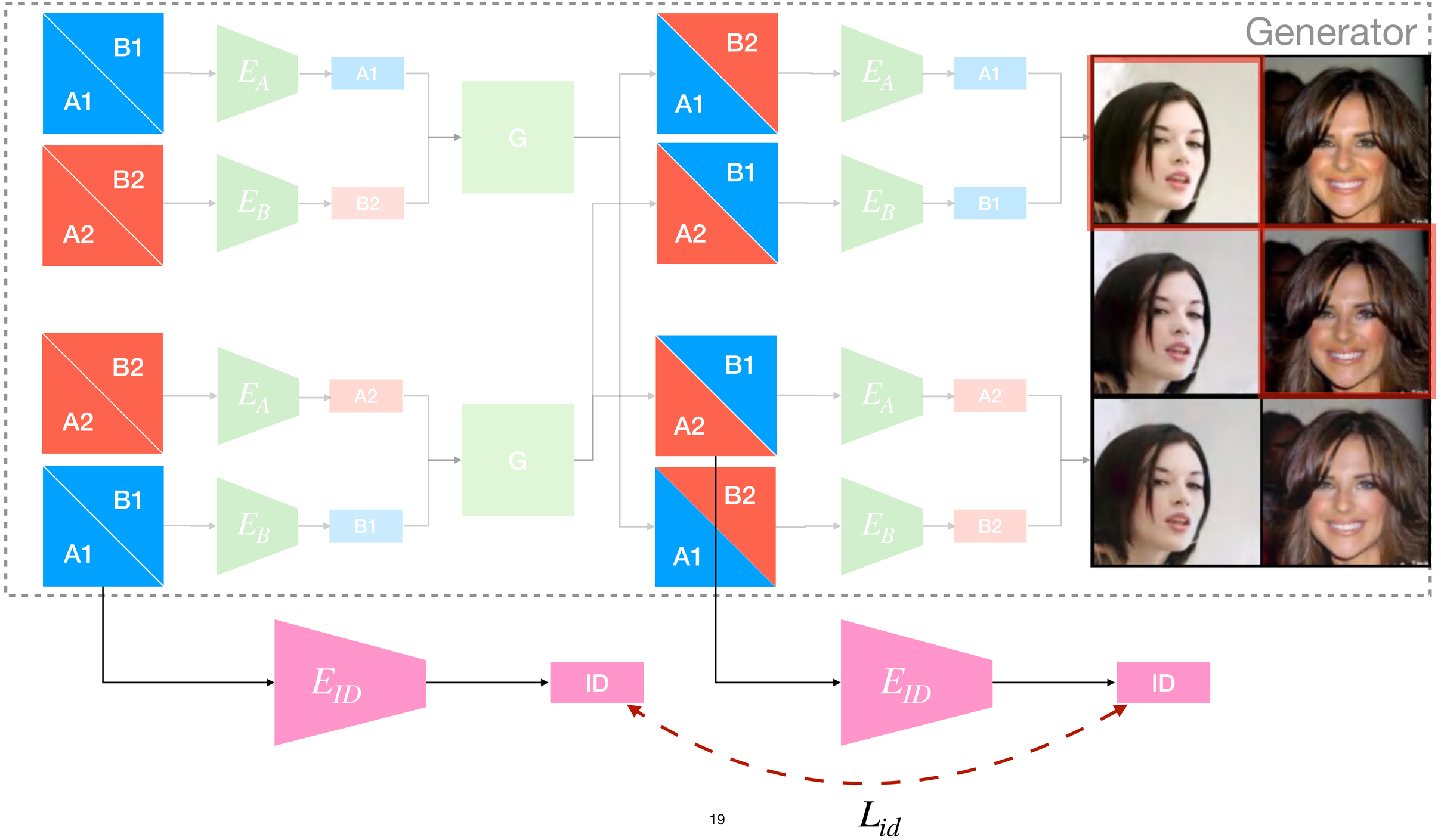
Timeline


March
April
May
June
Kickoff
Development of Ideas
Naive Approach
Integrate **FUNIT** into **G2G**
Add **GAN** Loss
Add **Identity** Loss

Identity Loss

- Use pretrained **Face Detector** as an identity encoder
- Minimise distance between produced representation




Generator
B1
A1
B2
A2
E_A
A1
E_B
B2
G
B2
A1
B1
A2
E_A
A1
E_B
B1
B2
A2
E_A
A2
B1
A1
E_B
B1
G
B1
A2
B2
A1
E_A
A2
E_B
B2
E_{ID}
ID
E_{ID}
ID
L_{id}

Overall Losses

Generator

$$L_{gen} = \alpha \cdot L_{vgg} + \gamma \cdot L_{cyc} + \lambda_g \cdot L_{gen} + \delta \cdot L_{id}$$

= VGG Perceptual Loss

= Cycle Consistency Losses

= Generator GAN Loss

= Identity Loss

Discriminator

$$L_{dis} = \zeta \cdot \left(\frac{L_{real} + L_{fake}}{2}\right) \qquad \text{with } \zeta = 0.2$$

Results

Original

Mixed

Reconstructed

Epochs: 0-8, Batch Size: 8, $\lambda_g = 1.0, \gamma = 10.0, \delta = 1.0$

Next Steps

- Increase weight of L_{ID} successively
- Quantify usefulness of G2G reassembly stage
- Idea: Erase facial landmarks (identity specific information) of image which represents *rest*

*

** Disentangling in Latent Space by Harnessing a Pretrained Generator, 2020, Yotam Nitzan et al.*

Any Questions?