Transfer Learning of CNNs for Texture Synthesis and Visual Recognition in Artistic Images

LASTIG Seminar

Nicolas Gonthier

November 18th 2022

Introduction

- Art Analysis

Introduction

- Art Analysis

- Texture Synthesis

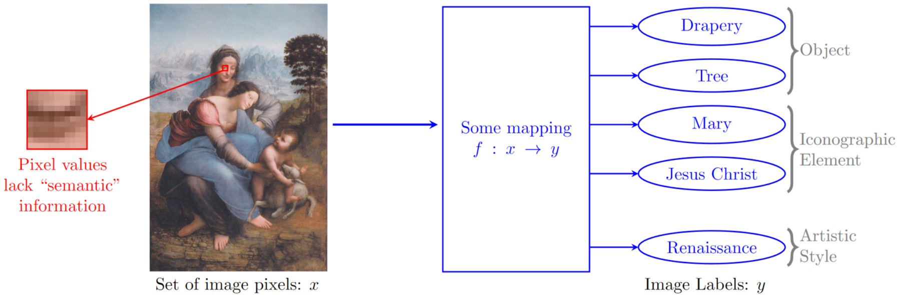
Outline

Introduction

Image Representation

How to obtain “good” image representations for image analysis and image synthesis ?

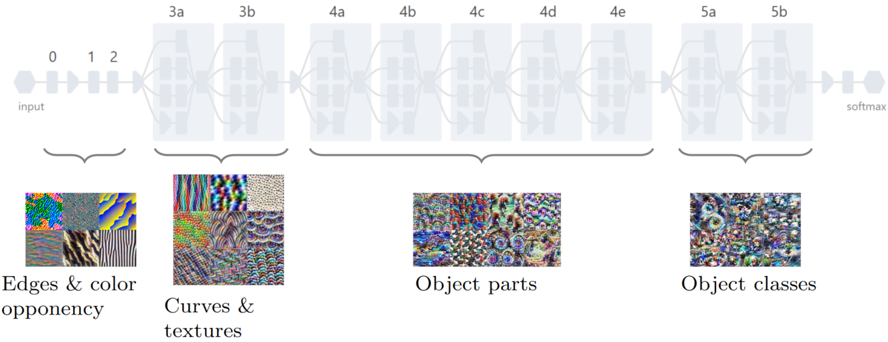
- Central problem in computer vision



- Transfer Learning of the parameters of a model f trained with supervised methods

Convolutional Neural Network (CNN)

- Feed-forward artificial neural network
- Use of convolutions
- Trained by stochastic gradient descent



- The CNN learns powerful internal representations during training
- Given an input image, one can extract these internal representations

Introduction to Transfer learning

Definition: Training a machine learning algorithm on a particular task while using knowledge the algorithm has already learned on a previous and related task.

Different Transfer Learning Approaches of CNNs

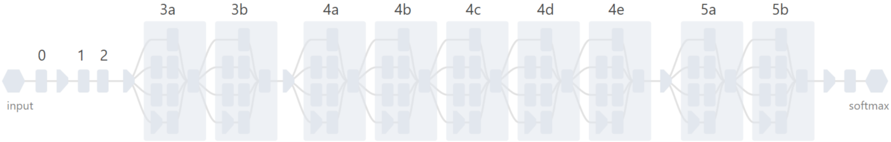
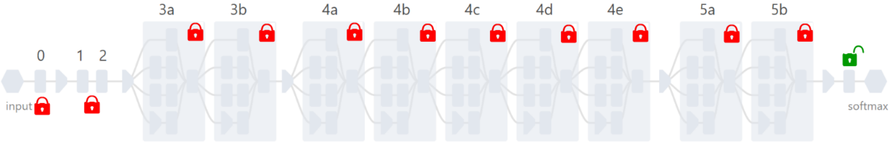


Figure: Convolutional Neural Network InceptionV1 model [Szegedy et al., 2015]

- Off-the-shelf Feature Extraction [Donahue et al., 2014]
- Fine-Tuning [Girshick et al., 2014]
- Training from scratch the same architecture

Off-the-shelf Feature Extraction



Pretrained on ImageNet New

Used by us for:

- Weakly Supervised Object Detection task
- Classification
- Texture Synthesis

Fine-Tuning

Pretrained on ImageNet

New

Used for:

- Classification

Training from scratch

Random initialization

Used for:

- Classification

Multiple Instance Model for Weakly Supervised Object Detection in Artworks

Weakly Supervised Object Detection Task Definition

Classification

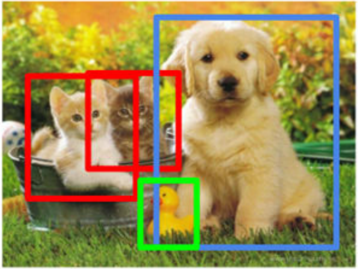
CAT

Weakly Supervised Object Detection Task Definition

Classification

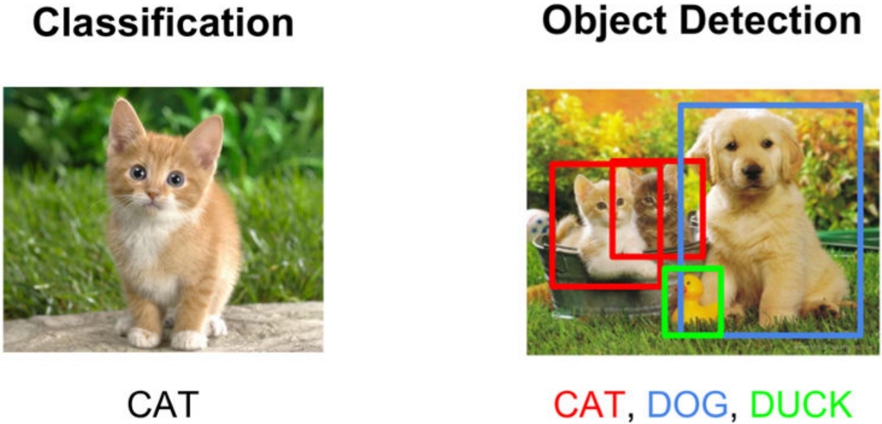
CAT

Object Detection



CAT, DOG, DUCK

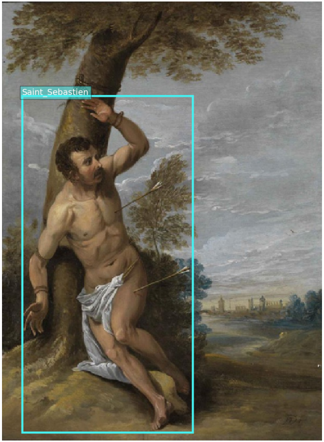
Weakly Supervised Object Detection Task Definition



Weakly Supervised Object Detection : only image level annotation during training

Motivation

Help to search artwork databases.
We would like to **localize** the object of interest



Saint Sebastian

Saint Sebastian

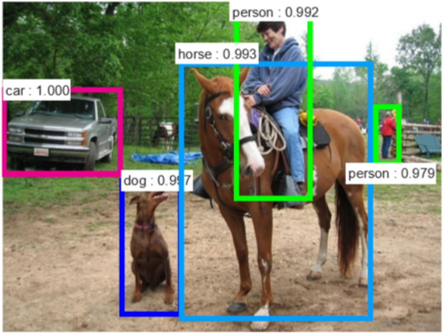
Motivation II

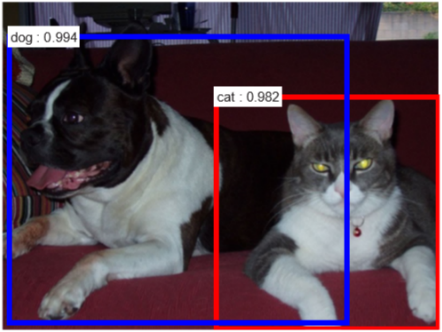
- Use only **image level annotation** → **Weakly supervised** setup
- Fast → No Fine Tuning
- Recognize new classes (not available in photography)

Figure: Example images from our IconArt database, for the Saint Sebastian category.

Transfer of a CNN

Use a Faster R-CNN network [Ren et al., 2015] **pre-trained on photography** as an off-the-shelf features extractor





Multiple Instance Learning

To solve this weakly supervised problem, we use the **Multiple Instance Learning** paradigm → Regions of an image = bag of elements

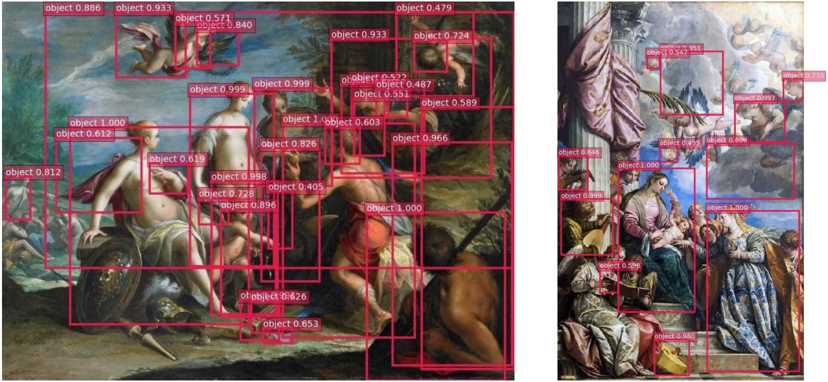


Figure: Some of the regions of interest generated by the region proposal part (RPN) of Faster R-CNN.

Multiple Instance Learning

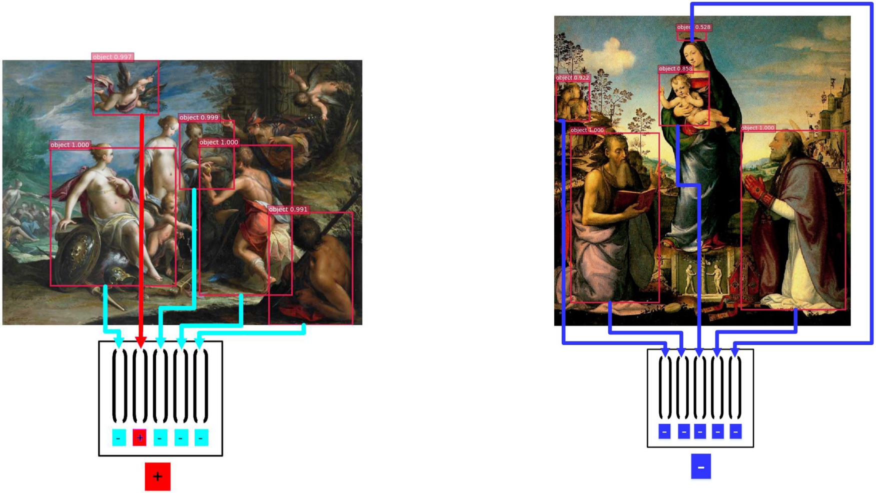
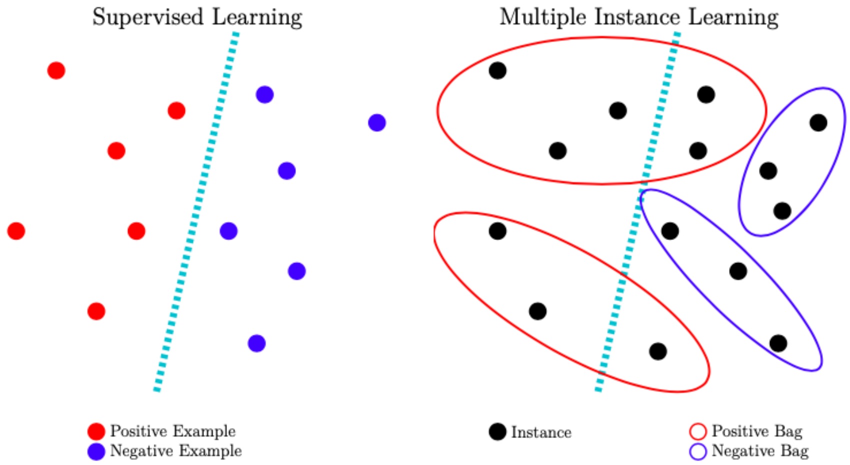
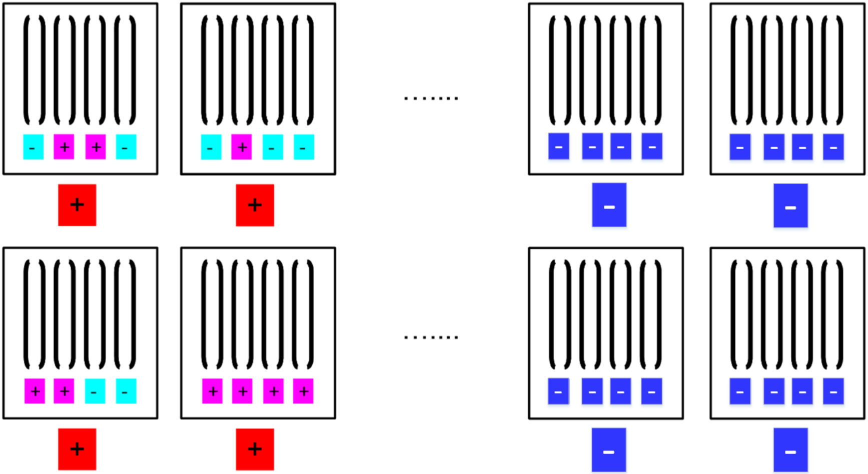


Figure: Illustration of positive and negative sets of detections (bounding boxes) for the *angel* category.

Multiple Instance Learning



Multiple Instance Learning



How to find the positive vectors in each positive bag?

How to choose the right region ?

- Classical MIL classifier: mi-SVM and MI-SVM [Andrews et al., 2003]
- Weakly Fine-Tuning the whole CNN: WSDDN, SPN and PCL [Bilen and Vedaldi, 2016, Zhu et al., 2017, Tang et al., 2018]
- Use the **highest objectness score** region: MAX [Crowley and Zisserman, 2016] and MAXA [Our]
- Use extra data from other domains: DT+ PL [Inoue et al., 2018]

Our Model: MI-max, a linear model

For each image i, we have:

$\left\{X_{i,k}\right\}_{\{1..K\}}$ feature vectors

$y_i = \pm 1$ a label

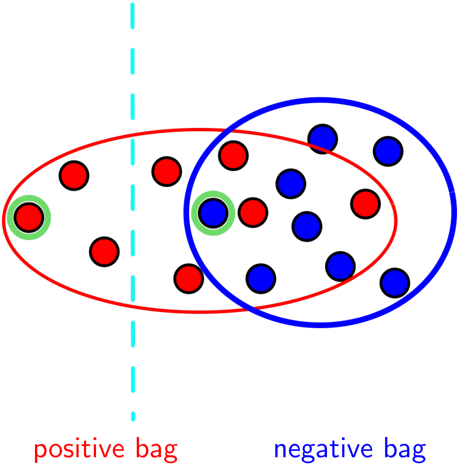
We look for $w \in \mathbb{R}^M$, $b \in \mathbb{R}$ minimizing:

$$\mathcal{L}(w,b) = \underbrace{\sum_{i=1}^{N} \frac{-y_i}{n_{y_i}} Tanh\left\{\max_{k\in\{1..K\}}\left(w^T X_{i,k} + b\right)\right\}}_{\text{classification loss}} \qquad \underbrace{+C * \|w\|^2}_{\text{regularisation term}} \qquad (1)$$

Simplified version of MI-SVM [Andrews et al., 2003]

Can be seen as a neural network without hidden layer [Zhou and Zhang, 2002]

Our Model: MI-max



- positive instance
- negative instance
- Instance used during training step

From MIL to WSOD

Use the objectness score $s_{i,k}$ of each Region of Interest.

$$\mathcal{L}^s(w,b) = \sum_{i=1}^{N} \frac{-y_i}{n_{y_i}} Tanh\left\{ \max_{k \in \{1..K\}} \left(\left(s_{i,k} + \epsilon \right) \left(w^T X_{i,k} + b \right) \right) \right\} + C * \|w\|^2 \tag{2}$$

With $\epsilon \geq 0$.

We do r restarts, and select the best couple $(w^\star, b^\star)$.

Test time score for a region x:

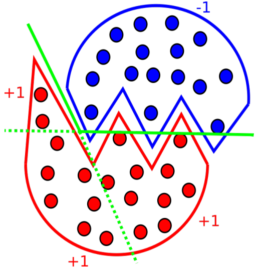
$$S(x) = Tanh\{(s(x) + \epsilon)\left(w^{\star T} x + b^\star\right)\} \tag{3}$$

Polyhedral MI-max model

Learn r hyperplanes in parallel:

$$f_w = \sum_{i=1}^{N} \frac{-y_i}{n_{y_i}} Tanh \left\{ \max_{k \in \{1..K\}} \left(s_{i,k} + \epsilon \right) \max_{j \in \{1...r\}} \left(\left(W_j^T X_{i,k} + b_j \right) \right) \right\} \qquad (4)$$

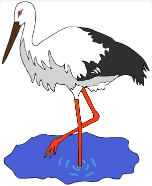
Polyhedral separability



Detection evaluation on Artistic Datasets

Watercolor2k

Comic2k



Clipart1k

[Inoue et al., 2018]

PeopleArt

[Westlake et al., 2016]

CASPA paintings

[Thomas and Kovashka, 2018]

IconArt

[Our]

Figure: Example images from the 6 art datasets used for evaluating the weakly supervised object detection.

Detection evaluation on Artistic Datasets II

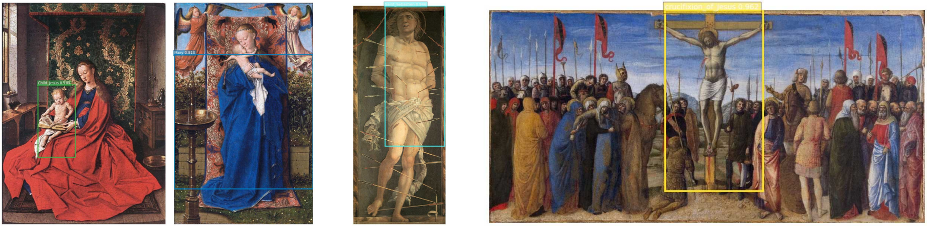
Table: Detection Mean Average Precision (%) with an IoU ≥ 0.5. Comparison on six art datasets of the proposed MI-max and Polyhedral MI-max methods to alternative approaches. The semi-supervised method is highlighted in green. The best weakly supervised method compared to others is highlighted in red.

Network	Method	Model	People-Art	Watercolor2k	Clipart1k	Comic2k	CASPA paintings	IconArt
SSD	Semi-supervised with DA	DT+PL	•	**54.3***	**46.0***	**54.3***	•	•
VGG16-IM	Weakly supervised fine-tuning	WSDDN	•	12.7	4.4	12.7	•	•
		SPN	10.0	7.1	3.8	1.2	0.7	7.7
		PCL	3.4	0.0	1.2	0.0	0.0	5.9
RES-152-COCO	Off-the-shelf Features extraction	MAX	25.9	34.3	16.9	11.9	9.8	3.7
		MAXA [Our]	48.9	43.9	22.0	19.8	14.6	12.0
		MI-SVM	13.3	21.8	19.3	13.0	2.5	4.0
		mi-SVM	5.6	5.3	6.2	4.6	1.2	2.8
		MI-max [Our]	55.5 ± 1.0	**49.5** ± 0.9	**38.4** ± 0.8	**27.0** ± 0.8	**16.2** ± 0.4	12.0 ± 0.9
		Polyhedral MI-max [Our]	**58.3** ± 1.2	46.6 ± 1.3	30.5 ± 2.3	23.3 ± 1.6	14.4 ± 0.7	**13.0** ± 2.2

Successful detections on CASPA paintings

Figure: Successful examples of animal detection using Polyhedral MI-max on CASPA paintings test set (there is no “person” class in the training set). We only show boxes whose scores are over 0.75, except for the elephant image.

Successful detections on IconArt dataset



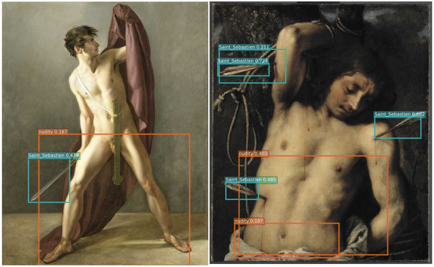
Jesus Child | Mary | Saint Sebastian | Crucifixion

Figure: Successful examples of detection of iconographic characters using Polyhedral MI-max on IconArt test set. We only show boxes whose scores are over 0.75.

Failure examples I

- Discriminative elements

Without score

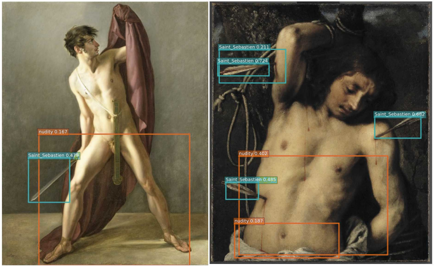


Saint Sebastian Nudity

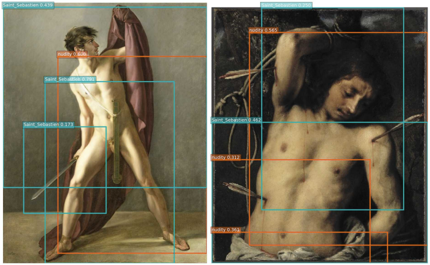
Failure examples I

- Discriminative elements

Without score



With score

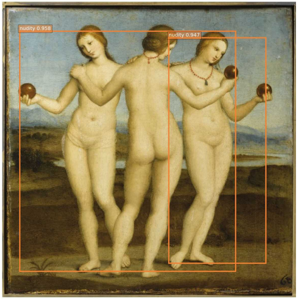


Saint Sebastian Nudity

Failure examples II

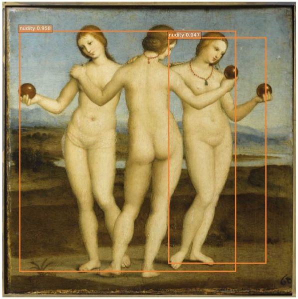
- Group of objects

  Nudity

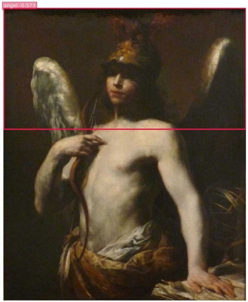


Failure examples II

- Group of objects
  Nudity

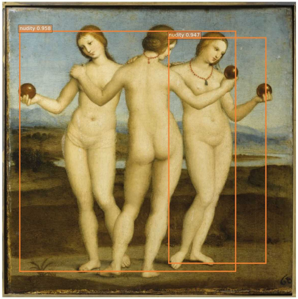


- Missing mode
  Angel score: -0.573

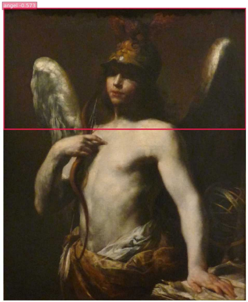


Failure examples II

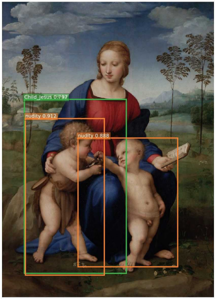
- Group of objects
  Nudity



- Missing mode
  Angel score: -0.573



- Confusing images
  Jesus Child Nudity



Cross Modalities Knowledge Transfer

Table: Mean AP (%) at IuO ⩾0.5 for the common classes between the source and target sets with the Polyhedral MI-max model. The results in parentheses, is the mean performance obtained by learning the detection on the same set (modality).

Source set \ Target set	PeopleArt	Watercolor2k	Comic2k	Clipart1k	CASPA paintings
PeopleArt	-	60.0 (59.2)	42.1 (39.5)	54.3 (55.4)	/
Watercolor2k	56.0 (57.3)	-	23.1 (24.1)	11.2 (24.6)	13.8 (18.3)
Comic2k	48.9 (57.3)	42.4 (46.6)	-	7.2 (24.6)	12.5 (18.3)
Clipart1k	52.0 (57.3)	36.7 (46.6)	19.6 (24.1)	-	7.7 (13.6)
CASPA paintings	/	27.5 (39.0)	9.9 (18.1)	4.2 (12.5)	-

Conclusion

Conclusion:

- Good results on a difficult task
- Fast solution
- The learned classifier can be transferred between modalities

Analyzing CNNs trained for Art classification tasks

Motivation

Transfer Learning of Deep Learning model trained on natural images has become a de facto method for art analysis applications:

- Replica [Seguin, 2018] for visual similarity search
- Oxford Painting Search [Crowley et al., 2018] for semantics recognition of arbitrary objects
- Style, artist or genre recognition [Lecoutre et al., 2017, Strezoski and Worring, 2017, Cetinic et al., 2018, Chen and Yang, 2019, Deng et al., 2020]

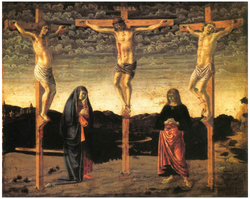
What are the effects of transfer learning for artistic images ?

Considered datasets

Name	Task	Number of classes	N_T	% for test set
ImageNet [Russakovsky et al., 2015]	Image Classification	1000	1.3M	~ 10%

Considered datasets

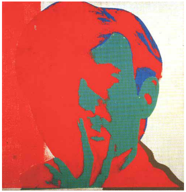
Name	Task	Number of classes	N_T	% for test set
ImageNet [Russakovsky et al., 2015]	Image Classification	1000	1.3M	~ 10%
RASTA [Lecoutre et al., 2017]	Style classification	25	80,000	20%



Early Renaissance

Impressionism

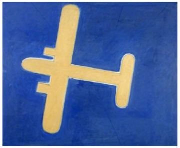
Ukiyo-e



Pop Art

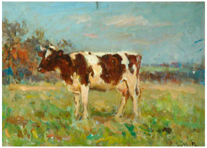
Considered datasets

Name	Task	Number of classes	N_T	% for test set
ImageNet [Russakovsky et al., 2015]	Image Classification	1000	1.3M	~ 10%
RASTA [Lecoutre et al., 2017]	Style classification	25	80,000	20%
Paintings [Crowley and Zisserman, 2014]	Object classification	10	8629	50%



Plane

Sheep



Cow

Train

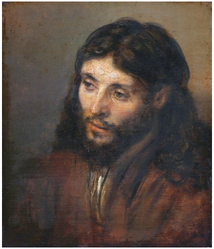
Considered datasets

Name	Task	Number of classes	N_T	% for test set
ImageNet [Russakovsky et al., 2015]	Image Classification	1000	1.3M	~ 10%
RASTA [Lecoutre et al., 2017]	Style classification	25	80,000	20%
Paintings [Crowley and Zisserman, 2014]	Object classification	10	8629	50%
IconArt	Object classification	7	5955	50%

Crucifixion | Mary

Saint Sebastian

Mary |Jesus Child | Angel



Performances of the different transfer methods

Method	Top-1	Top-3	Top-5
Off-the-shelf Feature extraction with InceptionV1 pretrained on ImageNet	30.95	58.71	74.10
Fine-Tuning of InceptionV1 pretrained on ImageNet	**55.18**	**82.25**	**91.06**
InceptionV1 trained from scratch	45.29	73.44	84.67

Table: Top-k accuracies (%) on RASTA dataset [Lecoutre et al., 2017] for different methods.

Similar results in [Cetinic et al., 2018, Sabatelli et al., 2018]

Feature Visualization

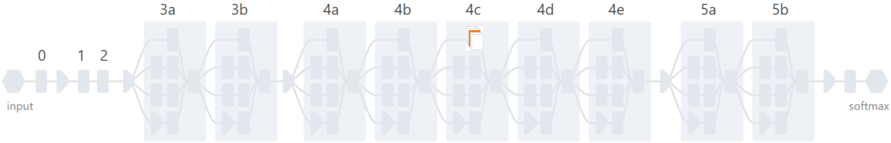


Figure: One individual channel is highlighted in orange.

- Feature Visualization by Optimization
- Maximal Activation Images

Feature Visualization by Optimization

Synthesize an image by maximizing the channel activation:
"Optimized Image"

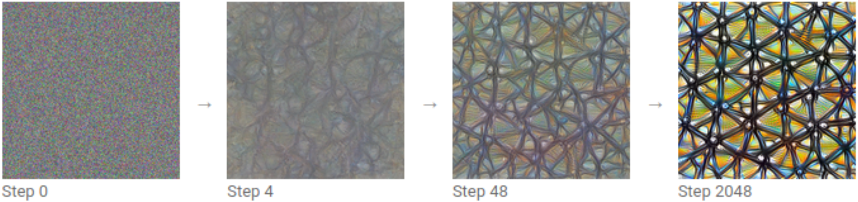


Figure: Feature Visualization by Optimization [Olah et al., 2017].

Feature Visualization by Optimization

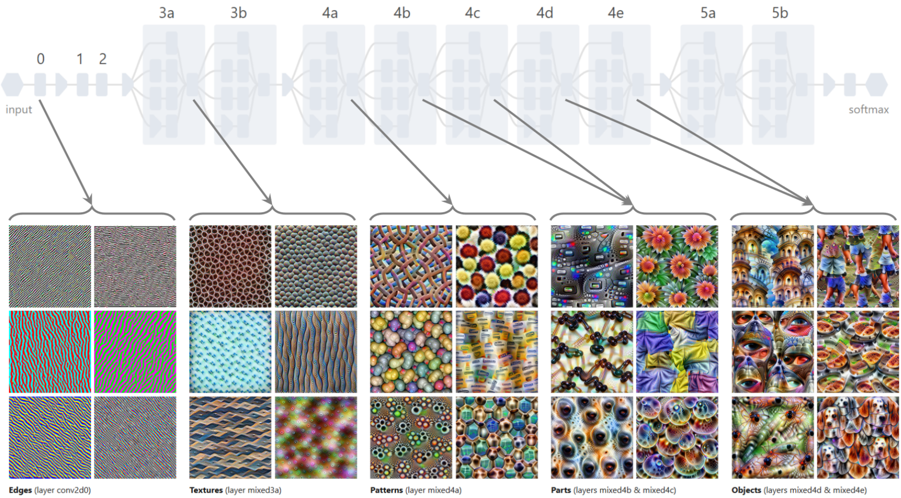
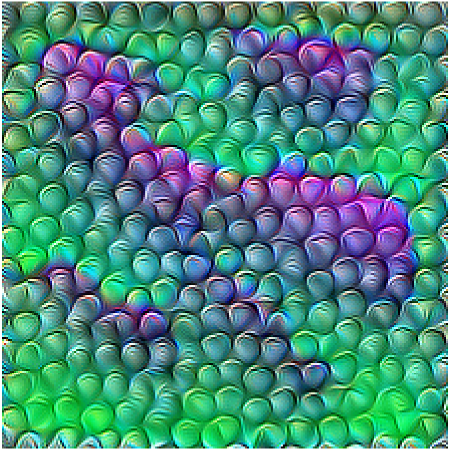


Figure: Feature Visualization by Optimization [Olah et al., 2017].

Low-level layers are not modified.

Imagenet Pretrained



RASTA Fine Tuned

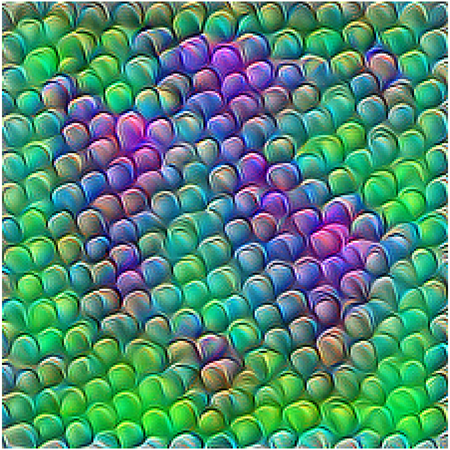


Figure: Optimized Images for channel mixed3a_3x3_pre_relu:12

Some detectors are already useful.

Imagenet Pretrained

RASTA Fine Tuned

Figure: Optimized Images for channel mixed4b_3x3_bottleneck_pre_relu:35

Some detectors are already useful.

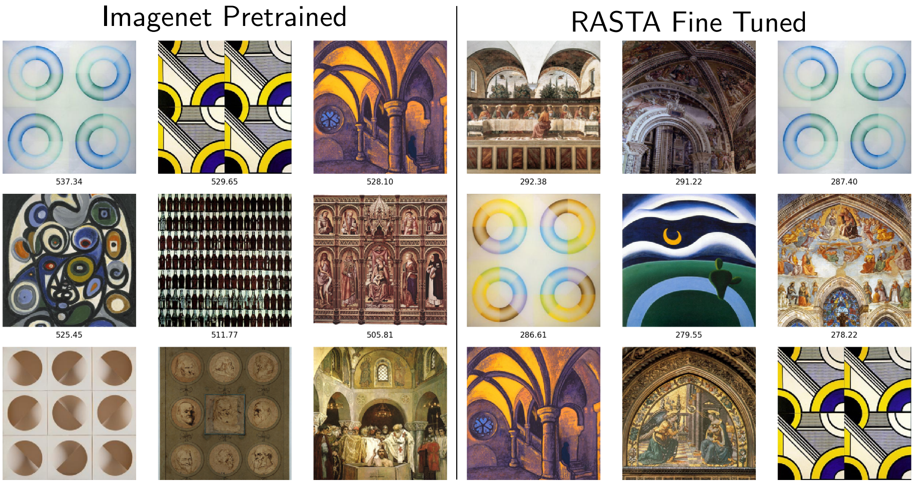


Figure: Maximal Activation Examples for channel mixed4b_3x3_bottleneck_pre_relu:35

Mid-level layers are adapted to the new dataset.

Imagenet Pretrained

RASTA Fine Tuned

Figure: Optimized Images for channel mixed4c_3x3 _bottleneck_pre_relu:78

Mid-level layers are adapted to the new dataset.

Figure: Maximal Activation Examples for channel mixed4c_3x3 _bottleneck_pre_relu:78

Mid-level layers are adapted to the new dataset.

Imagenet Pretrained | RASTA Fine Tuned

Figure: Optimized Images for channel mixed4d_3x3_pre_relu:52

Mid-level layers are adapted to the new dataset.

Imagenet Pretrained

RASTA Fine Tuned

Figure: Maximal Activation Examples for channel mixed4d_pool_reduce_pre_relu:63

The learned features have a high variability.

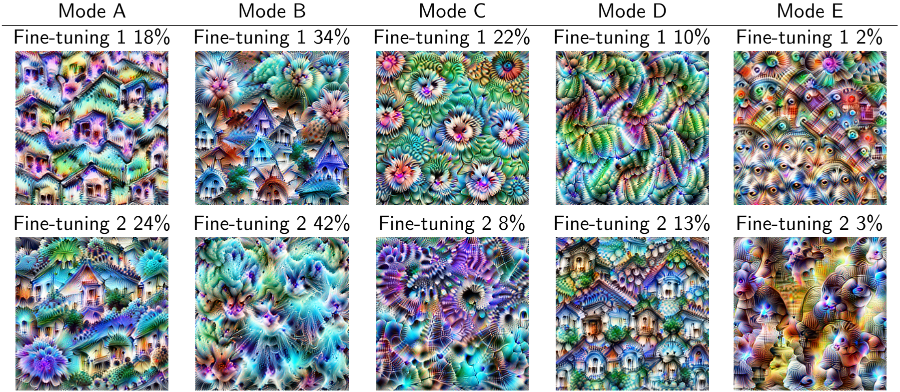


Figure: Same channel with different training (mixed4d_3x3_pre_relu:52), the overlapping ratio is displayed in %. Each mode corresponds to a different set of hyperparameters.

High-level layers cluster images of the same class.

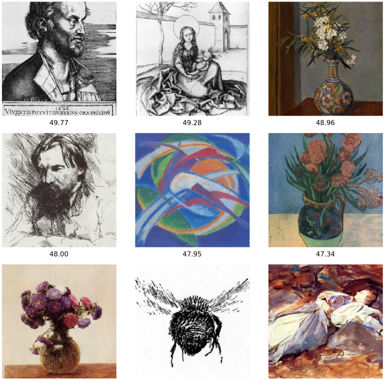
Imagenet Pretrained

RASTA Fine Tuned

Figure: Optimized Images for channel mixed5b_pool _reduce_pre_relu:92.

High-level layers cluster images of the same class.

Imagenet Pretrained



Realism 17%
Post-Impressionism 10%
Neoclassicism 10%

RASTA Fine Tuned

Ukiyo-e 82 %
Northern_Renaissance 14 %
Early_Renaissance 3 %

Figure: Maximal Activation Examples for channel mixed5b_pool_reduce_pre_relu:92 with the Top 100 composition.

The feature visualization is less interpretable with a training from scratch.

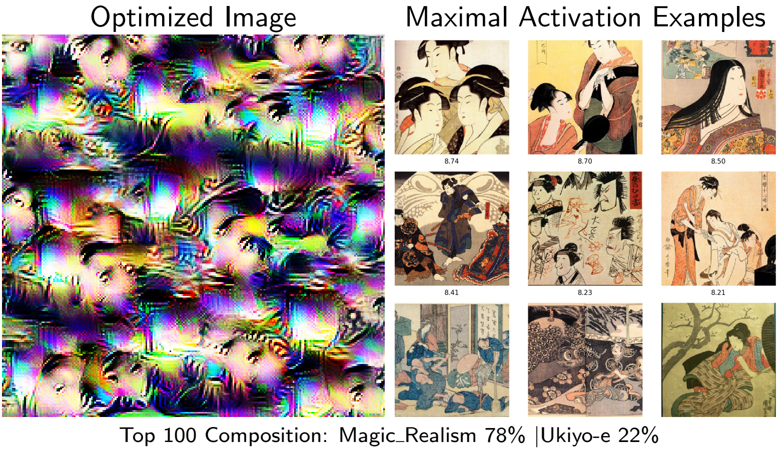
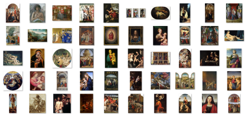
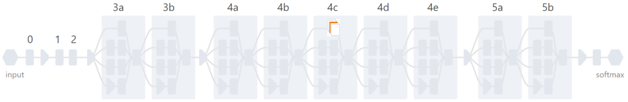


Figure: Optimized Image and Maximal Activation Examples for channel mixed4:16 for a model trained from scratch.

Maximal Activation Images

We look at the images with the maximal activation for a particular channel.

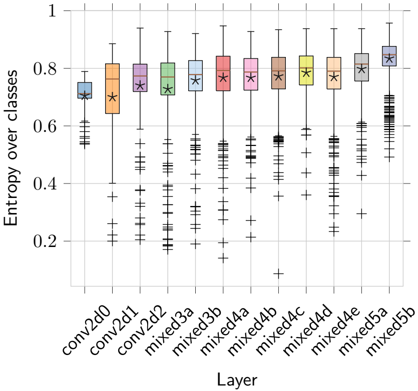




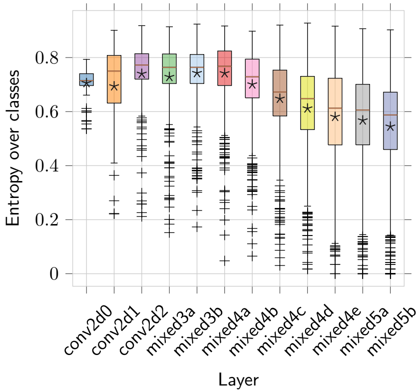
...

Compute the class entropy and the overlapping ratio (before and after fine-tuning)

Changes in the fine-tuned model.



(a) ImageNet Pretraining.



(b) Fine Tuned.

Figure: Boxplots of Entropy over classes on the top 100 maximal activation images for the model fine-tuned on RASTA. For each box, the horizontal line corresponds to the average result and the star to the median.

Overlapping ratio before and after the fine-tuning.

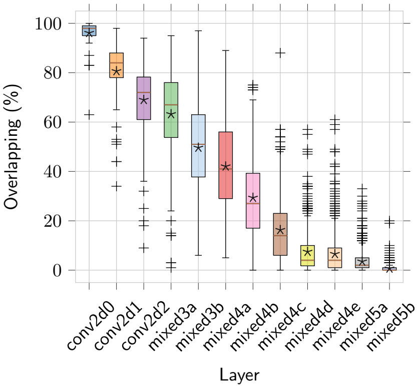
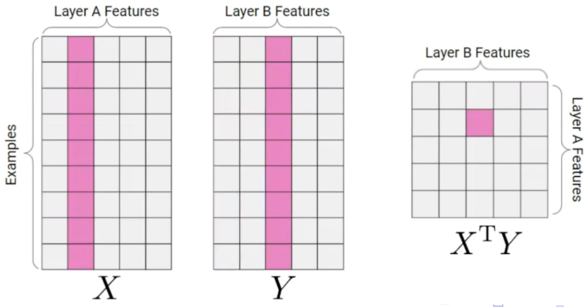


Figure: Boxplots of the overlapping ratio metrics on the top 100 maximal activation images before and after the fine-tuning on RASTA . For each box, the horizontal orange line corresponds to the average result and the star to the median. The crosses are outliers (i.e. points outside 1.5 times the interquartile range).

Networks comparison

A feature similarity index named Centered Kernel Alignment (CKA) [Cortes et al., 2012, Kornblith et al., 2019]: normalized sum of the squared dot products (similarity) between features.

$$CKA = \frac{\left\| X^{\mathrm{T}} Y \right\|_{\mathrm{F}}^{2}}{\left\| X^{\mathrm{T}} X \right\|_{\mathrm{F}} \left\| Y^{\mathrm{T}} Y \right\|_{\mathrm{F}}} \quad (5)$$



Feature Similarity between networks.

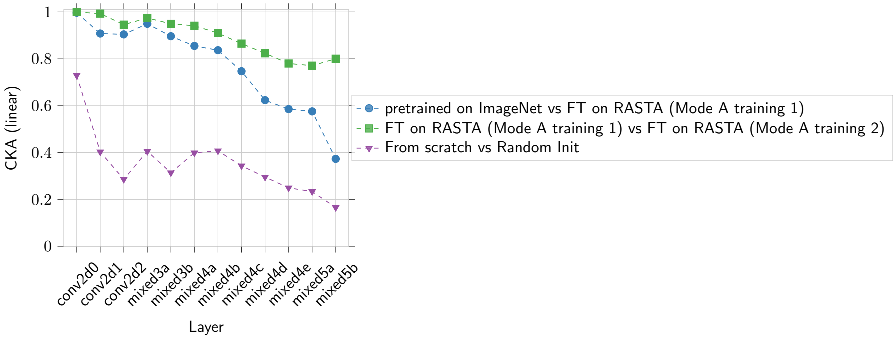


Figure: CKA (defined in eq. 5) computed on RASTA test set for different models trained or fine-tuned on RASTA train set.

- The fine-tuning will make converge the high level layers to a given neighborhood

From One Art dataset to another.

Table: Mean Average Precision on:
- Paintings [Crowley and Zisserman, 2014]
- IconArt

Method	Paintings	IconArt
Fine-Tuning of InceptionV1 pretrained on ImageNet	0.65	0.59
Fine-Tuning of InceptionV1 pretrained on ImageNet and RASTA	**0.66**	**0.67**

Similar results in [Sabatelli et al., 2018]

From One Art dataset to another.

Table: Mean Average Precision on:
- Paintings [Crowley and Zisserman, 2014]
- IconArt

Method	Paintings	IconArt
Fine-Tuning of InceptionV1 pretrained on ImageNet	0.65	0.59
Fine-Tuning of InceptionV1 pretrained on ImageNet and RASTA	**0.66**	**0.67**

Similar results in [Sabatelli et al., 2018]

Table: Mean CKA between the model pretained on ImageNet and the one fine-tuned on Paintings [Crowley and Zisserman, 2014] or IconArt.

mean CKA of a pair of nets	Paintings	IconArt
Pretrained on ImageNet & FT on small art dataset	0.91	0.90
Pretrained on ImageNet & FT on RASTA + FT on small dataset	0.76	0.73

Some detectors may be adapted to the IconArt dataset.

Imagenet Pretrained | IconArt Fine Tuned | RASTA Fine Tuned | RASTA and IconArt Fine Tuned

Figure: Optimized Images for channel mixed4c_3x3_bottleneck_pre_relu:78.

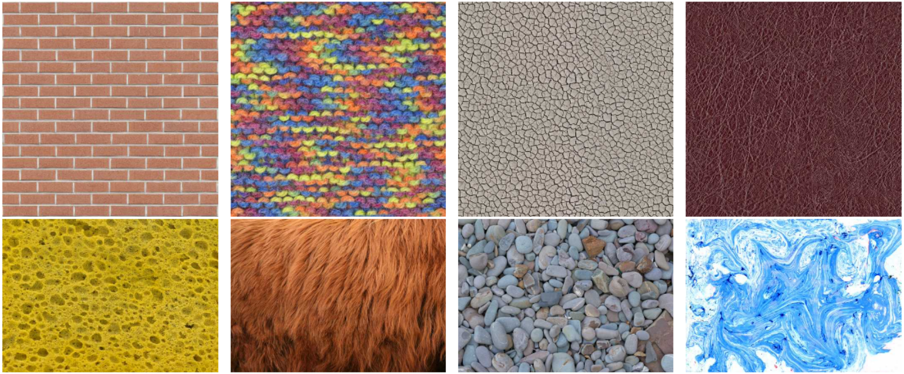
Conclusion

Conclusion:

- Fine Tuning an ImageNet pretrained model provides better results then other transfer methods
- Pretraining on ImageNet plus Artistic dataset may help for art analysis application
- Feature Visualization helps to understand what happens during fine-tuning

Texture Synthesis with CNNs

Texture Image



Texture Synthesis with exemplar

Definition: Given a reference texture, texture synthesis aims at producing more texture images which are "visually similar" to the reference.

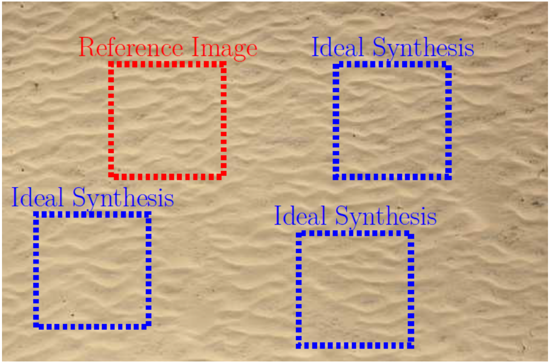
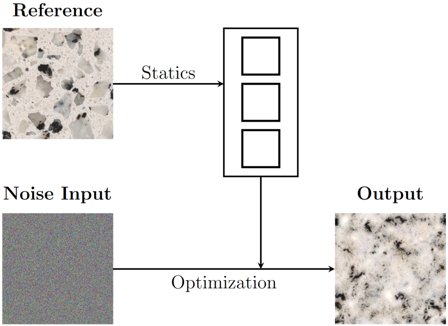


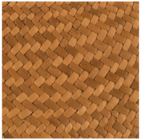
Figure: Examplar of a reference texture with ideal synthesis.

Texture Synthesis with CNNs [Gatys et al., 2015]

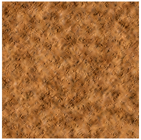
Statistics-based methods.



Reference Image

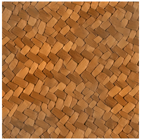


[Heeger and Bergen, 1995]



[Portilla and Simoncelli, 2000]

[Gatys et al., 2015]

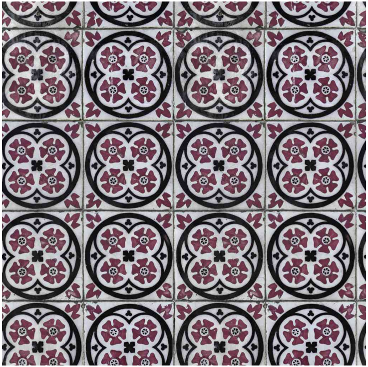


Motivation

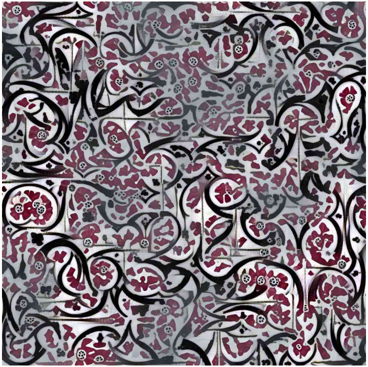
Limitations of [Gatys et al., 2015]:

- Large scale regularity especially in high resolution image
- How to model an image

Reference

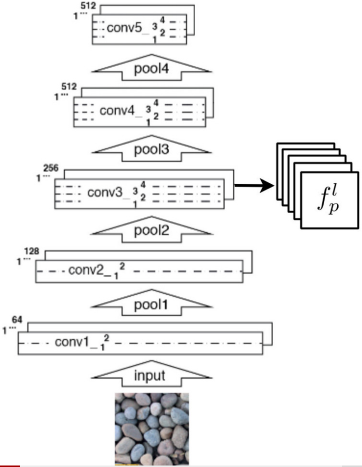


[Gatys et al., 2015]



Texture Model [Gatys et al., 2015]

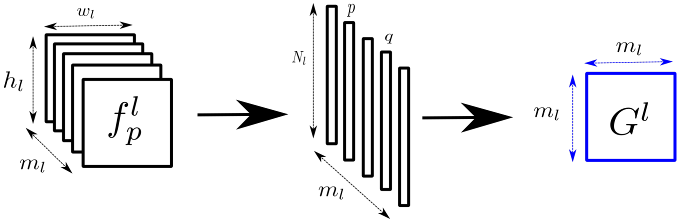
Texture features: Given an exemplar texture $I \in \mathbb{R}^N$, we compute the m_l feature maps $f_p^l \in \mathbb{R}^{h_l \times w_l}$ of the l-th layer of a VGG19 **pretrained** on ImageNet



Texture Model

We compute G^l the Gram matrix [Gatys et al., 2015] of the feature maps of the layer l:

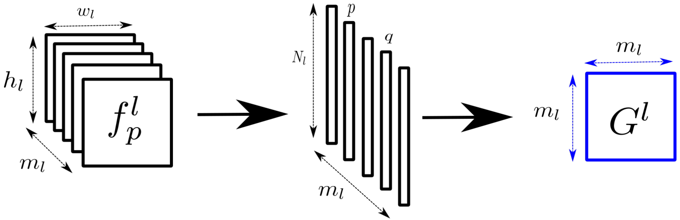
$$G^l_{p,q} = \frac{1}{N_l^2} \langle f^l_p | f^l_q \rangle$$



Texture Model

We compute G^l the Gram matrix [Gatys et al., 2015] of the feature maps of the layer l:

$$G^l_{p,q} = \frac{1}{N_l^2}\langle f^l_p | f^l_q \rangle$$



We synthesis $\tilde{I}$ by minimizing :

$$\mathcal{L}(I, \tilde{I}) = \sum_{l=1}^{L} \omega_l \| G^l - \tilde{G}^l \|^2_{\mathcal{F}} \tag{6}$$

by **gradient descent** with back-propagation through the CNN.

Improvements of the method

- Speed Up the synthesis:
  - Feed forward generators [Ulyanov et al., 2016, Ulyanov et al., 2017, Risser, 2020]
  - GAN [Jetchev et al., 2016, Darzi et al., 2020]
- Add a corrective term to the loss function:

$$\mathcal{L} = \mathcal{L}_{Gram} + \beta \mathcal{L}_{corrective}$$

  - Spectrum constraints [Liu et al., 2016]
  - Shift correlation [Berger and Memisevic, 2017]
  - Multiple constraints (total variation, autocorrelation, extended correlation) [Sendik and Cohen-Or, 2017]
  - Histogram matching [Risser et al., 2017, Heitz et al., 2020, Risser, 2020]
- High resolution images
  - Gaussian Pyramid [Snelgrove, 2017]

Multi-resolution strategy

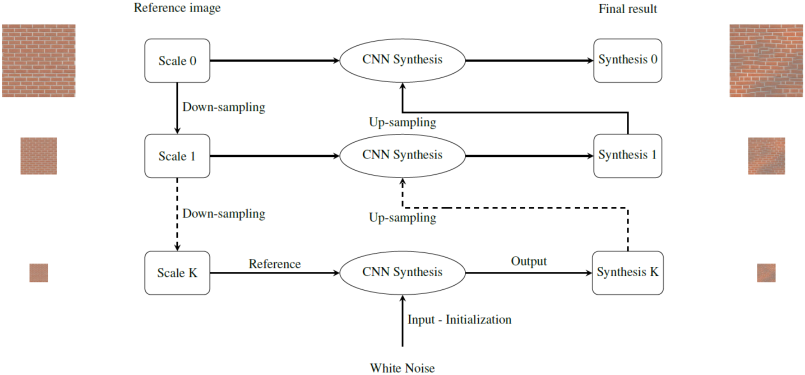


Figure: Illustration of synthesis results at K different scales, named **MRInit**.

Classical idea presented in e.g.

[Kwatra et al., 2005, Risser et al., 2017, Galerne et al., 2018, Risser, 2020].
Alternative multi-resolution framework:
[Heeger and Bergen, 1995, Portilla and Simoncelli, 2000, Snelgrove, 2017].

Spectrum Transferring [Liu et al., 2016]

We impose the spectrum (modulus of the Fourier transform) of I to $\tilde{I}$ by adding this term to the loss function:

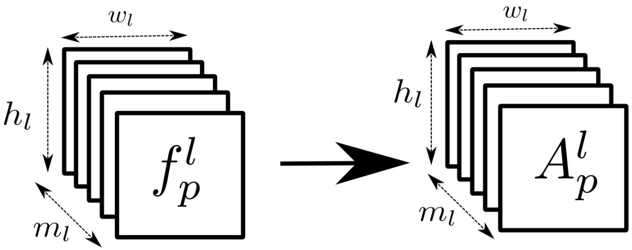
$$\mathcal{L}_{spe} = \frac{1}{2N} \| |\mathcal{F}(\tilde{I})| - |\mathcal{F}(I)| \|^2, \tag{7}$$

Used by [Galerne et al., 2011, Tartavel et al., 2015].

Autocorrelation of the feature maps

We replace the Gram Matrix by the autocorrelation of each of the feature map p. We impose the squared modulus of the Fourier Transform (equivalent to the autocorrelation):

$$A_p^l = \frac{1}{N_l^2} \mid \mathcal{F}(f_p^l) \mid^2 \tag{8}$$



Idea inspired by [Portilla and Simoncelli, 2000]

Parameters Setup

For the experiments, all the images are of size 1024×1024.
We will compare different methods:

- [Gatys et al., 2015]
- Multi-resolution strategy of [Snelgrove, 2017]
- Gram with our multi-resolution strategy (MRInit)
- Gram + Spectrum Image [Liu et al., 2016] with our multi-resolution strategy
- Autocorrelation with our multi-resolution strategy

With $K = 2$ for our method and $K = 3$ for [Snelgrove, 2017].

Reference

[Gatys et al., 2015]

[Snelgrove, 2017]

Gram + MRInit [Our]

Gram + Spectrum + MRInit [Our]

Autocorr + MRInit [Our]

Reference

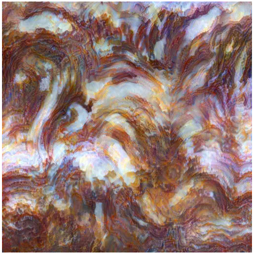
[Gatys et al., 2015]

[Snelgrove, 2017]

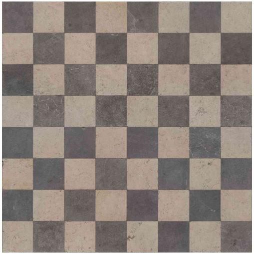
Gram + MRInit [Our]

Gram + Spectrum + MRInit [Our]

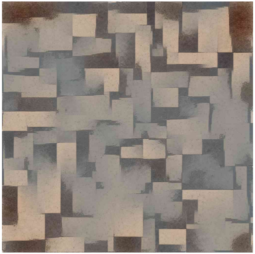
Autocorr + MRInit [Our]



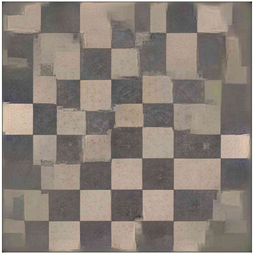
Reference



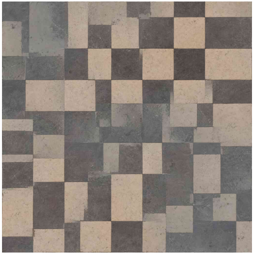
[Gatys et al., 2015]



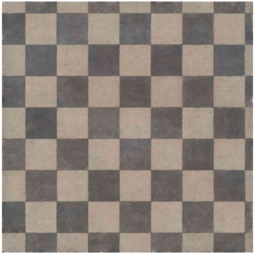
[Snelgrove, 2017]



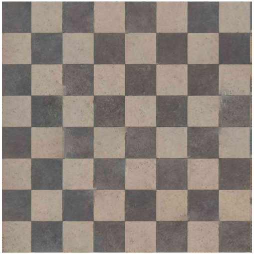
Gram + MRInit [Our]



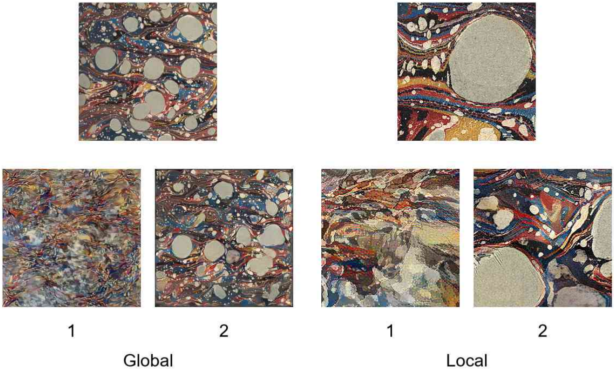
Gram + Spectrum + MRInit [Our]



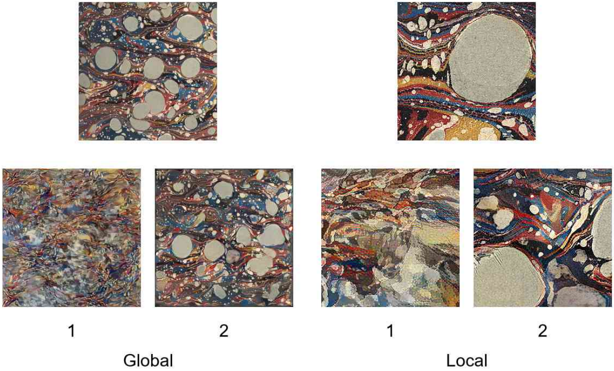
Autocorr + MRInit [Our]



User Study : Perceptual Test



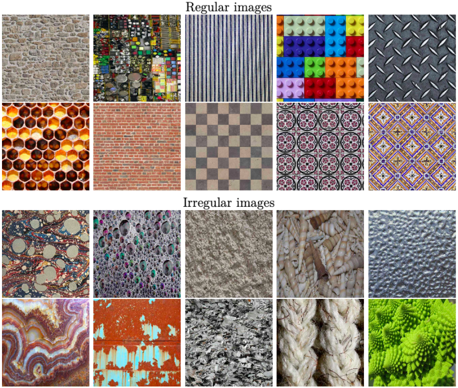
User Study : Perceptual Test



Select the best synthesis for each scale.

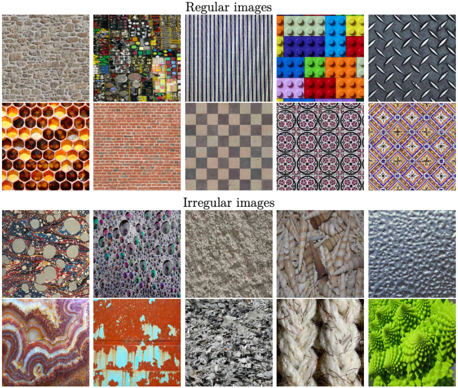
User Study : Perceptual Test

We collected 3170 votes between pairs of images from 20 different reference images.



User Study : Perceptual Test

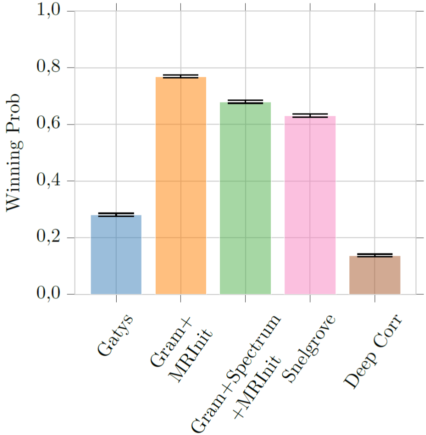
We collected 3170 votes between pairs of images from 20 different reference images.



We remove the Autocorrelation method due to a high variability in the results.

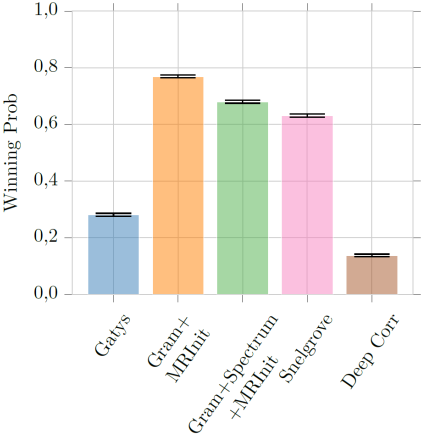
Perceptual Test Results

General performance:

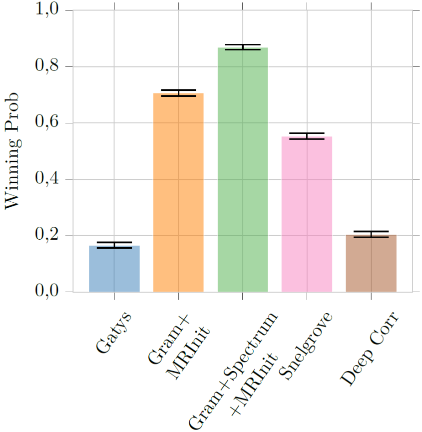


Perceptual Test Results

General performance:



Global scale for regular textures:



Conclusion

- We propose a simple way to synthesise high definition images based on [Gatys et al., 2015]
- The results are improved with new designs of the loss function

Conclusion

Thank you for your attention.

References I

- [Andrews et al., 2003] Andrews, S., Tsochantaridis, I., and Hofmann, T. (2003).
  Support vector machines for multiple-instance learning.
  In *Advances in neural information processing systems*, pages 577–584.
- [Berger and Memisevic, 2017] Berger, G. and Memisevic, R. (2017).
  Incorporating long-range consistency in CNN-based texture generation.
  In *ICLR*.
- [Bilen and Vedaldi, 2016] Bilen, H. and Vedaldi, A. (2016).
  Weakly Supervised Deep Detection Networks.
  In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2846–2854. IEEE.
- [Cetinic et al., 2018] Cetinic, E., Lipic, T., and Grgic, S. (2018).
  Fine-tuning Convolutional Neural Networks for Fine Art Classification.
  Expert Systems with Applications.
- [Chen and Yang, 2019] Chen, L. and Yang, J. (2019).
  Recognizing the Style of Visual Arts via Adaptive Cross-layer Correlation.
  In *Proceedings of the 27th ACM International Conference on Multimedia*, page 9, Nice,France.
- [Cortes et al., 2012] Cortes, C., Mohri, M., and Rostamizadeh, A. (2012).
  Algorithms for Learning Kernels Based on Centered Alignment.
  Journal of Machine Learning Research, 13:795–828.
- [Crowley et al., 2018] Crowley, E. J., Cot, E., and Zisserman, A. (2018).
  Oxford Painting Search.
- [Crowley and Zisserman, 2014] Crowley, E. J. and Zisserman, A. (2014).
  In search of art.
  In *Workshop at the European Conference on Computer Vision*, pages 54–70. Springer.

References II

- [Crowley and Zisserman, 2016] Crowley, E. J. and Zisserman, A. (2016).
  The Art of Detection.
  In *European Conference on Computer Vision*, pages 721–737. Springer.
- [Darzi et al., 2020] Darzi, A., Lang, I., Taklikar, A., Averbuch-Elor, H., and Avidan, S. (2020).
  Co-occurrence Based Texture Synthesis.
  arXiv:2005.08186 [cs].
  arXiv: 2005.08186 Version: 1.
- [Deng et al., 2020] Deng, Y., Tang, F., Dong, W., Ma, C., Huang, F., Deussen, O., and Xu, C. (2020).
  Exploring the Representativity of Art Paintings.
  IEEE Transactions on Multimedia, pages 1–1.
  Conference Name: IEEE Transactions on Multimedia.
- [Donahue et al., 2014] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2014).
  DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition.
  In *International conference on machine learning*, pages 647–655.
  arXiv: 1310.1531.
- [Galerne et al., 2011] Galerne, B., Gousseau, Y., and Morel, J.-M. (2011).
  Micro-Texture Synthesis by Phase Randomization.
  Image Processing On Line, 1:213–237.
- [Galerne et al., 2018] Galerne, B., Leclaire, A., and Rabin, J. (2018).
  A Texture Synthesis Model Based on Semi-discrete Optimal Transport in Patch Space.
  SIAM Journal on Imaging Sciences, 11(4):2456–2493.

References III

- [Gatys et al., 2015] Gatys, L. A., Ecker, A. S., and Bethge, M. (2015).
  Texture Synthesis Using Convolutional Neural Networks.
  In *Advances in neural information processing systems*, pages 262–270.
  arXiv: 1505.07376.
- [Girshick et al., 2014] Girshick, R. B., Donahue, J., Darrell, T., and Malik, J. (2014).
  Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation.
  In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587.
- [Heeger and Bergen, 1995] Heeger, D. J. and Bergen, J. R. (1995).
  Pyramid-based texture analysis/synthesis.
  In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 229–238. ACM.
- [Heitz et al., 2020] Heitz, E., Vanhoey, K., Chambon, T., and Belcour, L. (2020).
  Pitfalls of the Gram Loss for Neural Texture Synthesis in Light of Deep Feature Histograms.
  arXiv:2006.07229 [cs].
  arXiv: 2006.07229 Version: 2.
- [Inoue et al., 2018] Inoue, N., Furuta, R., Yamasaki, T., and Aizawa, K. (2018).
  Cross-Domain Weakly-Supervised Object Detection through Progressive Domain Adaptation.
  In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*. IEEE.
  arXiv: 1803.11365.
- [Jetchev et al., 2016] Jetchev, N., Bergmann, U., and Vollgraf, R. (2016).
  Texture Synthesis with Spatial Generative Adversarial Networks.
  arXiv:1611.08207 [cs, stat].
  arXiv: 1611.08207 Version: 4.

References IV

- [Kornblith et al., 2019] Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. (2019).
  Similarity of Neural Network Representations Revisited.
  In *International Conference on Machine Learning*, volume 97, pages 3519–3529, Long Beach, California, USA.
  arXiv: 1905.00414.
- [Kwatra et al., 2005] Kwatra, V., Essa, I., Bobick, A., and Kwatra, N. (2005).
  Texture optimization for example-based synthesis.
  ACM Transactions on Graphics, 24(3):795–802.
- [Lecoutre et al., 2017] Lecoutre, A., Negrevergne, B., and Yger, F. (2017).
  Recognizing Art Style Automatically in painting with deep learning.
  In *Asian conference on machine learning*, JMLR: Workshop and Conference Proceedings, pages 327–342.
- [Liu et al., 2016] Liu, G., Gousseau, Y., and Xia, G.-S. (2016).
  Texture Synthesis Through Convolutional Neural Networks and Spectrum Constraints.
  In *23rd International Conference on Pattern Recognition*, pages 3234–3239. IEEE.
  arXiv: 1605.01141.
- [Olah et al., 2017] Olah, C., Mordvintsev, A., and Schubert, L. (2017).
  Feature Visualization.
  Distill, 2(11):e7.
- [Portilla and Simoncelli, 2000] Portilla, J. and Simoncelli, E. P. (2000).
  A parametric texture model based on joint statistics of complex wavelet coefficients.
  International journal of computer vision, 40(1):49–70.
- [Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015).
  Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.
  Advances in neural information processing systems, pages 91–99.
  arXiv: 1506.01497.

References V

- [Risser, 2020] Risser, E. (2020).
  Optimal Textures: Fast and Robust Texture Synthesis and Style Transfer through Optimal Transport.
  arXiv:2010.14702 [cs].
  arXiv: 2010.14702 Version: 1.
- [Risser et al., 2017] Risser, E., Wilmot, P., and Barnes, C. (2017).
  Stable and Controllable Neural Texture Synthesis and Style Transfer Using Histogram Losses.
  arXiv:1701.08893 [cs].
  arXiv: 1701.08893 Version: 2.
- [Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015).
  ImageNet Large Scale Visual Recognition Challenge.
  International Journal of Computer Vision, 115(3):211–252.
  arXiv: 1409.0575.
- [Sabatelli et al., 2018] Sabatelli, M., Kestemont, M., Daelemans, W., and Geurts, P. (2018).
  Deep Transfer Learning for Art Classification Problems.
  In *Workshop on Computer Vision for Art Analysis ECCV*, pages 1–17, Munich.
- [Seguin, 2018] Seguin, B. (2018).
  The Replica Project: Building a Visual Search Engine for Art Historians.
  XRDS, 24(3):24–29.
- [Sendik and Cohen-Or, 2017] Sendik, O. and Cohen-Or, D. (2017).
  Deep correlations for texture synthesis.
  ACM Transactions on Graphics (TOG), 36(5):1–15.

References VI

- [Snelgrove, 2017] Snelgrove, X. (2017).
  High-resolution multi-scale neural texture synthesis.
  In *SIGGRAPH Asia*, pages 1–4. ACM Press.
- [Strezoski and Worring, 2017] Strezoski, G. and Worring, M. (2017).
  OmniArt: Multi-task Deep Learning for Artistic Data Analysis.
  arXiv:1708.00684 [cs].
  arXiv: 1708.00684 Version: 1.
- [Szegedy et al., 2015] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015).
  Going deeper with convolutions.
  In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.
- [Tang et al., 2018] Tang, P., Wang, X., Bai, S., Shen, W., Bai, X., Liu, W., and Yuille, A. (2018).
  PCL: Proposal Cluster Learning for Weakly Supervised Object Detection.
  IEEE transactions on pattern analysis and machine intelligence.
  arXiv: 1807.03342.
- [Tartavel et al., 2015] Tartavel, G., Gousseau, Y., and Peyré, G. (2015).
  Variational Texture Synthesis with Sparsity and Spectrum Constraints.
  Journal of Mathematical Imaging and Vision, 52(1):124–144.
- [Thomas and Kovashka, 2018] Thomas, C. and Kovashka, A. (2018).
  Artistic Object Recognition by Unsupervised Style Adaptation.
  In *Asian Conference on Computer Vision*, pages 460–476, Cham. Springer.
  arXiv: 1812.11139.

References VII

- [Ulyanov et al., 2016] Ulyanov, D., Lebedev, V., Vedaldi, A., and Lempitsky, V. (2016).
  Texture Networks: Feed-forward Synthesis of Textures and Stylized Images.
  ICML, 1(2):4.
  arXiv: 1603.03417.
- [Ulyanov et al., 2017] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2017).
  Improved Texture Networks: Maximizing Quality and Diversity in Feed-forward Stylization and Texture Synthesis.
  In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6924–6932.
  arXiv: 1701.02096.
- [Westlake et al., 2016] Westlake, N., Cai, H., and Hall, P. (2016).
  Detecting people in artwork with CNNs.
  In Hua, G. and Jégou, H., editors, *Computer vision – ECCV 2016 workshops*, pages 825–841, Cham. Springer International Publishing.
  tex.ids: westlake_detecting_2016 arXiv: 1610.08871.
- [Zhou and Zhang, 2002] Zhou, Z.-H. and Zhang, M.-L. (2002).
  Neural Networks for Multi-Instance Learning.
  In *Proceedings of the International Conference on Intelligent Information Technology*, pages 455–459, Beijing, China.
- [Zhu et al., 2017] Zhu, Y., Zhou, Y., Ye, Q., Qiu, Q., and Jiao, J. (2017).
  Soft Proposal Networks for Weakly Supervised Object Localization.
  In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1859–1868.

Publications

The material reported in this presentation was the subject of the following publications:

- Gonthier N., Gousseau Y., Ladjal S. *High resolution neural texture synthesis with long range constraints*; Journal of Mathematical Imaging and Vision 2022.
- Gonthier N., Ladjal S., Gousseau Y. *Multiple instance learning on deep features for weakly supervised object detection with extreme domain shifts*; Submission at Computer Vision and Image Understanding 2021.
- Gonthier N., Gousseau Y., Ladjal S. *An analysis of the transfer learning of convolutional neural networks for artistic images*; Workshop on Fine Art Pattern Extraction and Recognition, ICPR, 2020.
- Gonthier N., Gousseau Y., Ladjal S. *Transfert d'apprentissage et visualisation de réseaux de neurones pour les images artistiques*; The Measurement of Images. Computational Approaches in the History and Theory of the Arts, DHNord 2020.
- Gonthier N., Gousseau Y., Ladjal S., Bonfait O. *Weakly Supervised Object Detection in Artworks*; Workshop on Computer Vision for Art Analysis, ECCV, 2018.