



Neural Art -- 電腦作畫

by Mark Chang

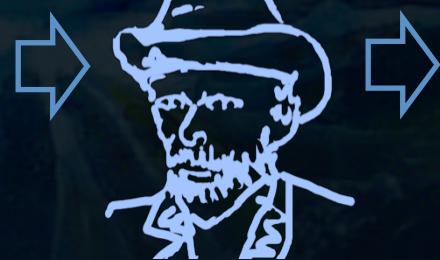
A Neural Algorithm of Artistic Style

- 作者：
 - Leon A. Gatys.
 - Alexander S. Ecker.
 - Matthias Bethge
- 所屬單位：
 - Werner Reichardt Centre for Integrative Neuroscience and Institute of Theoretical Physics, University of Tübingen, Germany.
 - Bernstein Center for Computational Neuroscience, Tübingen, Germany.

如何作畫？



畫家



大腦



景物

畫風

畫作



電腦



類神經網路



大綱

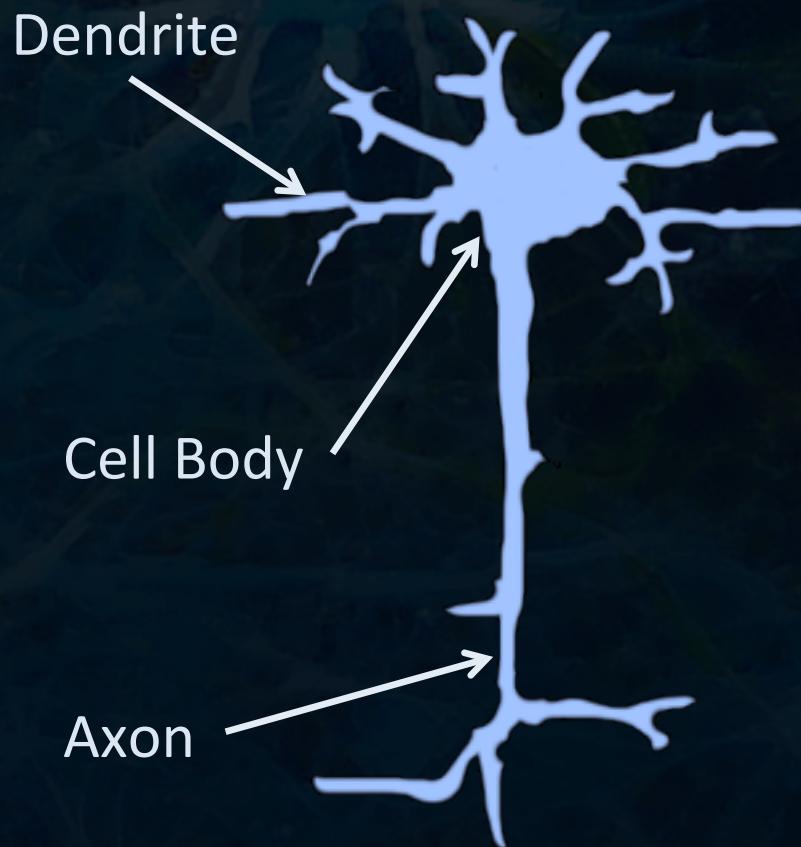
- 人類視覺
- 電腦視覺
- 電腦作畫
- 作品展示

人類視覺

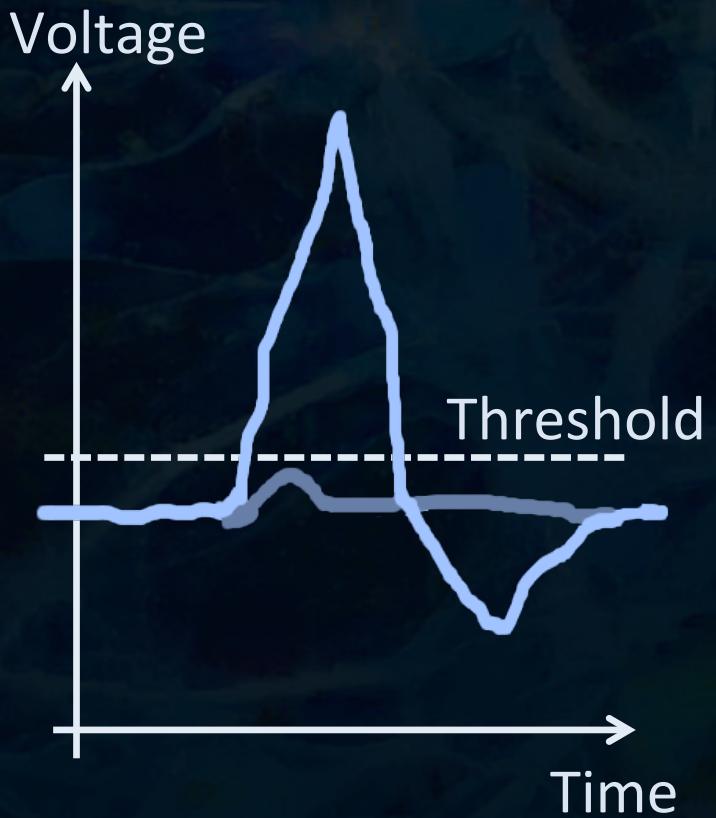
- 神經元
- 視覺傳遞途徑
- 錯覺

神經元

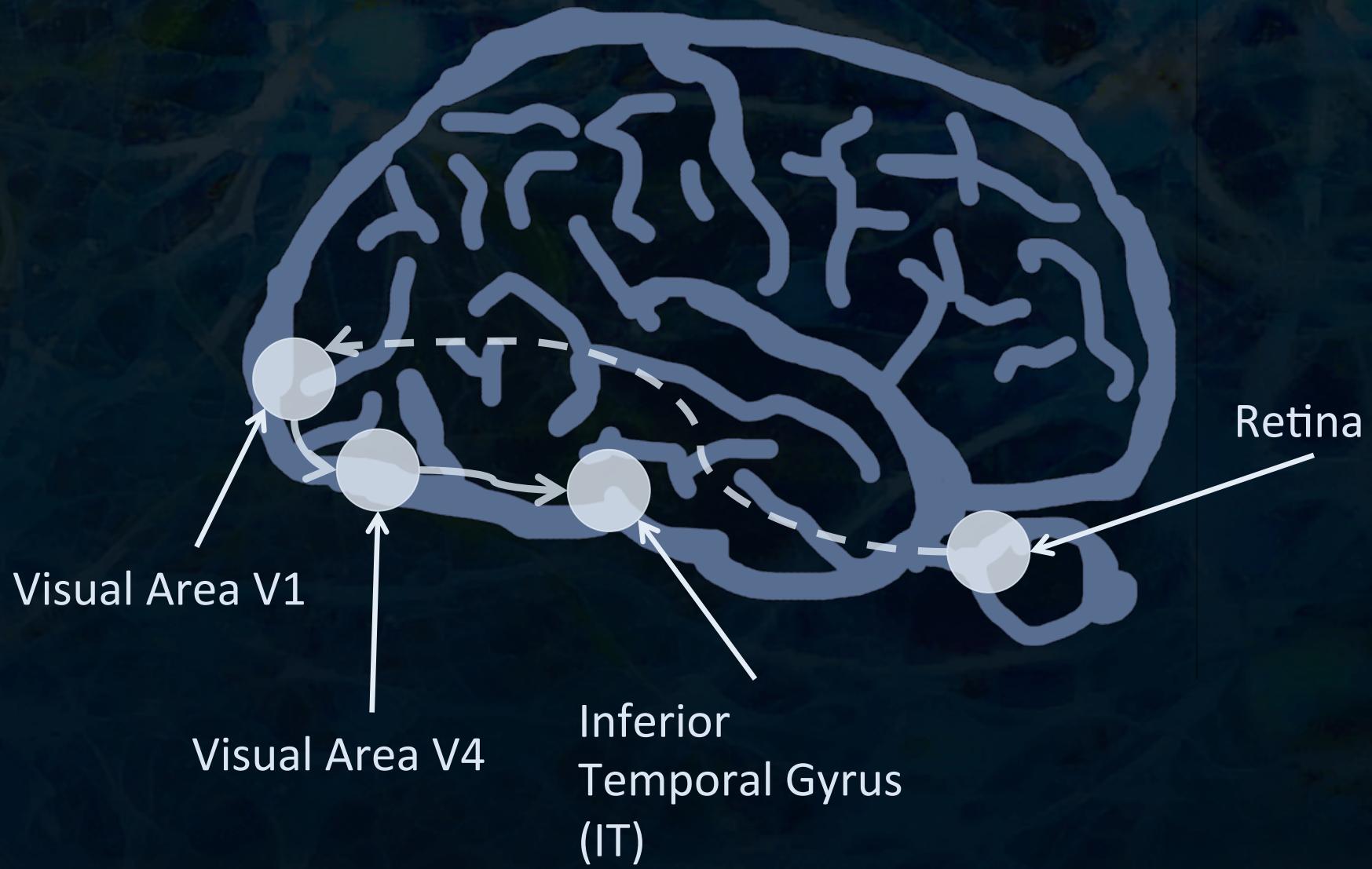
- Neuron



- Action Potential



視覺傳遞途徑



視覺傳遞途徑



Visual Area V1

Receptive Fields



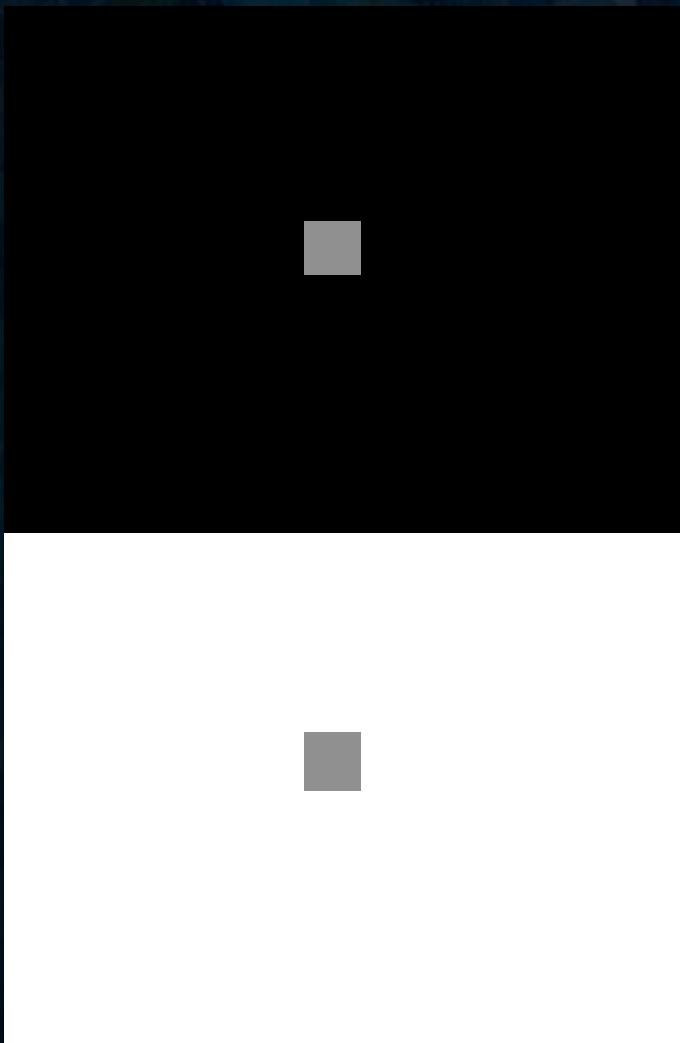
Visual Area V4



Inferior
Temporal
Gyrus (IT)



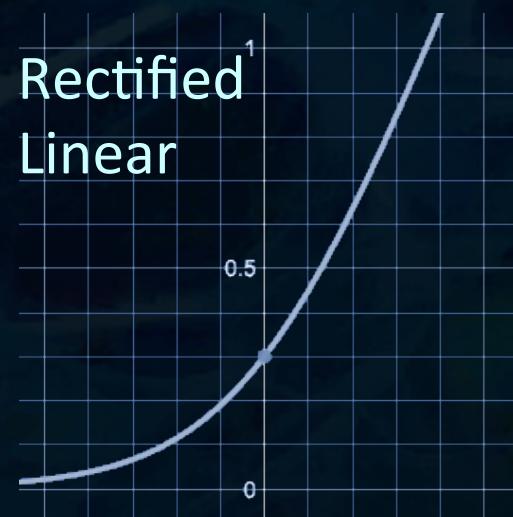
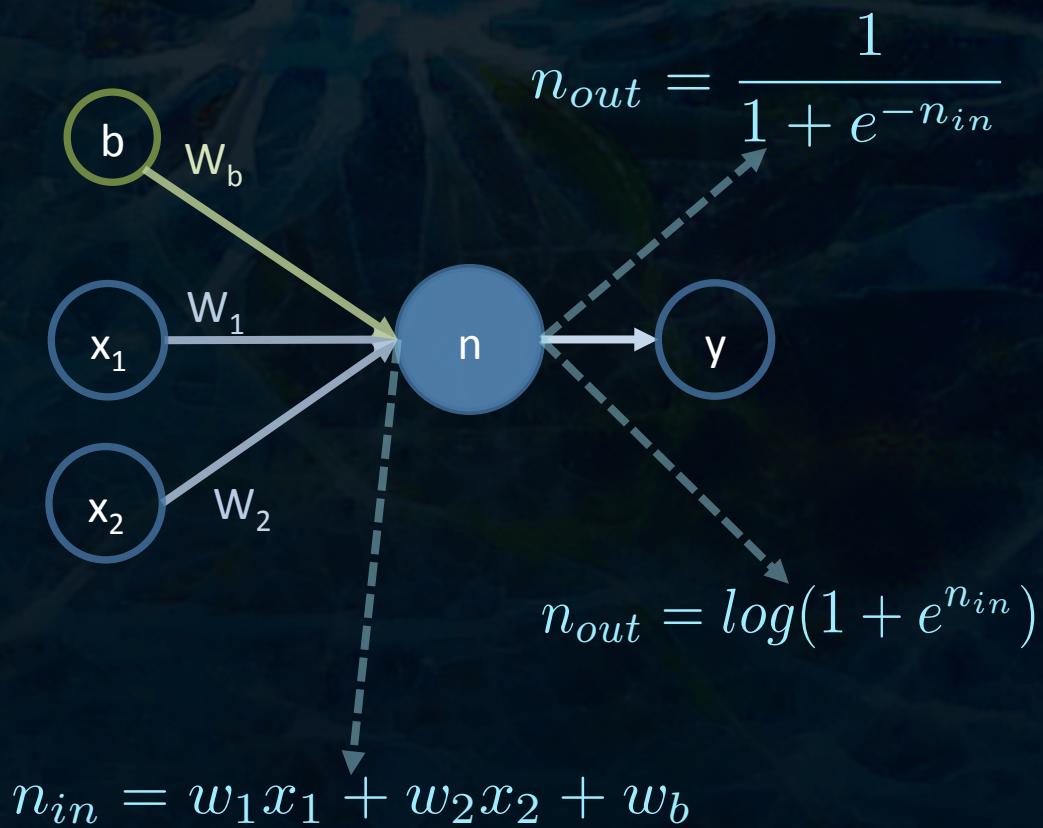
錯覺



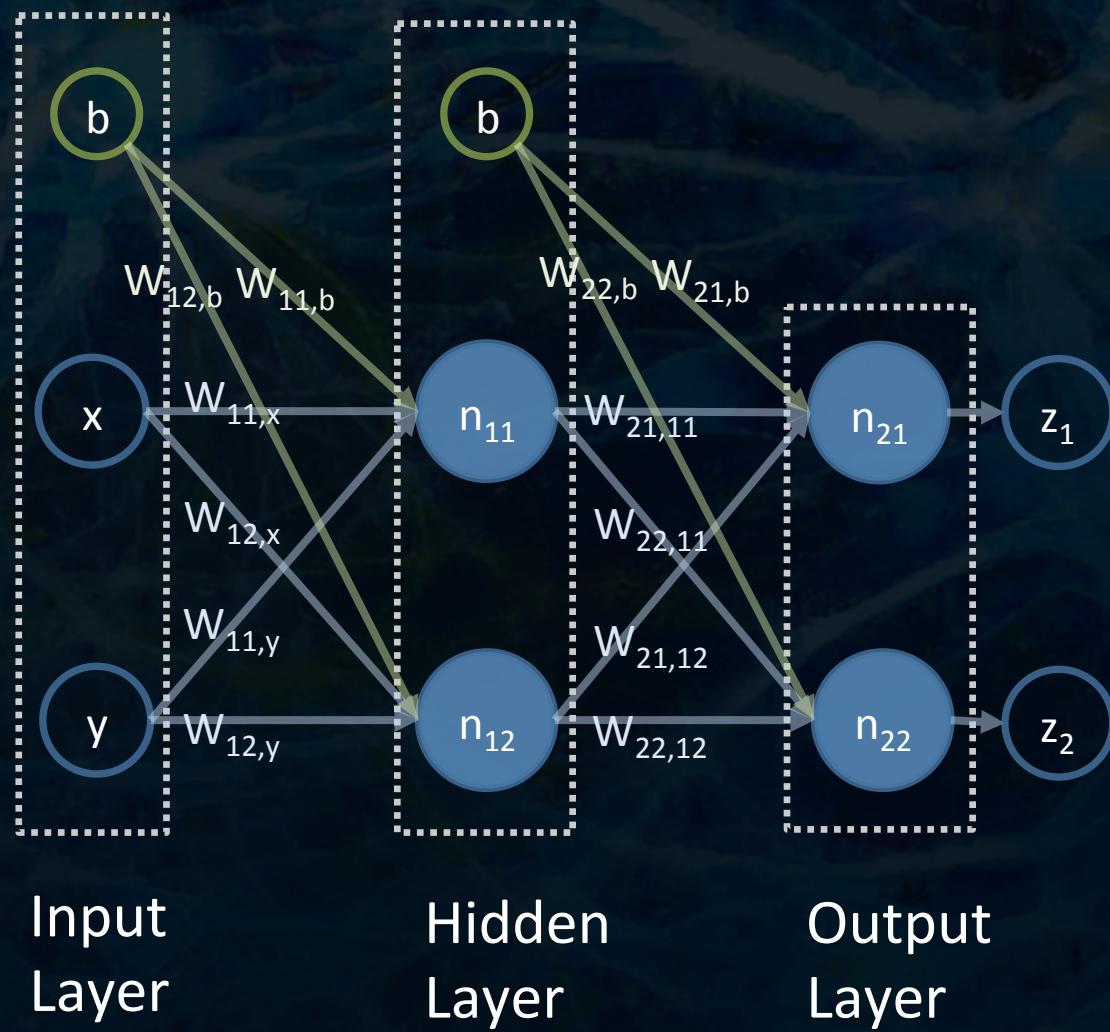
電腦視覺

- Neural Networks
- Convolutional Neural Networks
- VGG 19

Neural Networks

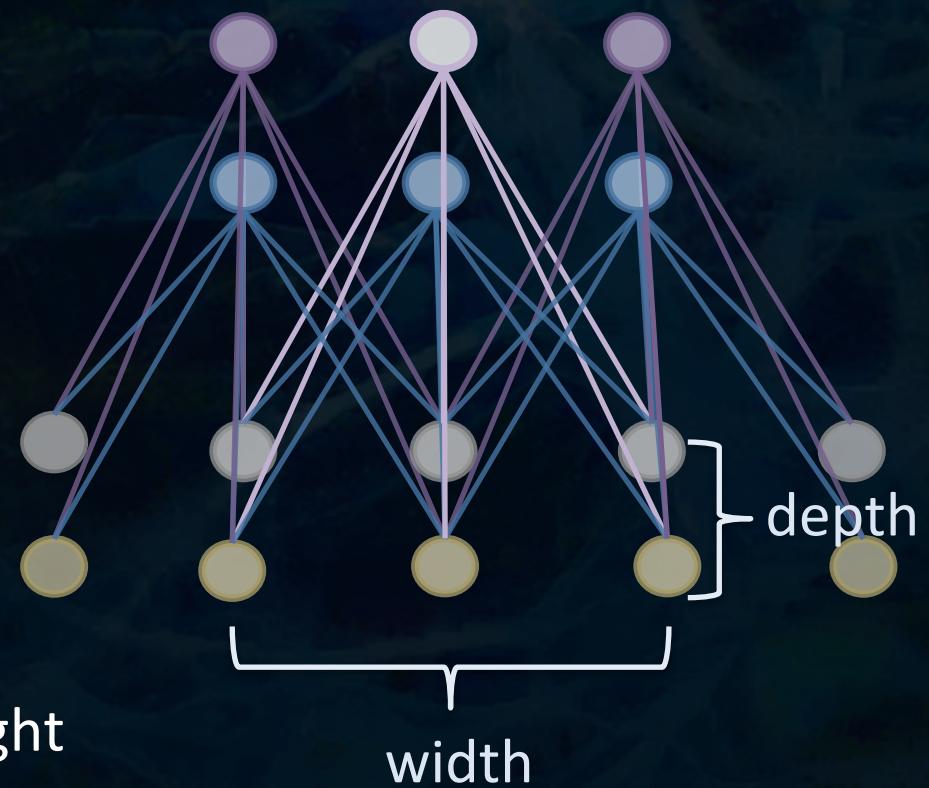
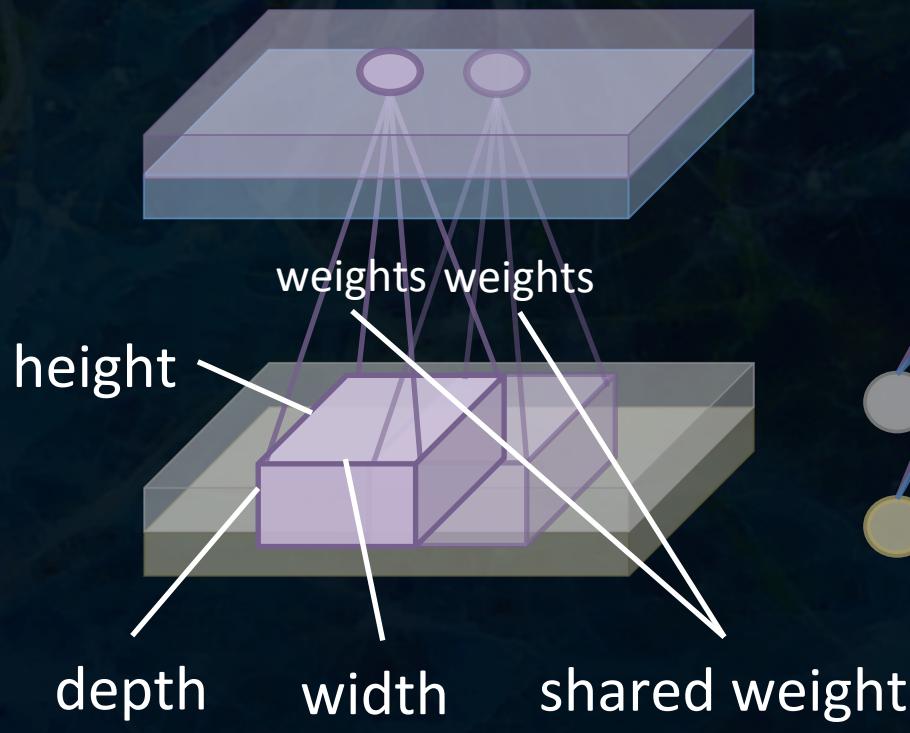


Neural Networks



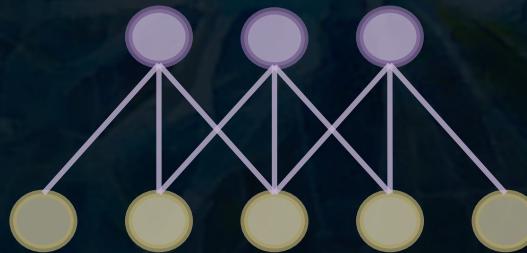
Convolutional Neural Networks

- Convolutional Layer

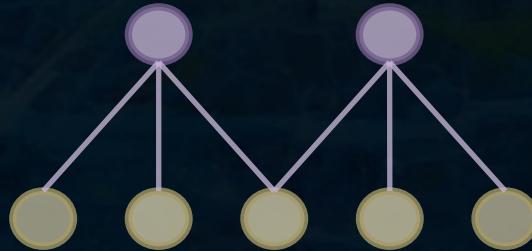


Convolutional Neural Networks

- Stride

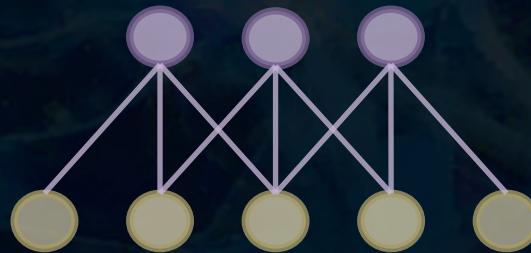


Stride = 1

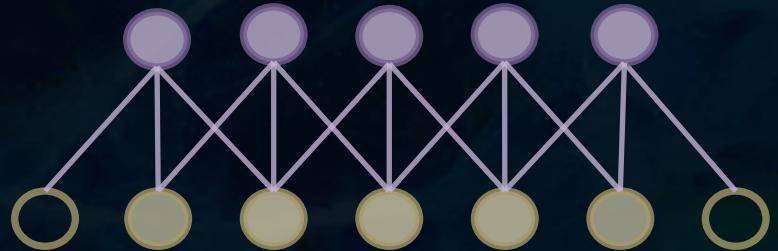


Stride = 2

- Padding



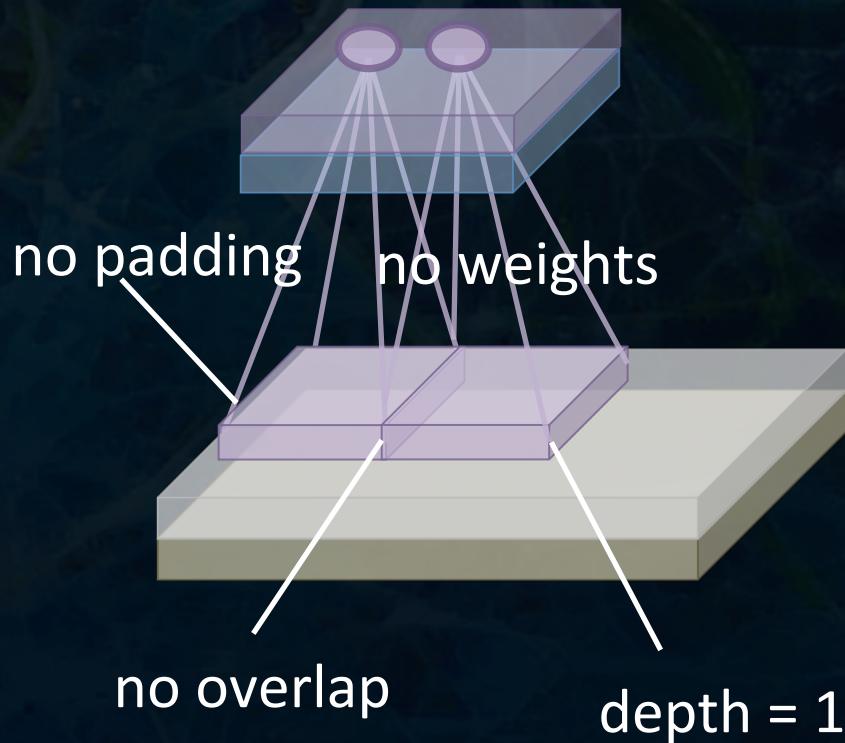
Padding = 0



Padding = 1

Convolutional Neural Networks

- Pooling Layer



Maximum
Pooling

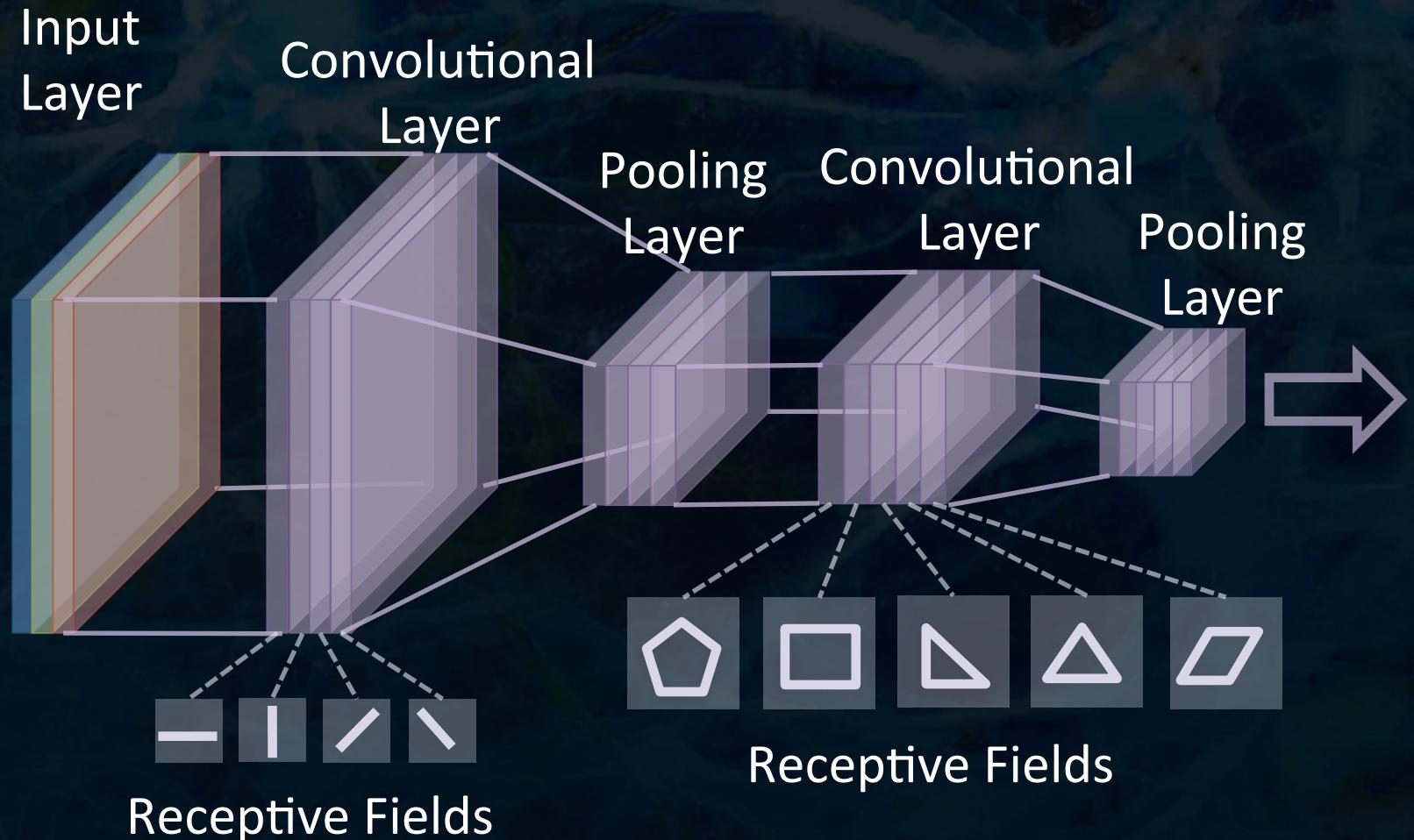
7	8
6	4

Average
Pooling

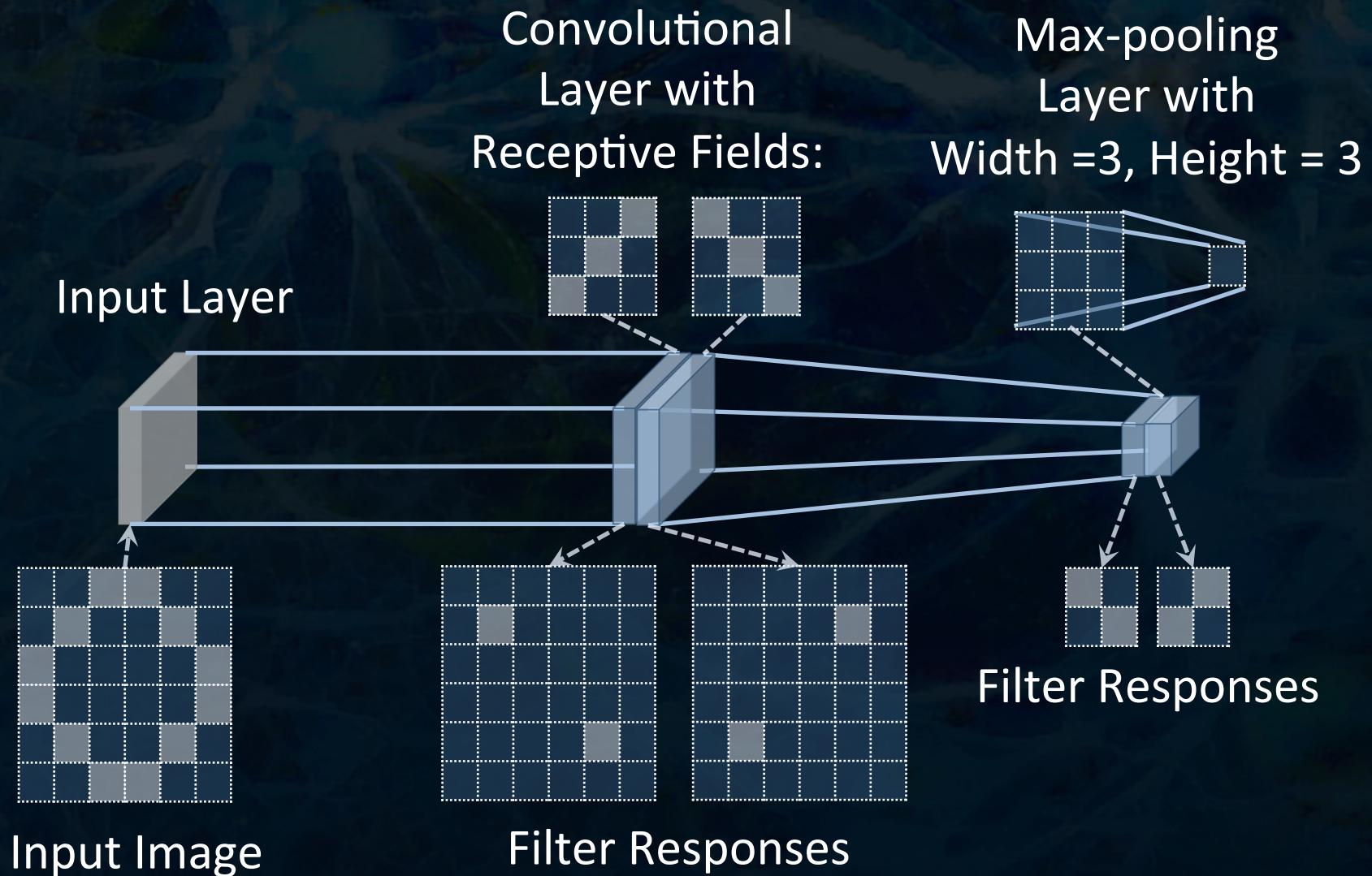
4	5
3	2

1	3	2	4
5	7	6	8
0	0	4	4
6	6	0	0

Convolutional Neural Networks



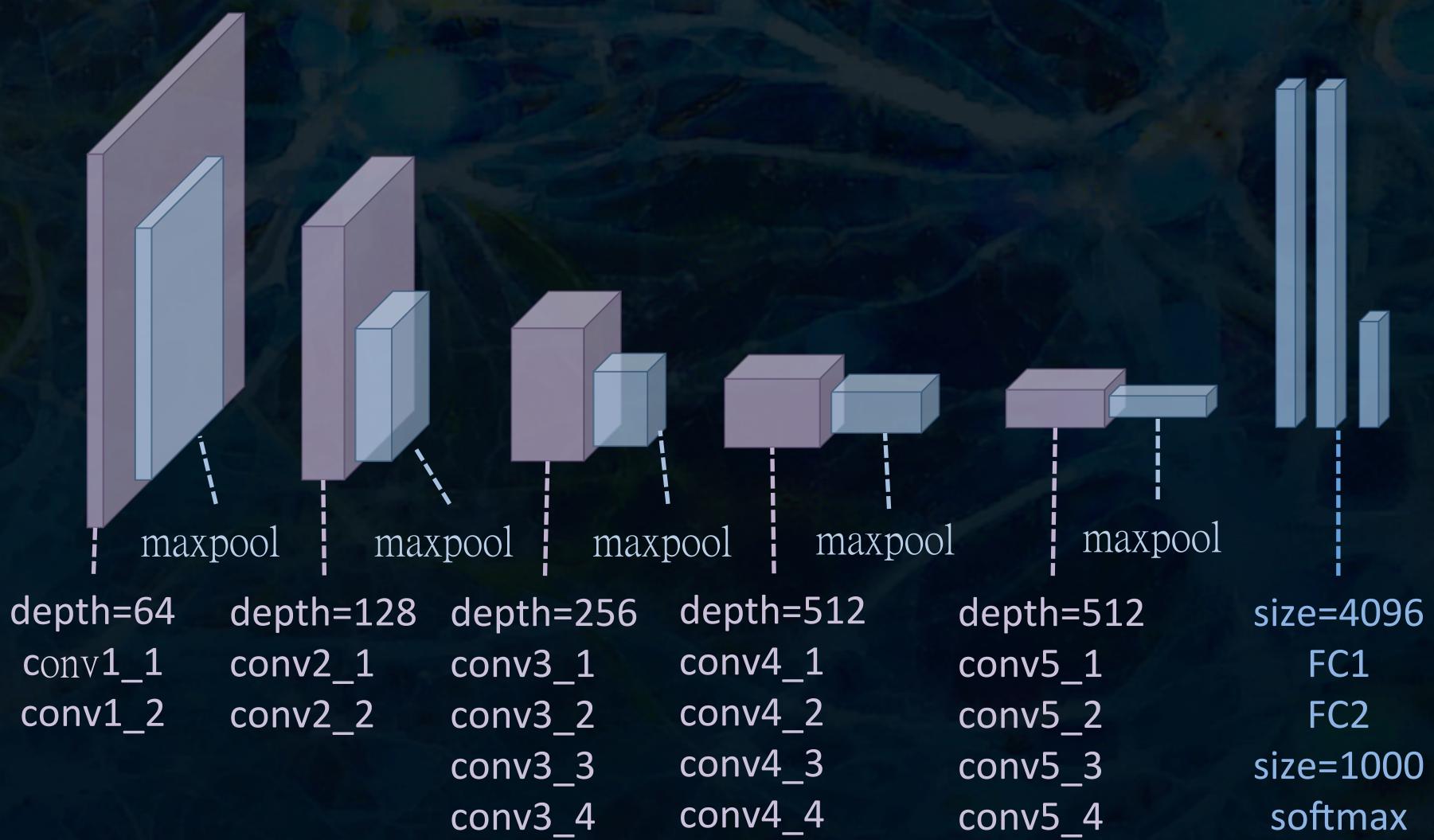
Convolutional Neural Networks



VGG 19

- Karen Simonyan & Andrew Zisserman. Very Deep Convolutional Networks for Large-scale Image Recognition.
- ImageNet Challenge 2014
- 19 (+5) layers
 - 16 Convolutional layers (width=3, height=3)
 - 5 Max-pooling layers (width=2, height=2)
 - 3 Fully-connected layers

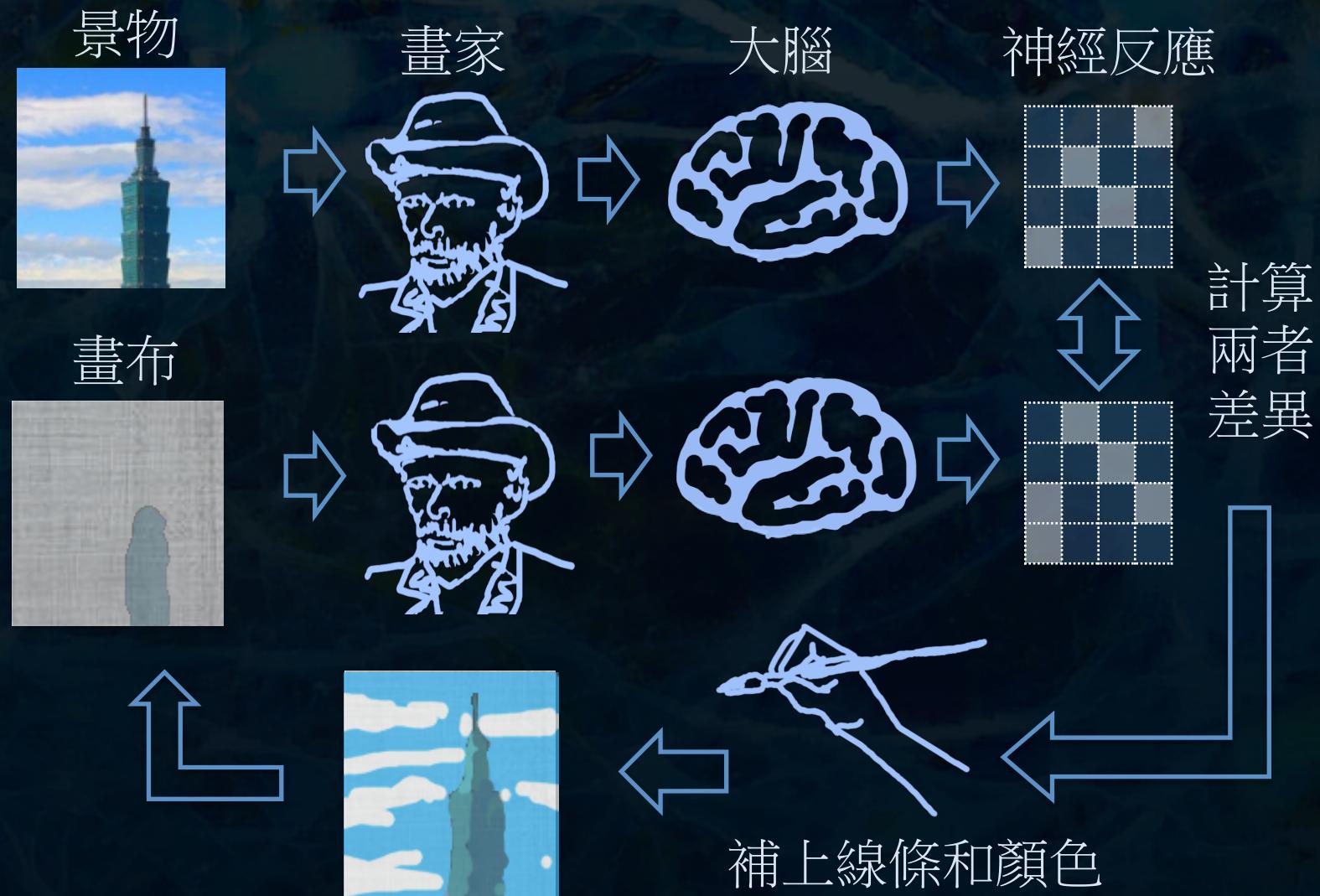
VGG 19



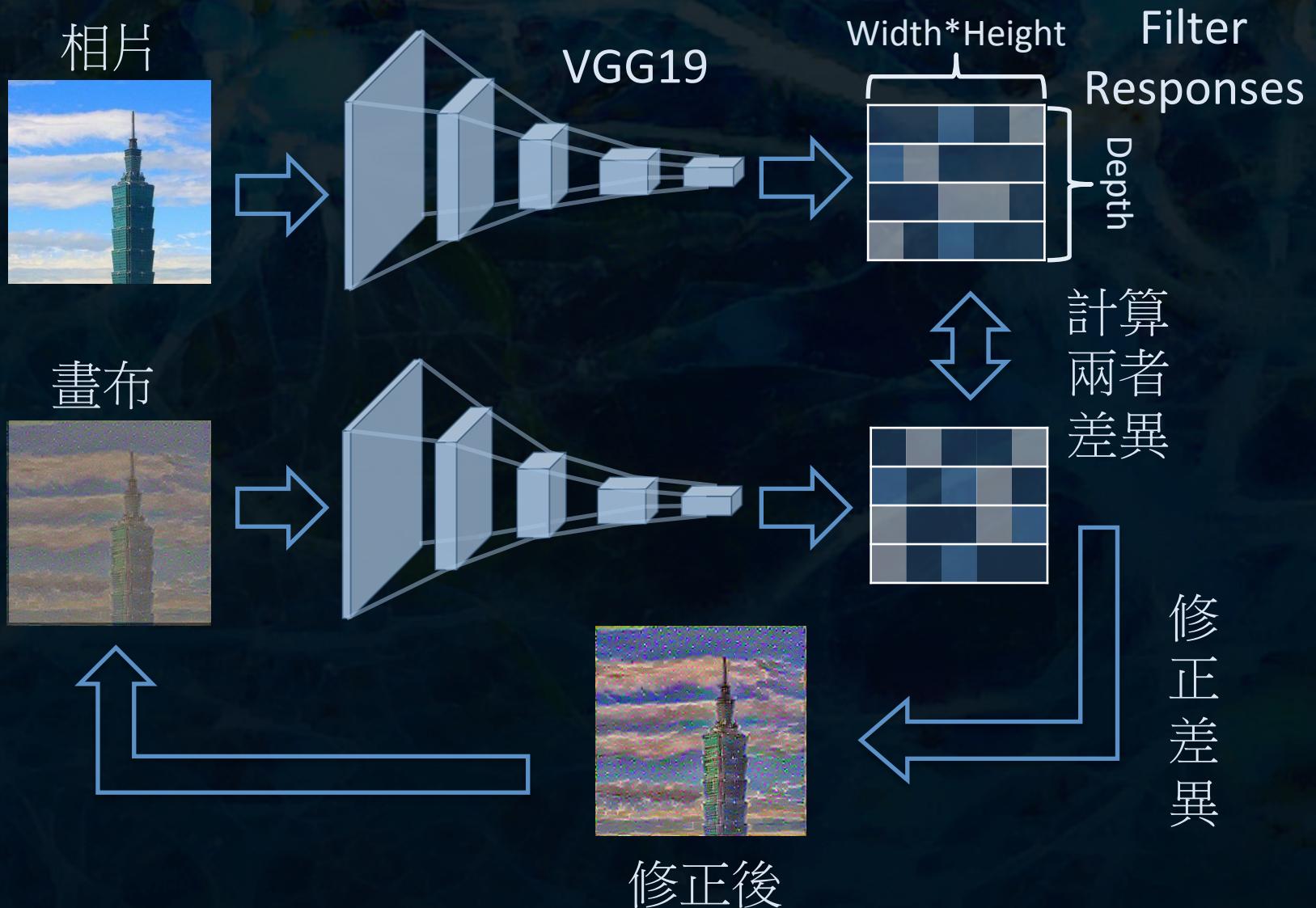
電腦作畫

- 內容產生
- 畫風產生
- 作品產生

內容產生



內容產生



內容產生

Input
Photo: \mathbf{p}



Layer l's Filter
Responses: P^l



Input
Canvas: \mathbf{x}



Layer l's Filter I
Responses: X^l



$$L_{content}(\mathbf{p}, \mathbf{x}, l) = \frac{1}{2} \sum_{i,j} (X_{i,j}^l - P_{i,j}^l)^2$$

$$\frac{\partial L_{content}(\mathbf{p}, \mathbf{x}, l)}{\partial X_{i,j}^l} = X_{i,j}^l - P_{i,j}^l$$

內容產生

- Backward Propagation

Input
Canvas:
 \mathbf{x}



$$\frac{\partial L_{content}}{\partial \mathbf{x}} = \frac{\partial L_{content}}{\partial X^l} \frac{\partial X^l}{\mathbf{x}}$$

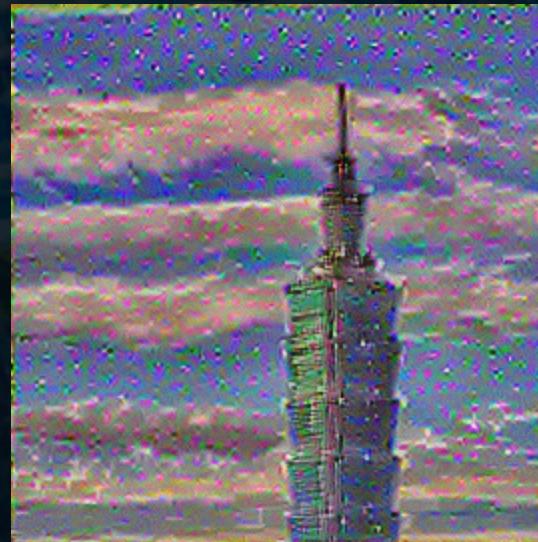
Update
Canvas



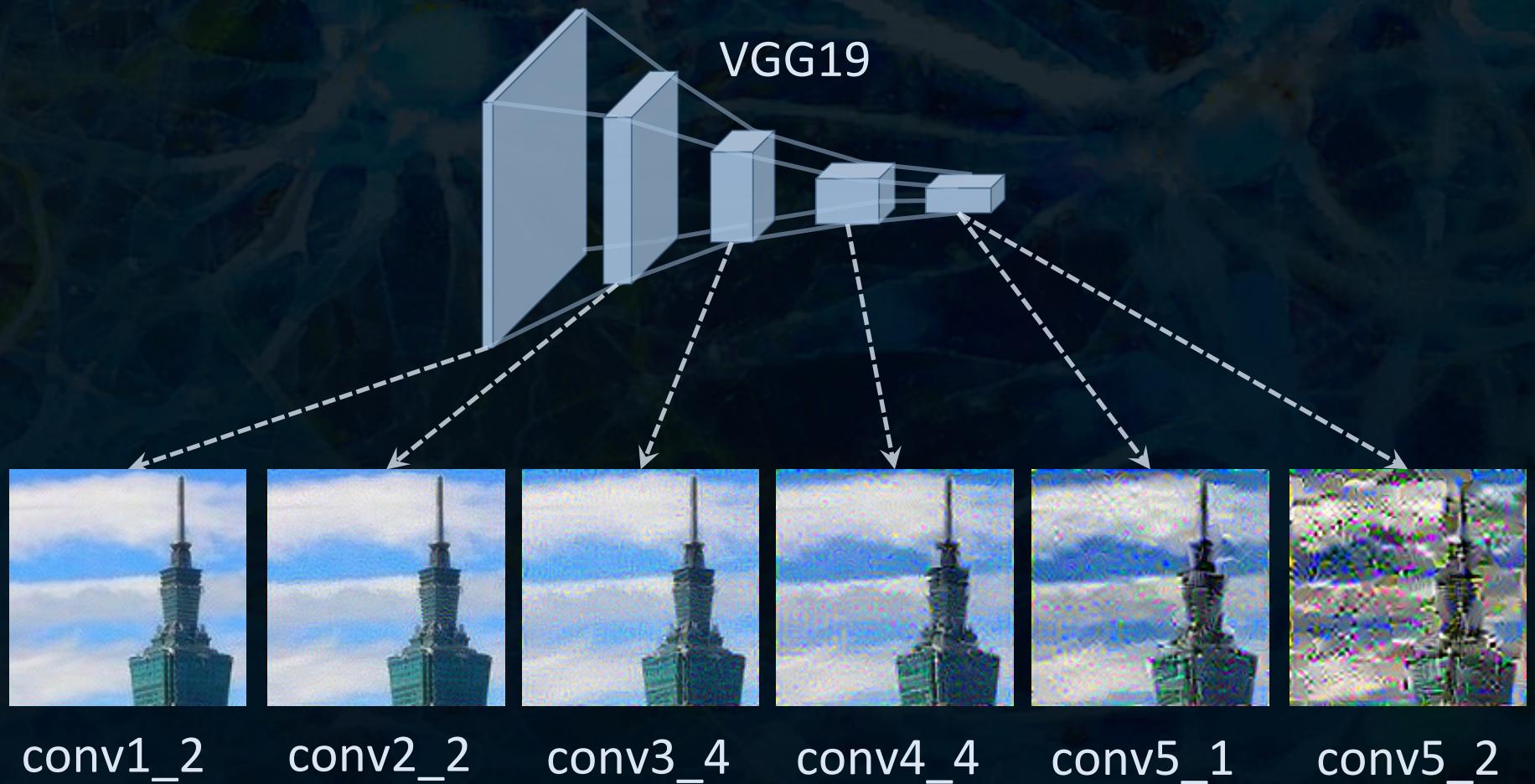
$$\mathbf{x} \leftarrow \mathbf{x} - \eta \frac{\partial L_{content}}{\partial \mathbf{x}}$$

Learning Rate

內容產生



內容產生

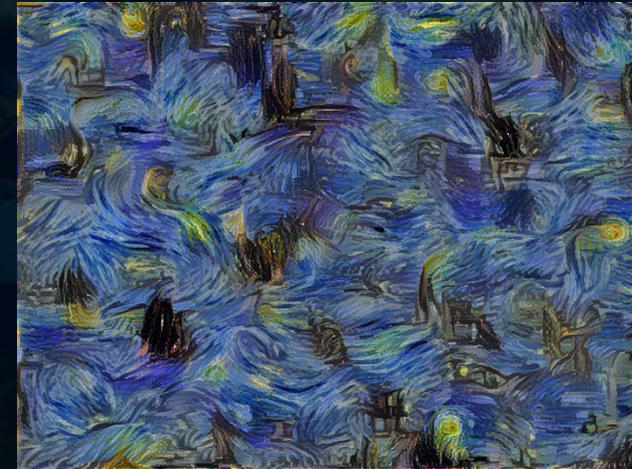


畫風產生

- “Style” is position-independent



style
extraction



畫風產生

畫作



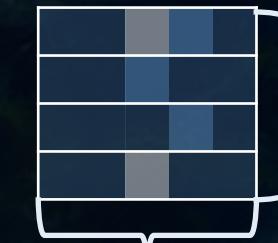
VGG19



Filter Responses



Gram Matrix



Depth

Width*Height

Position-
dependent

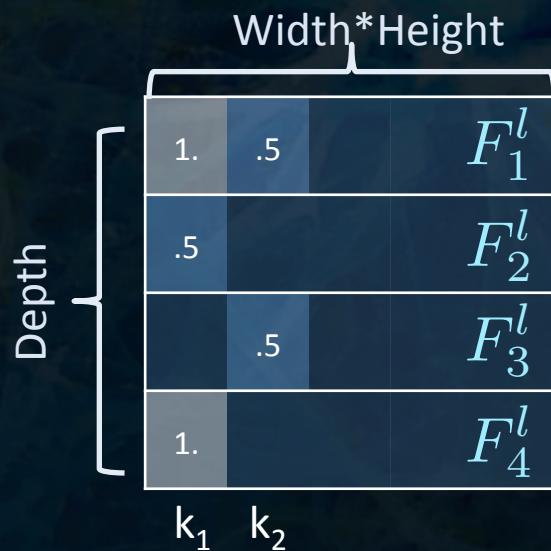


Depth

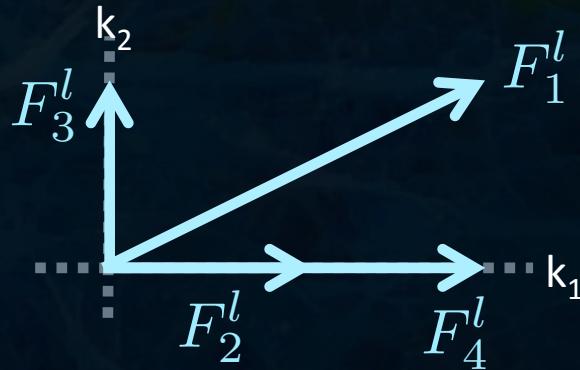
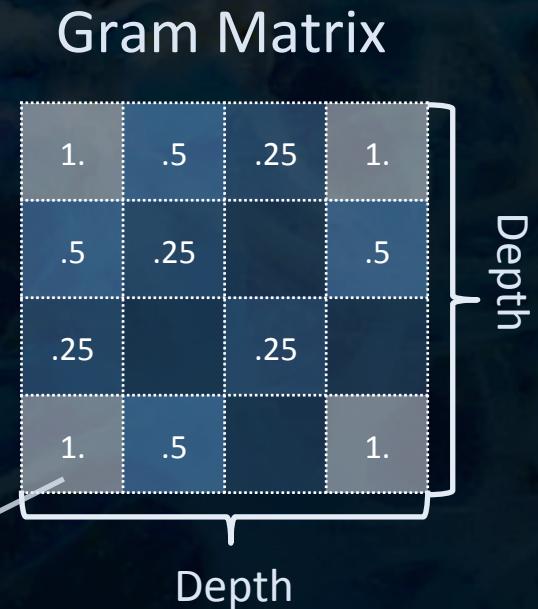
Position-
independent

畫風產生

Layer l's Filter Responses



$$G$$
$$G_{i,j}^l = F_i^l \cdot F_j^l$$

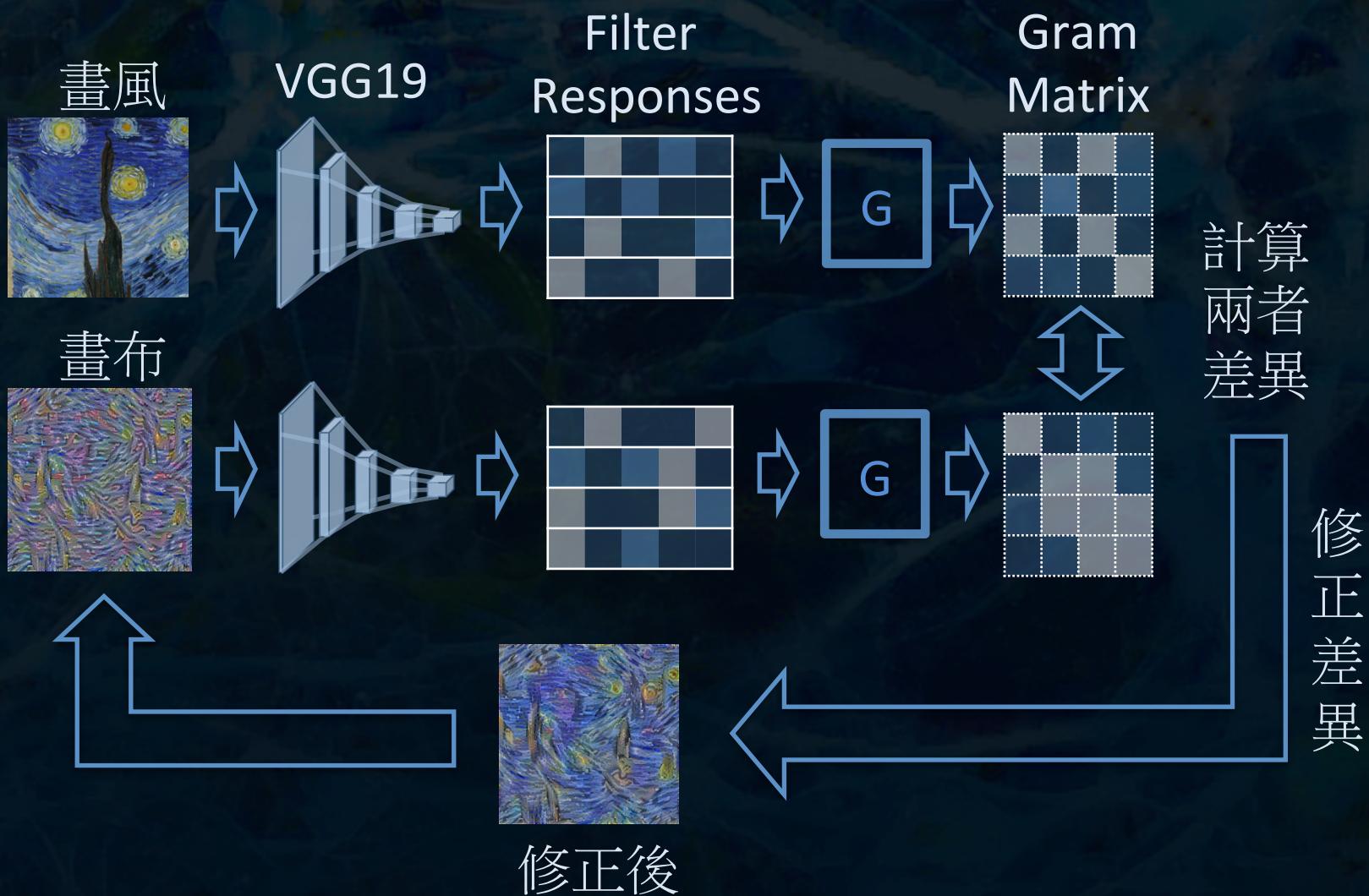


$$G_{4,1}^l = F_4^l \cdot F_1^l$$

$$= 1 \times 1 + 0 \times 0.5 + 0 \times 0 + \dots$$

$$= 1$$

畫風產生



畫風產生

Input
Artwork: \mathbf{a} Layer l's
Gram Matrix



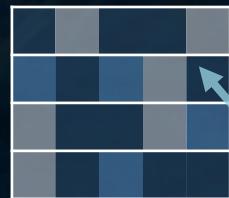
$$A_{i,j}^l$$


Input
Canvas: \mathbf{x} Layer l's
Gram Matrix



$$X_{i,j}^l$$


Layer l's
Filter Responses

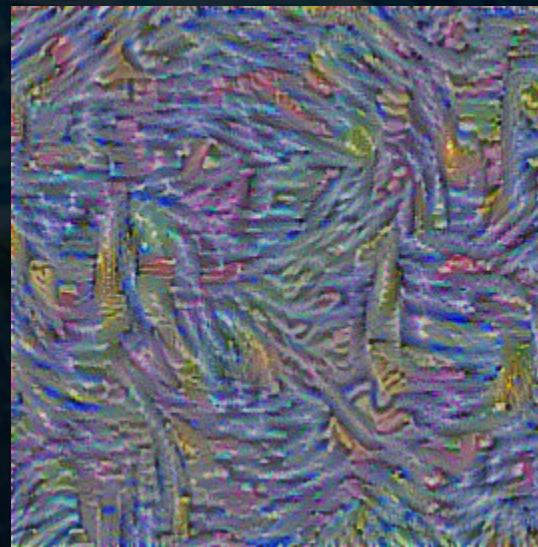
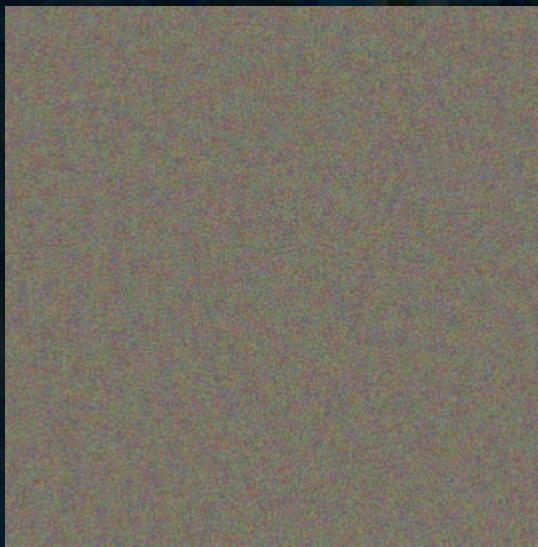


$$F_{i,j}^l$$

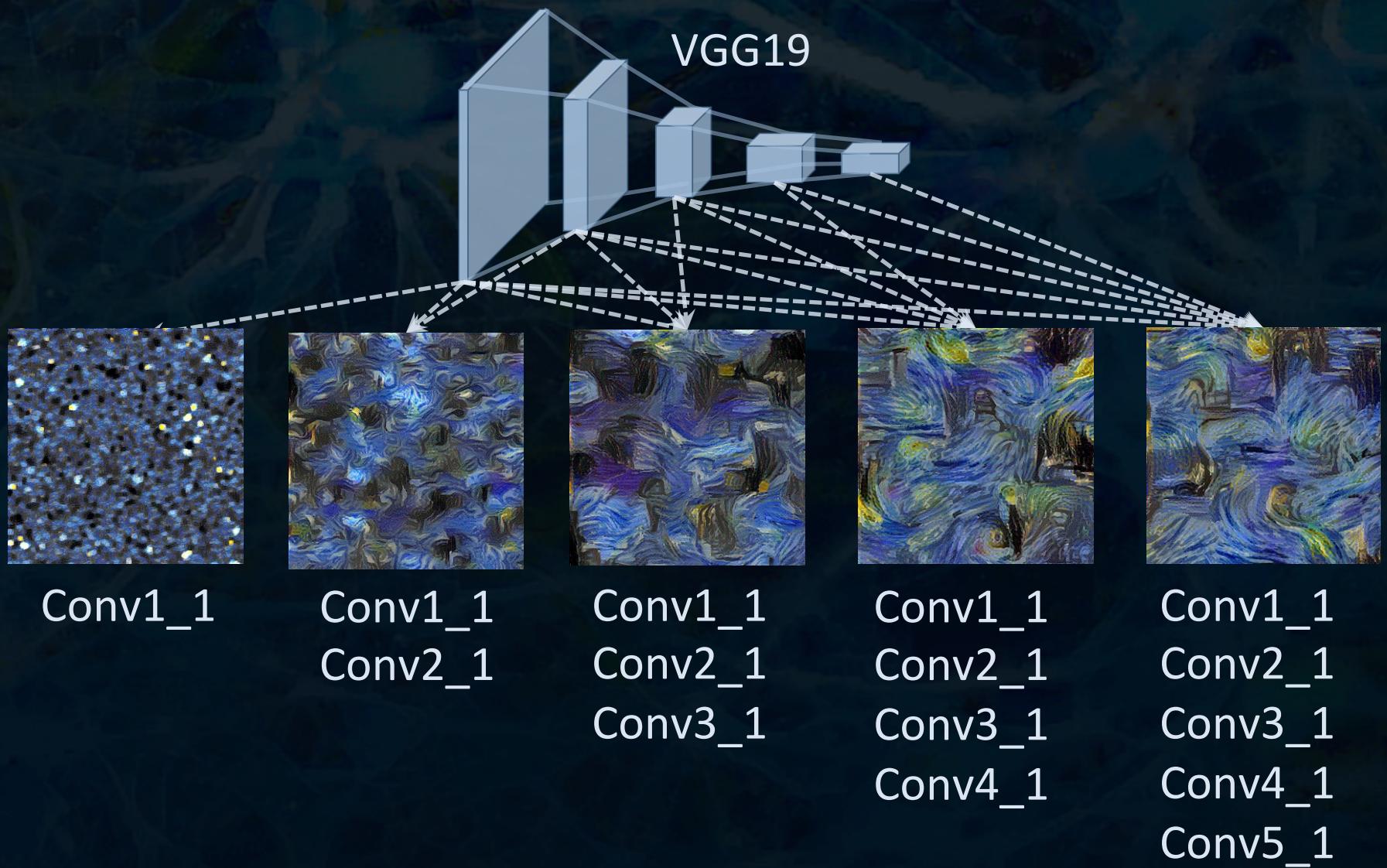

$$L_{style}(\mathbf{a}, \mathbf{x}, l) = \frac{1}{2} \sum_{i,j} (X_{i,j}^l - A_{i,j}^l)^2$$

$$\frac{\partial L_{style}(\mathbf{a}, \mathbf{x}, l)}{\partial F_{i,j}^l} = ((F^l)^T (X^l - A^l))_{j,i}$$

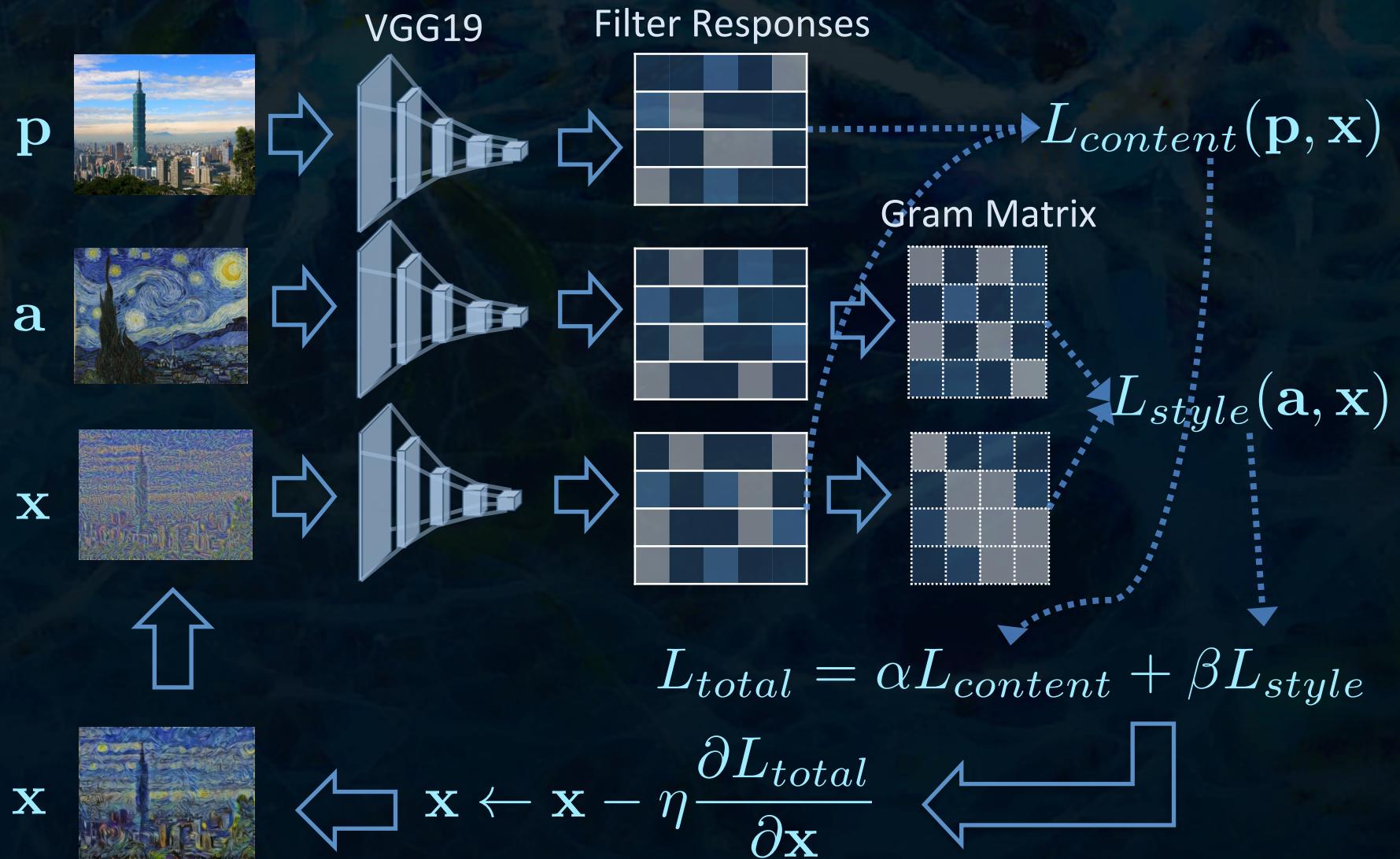
畫風產生



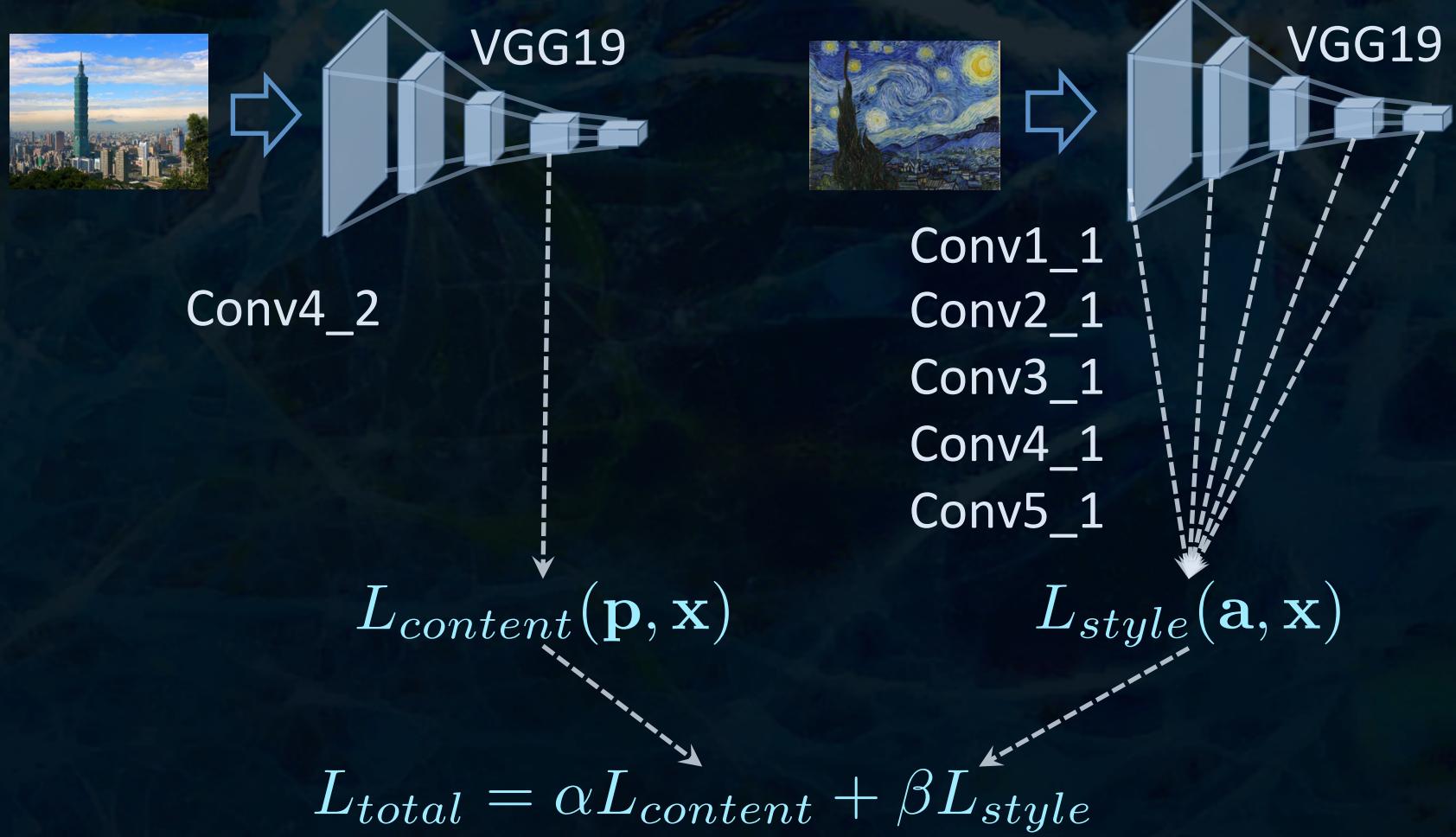
畫風產生



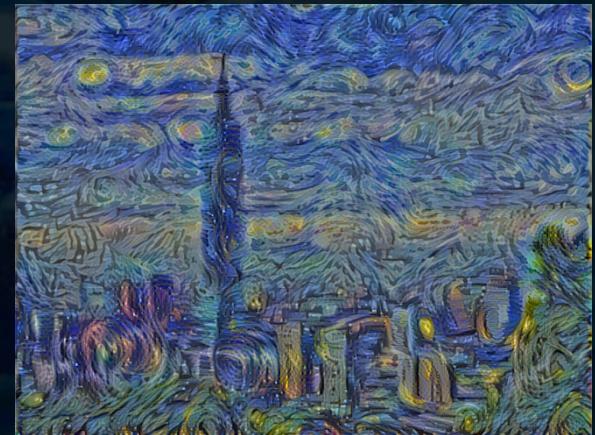
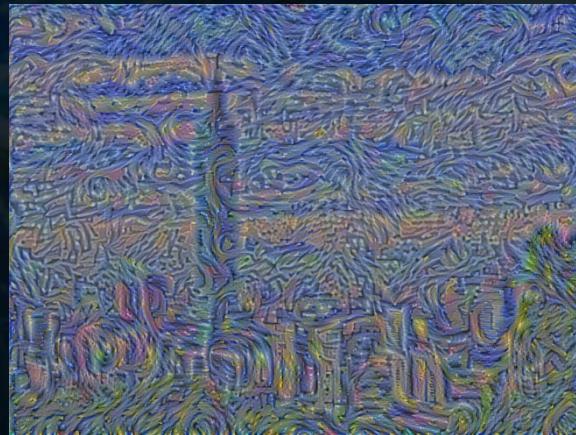
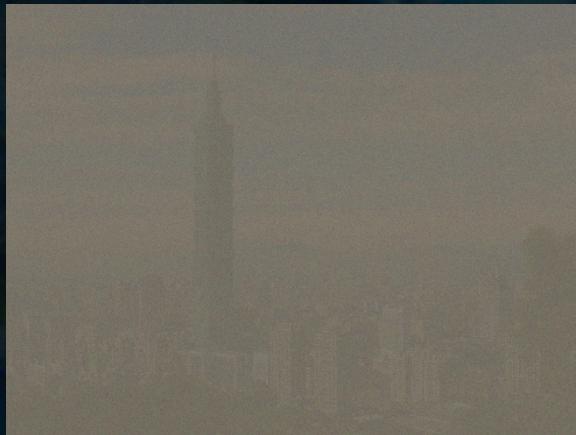
作品產生



作品產生



作品產生

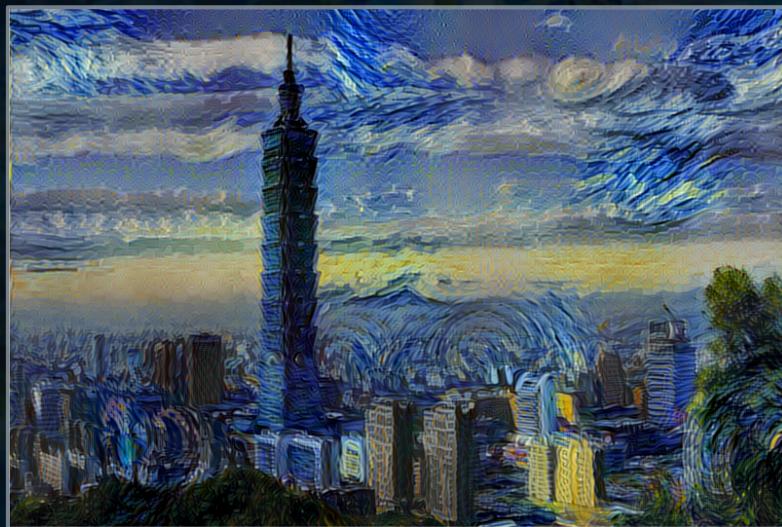


作品展示

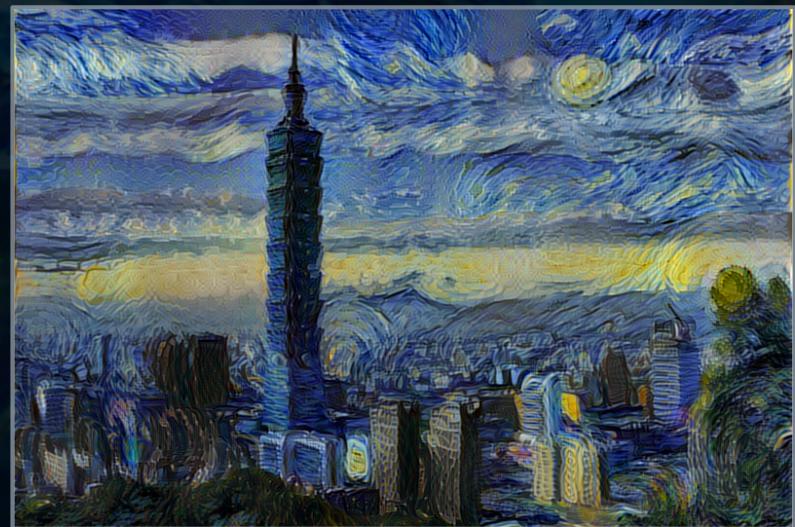
- 內容 v.s. 畫風
- 不同起始狀態
- 不同VGG Layers
- 素描、水彩
- 詩中有畫、畫中有詩

內容 v.s. 畫風

$\frac{\alpha}{\beta}$



0.15



0.05



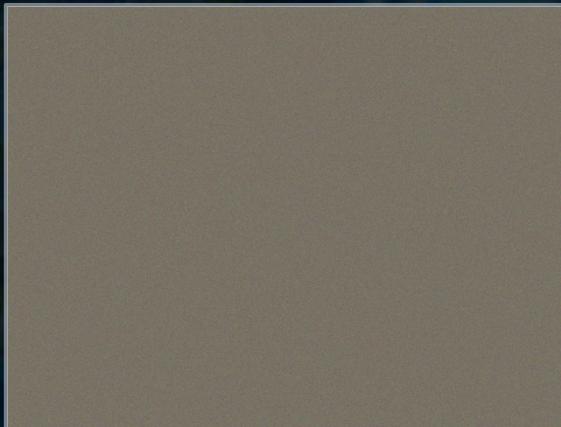
0.02



0.007

不同起始狀態

noise



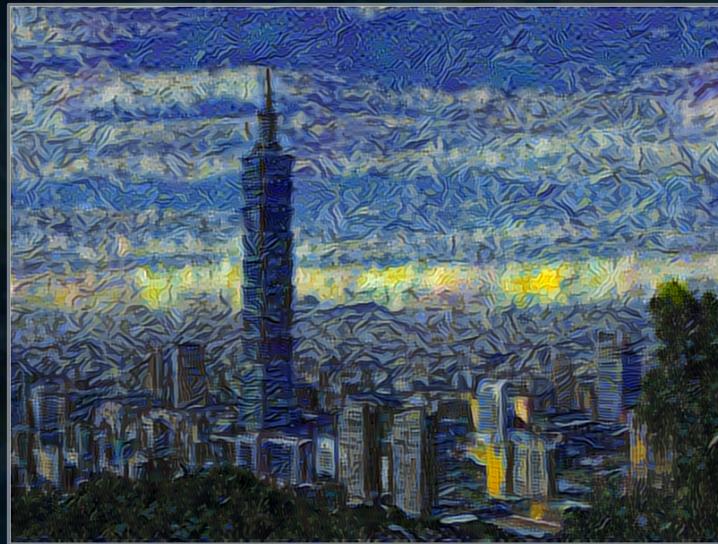
$0.9 * \text{noise} + 0.1 * \text{photo}$



photo



不同VGG Layers



Conv1_1
Conv2_1



$$\frac{\alpha}{\beta} = 0.002$$

Conv1_1
Conv2_1
Conv3_1

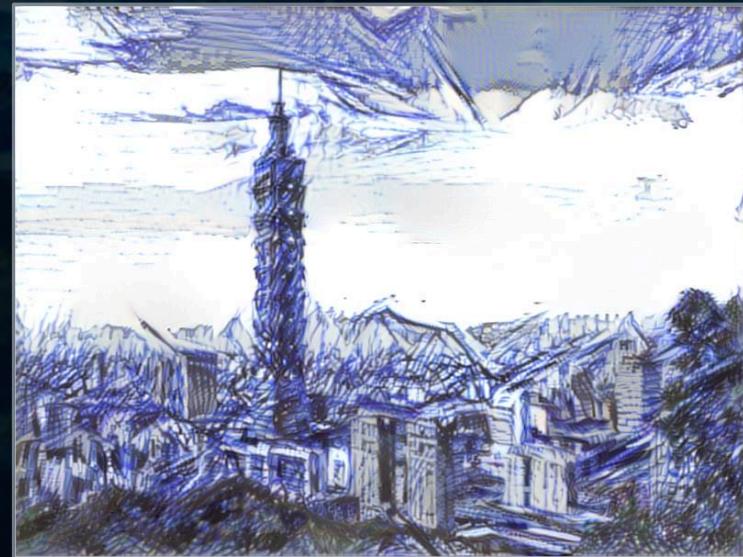
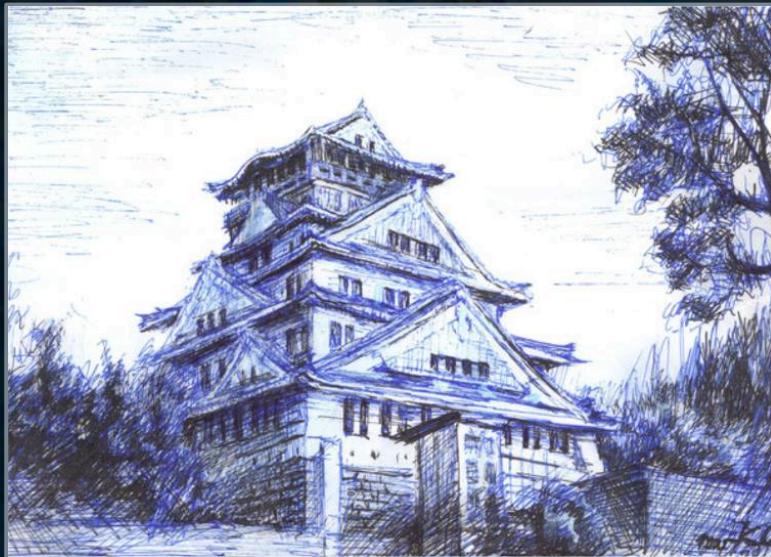


Conv1_1
Conv2_1
Conv3_1
Conv4_1



Conv1_1
Conv2_1
Conv3_1
Conv4_1
Conv5_1

素描、水彩



詩中有畫、畫中有詩



延伸閱讀

- A Neural Algorithm of Artistic Style
 - <http://arxiv.org/abs/1508.06576>
- Texture Synthesis Using Convolutional Neural Networks
 - <http://arxiv.org/abs/1505.07376>
- Convolutional Neural Network
 - <http://cs231n.github.io/convolutional-networks/>
- Neural Network Back Propagation
 - <http://cpmarkchang.logdown.com/posts/277349-neural-network-backward-propagation>
- 電腦賦詩：
 - <http://www.slideshare.net/ckmarkohchang/computational-poetry>

程式碼

- Python Tensorflow
 - https://github.com/ckmarkoh/neuralart_tensorflow
- Python Theano
 - <https://github.com/woonketwong/artify>
- Python Theano (ipython notebook)
 - <https://github.com/Lasagne/Recipes/blob/master/examples/styletransfer/Art%20Style%20Transfer.ipynb>
- Python deeppy
 - https://github.com/andersbll/neural_artistic_style

圖片來源



- <http://www.taipei-101.com.tw/upload/news/201502/2015021711505431705145.JPG>



- https://github.com/andersbli/neural_artistic_style/blob/master/images/starry_night.jpg?raw=true

特別感謝

- 臺大資工imlab