Unpaired Visual Domain Translation using Advanced - GAN

IIT2015032 Rohan MR

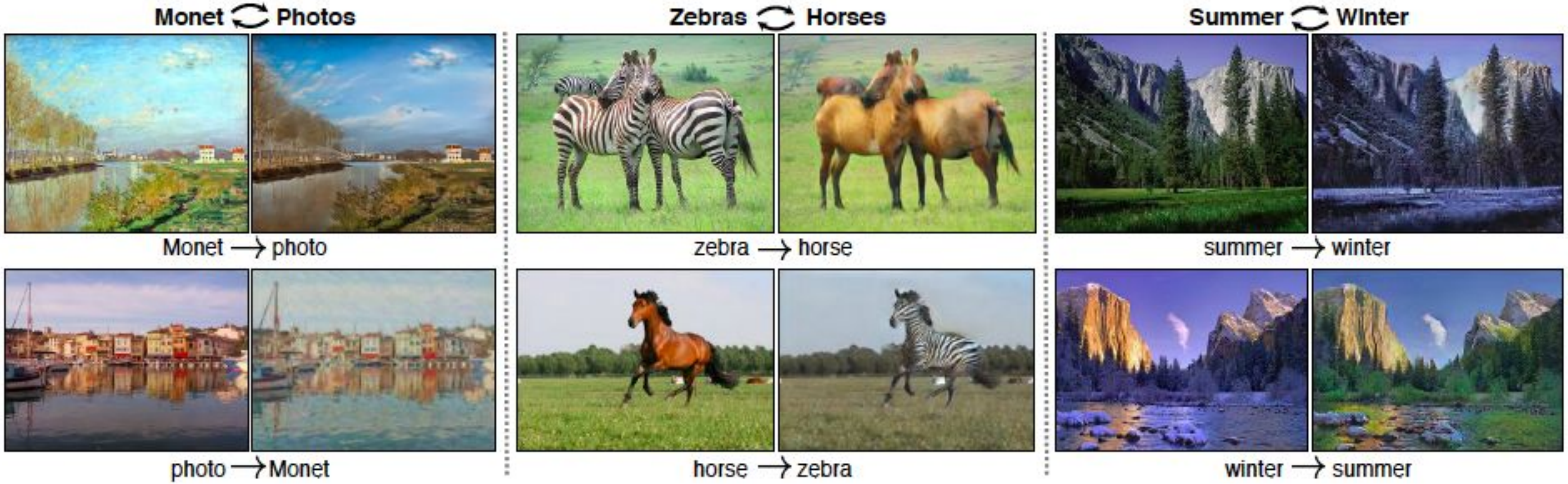
IIT2015042 Raghav Saboo

IIT2015044 Jayesh Chaudhari

Under supervision of -
Dr K.P.Singh

Visual Domain Translation

- Image-to-Image translation is a class of vision and graphics problems where the goal is to learn the mapping between an input image and an output image using a training set of aligned image pairs.
- However, for many tasks, training data will not be available.

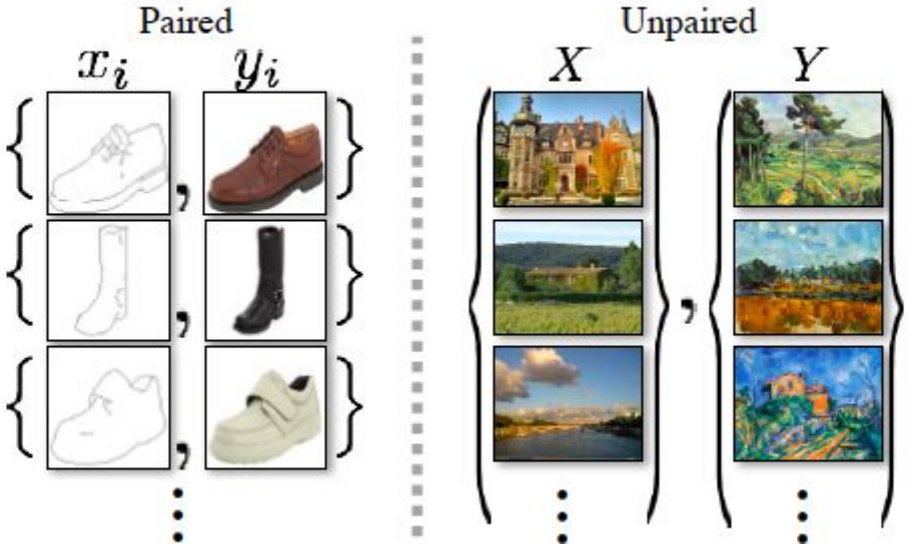


Problem Statement -

To translate images and videos from one domain to its related another domain by training GAN using unpaired image and video datasets.

We present an approach for learning to translate an image from a source domain X to a target domain Y in the absence of paired examples.Our goal is to learn a mapping $G : X \rightarrow Y$ such that the distribution of images from $G(X)$ is indistinguishable from the distribution Y using an adversarial loss. Because this mapping is highly under-constrained, we couple it with an inverse mapping $F:Y \rightarrow X$ and introduce a cycle consistency loss to enforce $F(G(X)) \approx X$(and vice versa).

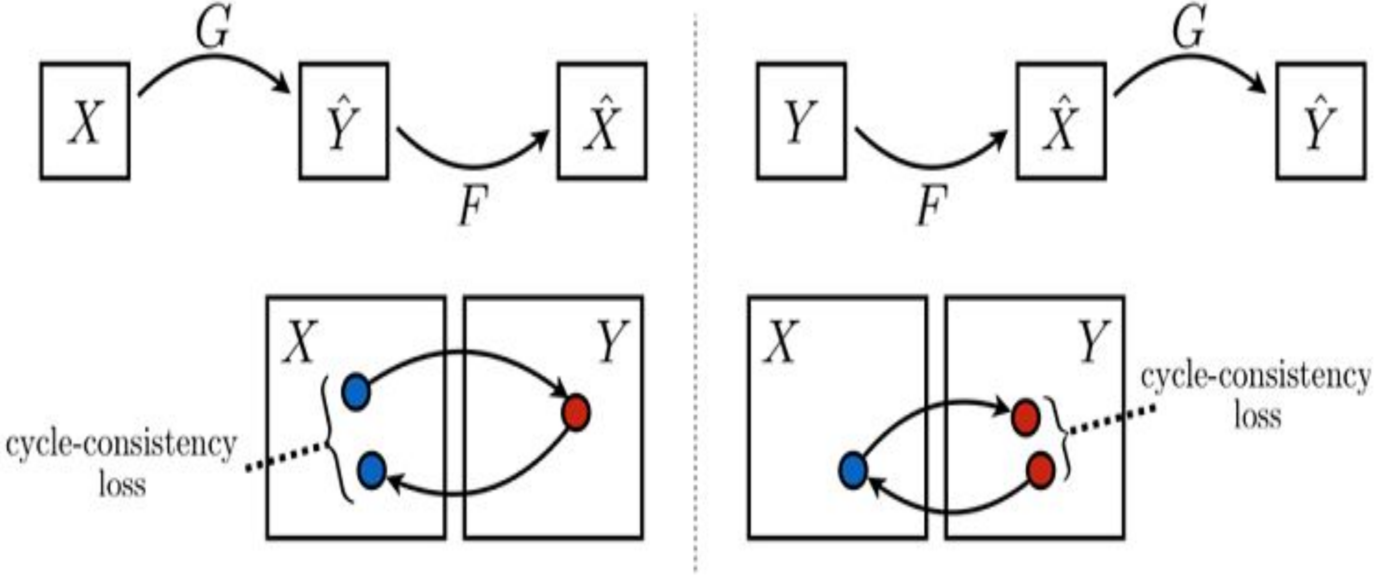
Paired vs Unpaired Image translation

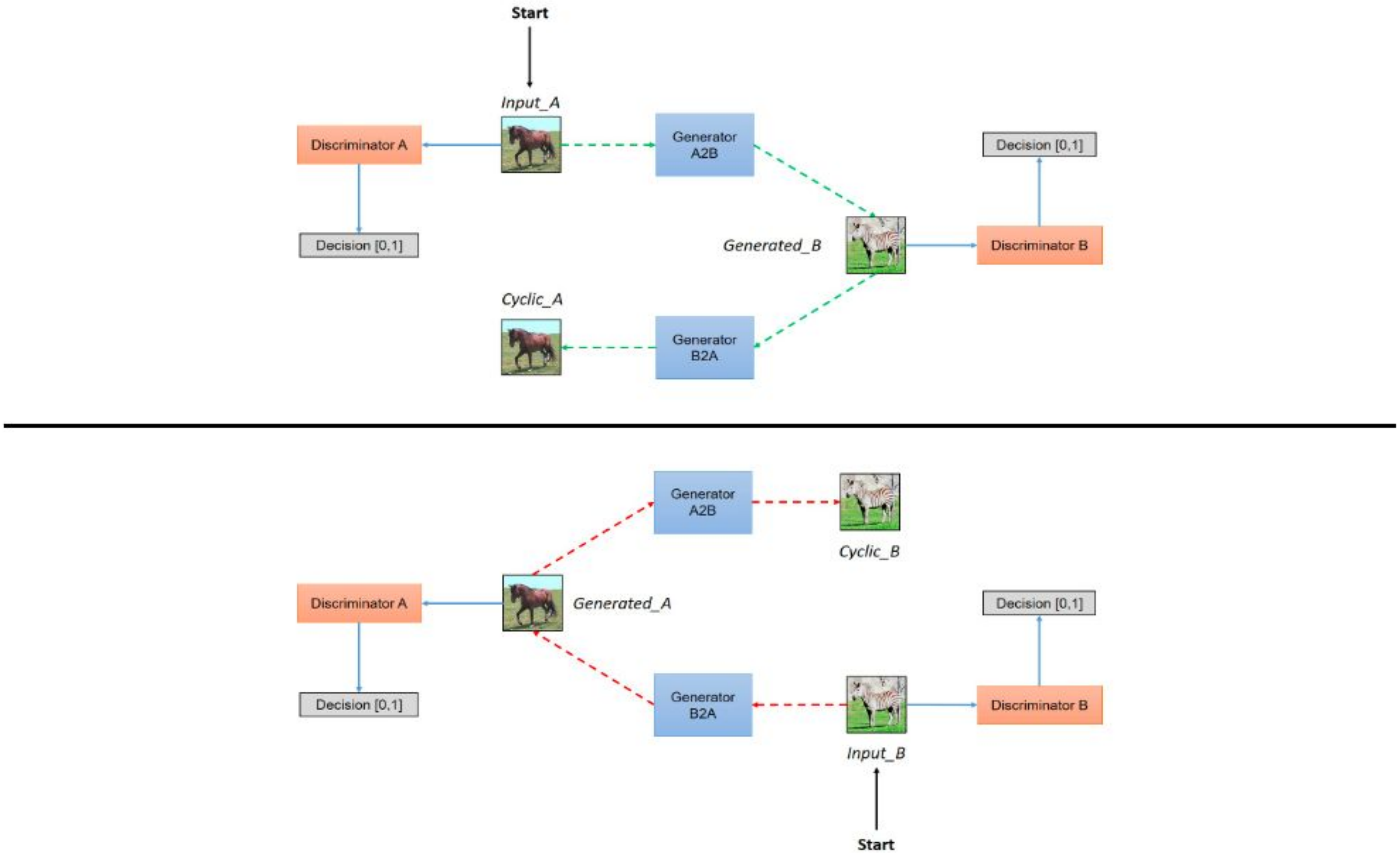


Modified GAN Architecture

$G : X \rightarrow Y$

$F : Y \rightarrow X$





Start
Input_A
Discriminator A
Decision [0,1]
Generator A2B
Generated_B
Discriminator B
Decision [0,1]
Cyclic_A
Generator B2A
Generator A2B
Cyclic_B
Discriminator A
Generated_A
Decision [0,1]
Decision [0,1]
Generator B2A
Discriminator B
Input_B
Start

Objective function for Modified GAN

The full objective function is -

$$\mathcal{L}(G, X, D_{\mathrm{X}}, D_{\mathrm{Y}}) = \mathcal{L}_{\mathrm{GAN}}(G, D_{\mathrm{Y}}, X, Y) + \mathcal{L}_{\mathrm{GAN}}(F, D_{\mathrm{X}}, Y, X) + \lambda\mathcal{L}_{\mathrm{cyc}}(G, F) \tag{5.1}$$

$$\mathcal{L}_{\mathrm{cyc}}(G, F) = E_{x\backsim p_{data}(X)}[\| F(G(x)) - x \|_1] + E_{y\backsim p_{data}(Y)}[\| G(F(y)) - y \|_1] \tag{5.2}$$

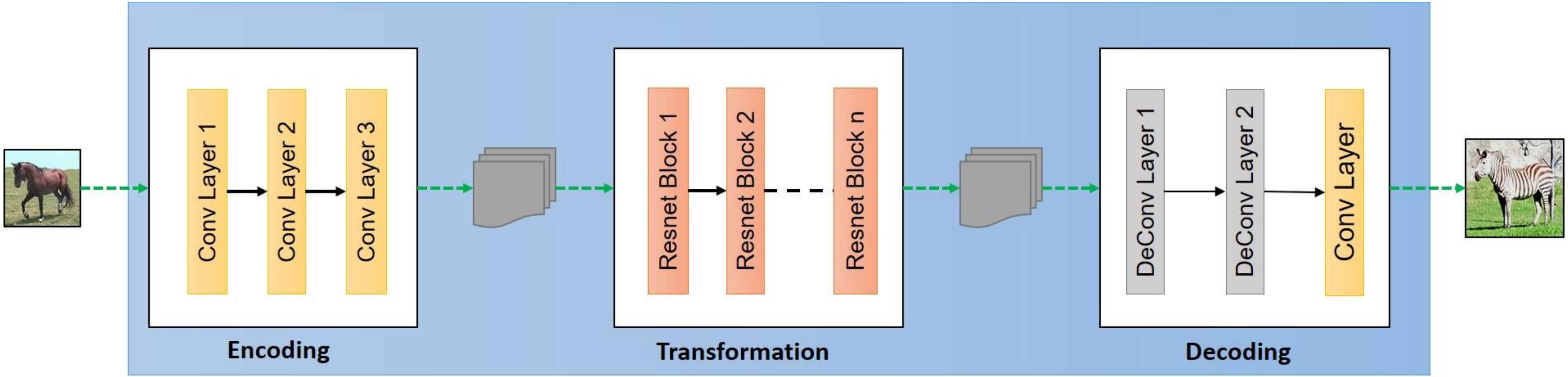
$$\mathcal{L}_{GAN}(G, D_{\mathrm{Y}}, X, Y) = E_{y\backsim p_{data}(Y)}[log D_Y(y)] + E_{x\backsim p_{data}(x)}[log(1 - D_Y G(x))] \tag{5.3}$$

$$G^{*} F^{*} = \arg\min_{G,F}\max_{D_X, D_Y} \mathcal{L}(G, F, D_X, D_Y)$$

Generator Model -

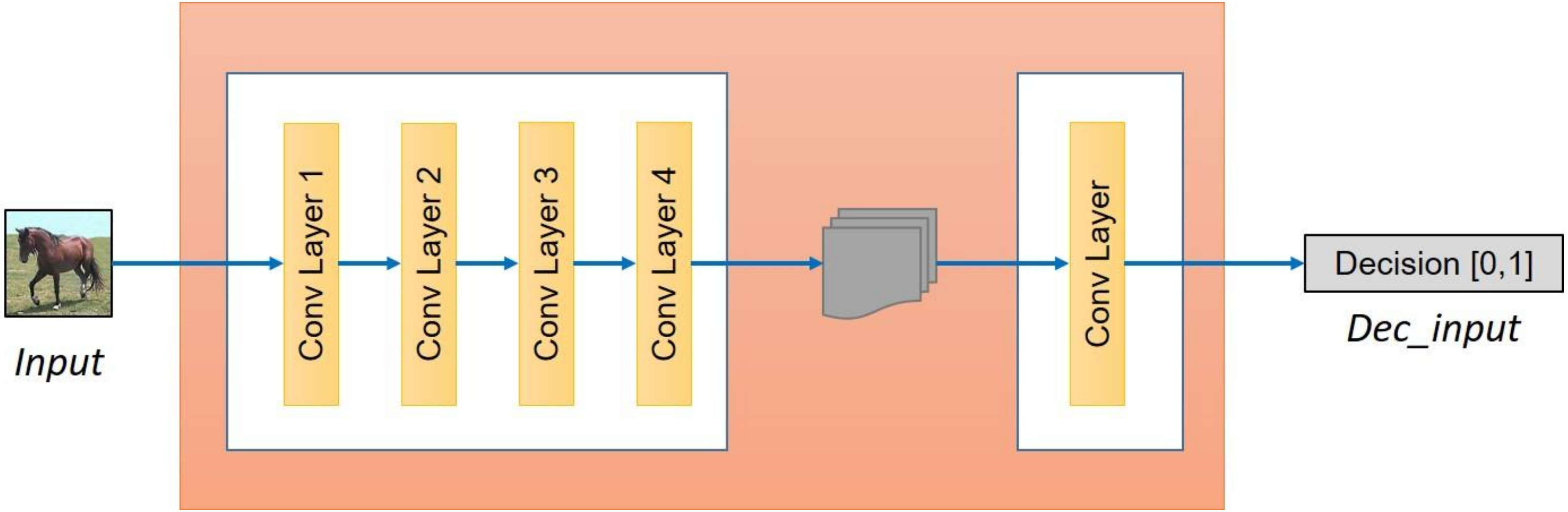
The generator has three components:

1. Encoder
2. Transformer
3. Decoder

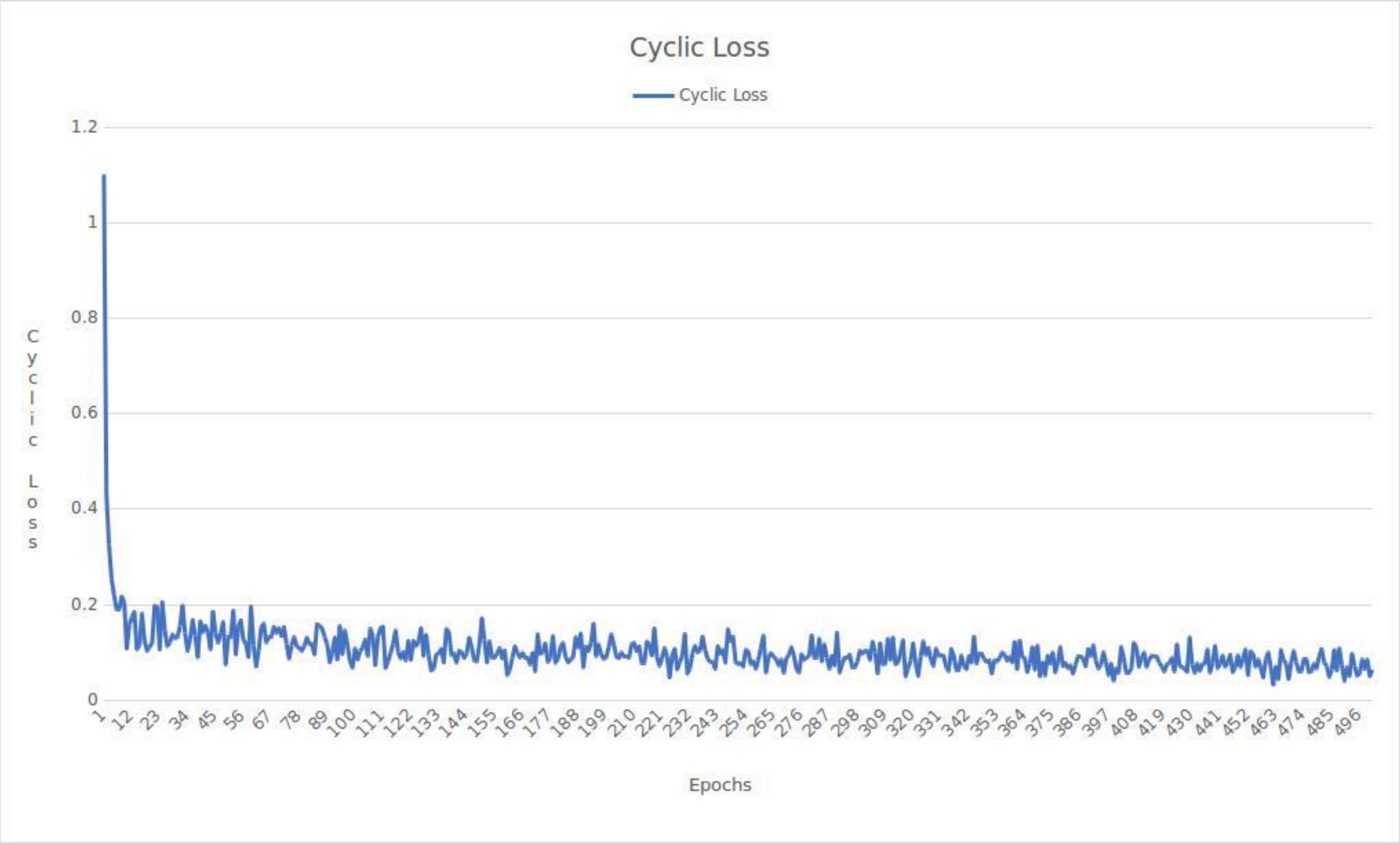


Discriminator Model -

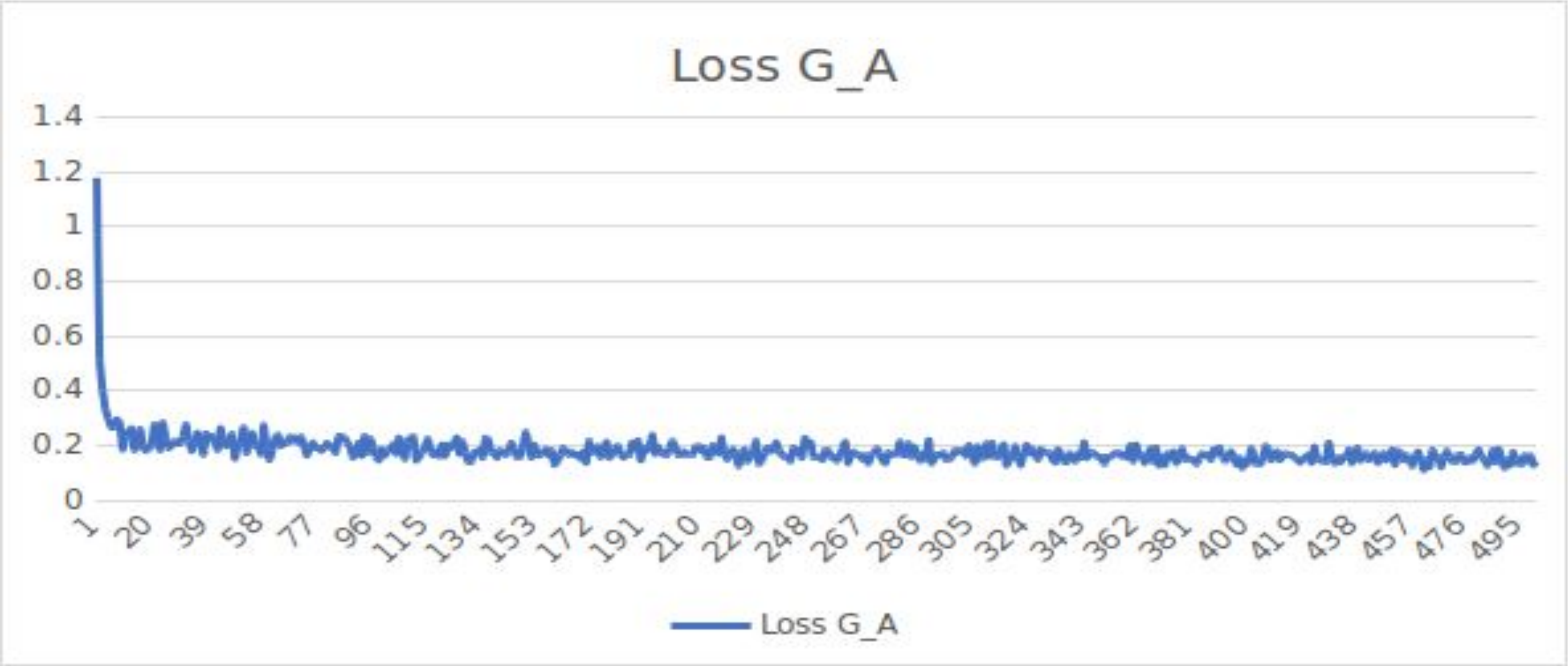
The discriminator would take an image as an input and try to predict if it is an original or the output from the generator.

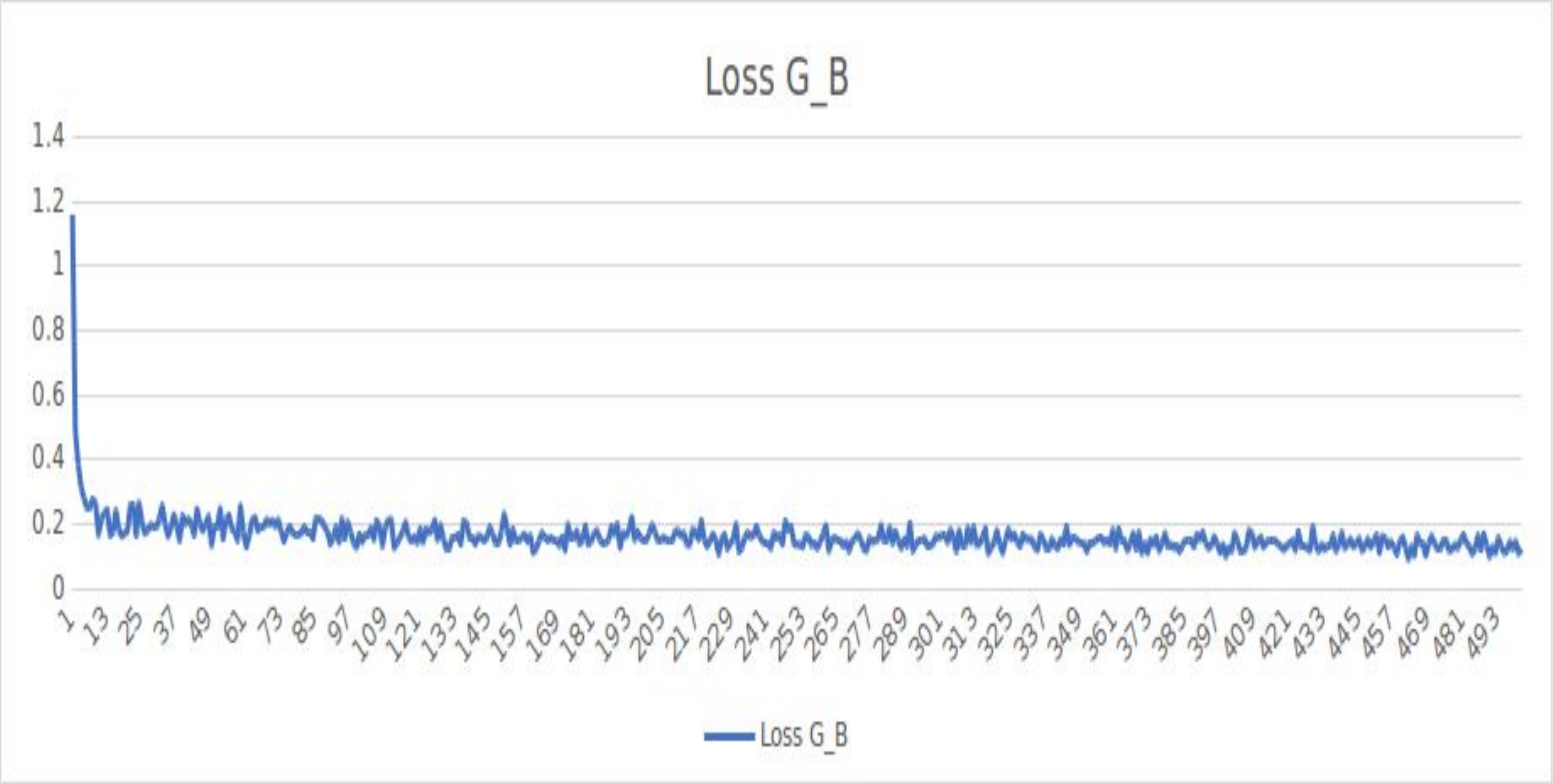


Results

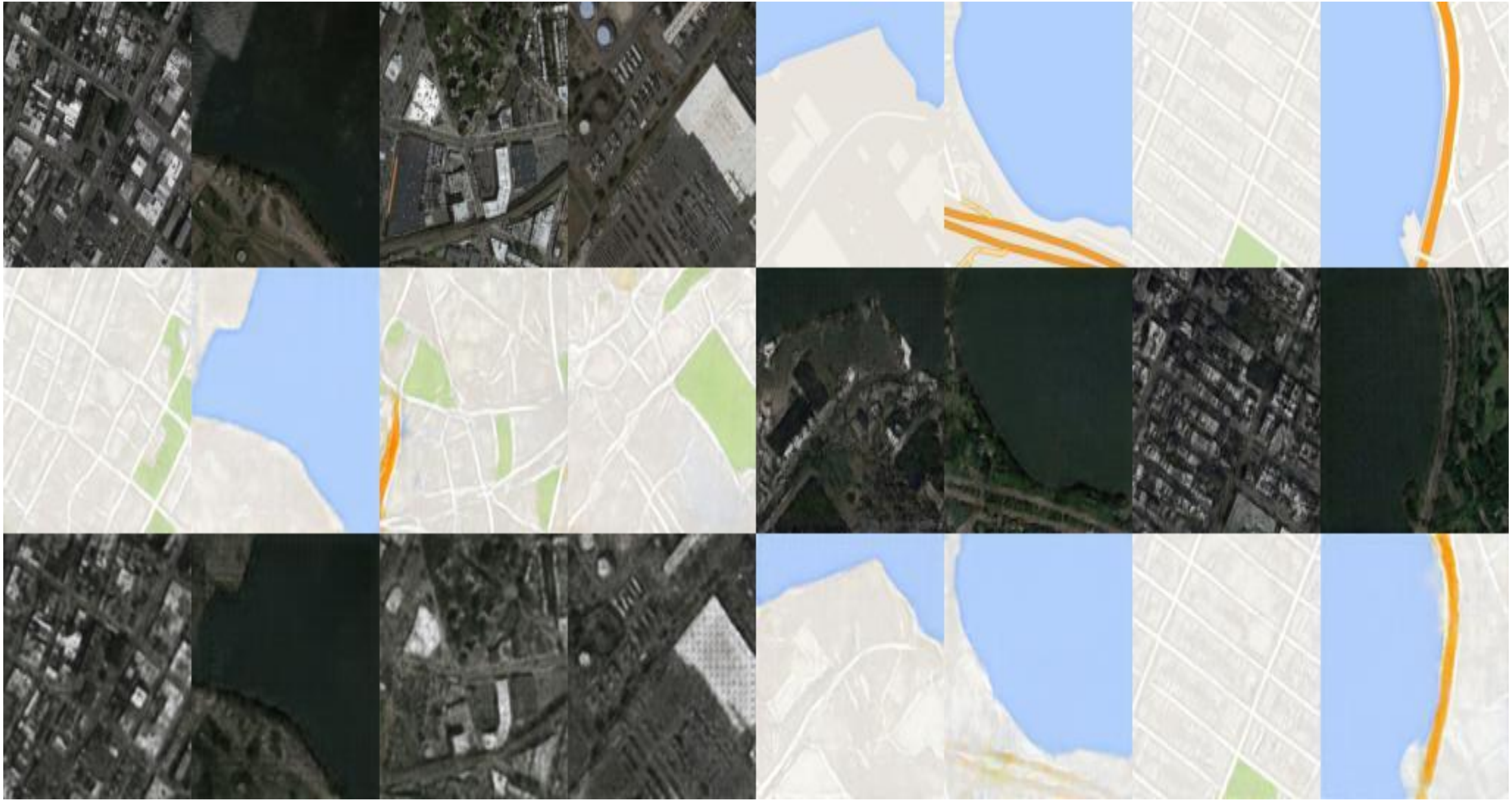


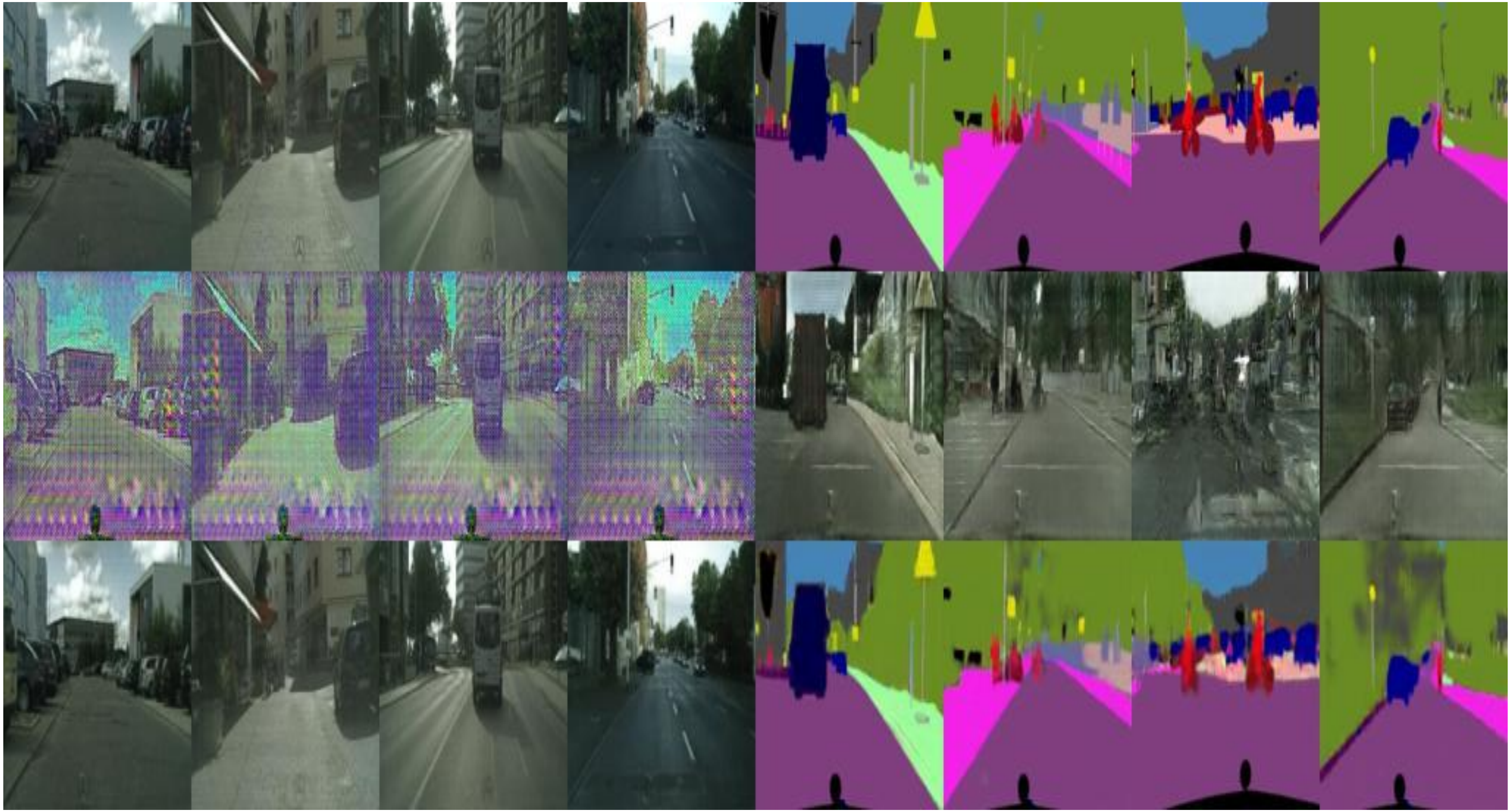
Cyclic Loss
Cyclic Loss
1.2
1
0.8
0.6
0.4
0.2
0
Cyclic Loss
1
12
23
34
45
56
67
78
89
100
111
122
133
144
155
166
177
188
199
210
221
232
243
254
265
276
287
298
309
320
331
342
353
364
375
386
397
408
419
430
441
452
463
474
485
496
Epochs


Loss G_A
1.4
1.2
1
0.8
0.6
0.4
0.2
0
1
20
39
58
77
96
115
134
153
172
191
210
229
248
267
286
305
324
343
362
381
400
419
438
457
476
495
Loss G_A



Loss G_B
1.4
1.2
1
0.8
0.6
0.4
0.2
0
1
13
25
37
49
61
73
85
97
109
121
133
145
157
169
181
193
205
217
229
241
253
265
277
289
301
313
325
337
349
361
373
385
397
409
421
433
445
457
469
481
493
Loss G_B





Video to Video translation modelling

- Apart from the image components in the above model, video components also need to be trained.
- Optical flow defines the perceived flow(pattern) continuously moving through a video.
- A similar (GAN) generator discriminator combination is added to train the network to learn if the video is real(stable) or fake.
- The generator for the image and the video component is however combined in the proposed model.

Thus the proposed mathematical model is as follows -

There are two generators G and F and four discriminators.G is a generator for mapping input video to output video and F does the mapping in the reverse direction.

We use two types of discriminators one is an image discriminator(D_{FI},D_{GI}) and the other is a video discriminator(D_{FV},D_{GV}).

x_t is t^{th} frame in the output video

$\tilde{x}_t$ is the t_{th} frame generated by G

s_t is the t_{th} frame in the input video

s_t is the t_{th} frame generated by F

w_{t-1} is the optical flow used to convert x_{t-1} to x_t

o_{t-1} is the optical flow used to convert s_{t-1} to s_t

w_{t-k}^{t-1} is the k-1 optical flow for k consecutive real images x_{t-k}^{t-1}

o_{t-k}^{t-1} is the k-1 optical flow for k consecutive input images s_{t-k}^{t-1}

Thus D_{GI} will output 1 for x_t and 0 for $\tilde{x}_t$ and D_{FI} will output 1 for s_t and 0 for $\tilde{s}_t$.

D_{GV} will try to make it's output 1 for (x_{t-k}^{t-1} , w_{t-k}^{t-2}) and 0 for ($\tilde{x}_{t-k}^{t-1}$, w_{t-k}^{t-2}) and D_{FV} will try to make it's output 1 for (s_{t-k}^{t-1} ,o_{t-k}^{t-2}) and 0 for ($\tilde{s}_{t-k}^{t-1}$,o_{t-k}^{t-2}).

Thus we have two types of losses adversarial loss which is the one used in Standard GAN and cycle loss used in Cycle GAN.

There are 4 adversarial losses -

$$\mathcal{L}(G, D_{GI}, S, X) = E_{\phi_I(x_1^T)}[logD_{GI}(x_i)] + E_{\phi_I(x_1^T)}[log(1 - D_{GI}(\tilde{x}_i))] \tag{5.5}$$

$$\mathcal{L}(G, D_{GV}, S, X) = E_{\phi_V(x_1^T, w_1^{T-1})}[logD_{GV}(x_{i-k}^{i-1}, w_{i-k}^{i-2})] + E_{\phi_V(x_1^T, w_1^{T-1})}[log(1 - D_{GV}(\tilde{x}_{i-k}^{i-1}, w_{i-k}^{i-2}))] \tag{5.6}$$

$$\mathcal{L}(F, D_{FI}, X, S) = E_{\phi_I(s_1^T)}[logD_{FI}(S_i)] + E_{\phi_I(s_1^T)}[log(1 - D_{FI}(\tilde{s}_i))] \tag{5.7}$$

$$\mathcal{L}(F, D_{FV}, X, S) = E_{\phi_V(s_1^T, o_1^{T-1})}[logD_{FV}(s_{i-k}^{i-1}, o_{i-k}^{i-2})] + E_{\phi_V(s_1^T, o_1^{T-1})}[log(1 - D_{FV}(\tilde{s}_{i-k}^{i-1}, o_{i-k}^{i-2}))] \tag{5.8}$$

Cycle loss -

$$\mathcal{L}_{cyc}(G, F) = \frac{1}{T}\sum_{t=1}^{T}[\| x_t - F(G(\tilde{x}_{t-l}^{t-1}, s_{t-l}^{t-1})) \|_1 + \| s_t - G(F(\tilde{s}_{t-l}^{t-1}, x_{t-l}^{t-1})) \|_1] \tag{5.9}$$

Final objective function -

$$G^*, F^* = \arg\min_{G,F}\ \max_{D_{GI}, D_{GV}, D_{FI}, D_{FV}} \mathcal{L}(G, F, D_{GI}, D_{GV}, D_{FI}, D_{FV}) \qquad (5.11)$$

Q & A