Pseudo-Cyclic Network for Unsupervised Colorization

R. Ratajczak*, C. Crispim-Junior, B. Fervers, E. Faure, L. Tougne

Laboratoire d’InfoRmatique en Image et Systèmes d’Information, UMR 5205 CNRS

Outline

Context

Methods

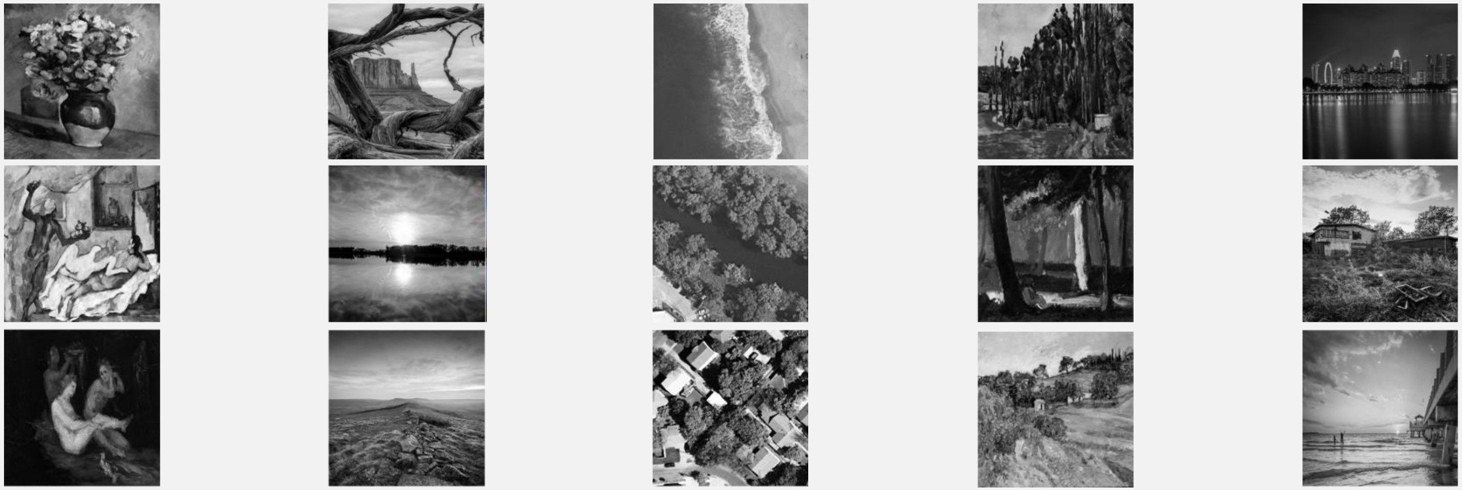
Experiments

Conclusion

Context

Hypothesis

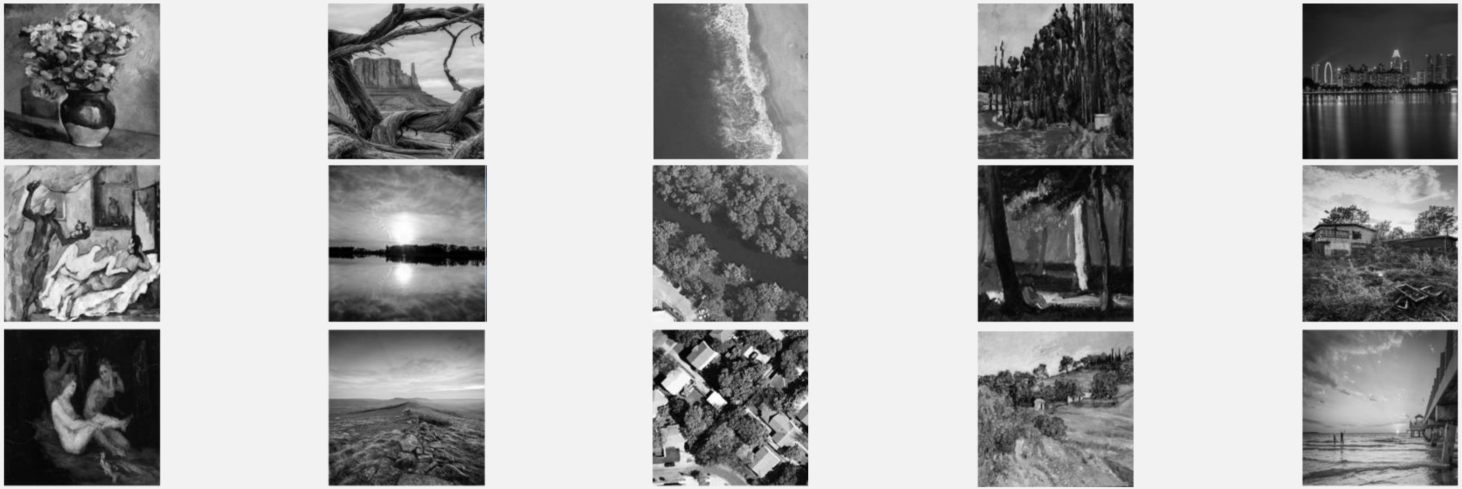
Grayscale → Difficult analysis



Hypothesis

Data are not always paired

Grayscale → Difficult analysis ← Mutable scenes Time & Scale

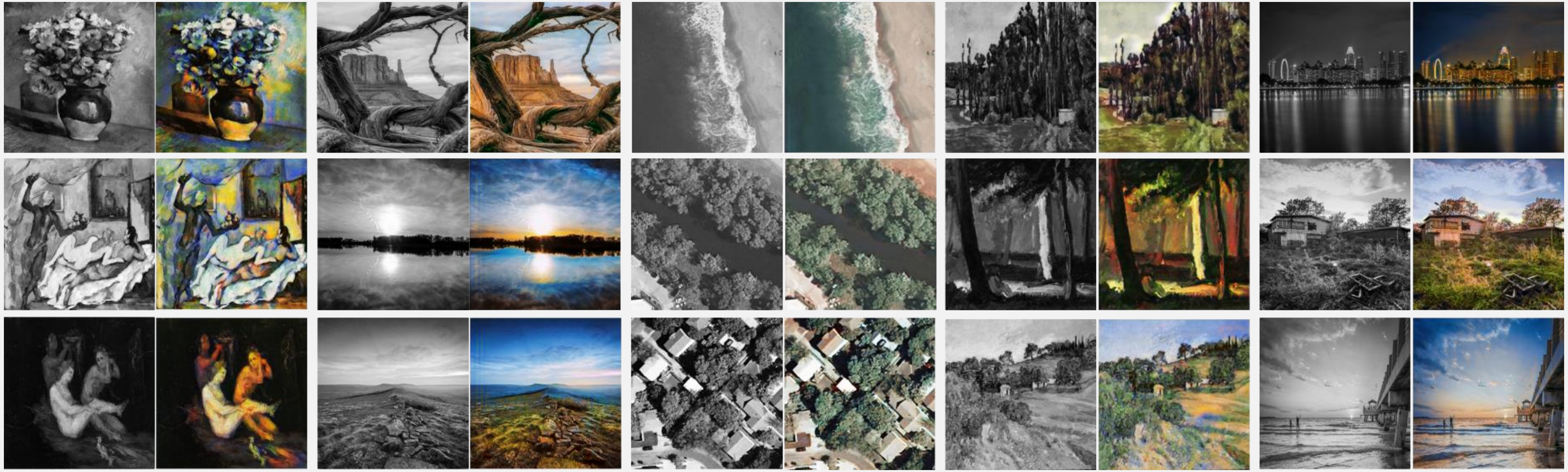


Hypothesis

Color could help

Data are not always paired

Grayscale → Difficult analysis ← Mutable scenes Time & Scale



Colorization

Main goals : Improve or modify visualizations, artistical or realistic

Domain A

Domain B

Grayscale → Generative process → Color

How to do this?

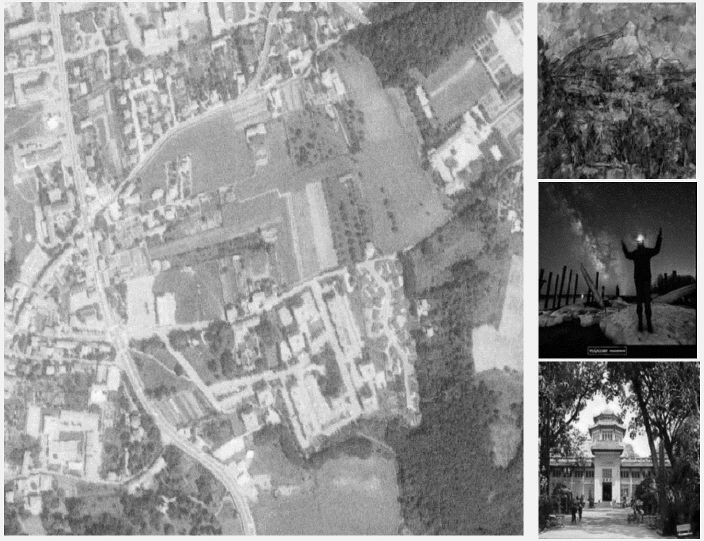
Colorization

Main goals : Improve or modify visualizations, artistical or realistic

Domain A

Grayscale → Generative process → Color

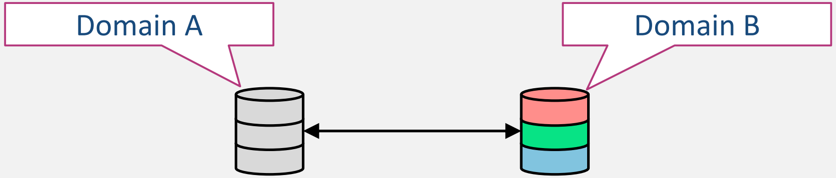
Domain B



How to do this?

Colorization as domain adaptation

Supervised



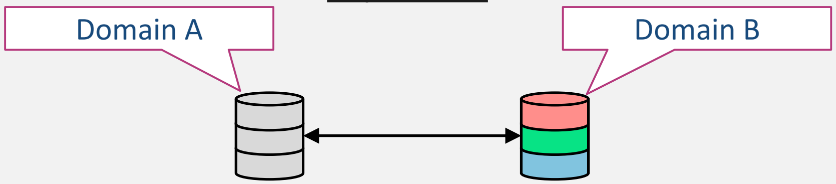
Needs paired data for training

Pros:
- Easier to train a model for colorization
- Results are usually more accurate

Cons:
- Data from both domains should be *paired*
- Trained models are more bias-prone

Colorization as domain adaptation

Supervised

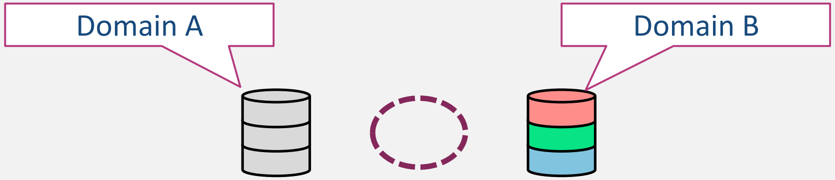


Needs paired data for training

Pros:
- Easier to train a model for colorization
- Results are usually more accurate

Cons:
- Data from both domains should be *paired*
- Trained models are more bias-prone

Unsupervised



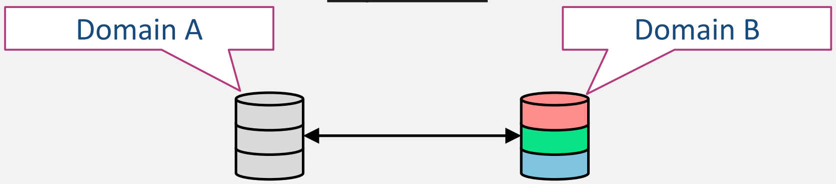
Does not need paired data for training

Pros:
- Allow to train a model with data from both domains
- Better handling of discrepancies (e.g., scales)

Cons:
- Data from both domains should/could be *related*
- Results are usually less accurate

Colorization as domain adaptation

Supervised

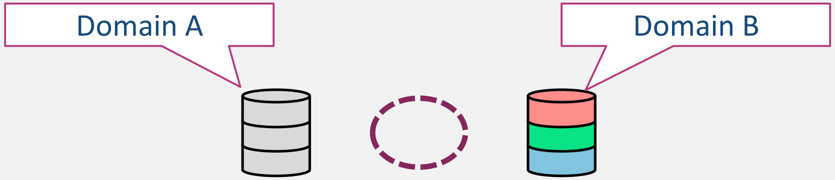


Needs paired data for training

Pros:
- Easier to train a model for colorization
- Results are usually more accurate

Cons:
- Data from both domains should be *paired*
- Trained models are more bias-prone

Unsupervised



Does not need paired data for training

Pros:
- Allow to train a model with data from both domains
- Better handling of discrepancies (e.g., scales)

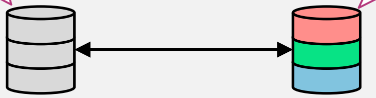
Cons:
- Data from both domains should/could be *related*
- Results are usually less accurate

Colorization as domain adaptation

Supervised

Domain A

Domain B



Needs paired data for training

Unsupervised

Domain A

Domain B

Does not need paired data for training

Deep Convolutional Neural Networks

Pros:
- Easier to train a model for colorization
- Results are usually more accurate

Cons:
- Data from both domains should be *paired*
- Trained models are more bias-prone

Pros:
- Allow to train a model with data from both domains
- Better handling of discrepancies (e.g., scales)

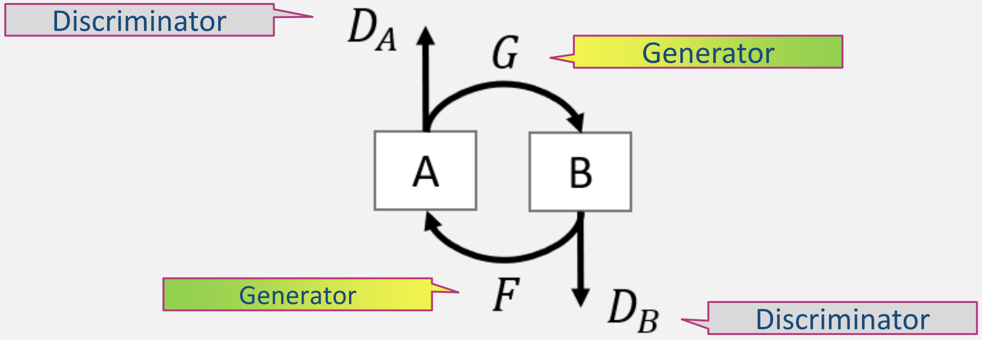
Cons:
- Data from both domains should/could be *related*
- Results are usually less accurate

Problematic

Could we use a handcrafted function to constrain the training of an **unsupervised** generative neural network for colorization?

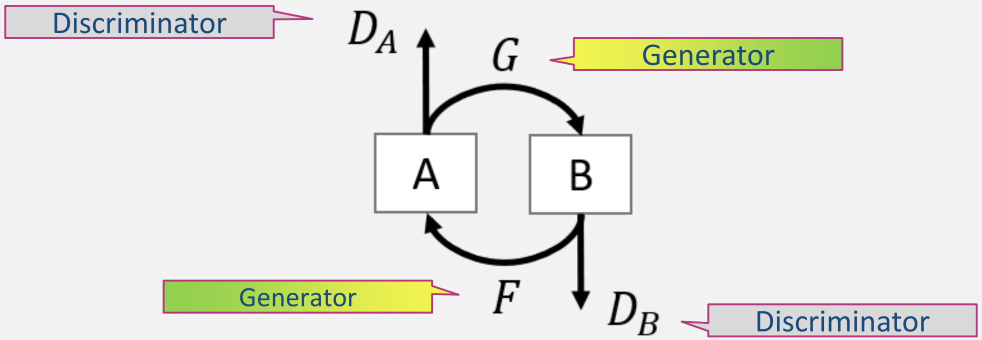
Methods

Adversarial Cyclic-Networks

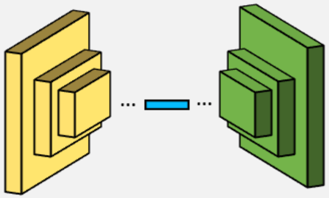


Popularized with **CycleGan** [Zhu et al., 2017]
Smaller versions like **Col-Cycle** followed [Ratajczak et al., 2019]

Adversarial Cyclic-Networks



Generators (G, F)

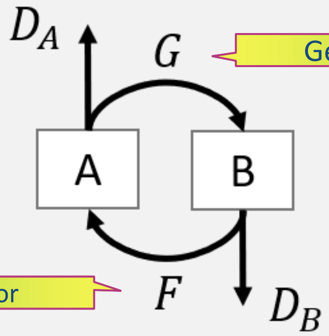


Popularized with **CycleGan** [Zhu et al., 2017]
Smaller versions like **Col-Cycle** followed [Ratajczak et al., 2019]

Adversarial Cyclic-Networks

Could we replace generator F with a well known handcrafted function ?

Discriminator

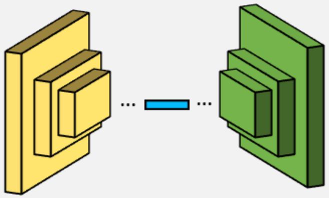


Generator

Generator

Discriminator

Generators (G, F)

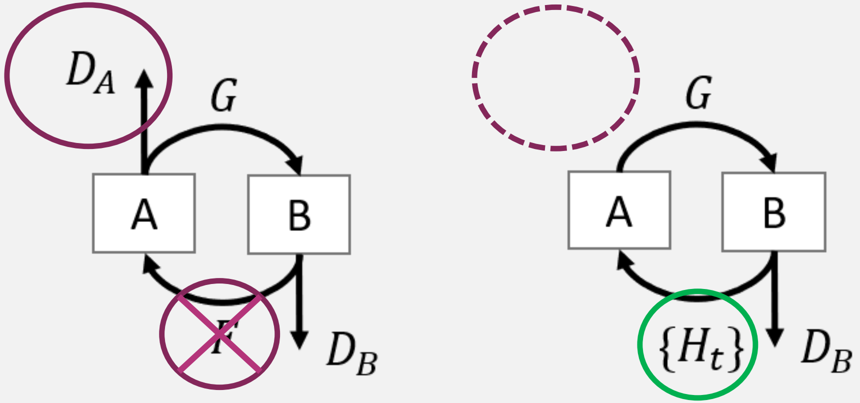


Encoder

Decoder

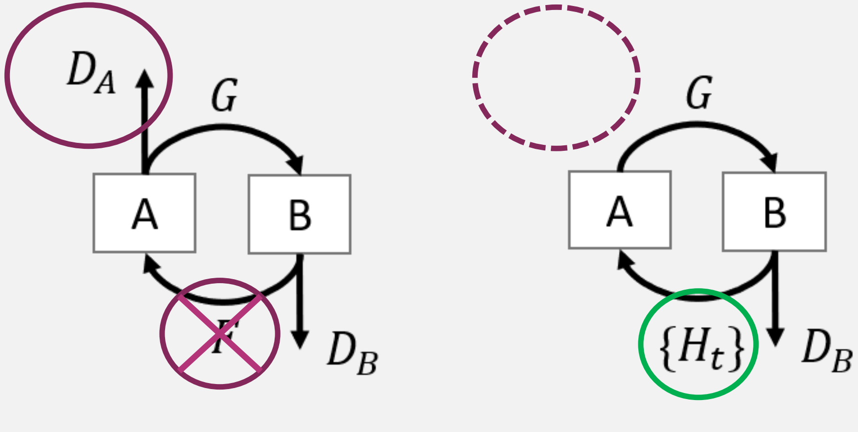
Popularized with **CycleGan** [Zhu et al., 2017]
Smaller versions like **Col-Cycle** followed [Ratajczak et al., 2019]

Rethinking the cycle for colorization



Replace F with well known Handcrafted Translation H_t

Rethinking the cycle for colorization



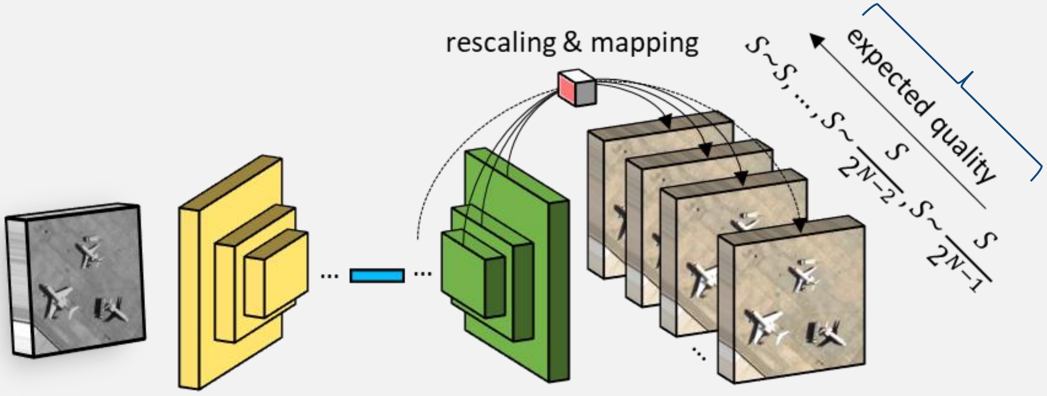
- ✓ Remove half the number of parameters
- ✓ Constrain the representation using H_t as a prior
- Color space B is RGB → linear & differentiable H_t

  H_t is the weighted sum of RGB channels

Replace F with well known Handcrafted Translation H_t

Output Spatial Pyramids (1/2)

Loss calculated with **all** scales

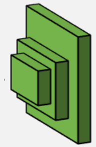


$$\frac{S}{2^{N-1}}, \frac{S}{2^{N-2}}, \dots, S = W \times H$$

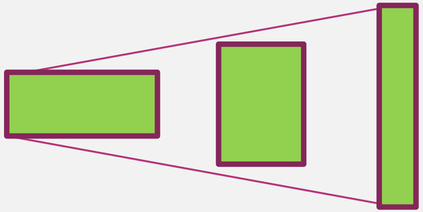
Single mapping function from deep features to domain B

⚠ Require same number of deep features as input

Output Spatial Pyramids (2/2)

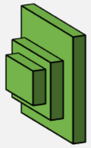


Usual scale / channel ratio



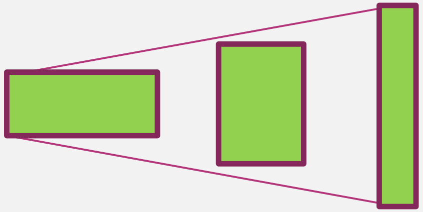
$H/4 \times W/4$	$H/2 \times W/2$	$H \times W$
C	$\frac{C}{2}$	$\frac{C}{4}$

Output Spatial Pyramids (2/2)



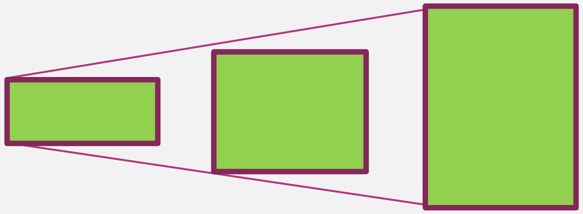
Generator: Decoder

Usual scale / channel ratio



$H/4 \times W/4$, C — $H/2 \times W/2$, $\frac{C}{2}$ — $H \times W$, $\frac{C}{4}$

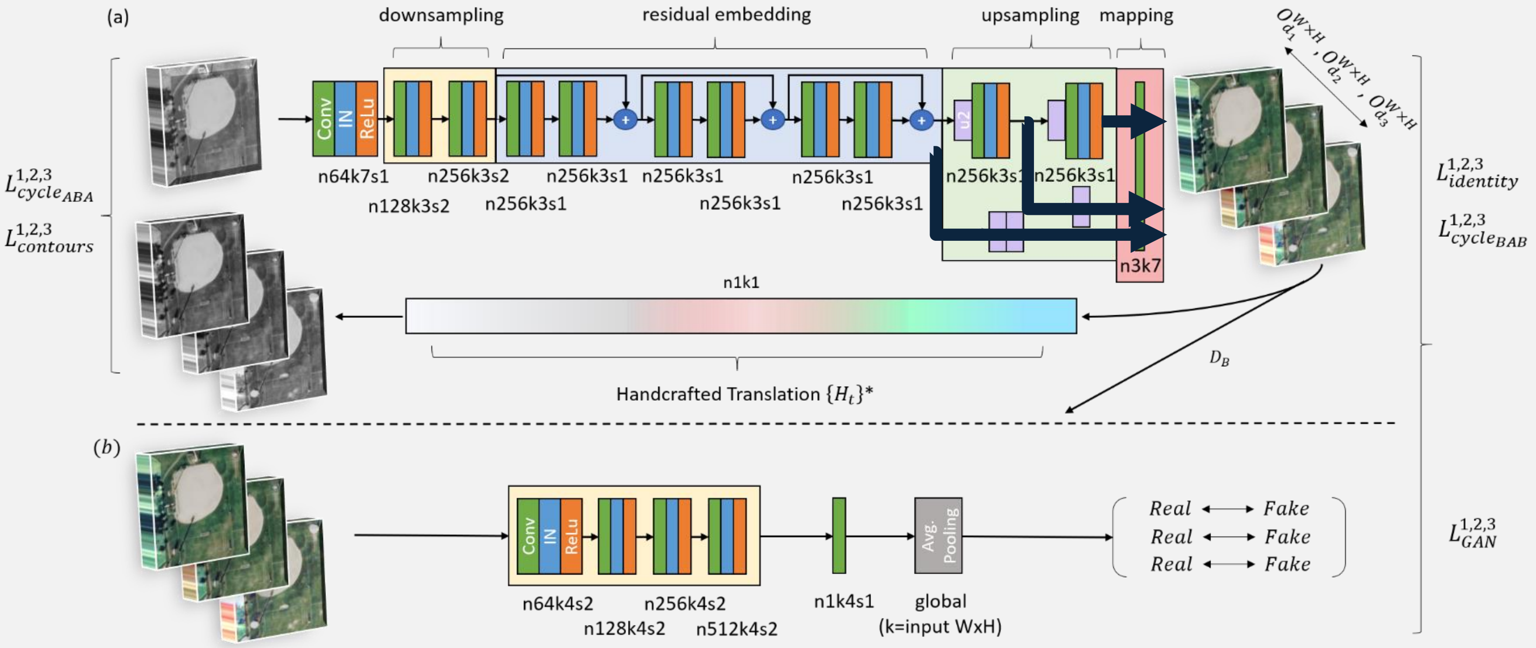
For the Output Spatial Pyramids



$H/4 \times W/4$, C — $H/2 \times W/2$, C — $H \times W$, C

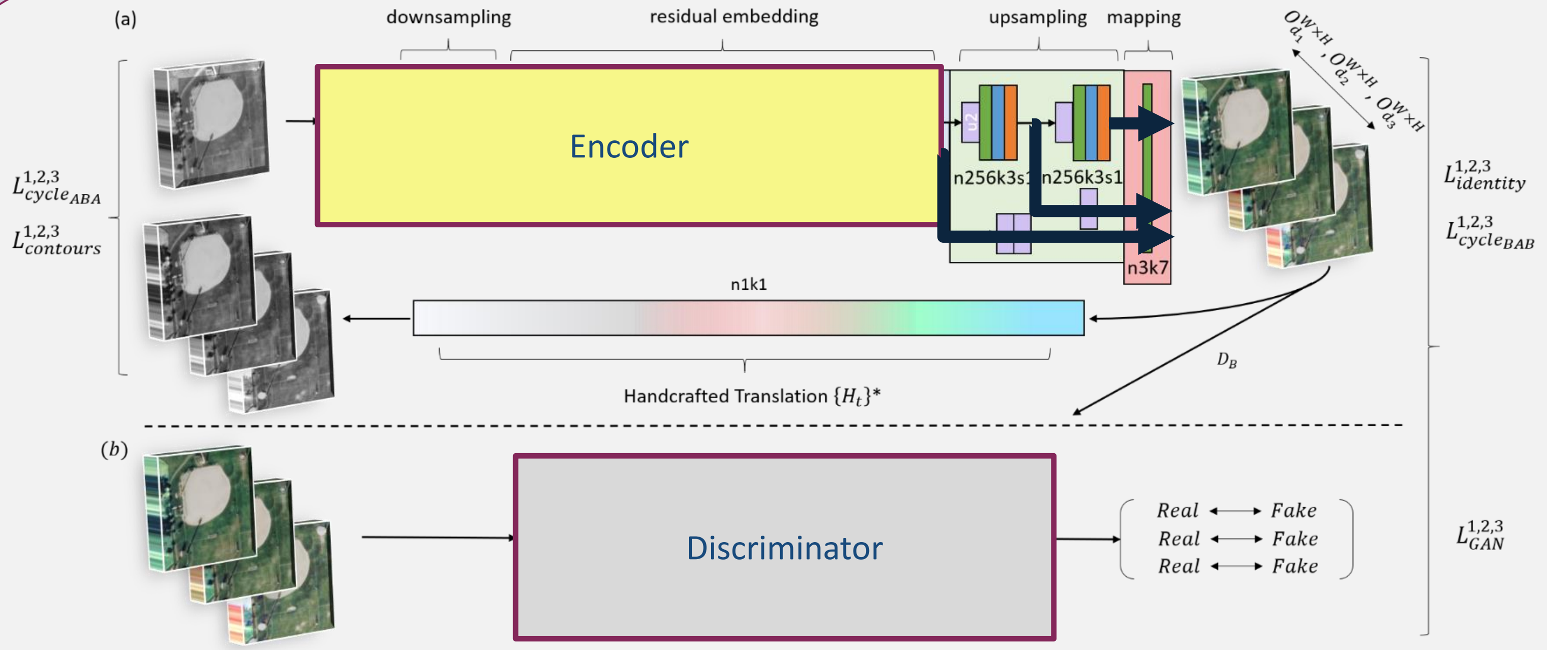
Need constant number of features in the decoder
Let’s keep it large to increase representativity

SpyncoGan



Let's put everything together! n=256 deep features in the decoder.

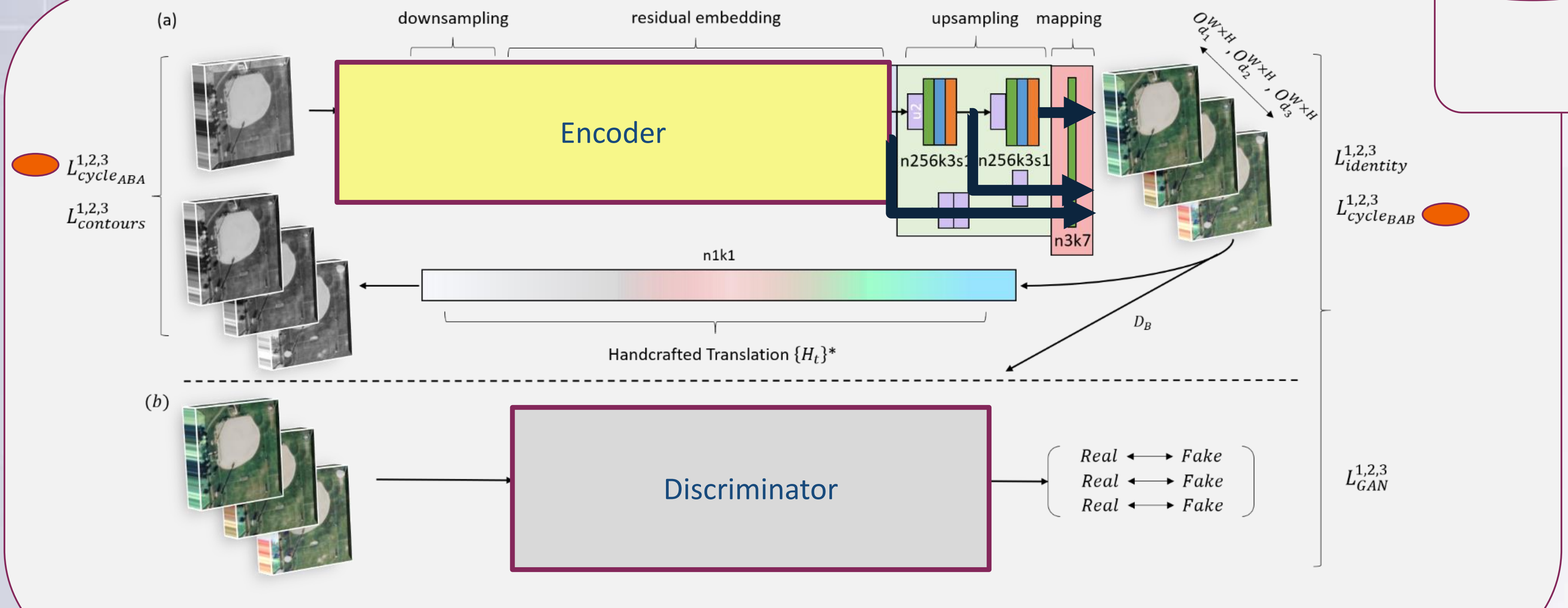
SpyncoGan



Let's put everything together! n=256 deep features in the decoder.

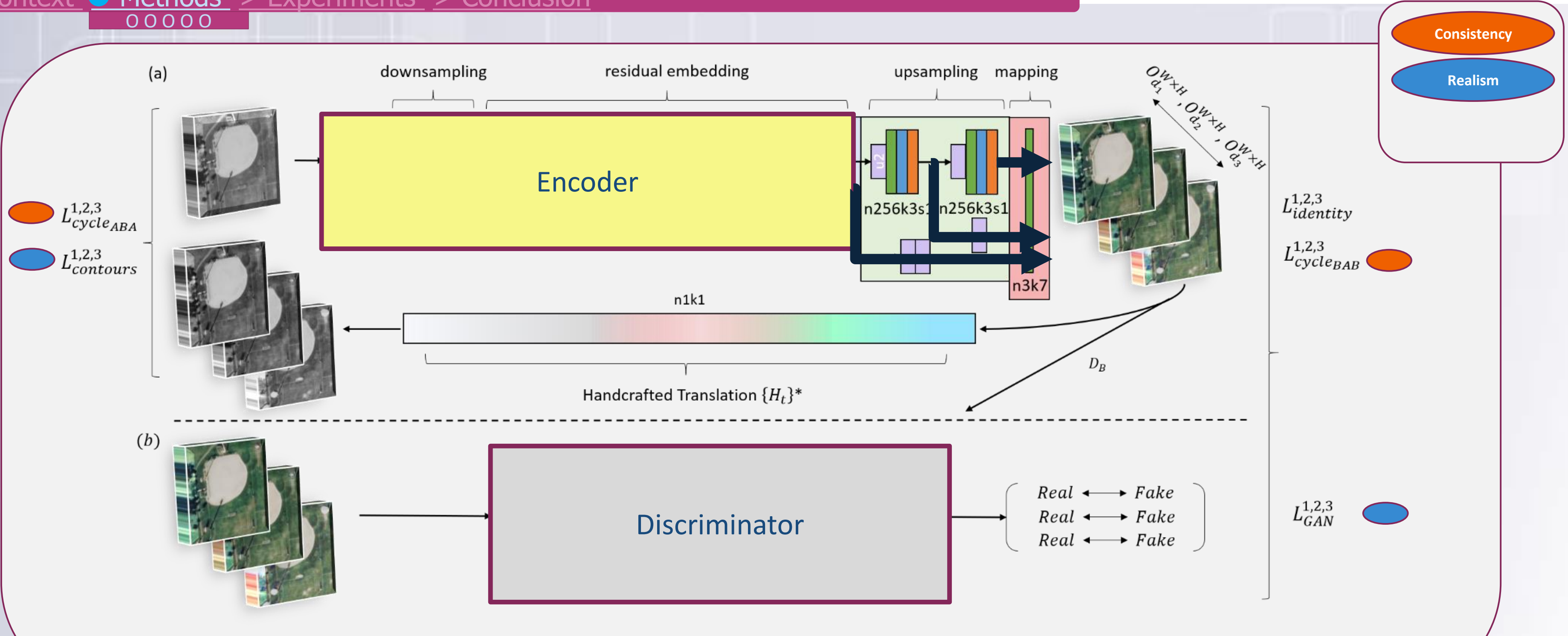
SpyncoGan

Consistency



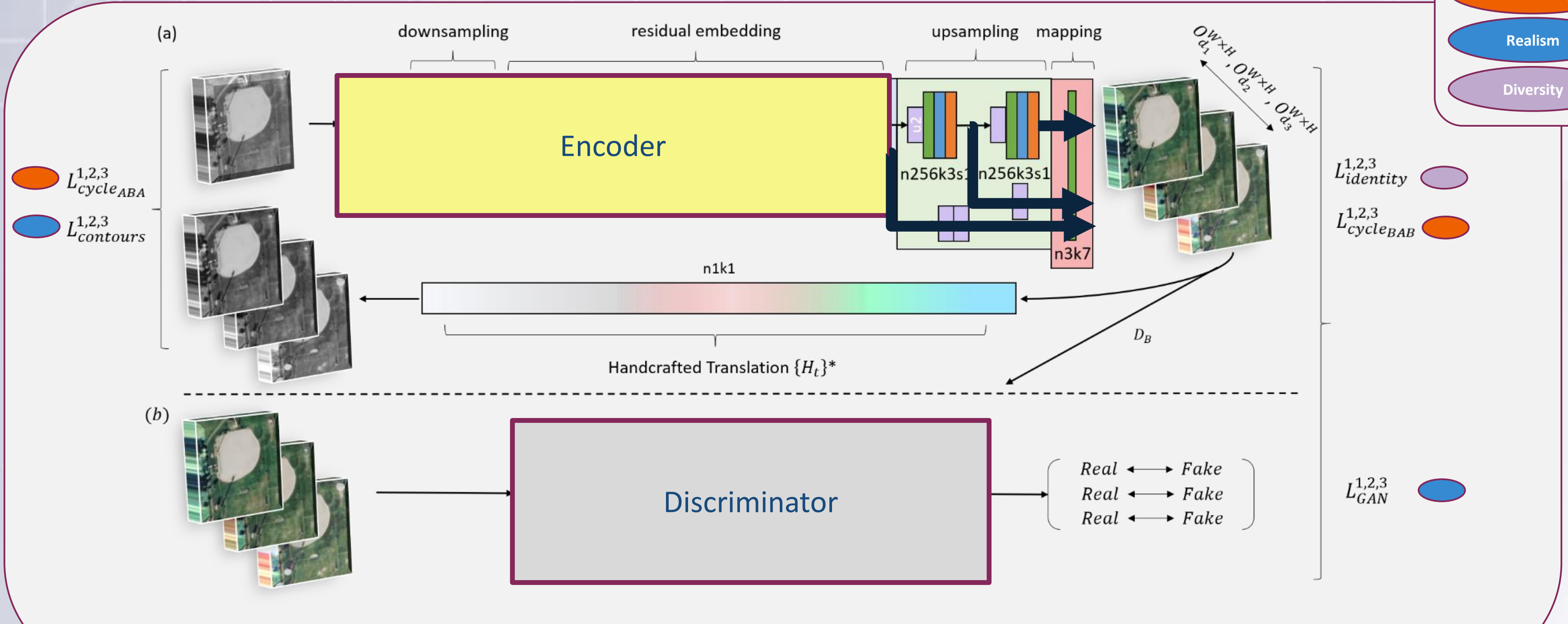
Let's put everything together! n=256 deep features in the decoder.

SpyncoGan



Let's put everything together! n=256 deep features in the decoder.

SpyncoGan

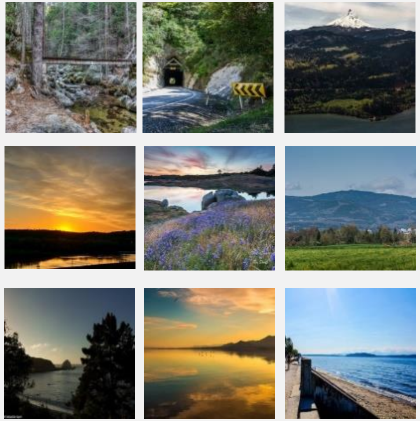


Let's put everything together! n=256 deep features in the decoder.

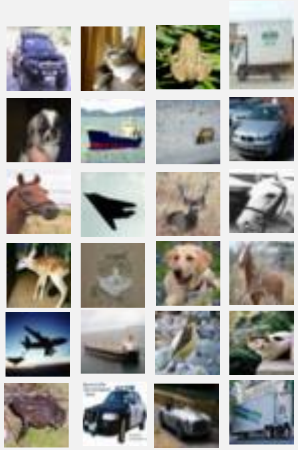
Experiments

Datasets

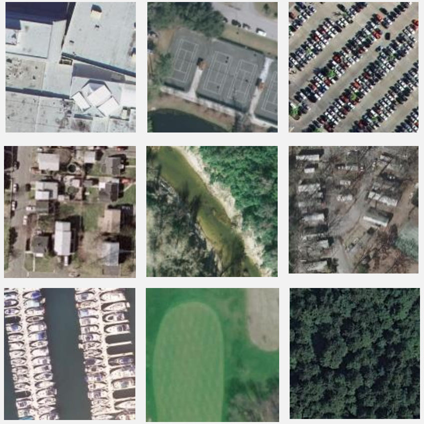
Cezanne paintings



Landscape photos



Cifar10

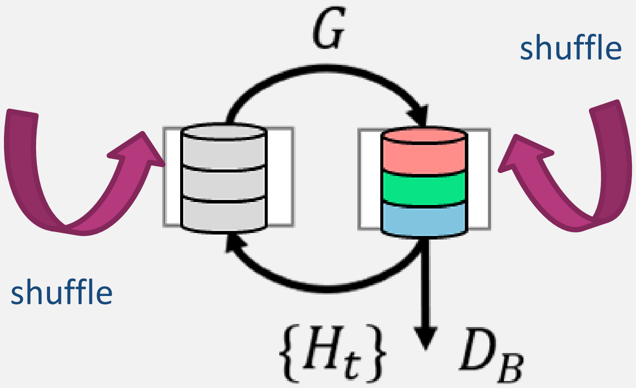


UCMerced Land Use

Data with real colors are used for evaluation.
Shuffled and supposed unpaired for training.

Evaluation

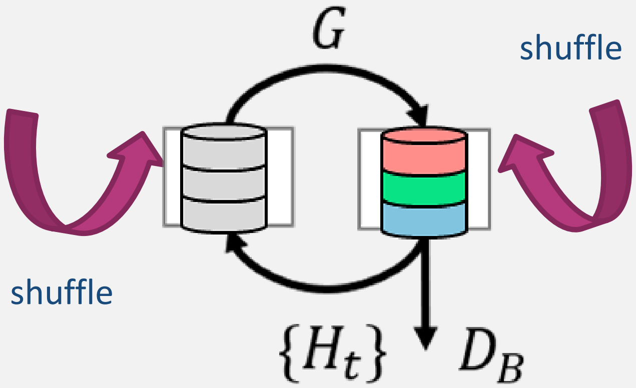
Train sets



Training, data supposed unpaired

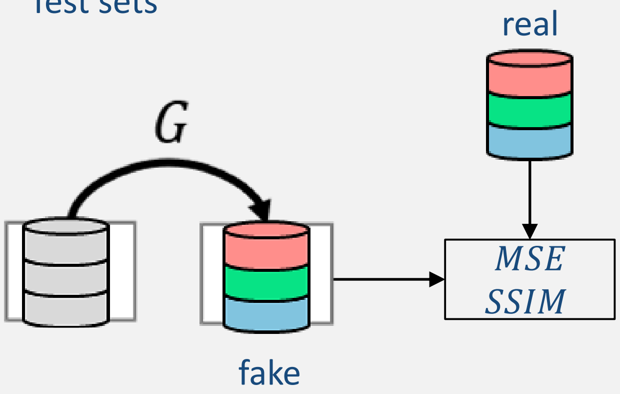
Evaluation

Train sets



Training, data supposed unpaired

Test sets



→ Ablation study
→ Comparison with the SOTA

Compare fake color images with real color images

Ablation (1/3)

Output Ablation

Dataset	Loss function	Avg. MSE ↓	Avg. SSIM (%) ↑
Cezanne paintings	$\mathcal{L}^{1,2,3}$	91.5	82
Landscape photos	$\mathcal{L}^{1,2,3}$	85.1	83
UCMerced Land Use	$\mathcal{L}^{1,2,3}$	83.1	85
Cifar-10	$\mathcal{L}^{1,2,3}$	86.8	89

Loss with all outputs $L^{1,2,3}$

Ablation (1/3)

Output Ablation

Dataset	Loss function	Avg. MSE ↓	Avg. SSIM (%) ↑
Cezanne paintings	$\mathcal{L}^1$	92.9	82
Cezanne paintings	$\mathcal{L}^{1,2,3}$	91.5	82
Landscape photos	$\mathcal{L}^1$	85.7	83
Landscape photos	$\mathcal{L}^{1,2,3}$	85.1	83
UCMerced Land Use	$\mathcal{L}^1$	85.5	86
UCMerced Land Use	$\mathcal{L}^{1,2,3}$	83.1	85
Cifar-10	$\mathcal{L}^1$	87.2	89
Cifar-10	$\mathcal{L}^{1,2,3}$	86.8	89

Loss with all outputs $L^{1,2,3}$

Loss with only last output L^1

Ablation (2/3)

Real Gray

Real Color

Intermediary outputs **(H/2, W/2).**
Top: without output ablation
Bottom: with output ablation

Ablation (3/3)

Contours Loss Ablation

Loss function	Ablation	Avg. MSE ↓	Avg. SSIM (%) ↑
$\mathcal{L}^1$	/	92.9	82
$\mathcal{L}^{1,2,3}$	/	91.5	82

With contours loss (on paintings)

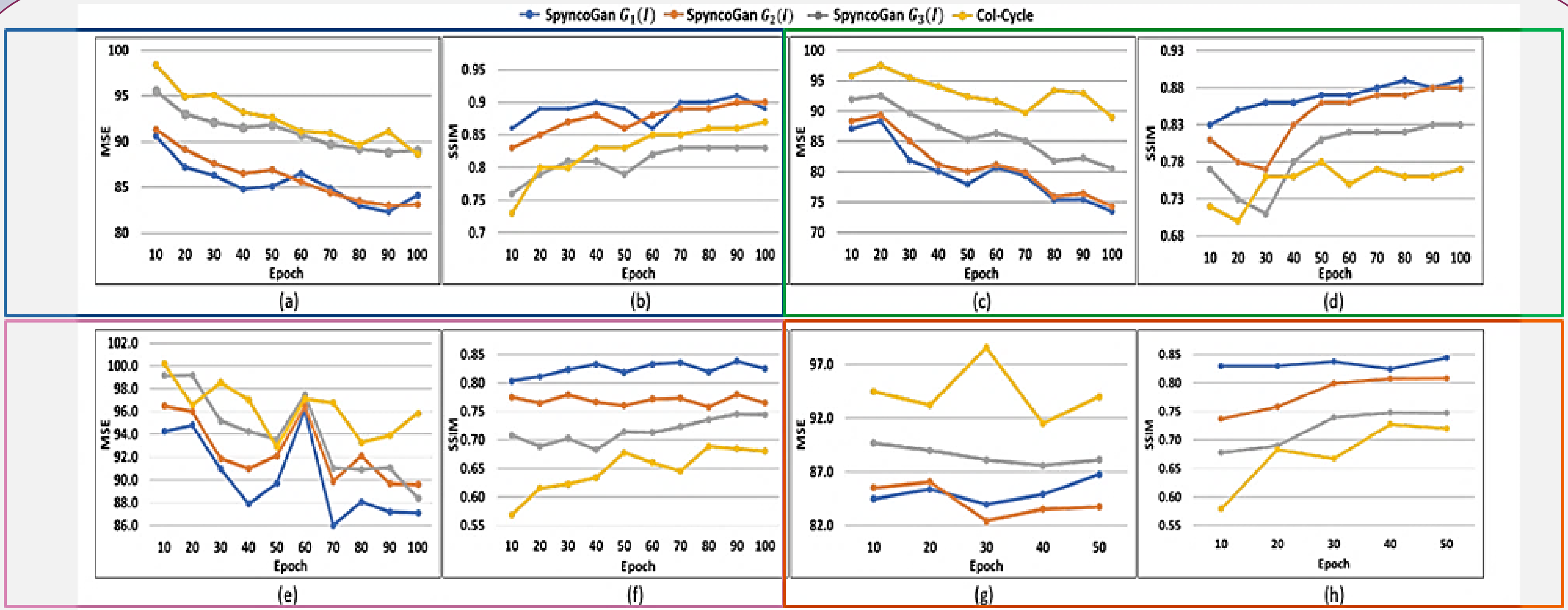
Ablation (3/3)

Contours Loss Ablation

Loss function	Ablation	Avg. MSE ↓	Avg. SSIM (%) ↑
$\mathcal{L}^1$	$\mathcal{L}^1_{contours}$	92.6	79
$\mathcal{L}^1$	/	92.9	82
$\mathcal{L}^{1,2,3}$	$\mathcal{L}^{1,2,3}_{contours}$	92.0	77
$\mathcal{L}^{1,2,3}$	/	91.5	82

Without contours loss (on paintings)
With contours loss (on paintings)

Comparison with SOTA



Conclusion

Conclusions

Handcrafted functions help to constrain deep neural networks for colorization

Output Spatial Pyramids are promising, but they require more memory

Training a classification network on colorized images improves generalization
(see the paper…)

Future works

Multispectral and Hyperspectral

Other generative tasks
(e.g., segmentation)

To see more results...

Supplementary materials

SCAN ME

http://tiny.cc/mvz5dz