Guided-Generative Network for noise detection in Monte-Carlo rendering

Jérôme Buisine, Fabien Teytaud, Samuel Delepoulle and Christophe Renaud

13th December 2021 - ICMLA

University of Littoral Opale Coast (Calais, France)

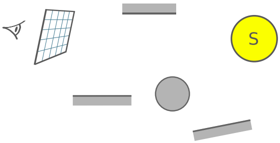
Plan

Plan

Context: Synthesis images

Geometric scenes 3D

- Objects
- Camera
- Light sources

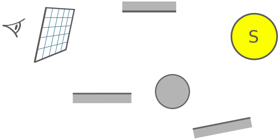


[1] https://github.com/mmp/pbrt-v4-scenes

Context: Synthesis images

Geometric scenes 3D

- Objects
- Camera
- Light sources



⇒

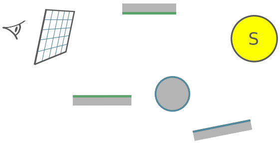
Photorealistic image [1]

[1] https://github.com/mmp/pbrt-v4-scenes

Context: Synthesis images

Geometric scenes 3D

- Objects
- Camera
- Light sources

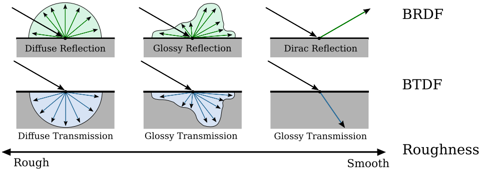


⇒

Photorealistic image [1]

Lighting simulation

- Material properties
- Lighting interactions

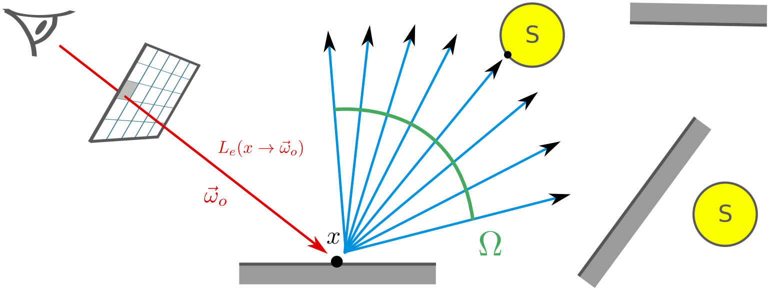


[1] https://github.com/mmp/pbrt-v4-scenes

Context: Global illumination

Rendering equation for global illumination [Kajiya, 1986]

$$L(x \rightarrow \vec{\omega}_o) = L_e(x \rightarrow \vec{\omega}_o) + \int_{\Omega} f_r(x, \vec{\omega}_i \rightarrow \vec{\omega}_o)\, L_i(x \leftarrow \vec{\omega}_i) \cos\theta_{\vec{\omega}_i} \mathrm{d}\omega_i$$



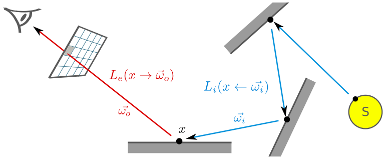
Context: Monte-Carlo method

Integration by Monte-Carlo method

$$F_N = \frac{1}{N}\sum_{j=1}^{N}\frac{f(x)}{p(x)} \approx \int_\Omega \frac{f(x)}{p(x)}p(x)\mathrm{d}x$$

with:

- N, number of samples;
- p, a probability density function.



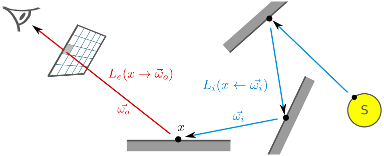
Context: Monte-Carlo method

Integration by Monte-Carlo method

$$F_N = \frac{1}{N}\sum_{j=1}^{N}\frac{f(x)}{p(x)} \approx \int_{\Omega}\frac{f(x)}{p(x)}p(x)\mathrm{d}x$$

with:

- N, number of samples;
- p, a probability density function.



Important notes

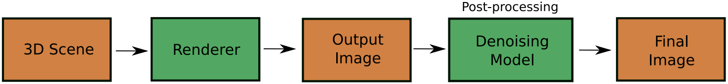
- Highly noticeable visual noise
- Requires a lot of computing time

Context: Perceptible residual noise

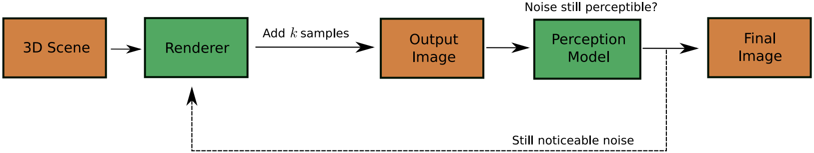
(a) 1 sample (b) 20 samples (c) 500 samples (d) 10 000 samples

Context: Taking noise into account

Post-processing treatment:



Stopping criterion when rendering:



Noise perception model

- Models based on machine learning

Plan

State of the art: Toward a perception model

Computer generated images base [Buisine et al., 2021b]

- 40 images of size 800×800
- Different levels of noise (number of samples per pixel)
- Images divided into 16 blocks $(16 \times 200 \times 200)$

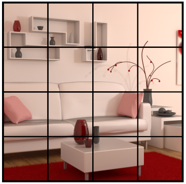
State of the art: Toward a perception model

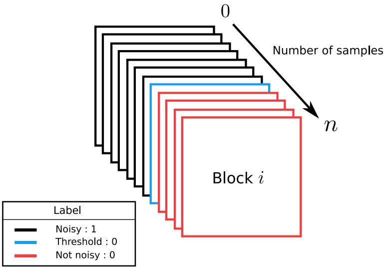
Computer generated images base [Buisine et al., 2021b]

- 40 images of size 800 × 800
- Different levels of noise (number of samples per pixel)
- Images divided into 16 blocks (16 × 200 × 200)

Labeling of data

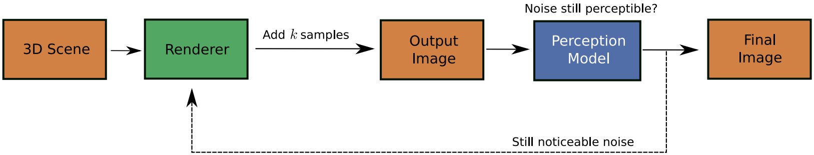
- Threshold collection (number of samples)
- Perception threshold (Mean Opinion Score)
- **Binary classification task**





State of the art: Toward a perception model

Interaction with a perception model:



Proposals for models of perception

- Support Vector Machine
  [Takouachet et al., 2017, Constantin et al., 2015, Constantin et al., 2016]
- Recurent Neural Network (RNN) [Buisine et al., 2021a]

Difficulty

- Find relevant input data for the perception model (noise characterization)

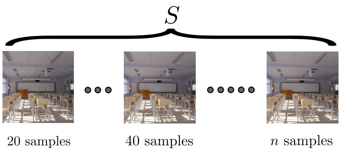
Plan

Plan

Guided-Generative Network: Motivation

Remarks from previous work

- Unknown reference image
- Characterization of noise in a scene seems difficult to generalize
- Use of sliding window [Buisine et al., 2021a]

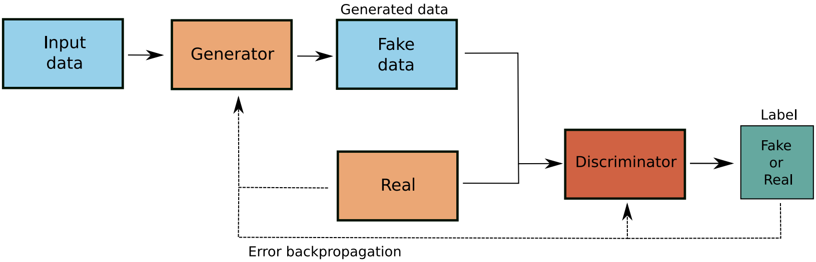


Considered solution

- Use of denoising methods
- Automated feature generation

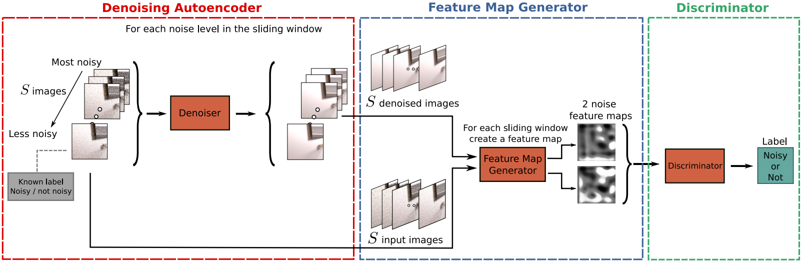
Guided-Generatic Network: GAN based approach

Generative Adversarial Network (GAN) [Goodfellow et al., 2014] **:**



Guided-Generative Network: Architecture

Guided-Generative Network (GGN)

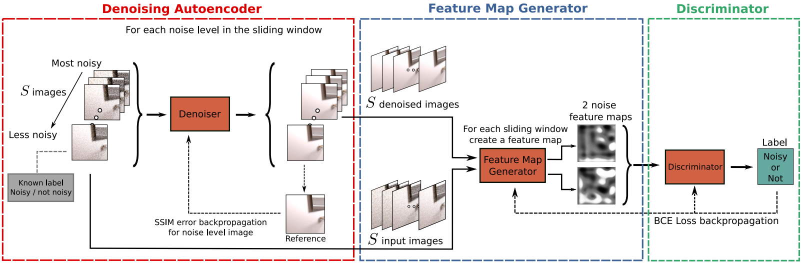


Model objectives

- Correctly denoise images of the sliding window [Ronneberger et al., 2015]
- Generate a feature map of the differences between the input and denoised images
- Interpret the difference in the feature maps

Guided-Generative Network: Architecture

Guided-Generative Network (GGN)



How to train a such model?

- Denoiser model learns from reference images using *Structural Similarity (SSIM)* metric [Wang et al., 2004]
- The two other models learn from the discriminator errors
- Use of the loss function *Binary Cross-Entropy* [Buja et al., 2005]

Plan

Guided-Generative Network: Training protocol

Training dataset

- 35 selected images from 40 availables
- 16 blocks of size 200×200
- 500 different noise levels
- Total of 280 000 labeled data

Training parameters

- Sliding window $S = 6$
- 30 epochs

Guided-Generative Network: Results overview

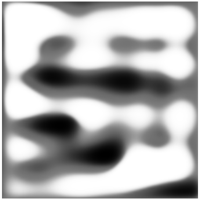
Denoising Model:

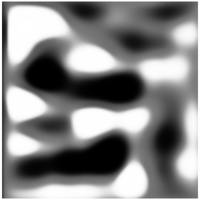

Noisy image
SSIM: 0.6433


Denoised image
SSIM: 0.9859

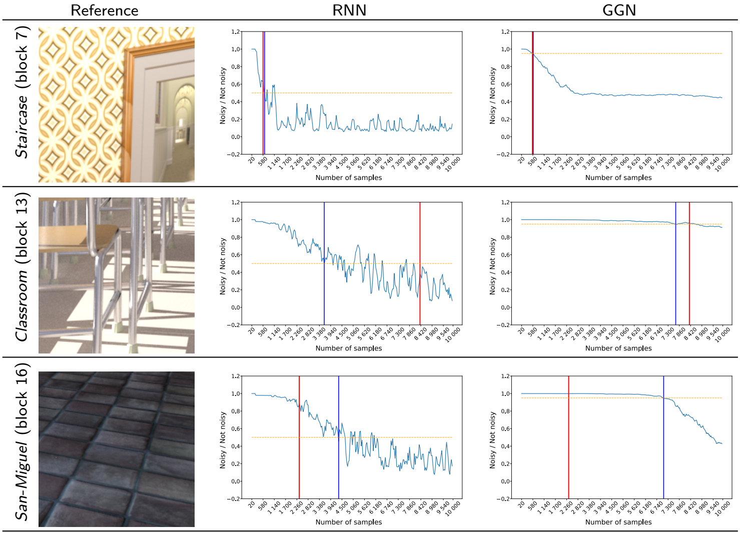

Référence
SSIM: 1.0

Feature Map Generator Model:


NFM of input images

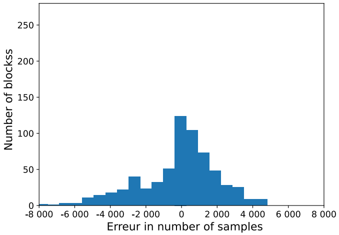

NFM of denoised images

Guided-Generative Network: Predictions

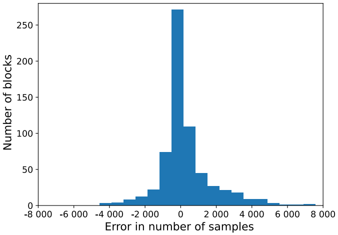


Guided-Generative Network: Study of the performance of the models

Distribution of differences between predicted and subjective thresholds:



(a) *RNN - Entropie SVD*



(b) GGN

Plan

Conclusion and future works

Contribution

- Complete architecture for the classification of synthesis images
- Generation of a feature map that guides the discriminator
- Discriminator with a property to be conservative rather than stopping too early

Conclusion and future works

Contribution

- Complete architecture for the classification of synthesis images
- Generation of a feature map that guides the discriminator
- Discriminator with a property to be conservative rather than stopping too early

Future works

- Overfitting reduction method for GAN [Yazici et al., 2020, Mukherjee et al., 2020]
- Study of the generality of the method for natural image classification tasks

Conclusion and future works

Thank you for your attention!

Buisine, J., Bigand, A., Synave, R., Delepoulle, S., and Renaud, C. (2021a).
Stopping criterion during rendering of computer-generated images based on svd-entropy.
Entropy, 23(1):75.

Buisine, J., Delepoulle, S., Synave, R., and Renaud, C. (2021b).
Subjective human thresholds over computer generated images.

Buja, A., Stuetzle, W., and Shen, Y. (2005).
Loss functions for binary class probability estimation and classification: Structure and applications.
Working draft, November, 3.

Constantin, J., Bigand, A., Constantin, I., and Hamad, D. (2015).
Image noise detection in global illumination methods based on FRVM.
164:82–95.

Constantin, J., Constantin, I., Bigand, A., and Hamad, D. (2016).
Perception of noise in global illumination based on inductive learning.
In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 5021–5028. IEEE.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014).
Generative adversarial networks.

Kajiya, J. T. (1986).
The rendering equation.
In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150.

Mukherjee, S., Xu, Y., Trivedi, A., and Ferres, J. L. (2020).
Protecting gans against privacy attacks by preventing overfitting.

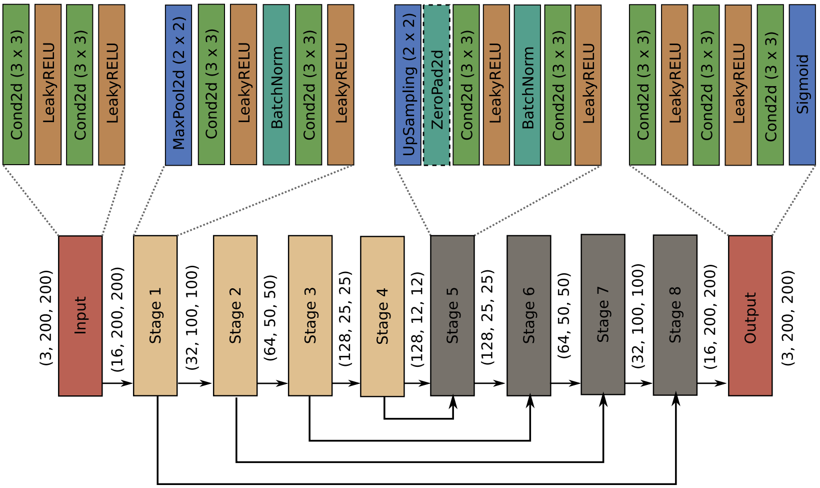
Ronneberger, O., Fischer, P., and Brox, T. (2015).
U-net: Convolutional networks for biomedical image segmentation.
In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, Lecture Notes in Computer Science, pages 234–241. Springer International Publishing.

Takouachet, N., Delepoulle, S., Renaud, C., Zoghlami, N., and Tavares, J. M. R. (2017).
Perception of noise and global illumination: Toward an automatic stopping criterion based on SVM.
69:49–58.

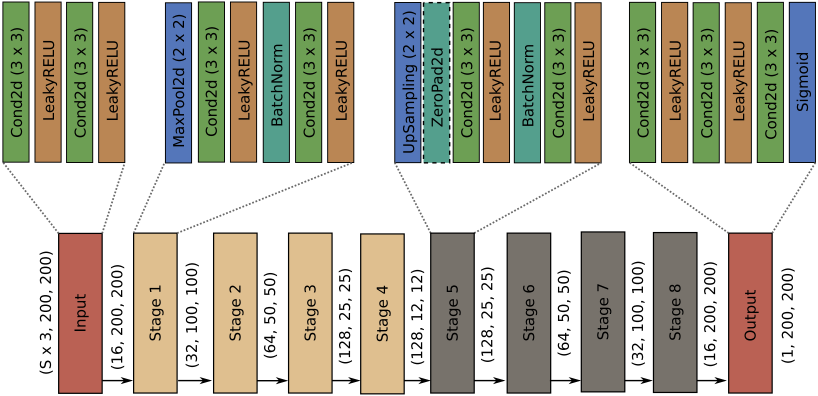
Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004).
Image quality assessment: from error visibility to structural similarity.
IEEE transactions on image processing, 13(4):600–612.

Yazici, Y., Foo, C.-S., Winkler, S., Yap, K.-H., and Chandrasekhar, V. (2020).
Empirical analysis of overfitting and mode drop in gan training.
In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1651–1655. IEEE.

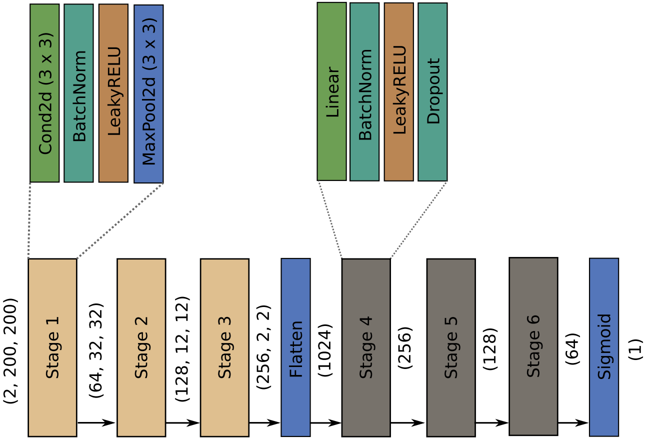
Guided-Generatic Network: Denoiser



Guided-Generatic Network: Feature Map Generator



Guided-Generatic Network: Discriminator



Guided-Generative Network: Performances

Comparison of classification thresholds:

	t threshold	Accuracy Train	AUC ROC Train	Accuracy Test	AUC ROC Test	Accuracy Global	AUC ROC Global
RNN	0.3	0.7619	0.9115	0.8122	0.9297	0.7682	0.9137
RNN	0.4	0.8085	0.9115	0.8456	0.9297	0.8131	0.9137
RNN	0.5	0.8281	0.9115	**0.8519**	0.9297	0.8311	0.9137
RNN	0.6	0.8329	0.9115	0.8385	0.9297	0.8336	0.9137
RNN	...	...	...	...	...	...	...
GGN	...	...	...	...	...	...	...
GGN	0.8	0.8809	0.9735	0.7709	0.8422	0.8672	0.9571
GGN	0.9	0.9067	0.9735	0.7858	0.8422	0.8916	0.9571
GGN	0.95	0.9204	0.9735	0.8013	0.8422	0.9055	0.9571
GGN	0.98	0.9201	0.9735	**0.8216**	0.8422	0.9078	0.9571

Perception Model : Validation experiment

Experiment parameters

- 28 selected images (threshold before 10 000 samples)
- Reconstructed images with thresholds predicted by the RNN and GGN model

Perception Model : Validation experiment

Experiment parameters

- 28 selected images (threshold before 10 000 samples)
- Reconstructed images with thresholds predicted by the RNN and GGN model

Perception Model : Results

Number of images and percentage of images considered as not noisy:

	RNN - SVD Entropy	GGN
Images	403/476	419/476
Pourcentage	85%	88%