

# Neural Style Transfer

Audrey Maurette

*Janvier 2026*

# 1. Introduction au Neural Style Transfer

# Introduction

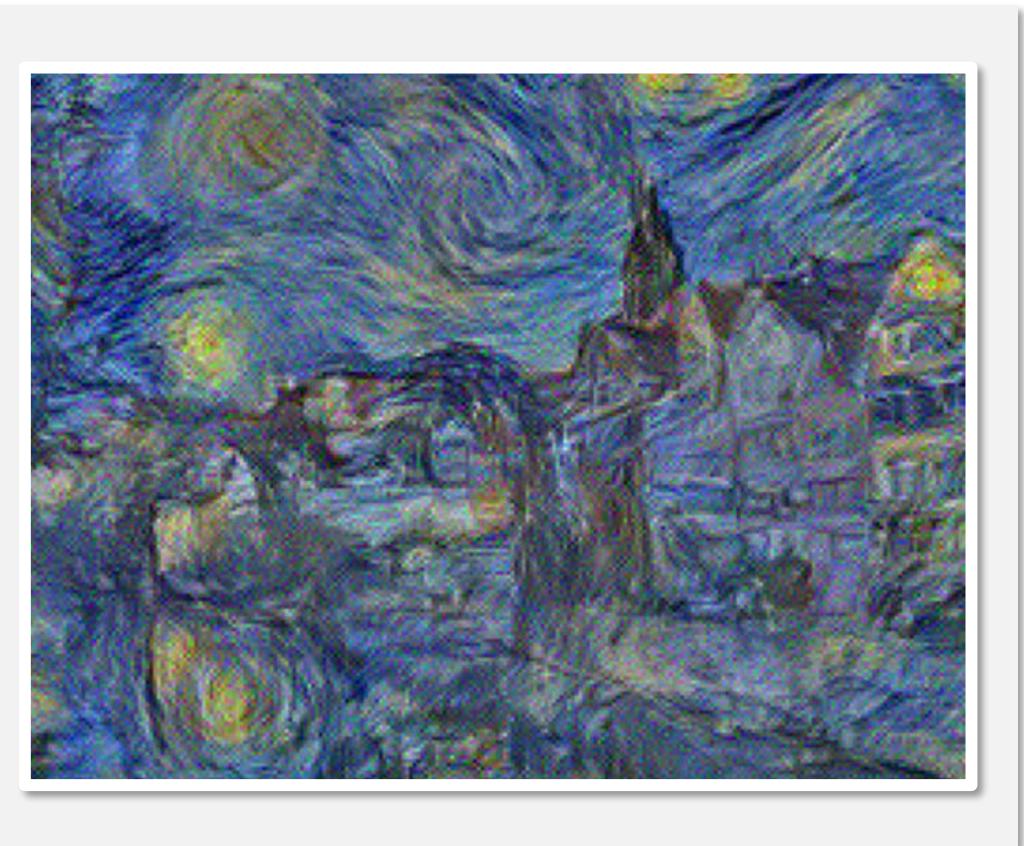
## A Neural Algorithm of Artistic Style

Leon A. Gatys,<sup>1,2,3\*</sup> Alexander S. Ecker,<sup>1,2,4,5</sup> Matthias Bethge<sup>1,2,4</sup>

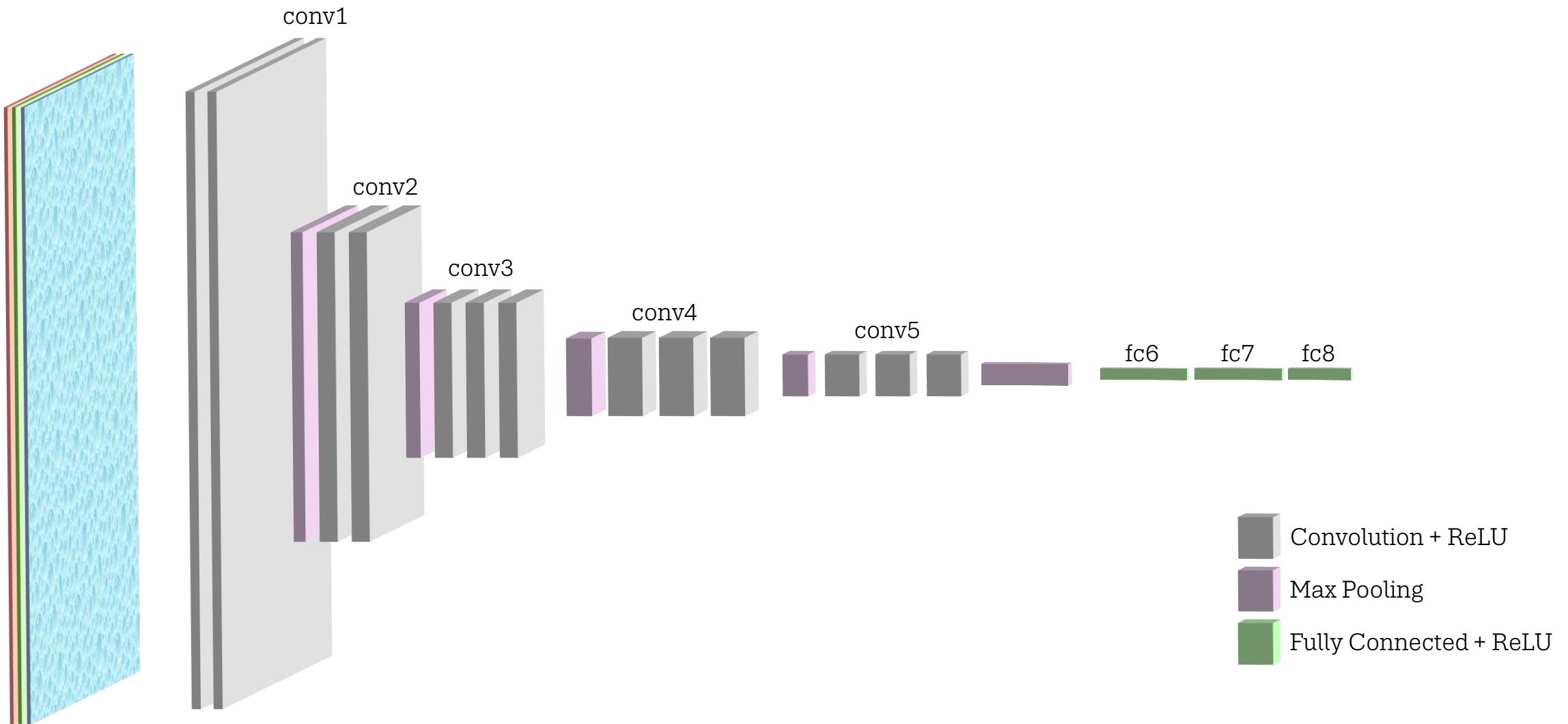
Le Neural Style Transfer est une méthode qui permet de générer une image en combinant le contenu d'une image avec le style artistique d'une autre.



CNN

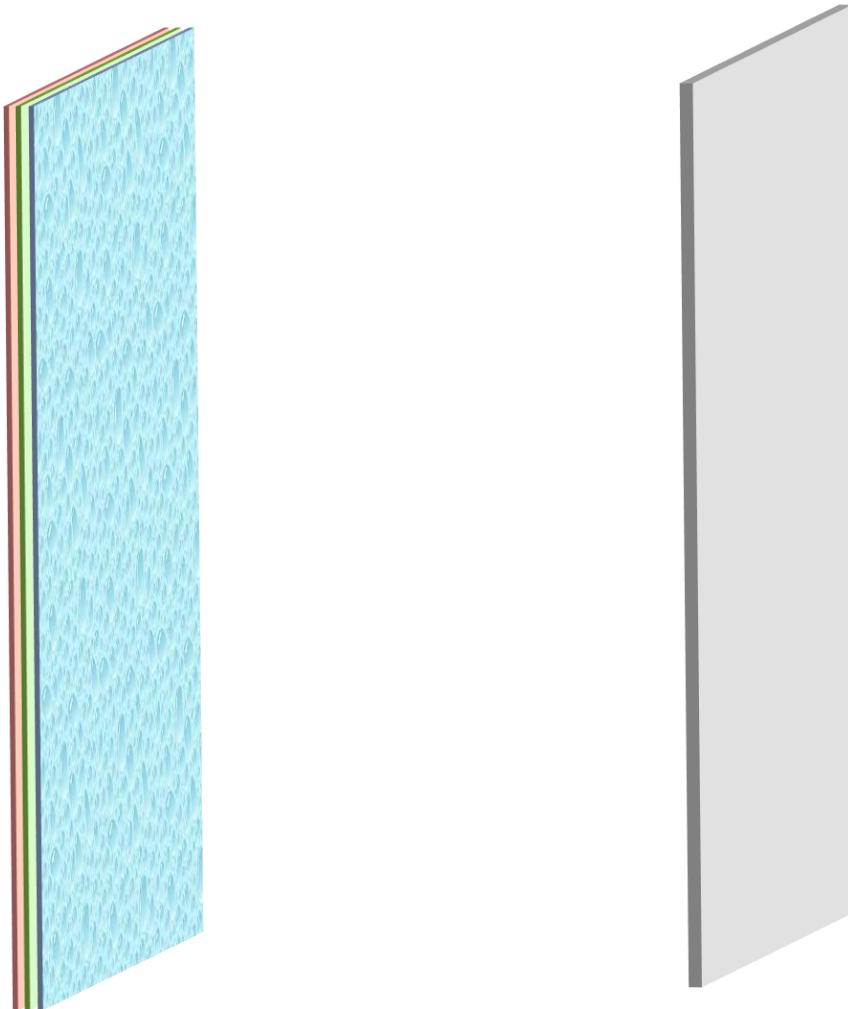


# Modèle utilisé : VGG19



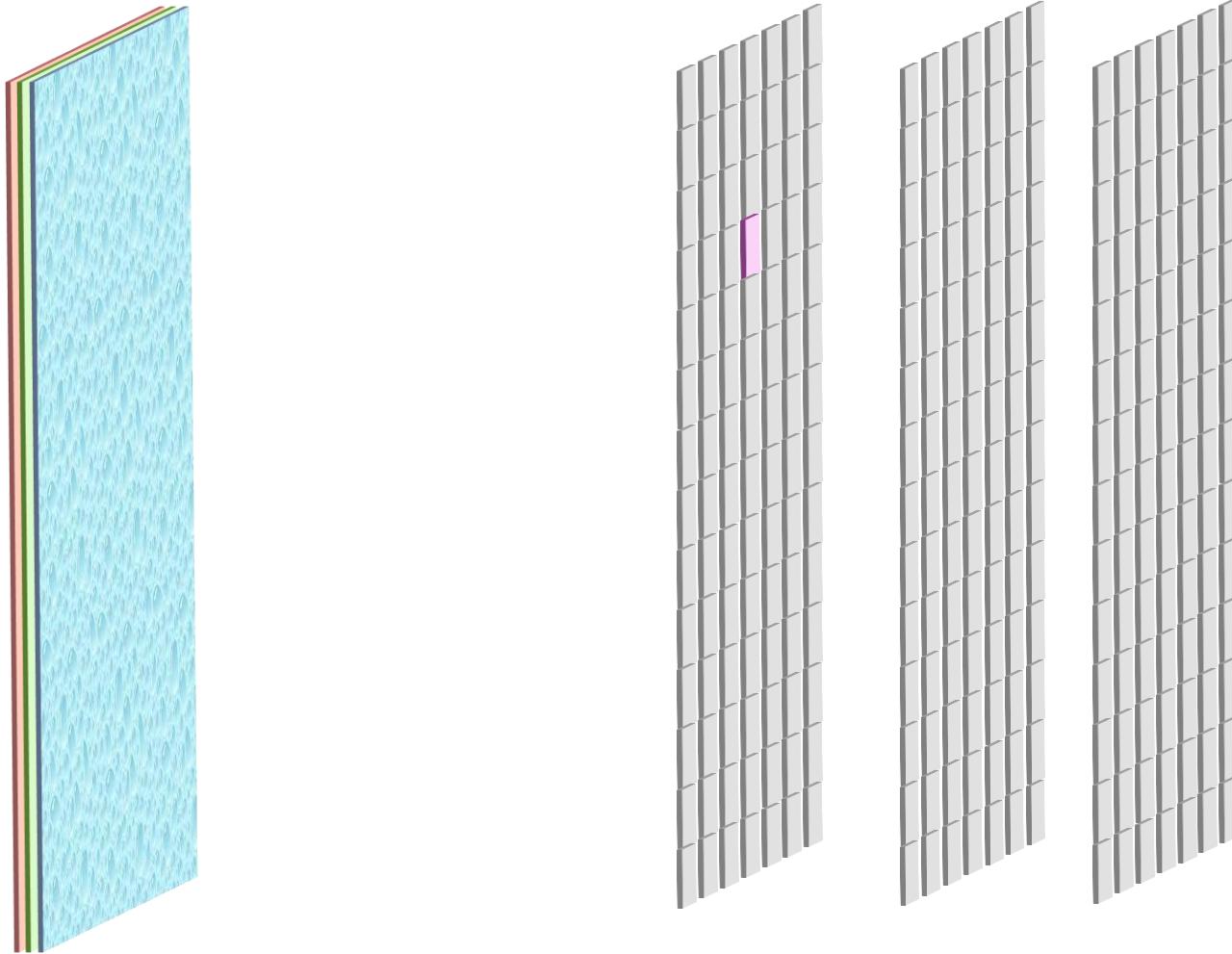
# Couche de convolution

Focus sur la couche conv1\_1



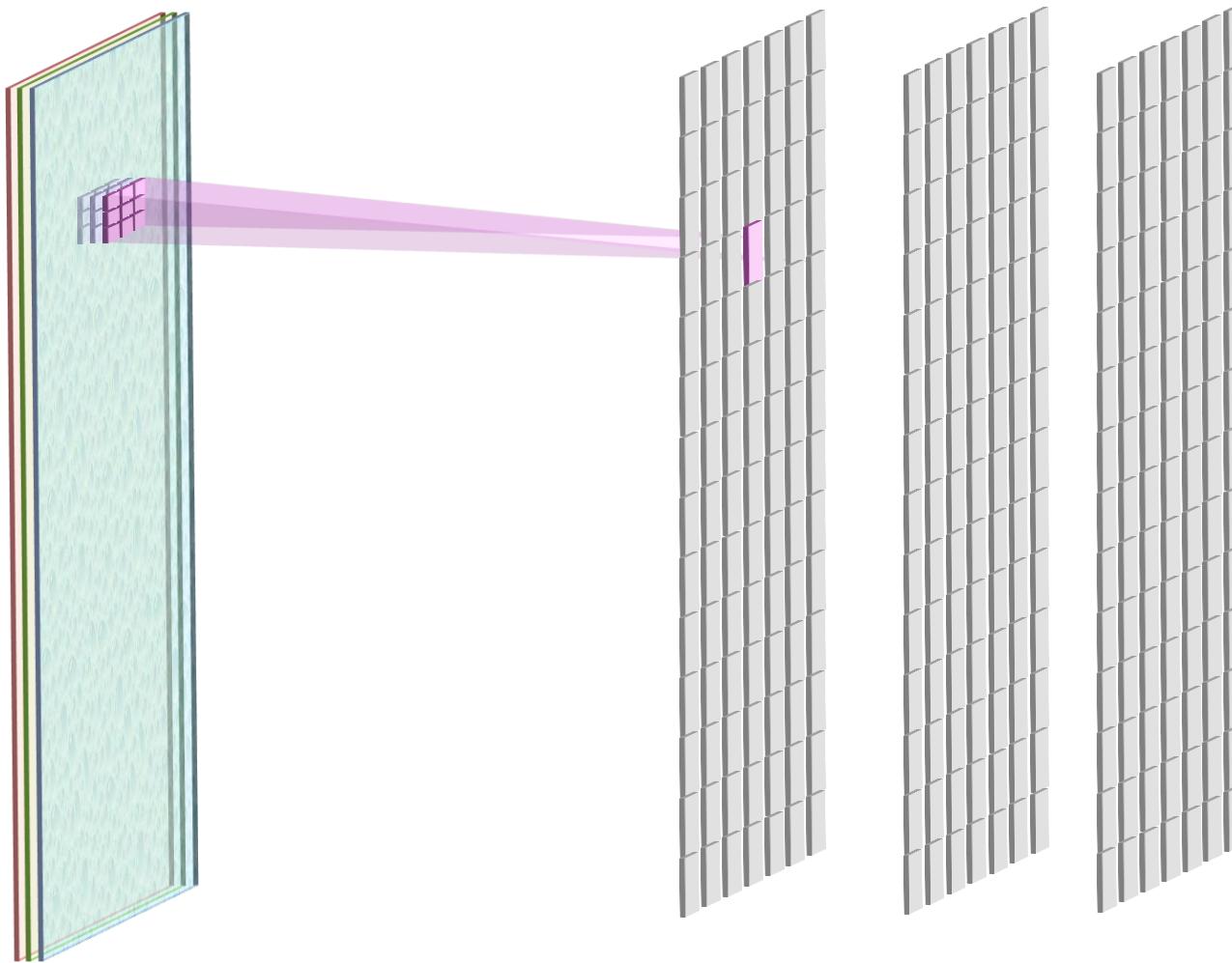
# Couche de convolution

Focus sur la couche conv1\_1



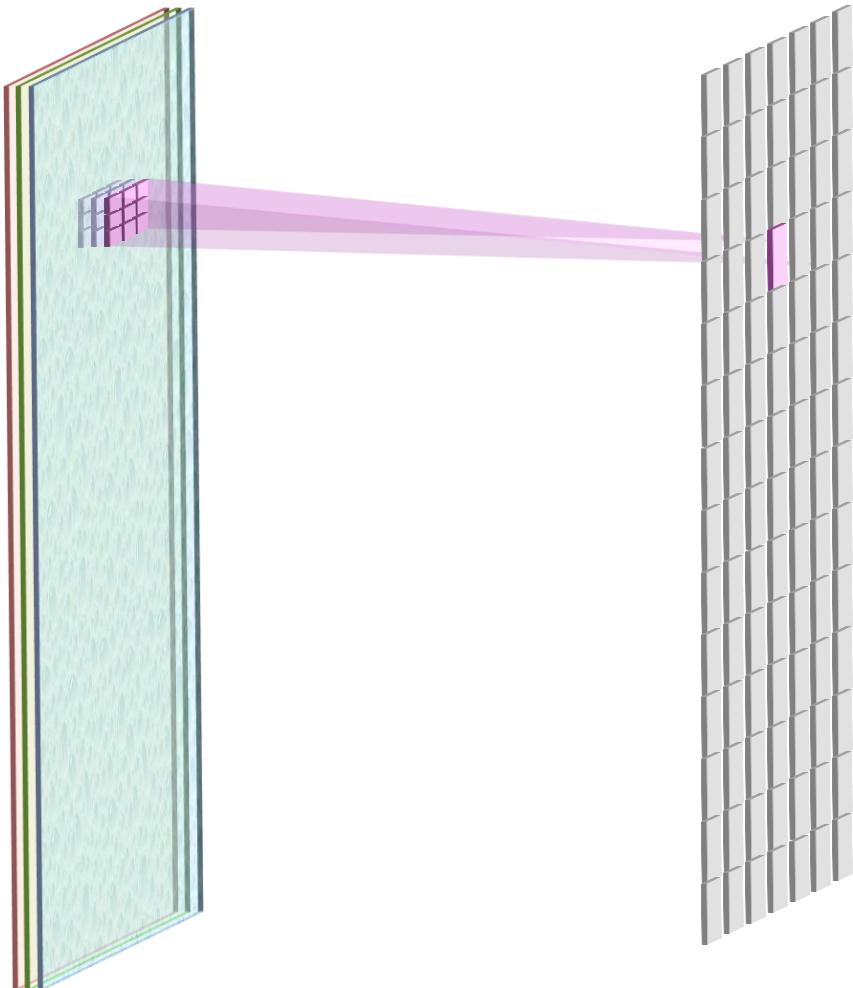
# Couche de convolution

Focus sur la couche conv1\_1



# Couche de convolution

Focus sur la couche conv1\_1



**Une couche de convolution = un ensemble de filtres**

La profondeur de sortie de la couche est égale au nombre de filtres



**Un filtre = ensemble de neurones identiques**

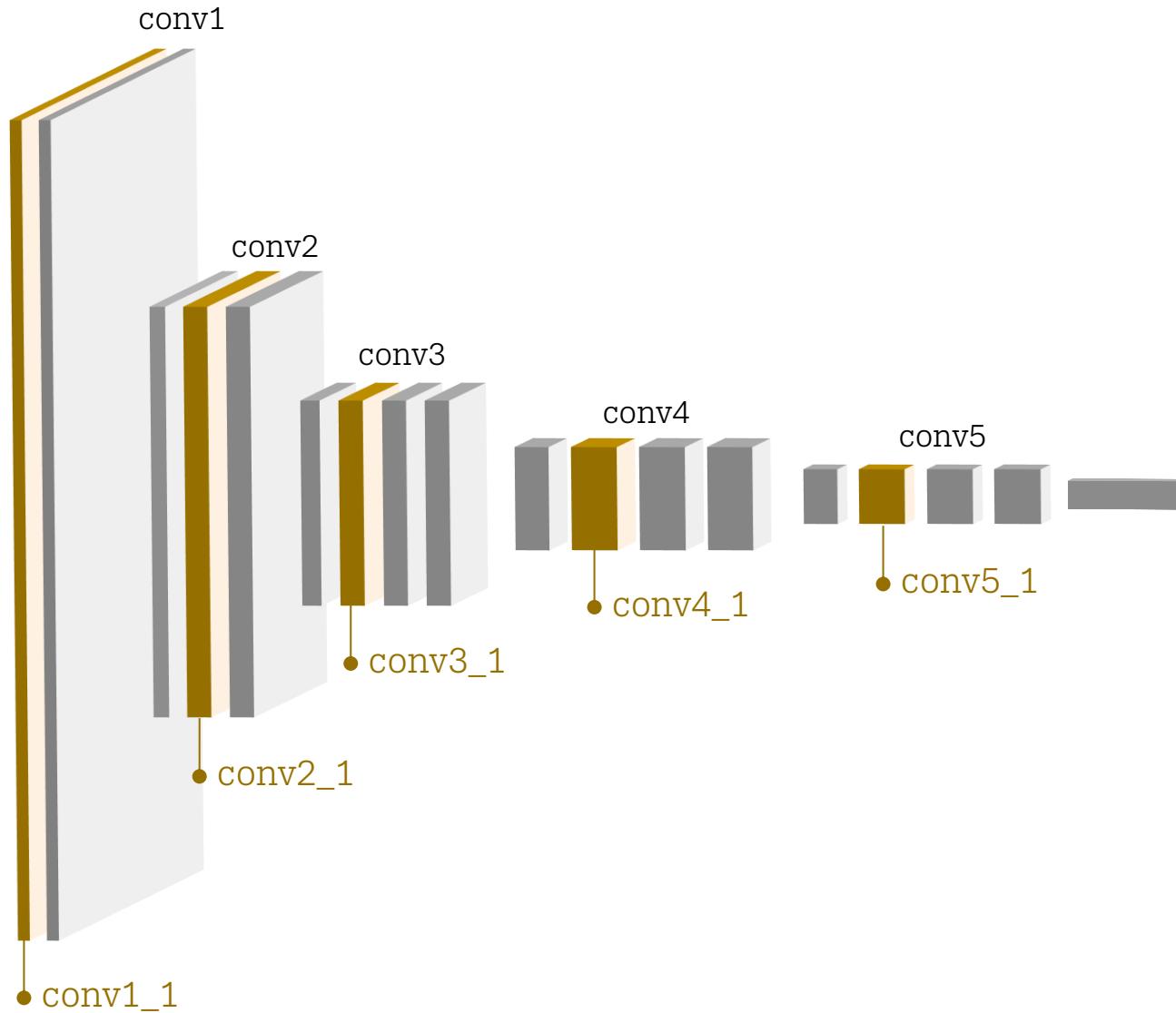
Chaque filtre correspond à la reconnaissance d'une caractéristique visuelle. L'application d'un filtre va ainsi produire une *feature maps*.



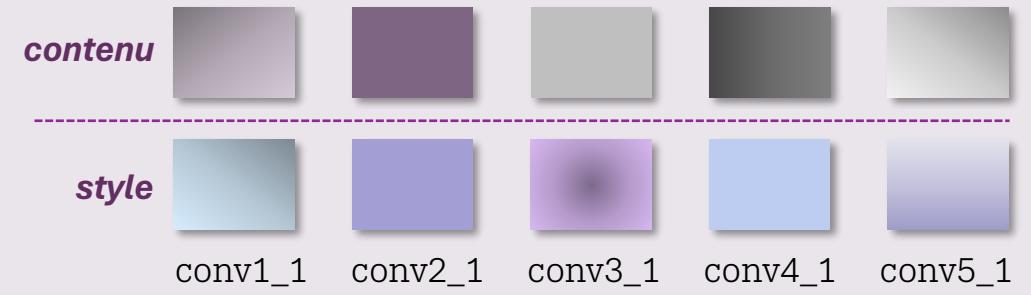
**Un neurone = connexion locale à l'image**

Chaque neurone est connecté à une petite zone de l'image, et s'applique toujours sur l'ensemble des canaux.

# Objectifs de l'étude



**1. Identifier le type d'information retenu au fur et à mesure des couches**



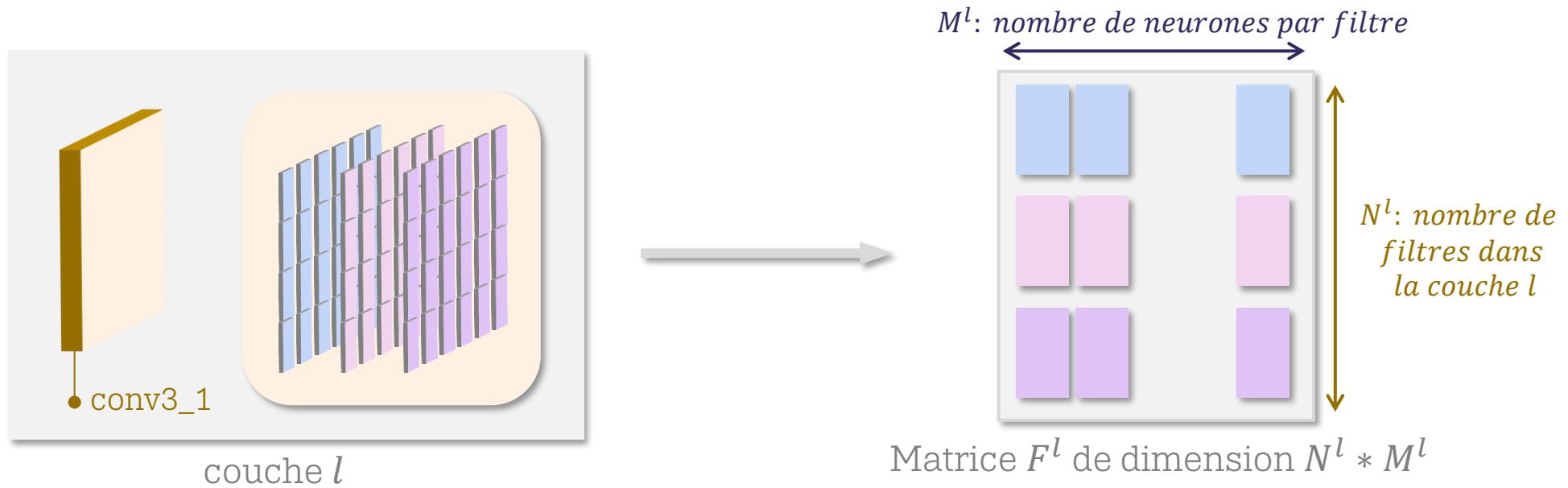
**2. Générer une image en combinant le contenu d'une image et le style d'une autre**



## 2. Représentations de contenu et style

# Analyse du contenu d'une image

Pour une couche  $l$ , on définit la matrice des activations des filtres :



On définit ainsi la perte de contenu entre deux images au niveau de la couche  $l$  :



image  $\vec{p}$  associée à  $P^l$

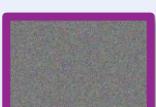
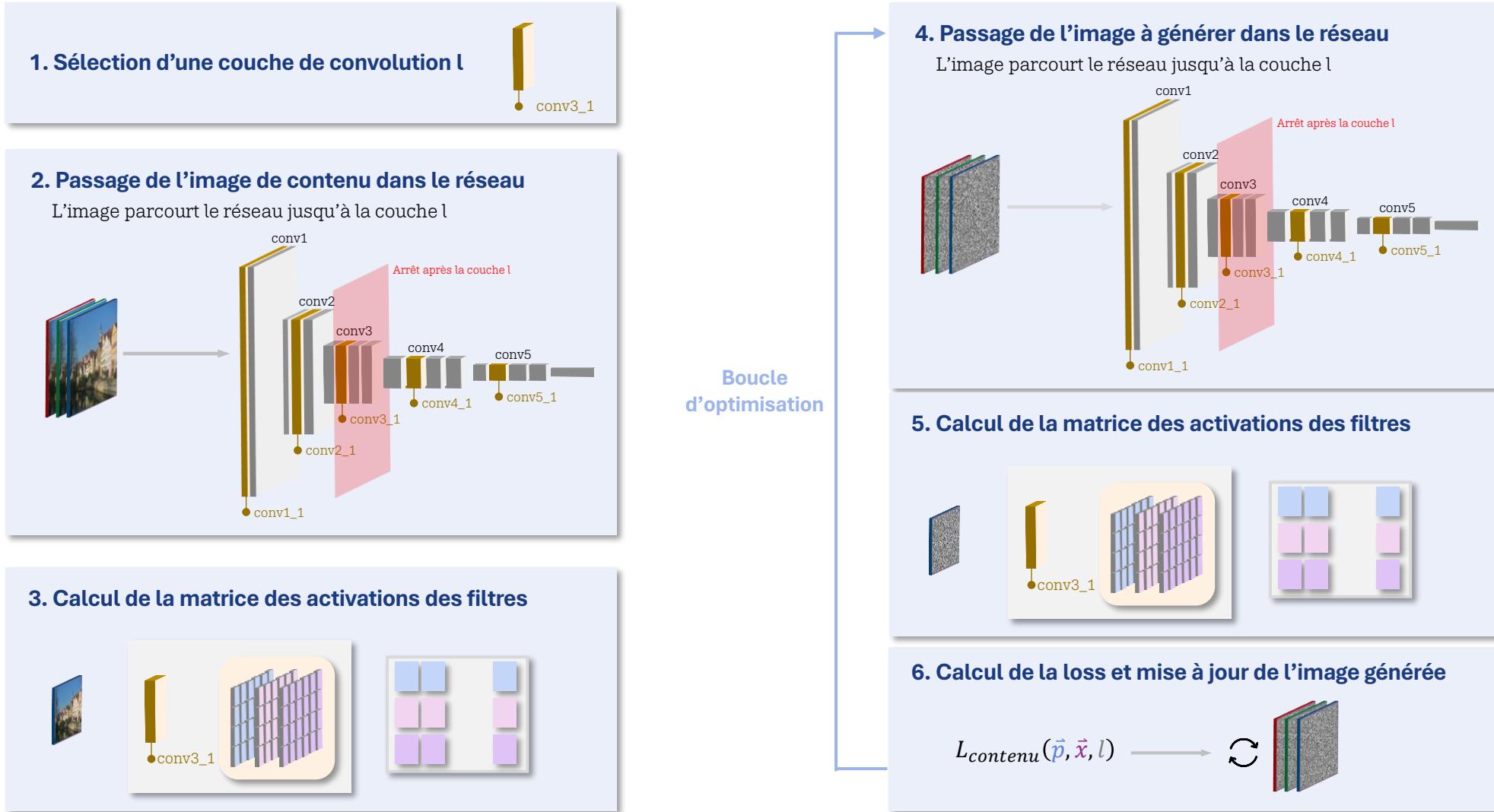


image  $\vec{x}$  associée à  $F^l$

$$L_{contenu}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{i,j}^l - P_{i,j}^l)^2$$

# Démarche à suivre pour l'analyse de contenu

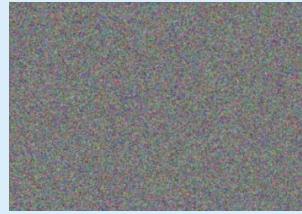


# Analyse de l'image de contenu

Image de contenu  
 $\vec{p}$



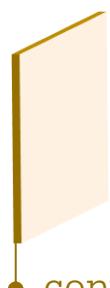
Initialisation de l'image générée  
 $\vec{x}$



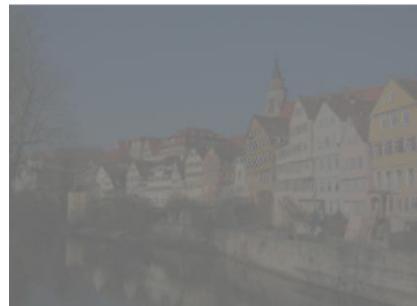
Perte associée

$$L_{contenu}(\vec{p}, \vec{x}, l) + L_{couleur}$$

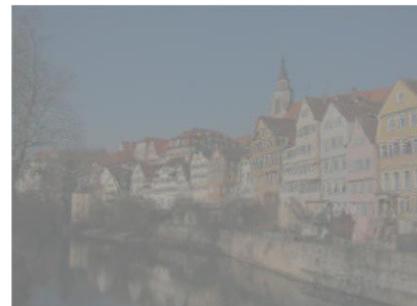
Résultats obtenus :



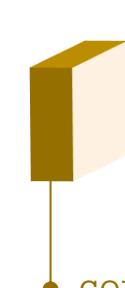
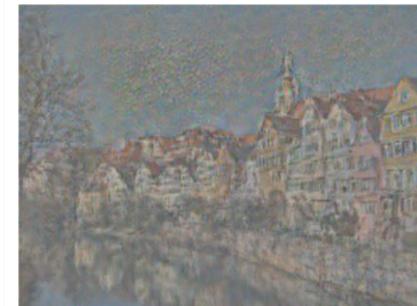
conv1\_1



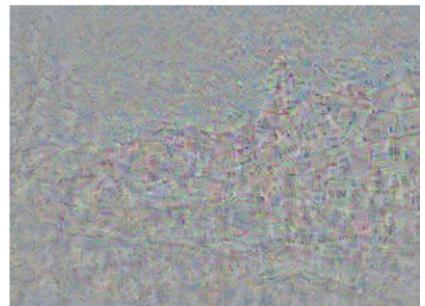
conv2\_1



conv3\_1



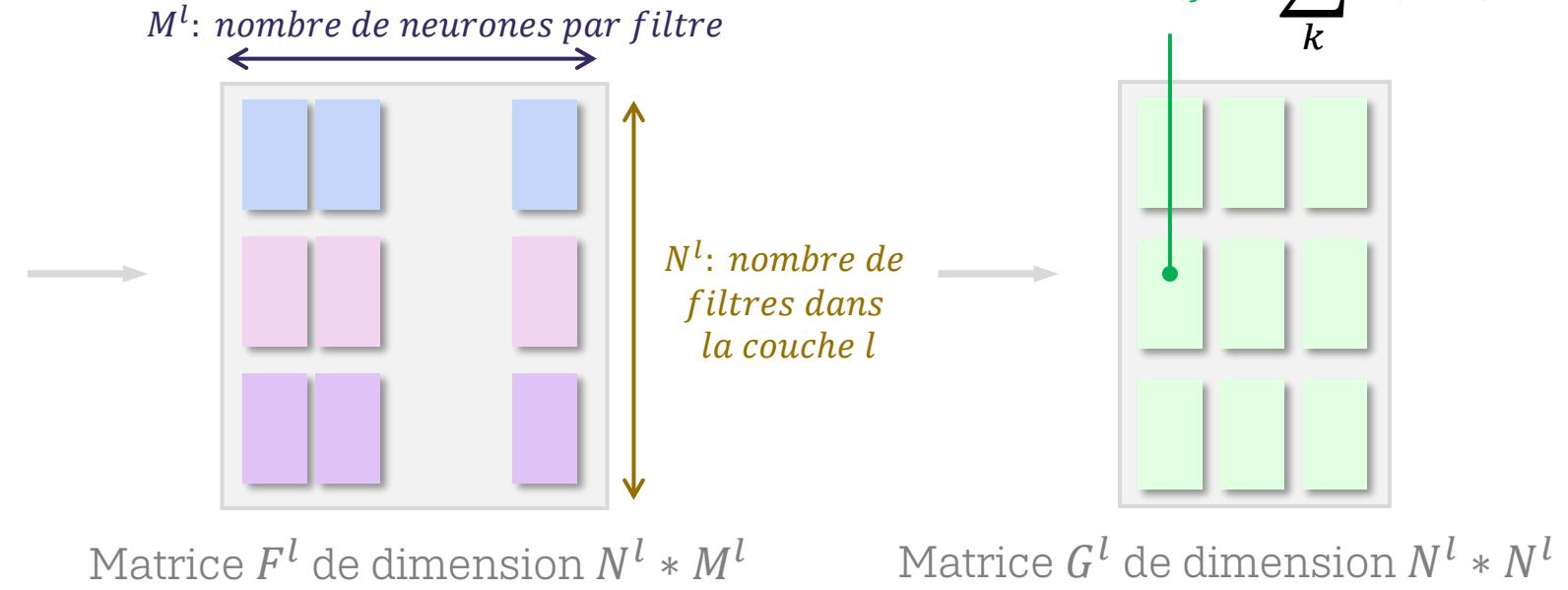
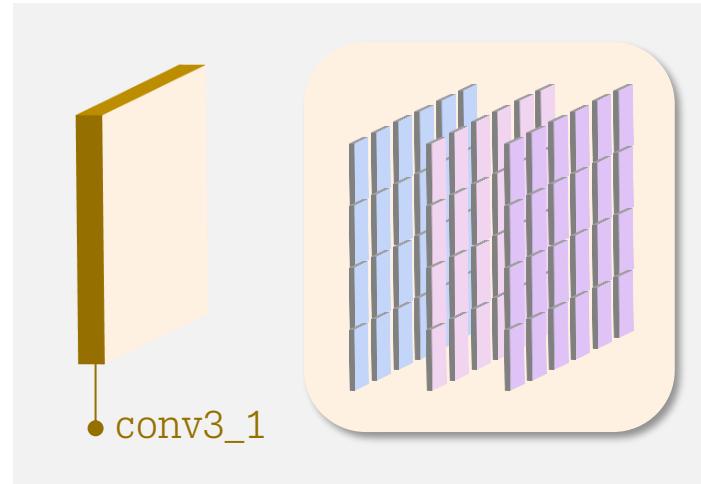
conv4\_1



conv5\_1

# Analyse du style d'une image

Pour une couche  $l$ , on définit la Gram Matrice :



On définit ainsi la contribution de la couche  $l$  à la perte de style entre deux images :



*image  $\vec{a}$*   
associée à la Gram Matrice  $A^l$

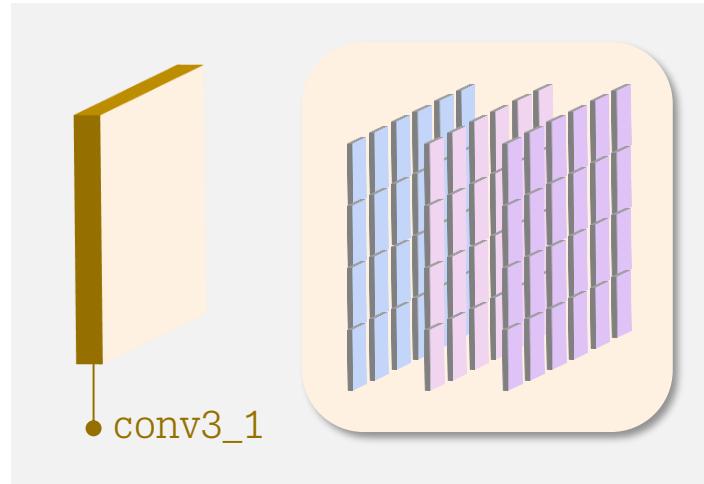


*image  $\vec{x}$*   
associée à la Gram Matrice  $G^l$

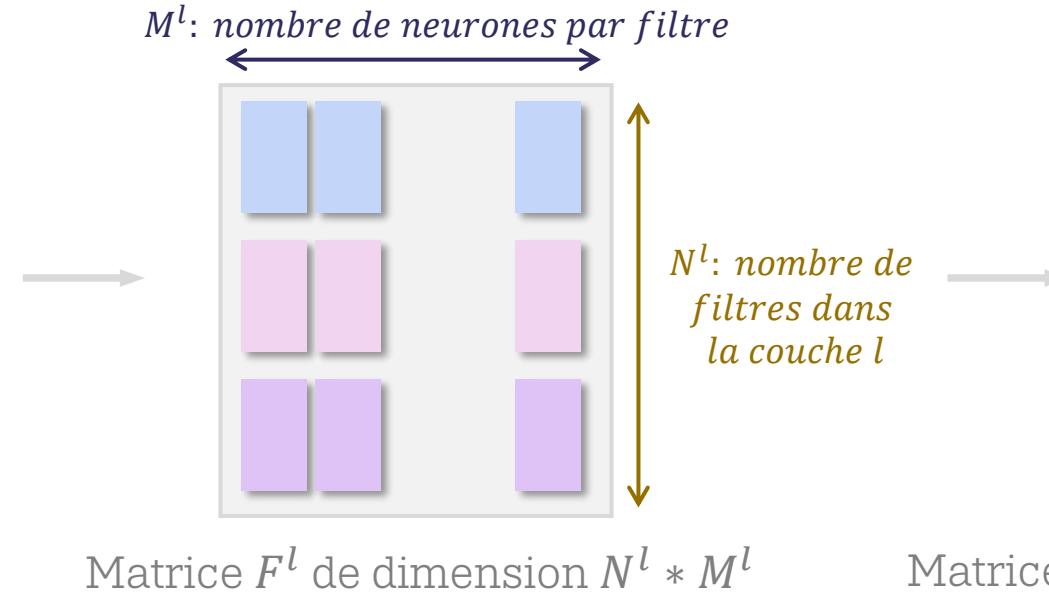
$$E_l(\vec{a}, \vec{x}, l) = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{i,j}^l - A_{i,j}^l)^2$$

# Analyse du style d'une image

Pour une couche  $l$ , on définit la Gram Matrice :



couche  $l$



Matrice  $F^l$  de dimension  $N^l * M^l$

Matrice  $G^l$  de dimension  $N^l * N^l$

Ainsi, la perte de style totale entre deux images est définie ainsi :



image  $\vec{a}$

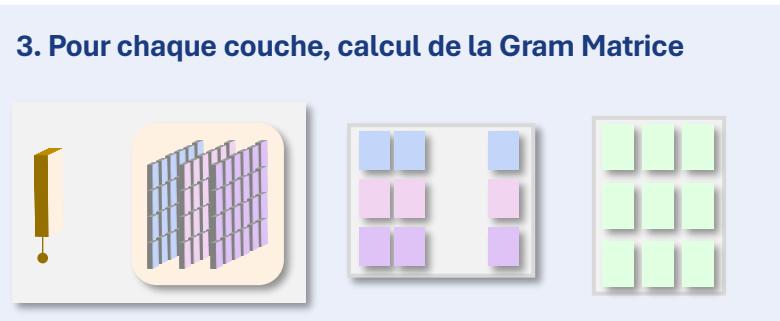
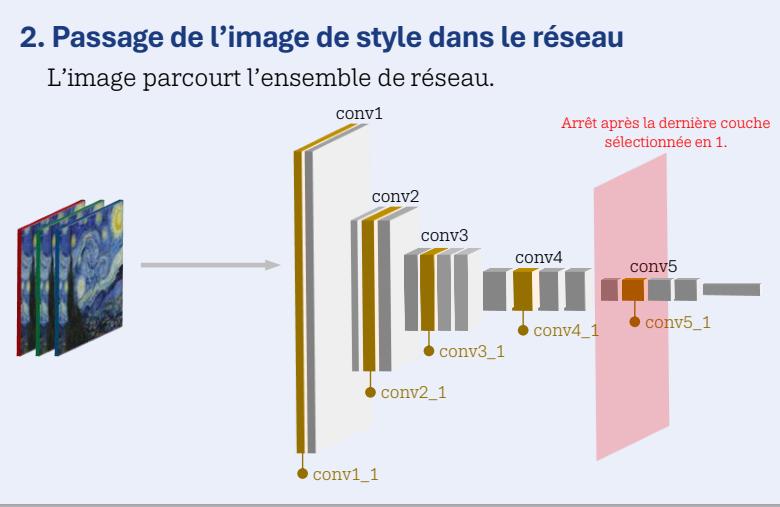


image  $\vec{x}$

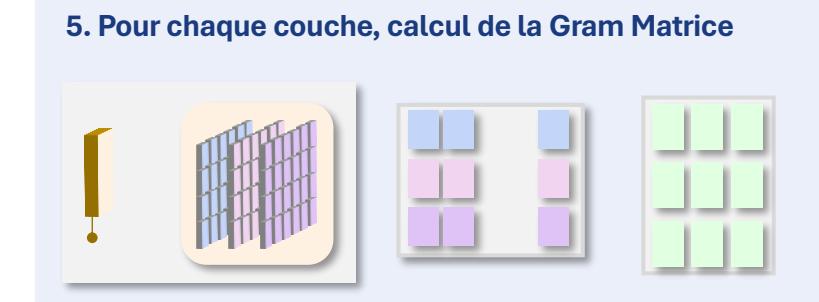
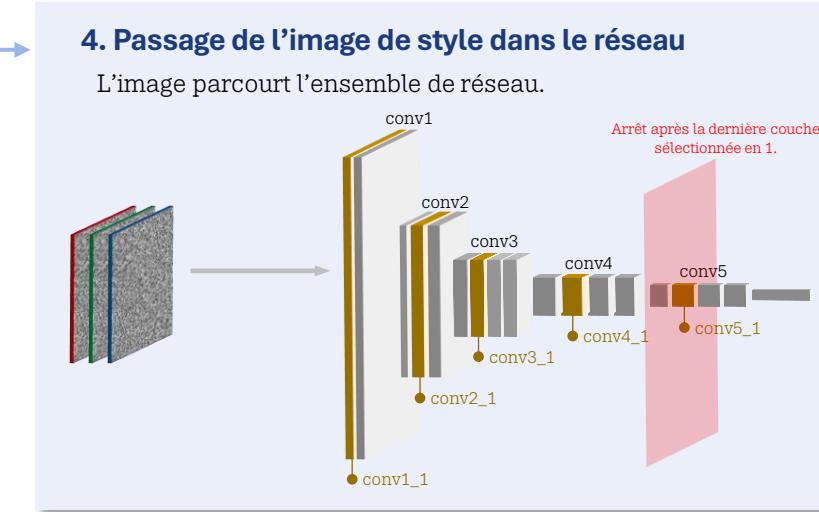
$$L_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^L w_l E_l$$

$w_l$  : poids associés à la contribution de chaque couche  $l$  à la loss totale

# Démarche à suivre pour l'analyse de style



Boucle  
d'optimisation



# Analyse de l'image de style



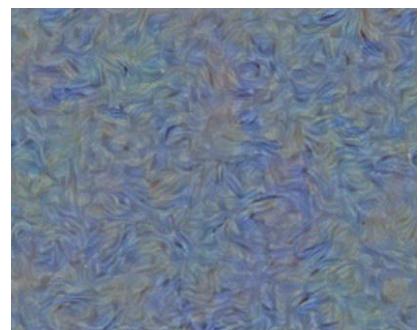
Perte associée

$$L_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^L w_l E_l + L_{couleur}$$

Résultats obtenus :

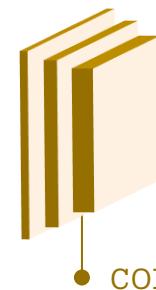


  
 $w1 = 1$   
 $w2 = 0$   
 $w3 = 0$   
 $w4 = 0$   
 $w5 = 0$   
conv1\_1



  
 $w1 = 1/2$   
 $w2 = 1/2$   
 $w3 = 0$   
 $w4 = 0$   
 $w5 = 0$   
conv2\_1



  
 $w1 = 1/3$   
 $w2 = 1/3$   
 $w3 = 1/3$   
 $w4 = 0$   
 $w5 = 0$   
conv3\_1



  
 $w1 = 1/4$   
 $w2 = 1/4$   
 $w3 = 1/4$   
 $w4 = 1/4$   
 $w5 = 0$   
conv4\_1



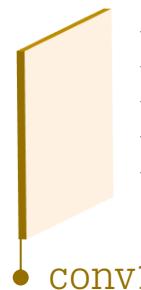
  
 $w1 = 1/5$   
 $w2 = 1/5$   
 $w3 = 1/5$   
 $w4 = 1/5$   
 $w5 = 1/5$   
conv5\_1

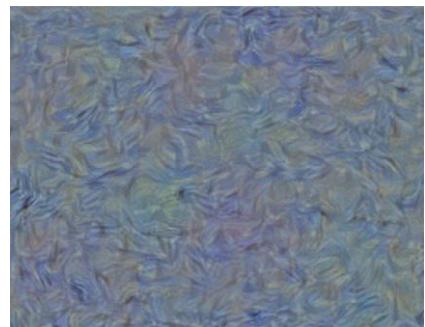
# Analyse de l'image de style



Résultats obtenus en changeant la combinaison des poids :



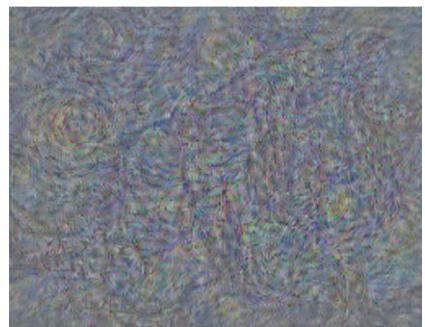
  
 $w1 = 1$   
 $w2 = 0$   
 $w3 = 0$   
 $w4 = 0$   
 $w5 = 0$   
conv1\_1



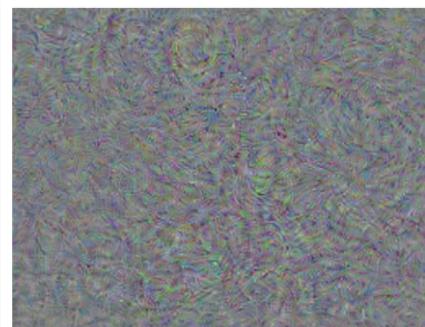
  
 $w1 = 0$   
 $w2 = 1$   
 $w3 = 0$   
 $w4 = 0$   
 $w5 = 0$   
conv2\_1



  
 $w1 = 0$   
 $w2 = 0$   
 $w3 = 1$   
 $w4 = 0$   
 $w5 = 0$   
conv3\_1



  
 $w1 = 0$   
 $w2 = 0$   
 $w3 = 0$   
 $w4 = 1$   
 $w5 = 0$   
conv4\_1



  
 $w1 = 0$   
 $w2 = 0$   
 $w3 = 0$   
 $w4 = 0$   
 $w5 = 1$   
conv5\_1

### 3. Création d'une image artistique

Analyse du contenu et du style d'une image par le réseau

# Pipeline de la création d'image artistique

Image de contenu  
 $\vec{p}$



Image de style  
 $\vec{a}$



Initialisation de l'image générée



Couche  $l$  de contenu associée à  $L_{contenu}(\vec{p}, \vec{x}, l)$

## Perte associée

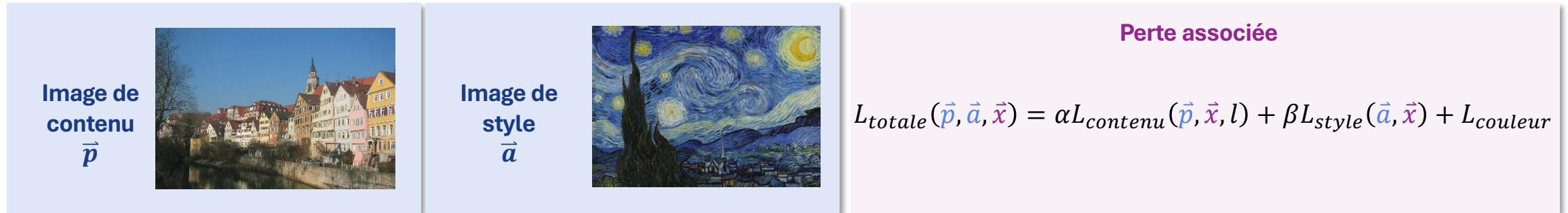
$$L_{totale}(\vec{p}, \vec{a}, \vec{x}) = \alpha L_{contenu}(\vec{p}, \vec{x}, l) + \beta L_{style}(\vec{a}, \vec{x}) + L_{couleur}$$

## Paramètres

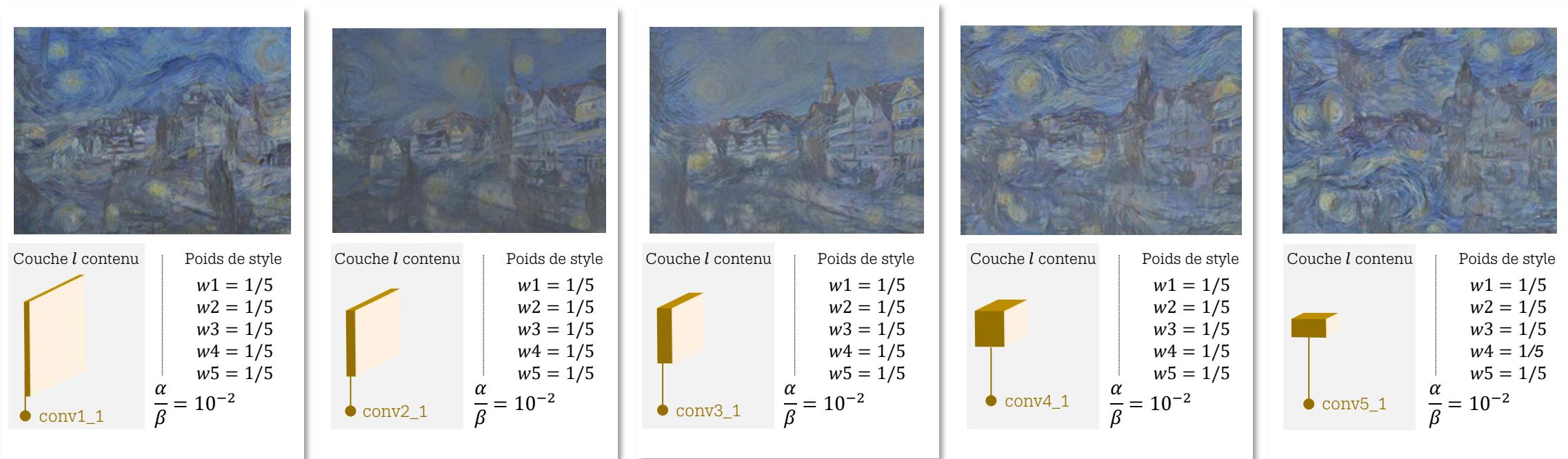
Combinaison des  $w_i$  associés à  $L_{style}(\vec{a}, \vec{x})$

$\frac{\alpha}{\beta}$  Balance entre contenu et style

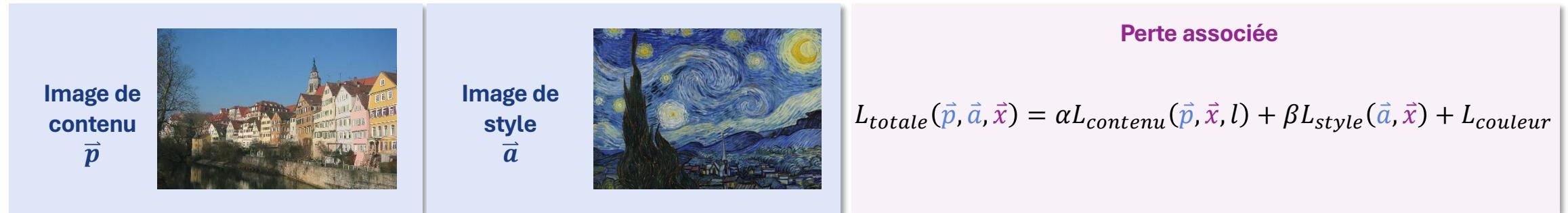
# Influence des paramètres : couche $l$



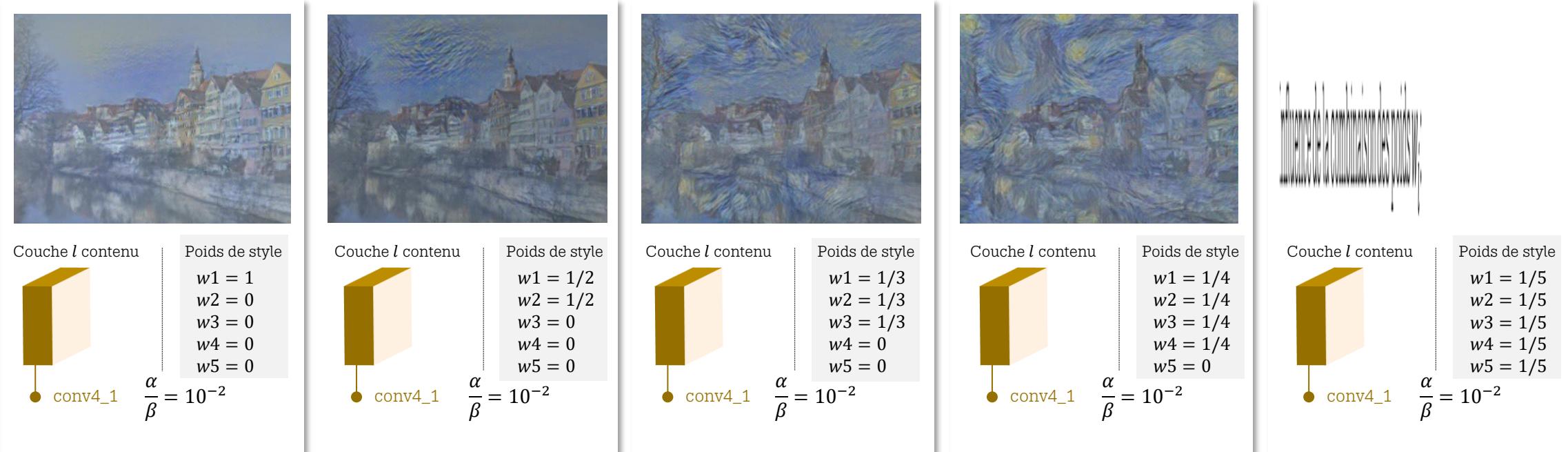
Influence de la couche  $l$  de contenu



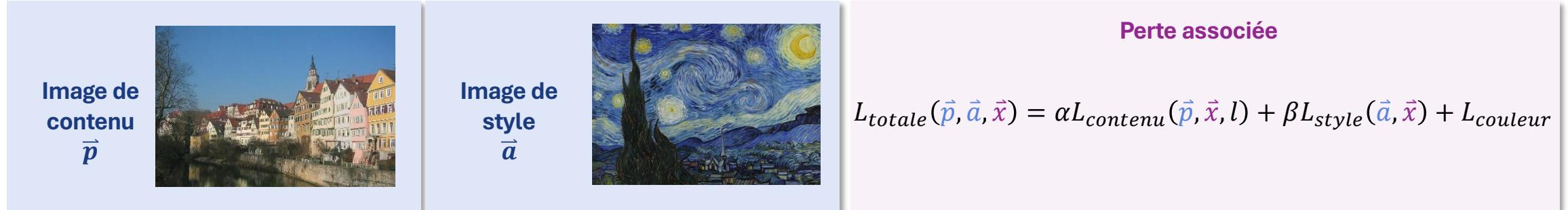
# Influence des paramètres : $w_i$



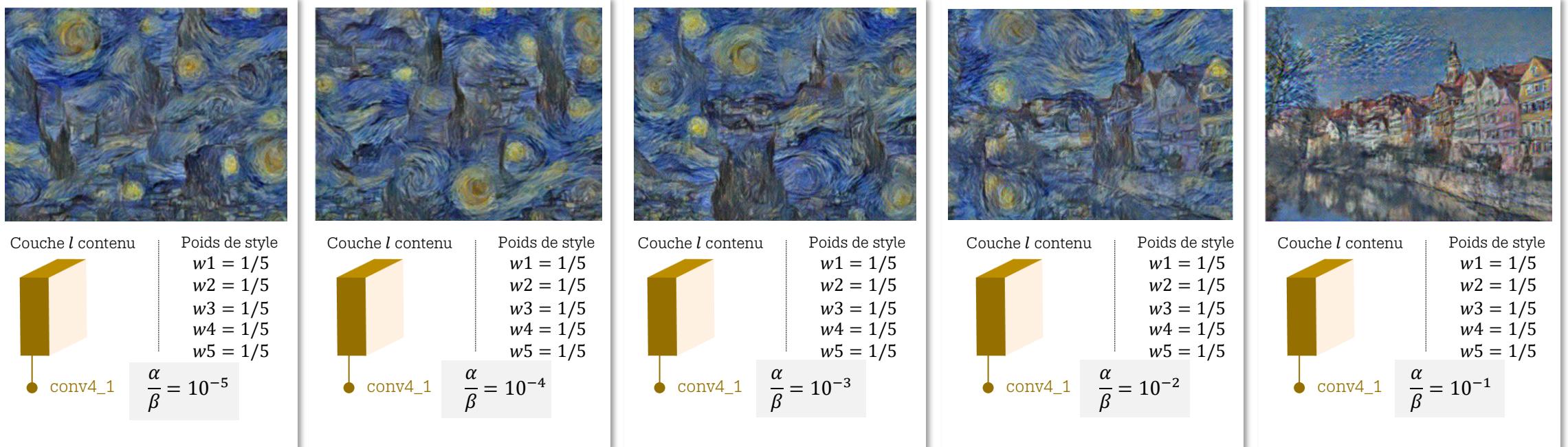
Influence de la combinaison des poids  $w_i$ :



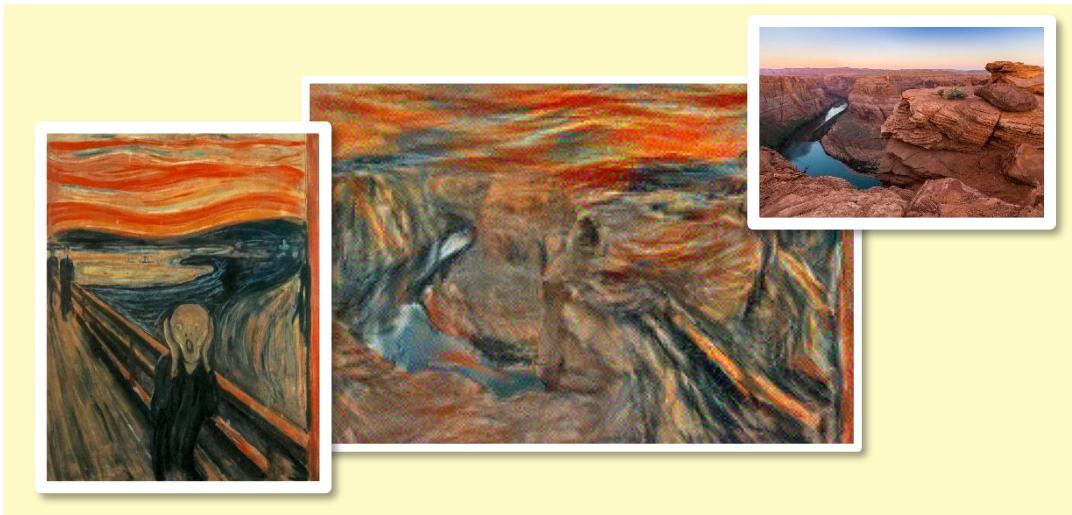
# Influence des paramètres : $\frac{\alpha}{\beta}$



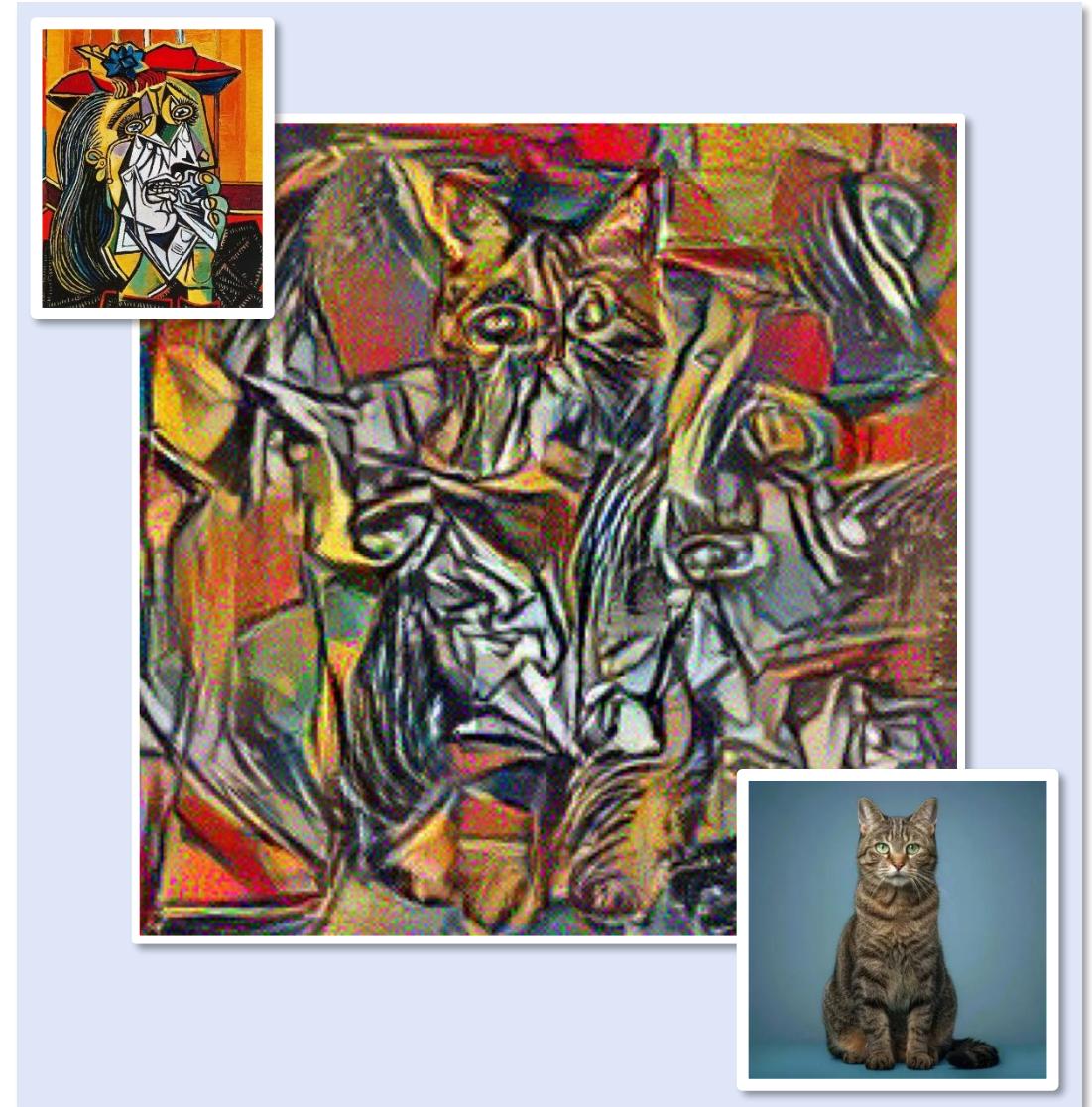
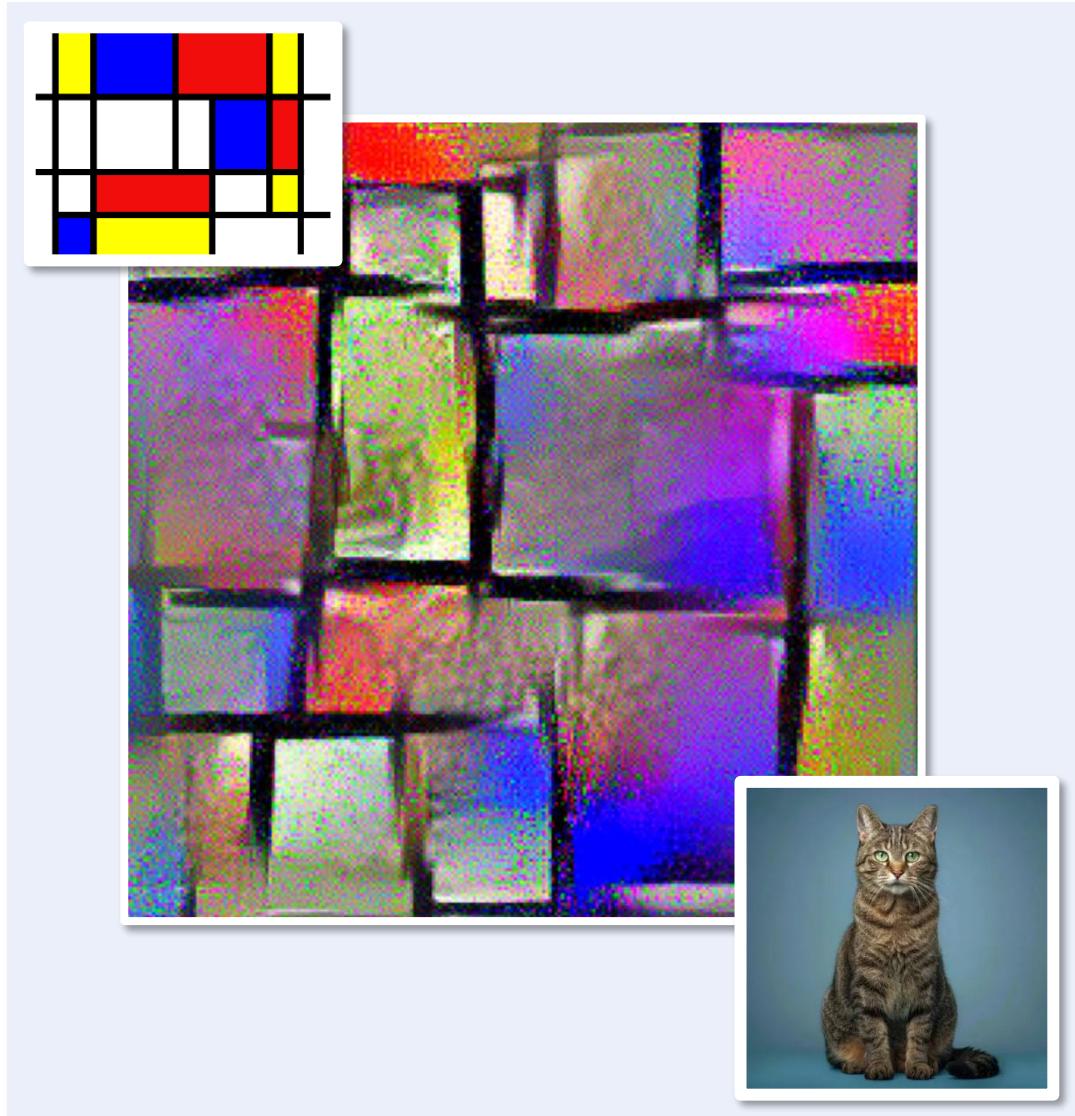
Influence de la balance  $\frac{\alpha}{\beta}$ :



# Exemples de création d'image



# Certains styles sont trop complexes



# Lien GitHub

[https://github.com/audrey1306/Neural\\_Style\\_Transfer.git](https://github.com/audrey1306/Neural_Style_Transfer.git)