

# Enter Title here

Université Paris-Saclay Machine Vision Project

Charbel Abi Hana  
 Department of Electrical Engineering  
 Université Paris-Saclay  
 Paris, France  
 charbel-a-h@outlook.com

**Abstract—**  
**Index Terms—**

## I. INTRODUCTION

hello [1]

## II. RELATED WORK

## III. METHODS

### A. Conditional Generative adversarial network(GANs)

$$\min_G \max_D \mathcal{V}_{\text{GAN}}(D, G) = \quad (1)$$

$$\mathbb{E}_{x \sim p_{\text{data}}(x)} [\log \{D(x)\}] + \mathbb{E}_{z \sim p_z(z)} [\log \{1 - D(G(z))\}].$$

where  $G : R^{100} \rightarrow R^{16,384}$

$$L_D = - \sum_{x \in \chi, z \in \zeta} \log(D(x)) + \log(1 - D(G(z))) \quad (6)$$

$$L_G = - \sum_{z \in \zeta} \log(D(G(z))) \quad (7)$$

### B. Model Architecture

#### 1) Generator:

$$h^{[i]} = \text{LeakyRELU}(W^{[i-1]}h^{[i-1]} + b[i-1]) \quad (2)$$

$\alpha$ , with  $h[i] \in R^{16 \times 2^i}$  and we output the vector  $o \in R^{16,384}$  via

$$o = \tanh(W^{[L]}h^{[L]} + b[L]) \quad (3)$$

Where L is the final layer.

2) *Discriminator*:  $h[0]$  denote the input image,  $W[j]$  and  $b[j]$  denoting the weight matrix and the bias vector in the L output layer, we have:

$$o = \text{sigmoid}(W^{[L]}h^{[L]} + b[L]) \quad (4)$$

#### 3) Classifier:

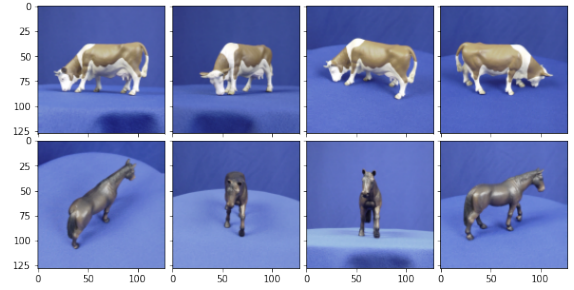


Fig. 1. Random Cows and Horses Images from Original Dataset

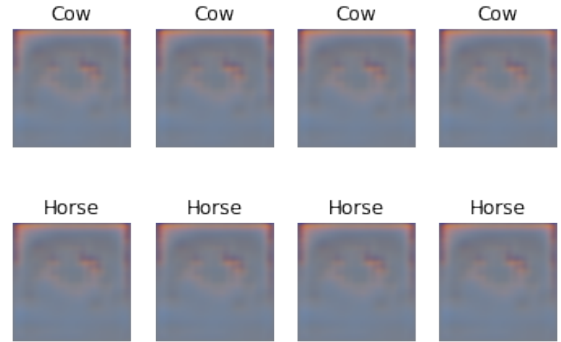


Fig. 2. Generated Output after 100 Steps

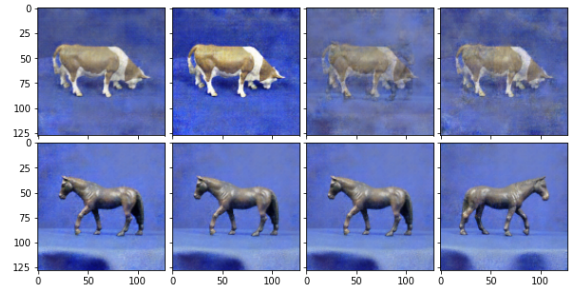


Fig. 3. Generated Output after 35000 Steps

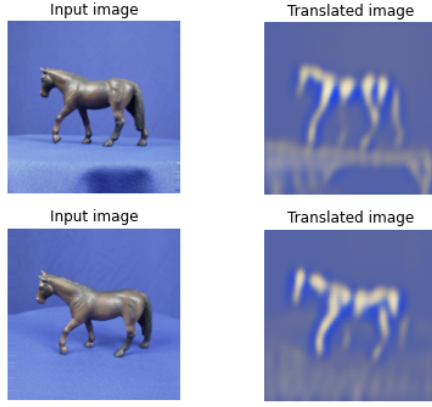


Fig. 4. Generated Translated Output after 1 Epoch

#### IV. EXPERIMENTS

##### A. Dataset

##### B. Evaluation Metrics

##### C. Experimentation Details

##### D. GAN Experimentation Results

##### E. CycleGAN Experimentation Results

- Pass real images through the generators and get the generated images
- Pass the generated images back to the generators to check if we can predict the original image from the generated image.
- Do an identity mapping of the real images using the generators.
- Pass the generated images in 1) to the corresponding discriminators.
- Calculate the generators total loss (adversarial + cycle + identity)
- Calculate the discriminators loss
- Update the weights of the generators
- Update the weights of the discriminators
- Return the losses in a dictionary.

##### F. Classifier Experimentation Results

Metric	Orig		Orig-250-GAN		Orig-500-GAN		Orig-1000-GAN	
	train	val	train	val	train	val	train	val
BCE-Loss	0.01	5	0.01	0.7	0.01	0.7	8	0.7
Accuracy	1.0	0.5	1.0	0.6	1.0	0.5	0.5	0.5
Precision	1.0	0.1	1.0	0.5	1.0	0.5	0.5	0.5
Recall	1.0	0.5	1.0	0.4	1.0	1.0	1.0	1.0

TABLE I

COMPILED TABLE OF CLASSIFICATION METRICS

#### V. CONCLUSION

##### REFERENCES

- [1] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.

Identify applicable funding agency here. If none, delete this.

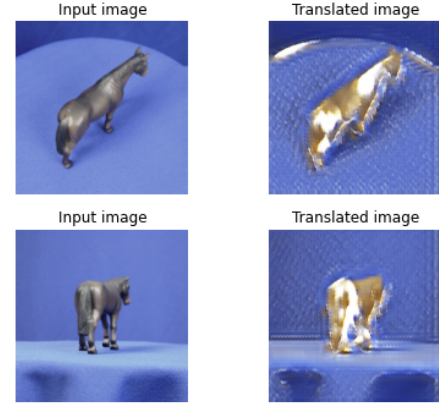


Fig. 5. Generated Output after 50 Epochs

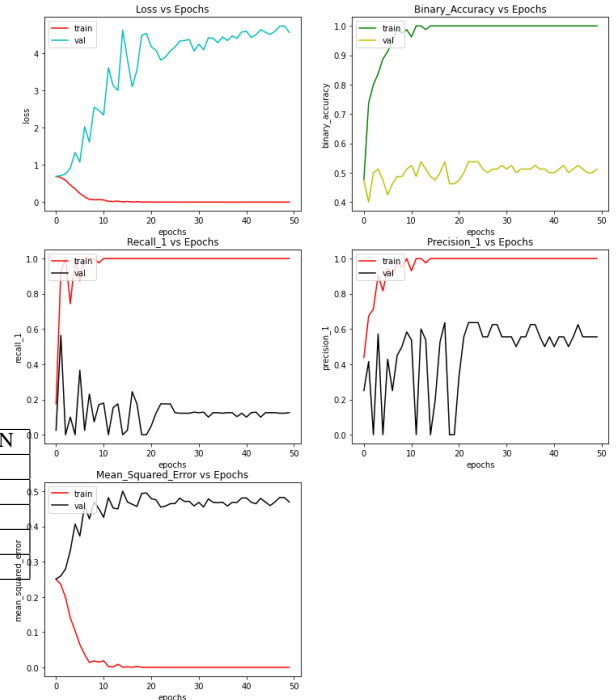


Fig. 6. Classification metrics for classifier trained on **original dataset**

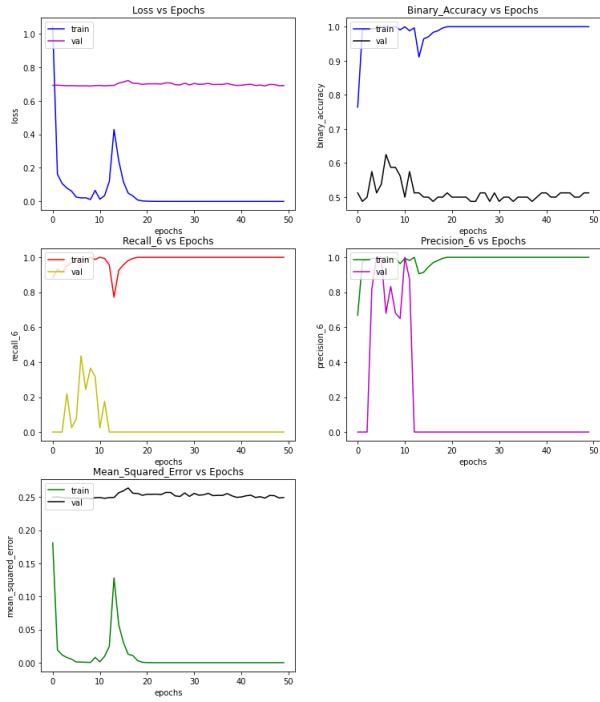


Fig. 7. Classification metrics for classifier trained on **original+250 GAN-based**

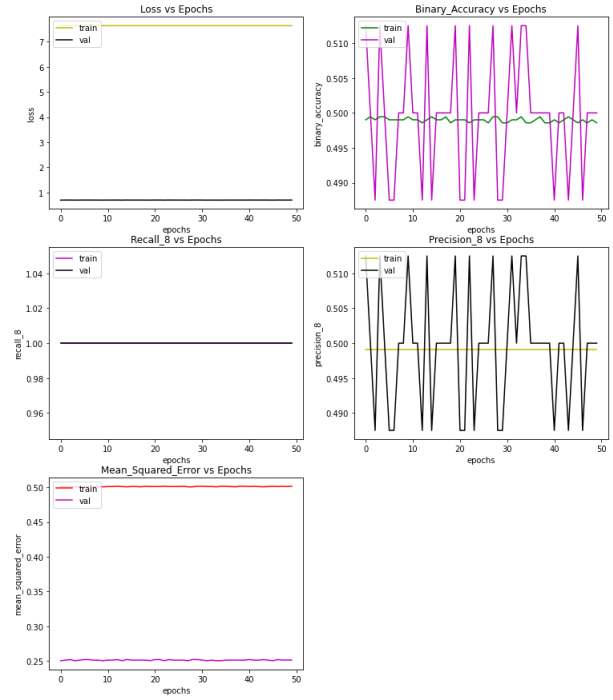


Fig. 9. Classification metrics for classifier trained on **original+250 GAN-based**

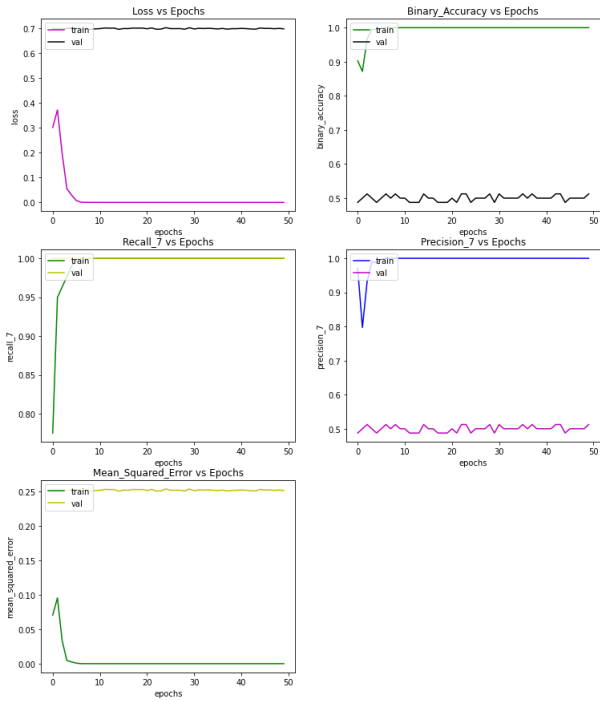


Fig. 8. Classification metrics for classifier trained on **original+250 GAN-based**