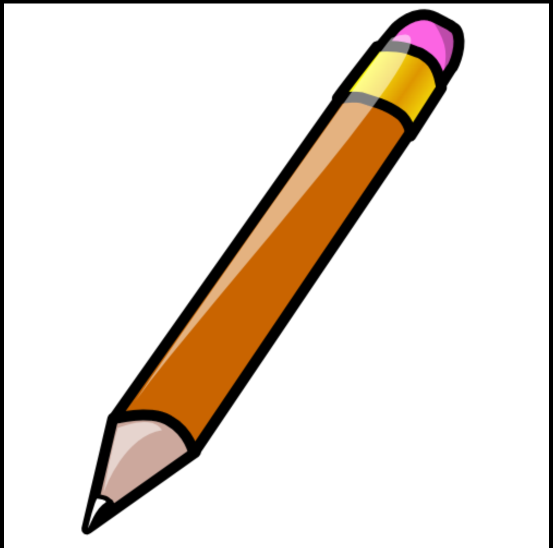
Combining Sketch and Tone for Pencil Drawing Production

Cewu Lu, Li Xu, Jiaya Jia,

The Chinese University of Hong Kong



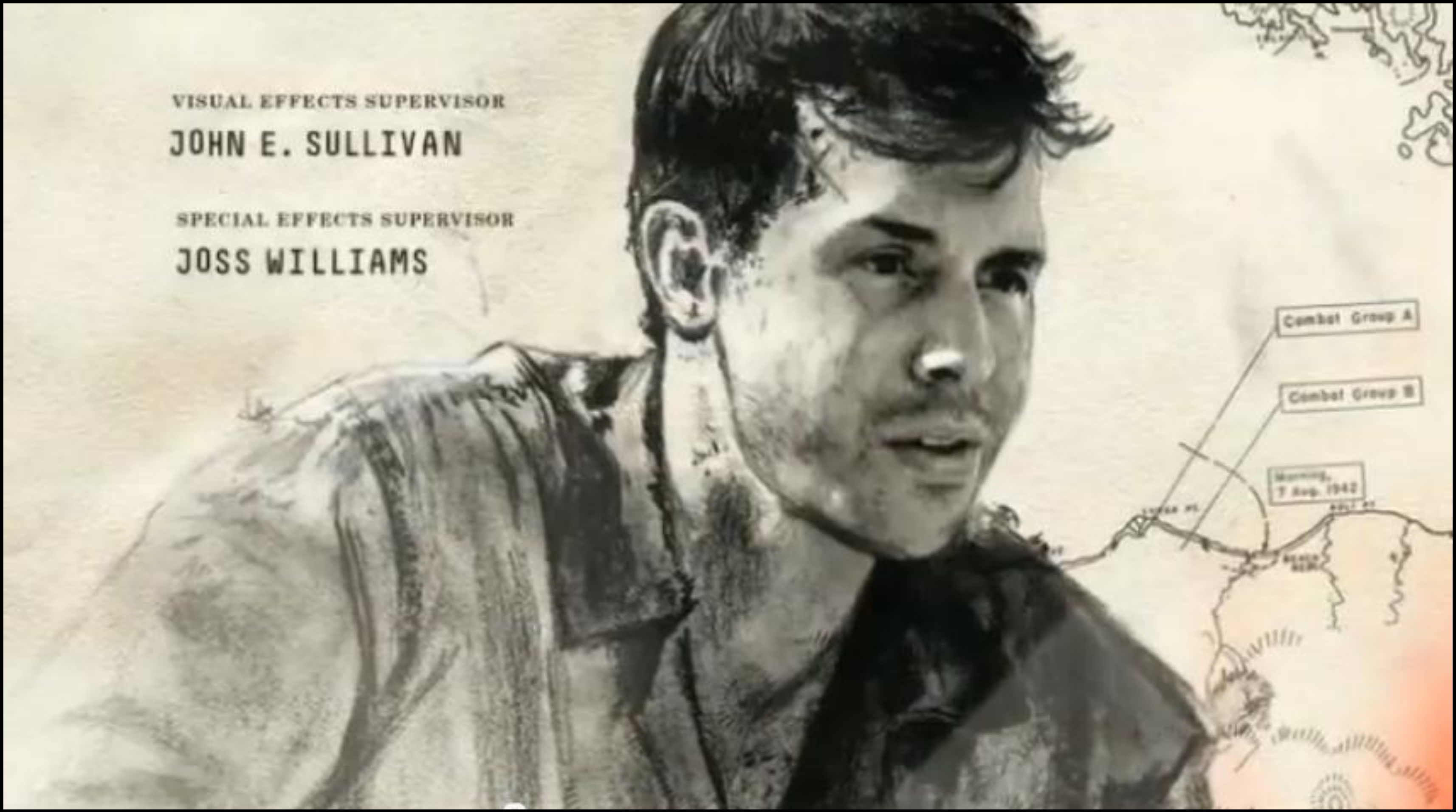
Pencil Sketch

Fundamental Pictorial Language

Popular Artistic Forms

High in real-work

Episode of <The Pacific>


VISUAL EFFECTS SUPERVISOR
JOHN E. SULLIVAN
SPECIAL EFFECTS SUPERVISOR
JOSS WILLIAMS
Combat Group A
Combat Group B
Morning,
7 Aug. 1942

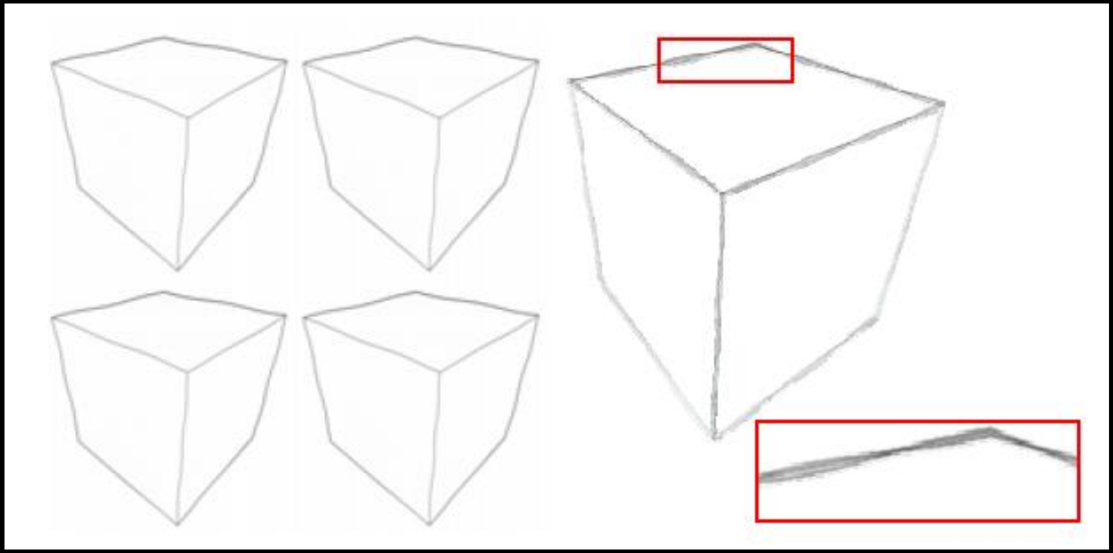
Outline

- Previous work
- Observation (Two stages)
- Framework (Pencil drawing prior included)
- Discussion
- Comparison and Results
- Conclusion

Outline

- **Previous Work**
- Observation (Two stages)
- Framework (Pencil drawing prior including)
- Discussion
- Comparison and Results
- Conclusion

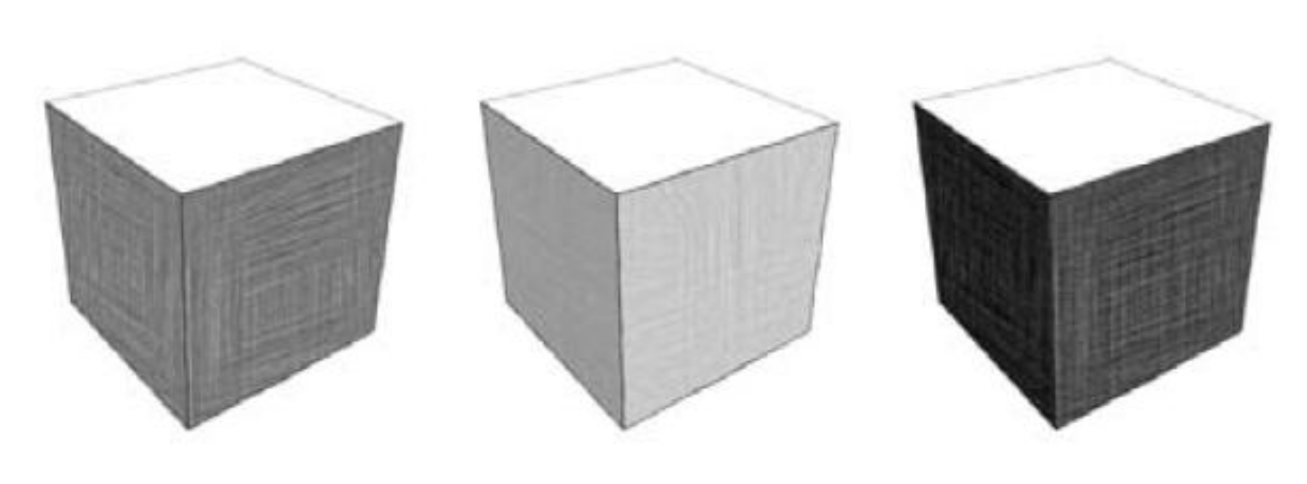
Previous Work (Model-based)



Accurate Line and Contour → Pencil Line Drawing

[Sousa and Buchanan 1999] [Lake et al. 2000] [Lee et al. 2006]

Previous Work (Model-based)

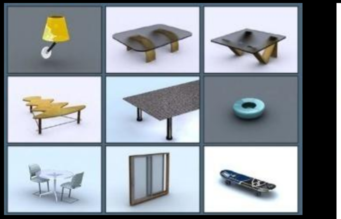


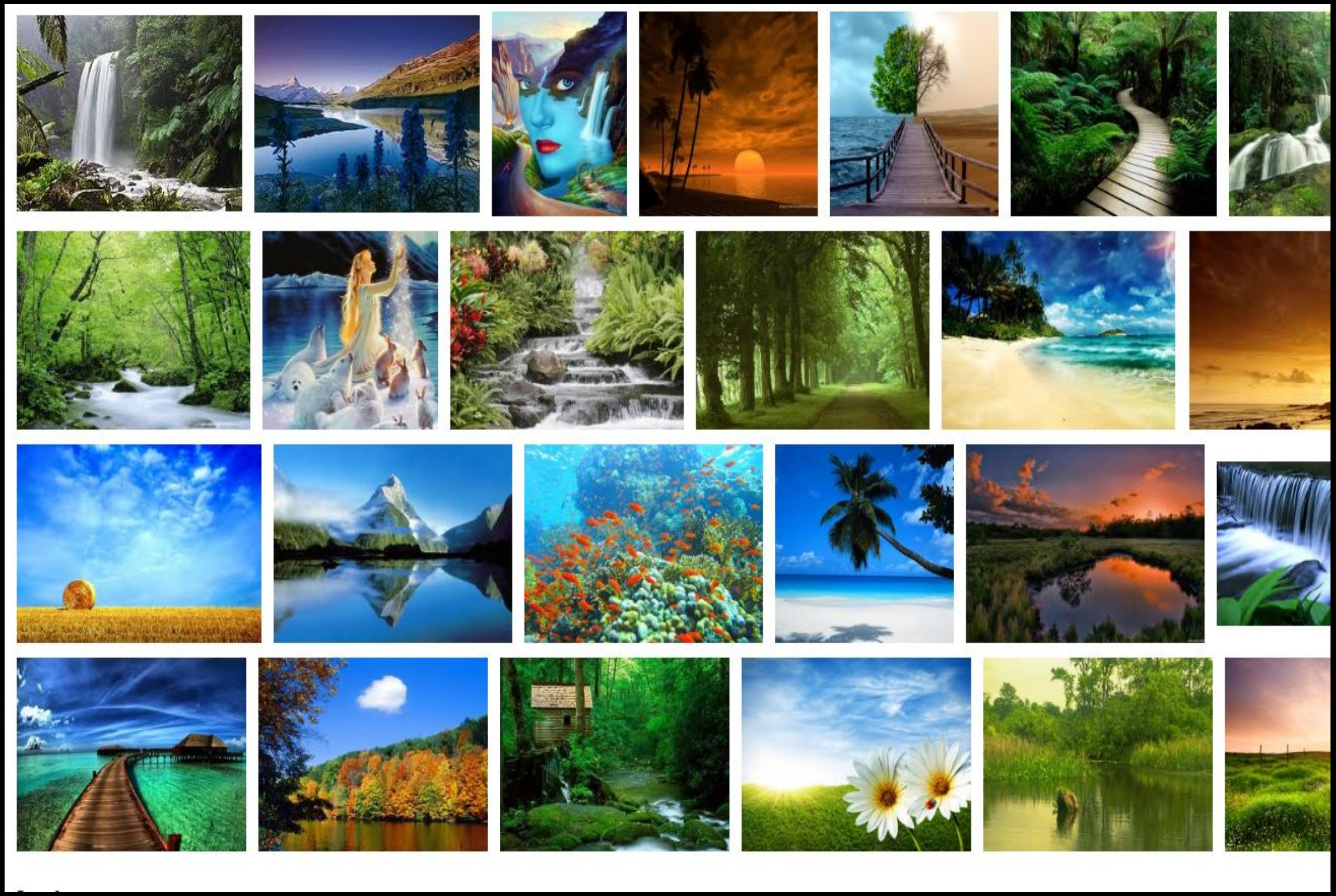
Surface Normal
Lighting Information
Geometry
Information

Automatic Hatching
Generation

Previous Work (Image-based)

Previous Work (Image-based)





Previous Work (Image-based)

- Li and Huang 2003
- Sun et al. 2007
- Gao et al. 2010

Previous Work (Image-based)

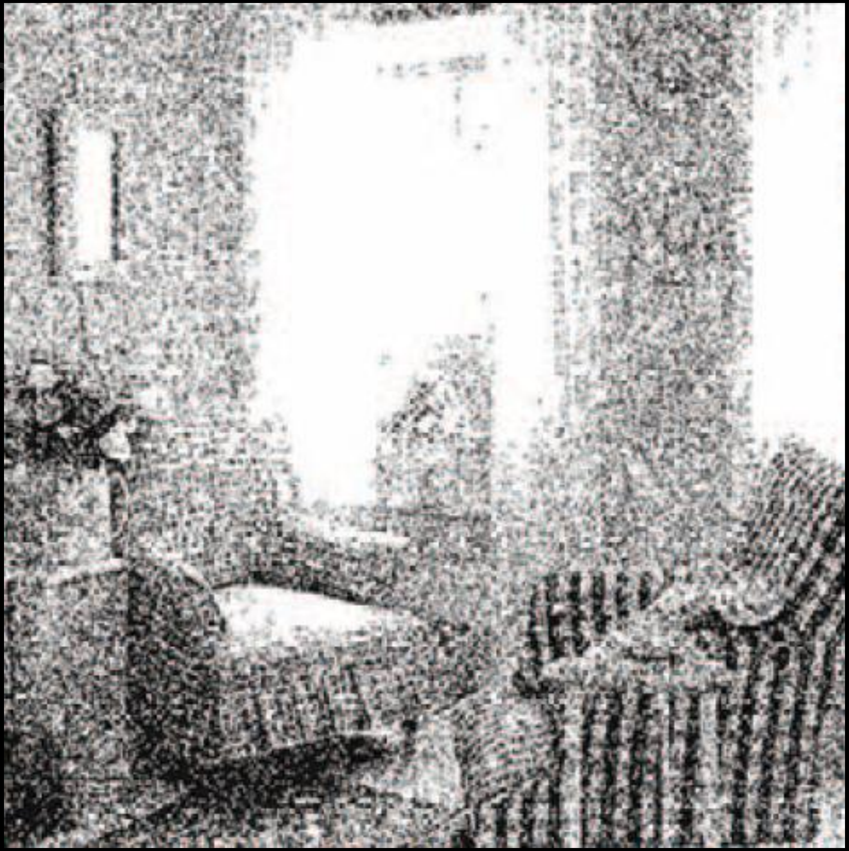
- Li and Huang 200
- Sun et al. 2007
- Gao et al. 2010

Previous Work (Image-based)

- Li and Huang. 2003
- Sun et al. 2007
- Gao et al. 2010

Previous Work (Image-based)

- Li and Huang. 2003
- Sun et al. 2007
- Gao et al. 2010



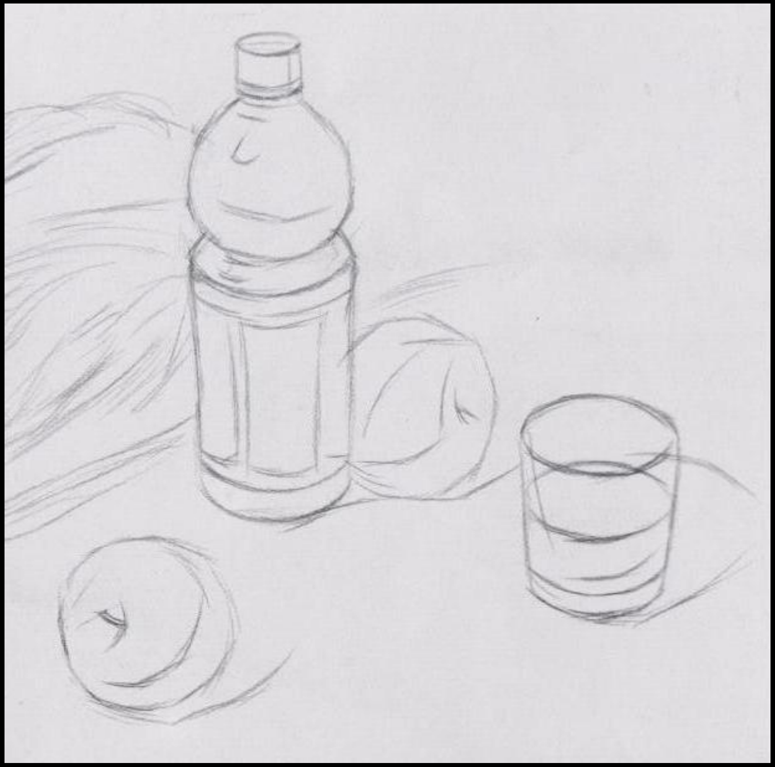
UNIVERSITY OF LONDON
SENATE HOUSE AND LIBRARY

PHILIPS
OLYMPUS
EPSON
TOSHIBA
TCL
MGM
HITACHI
Allianz
DBAY

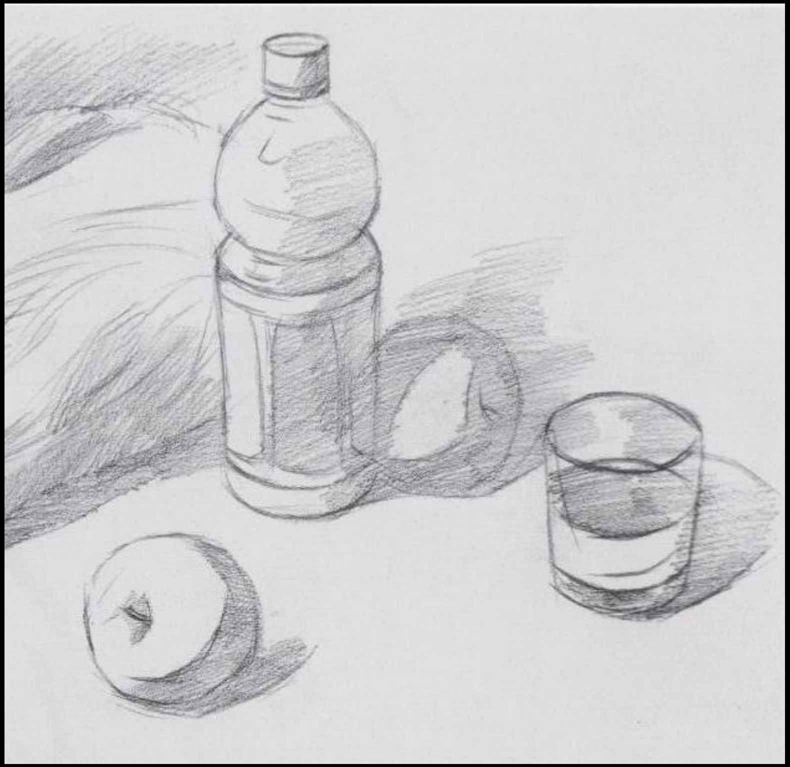
Outline

- Previous work
- **Observation (Two stages)**
- Framework (Pencil drawing prior including)
- Discussion
- Comparison and Results
- Conclusion

Observation



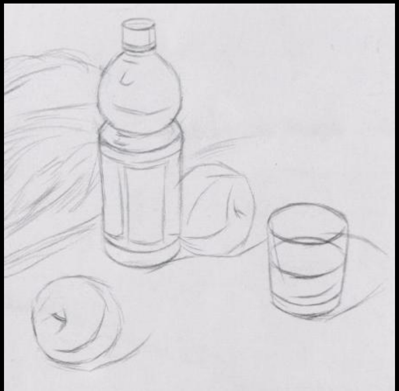
First Stage



Second Stage

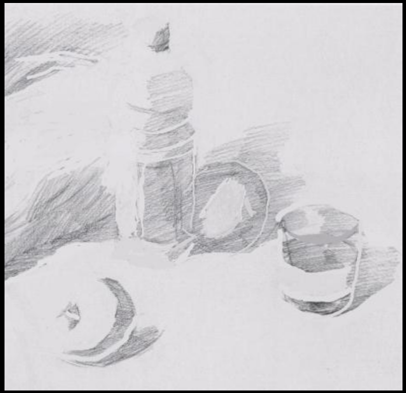
Human Drawing Procedure

Observation



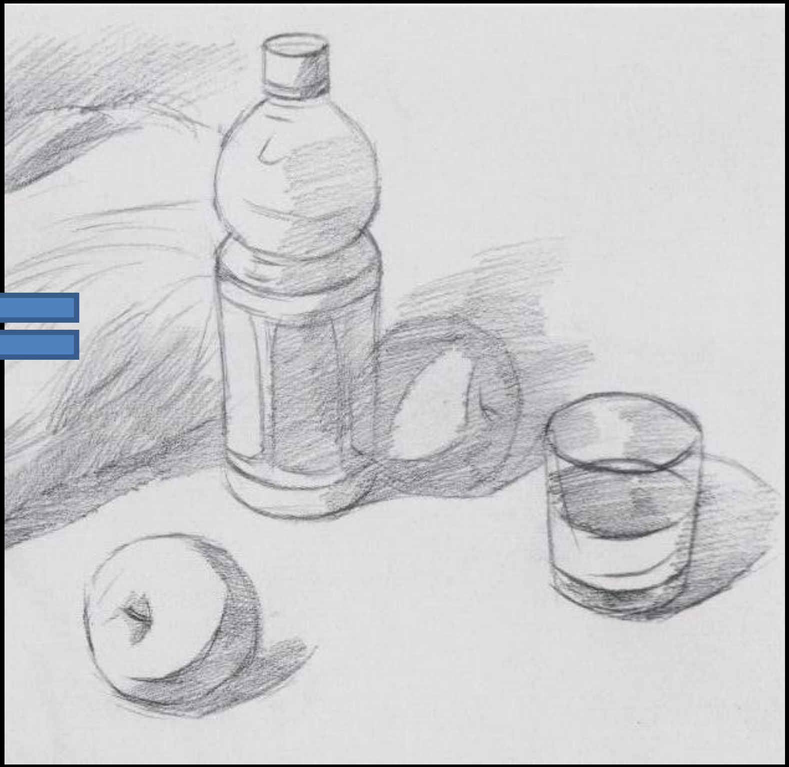
Structure

+

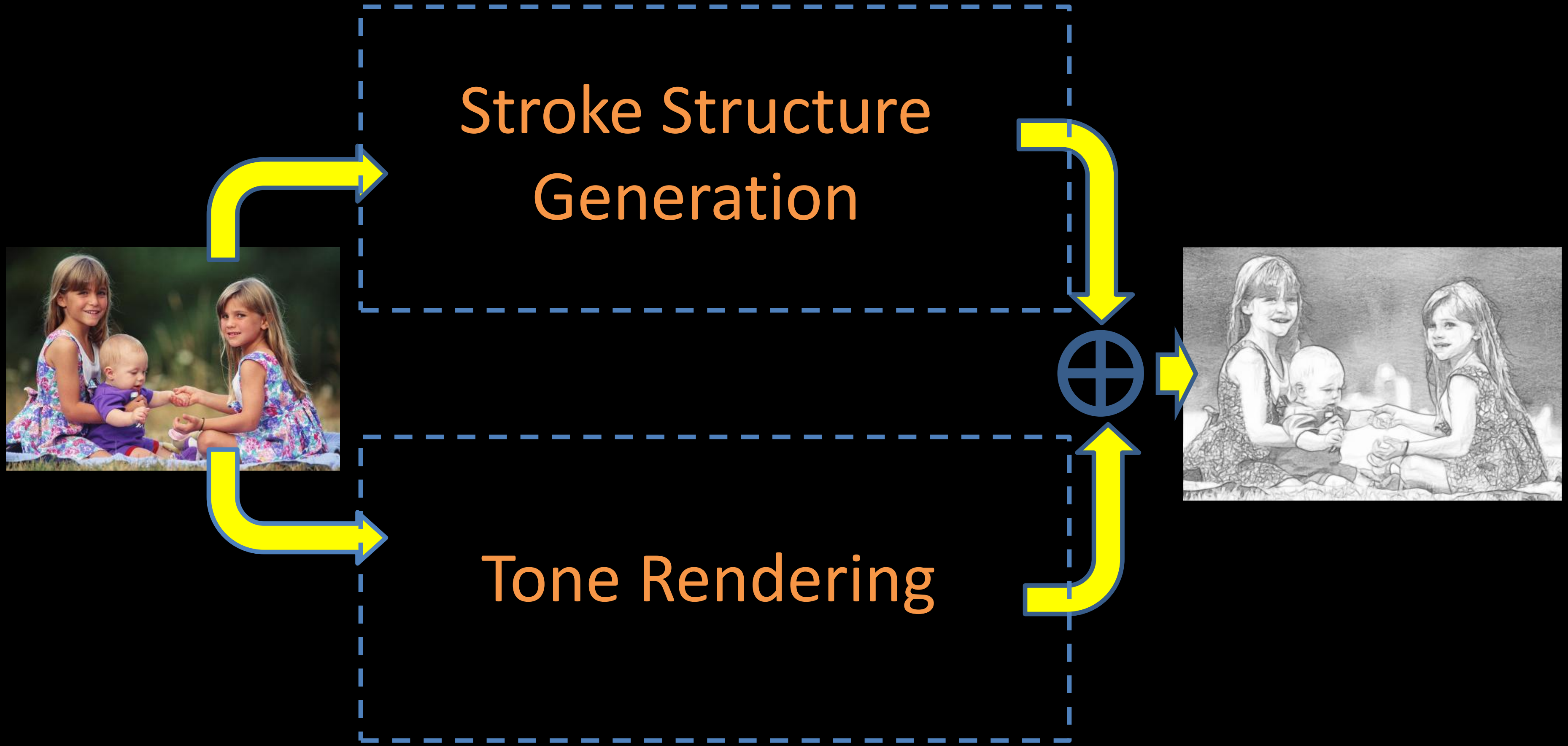


Tone

=



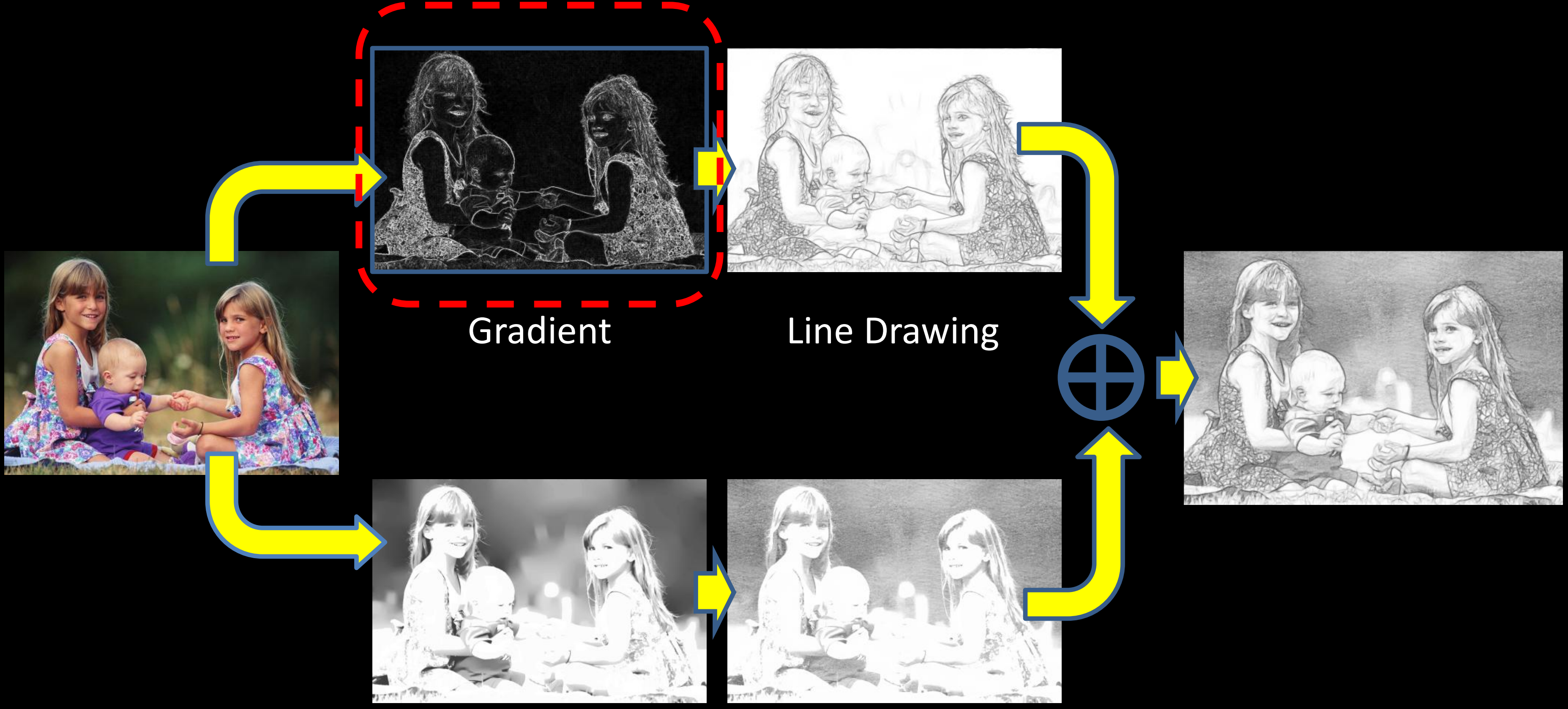
Our Framework



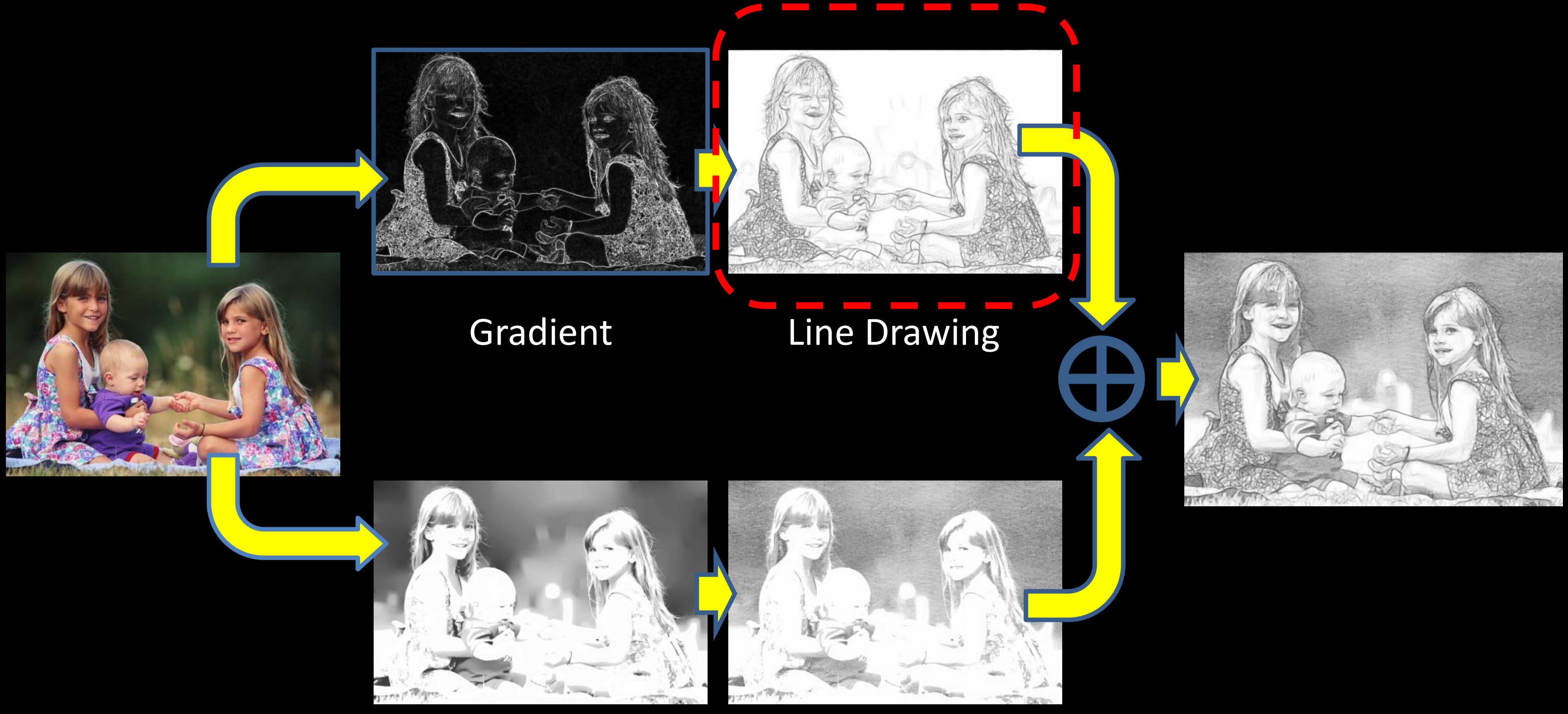
Stroke Structure Generation

Tone Rendering

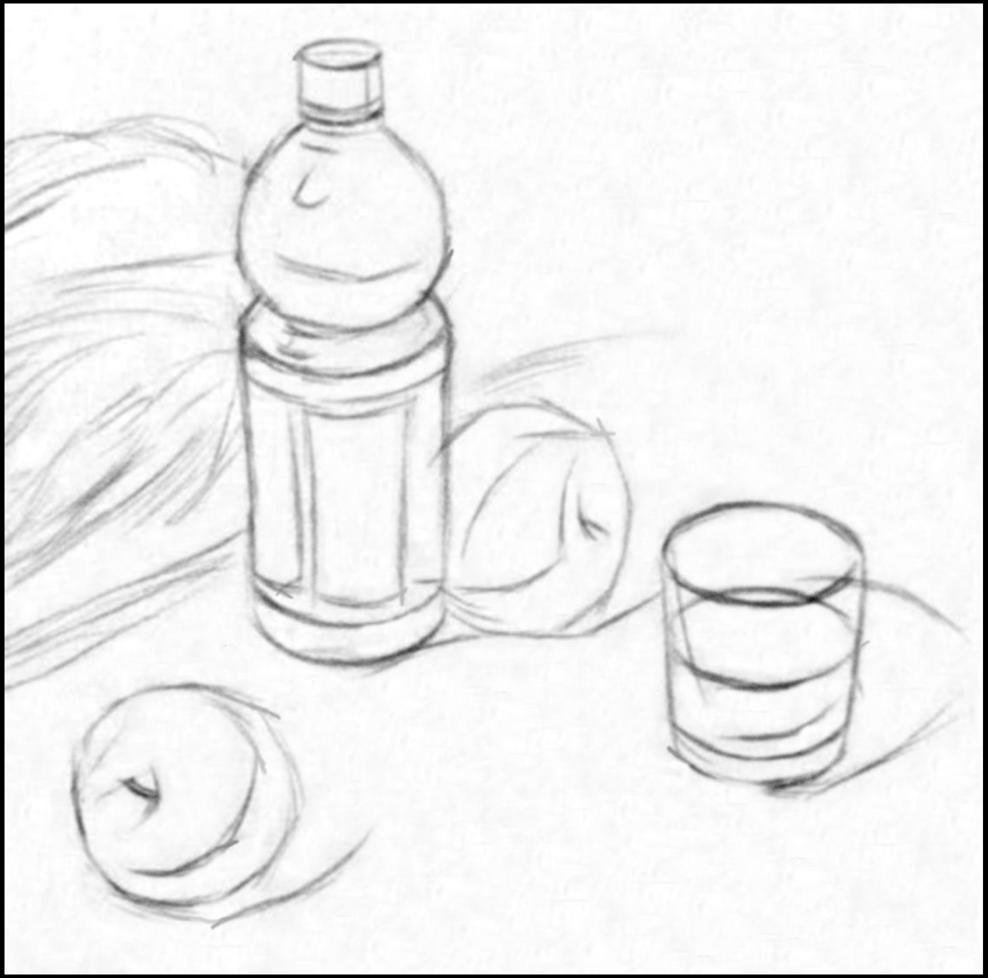
Our Framework



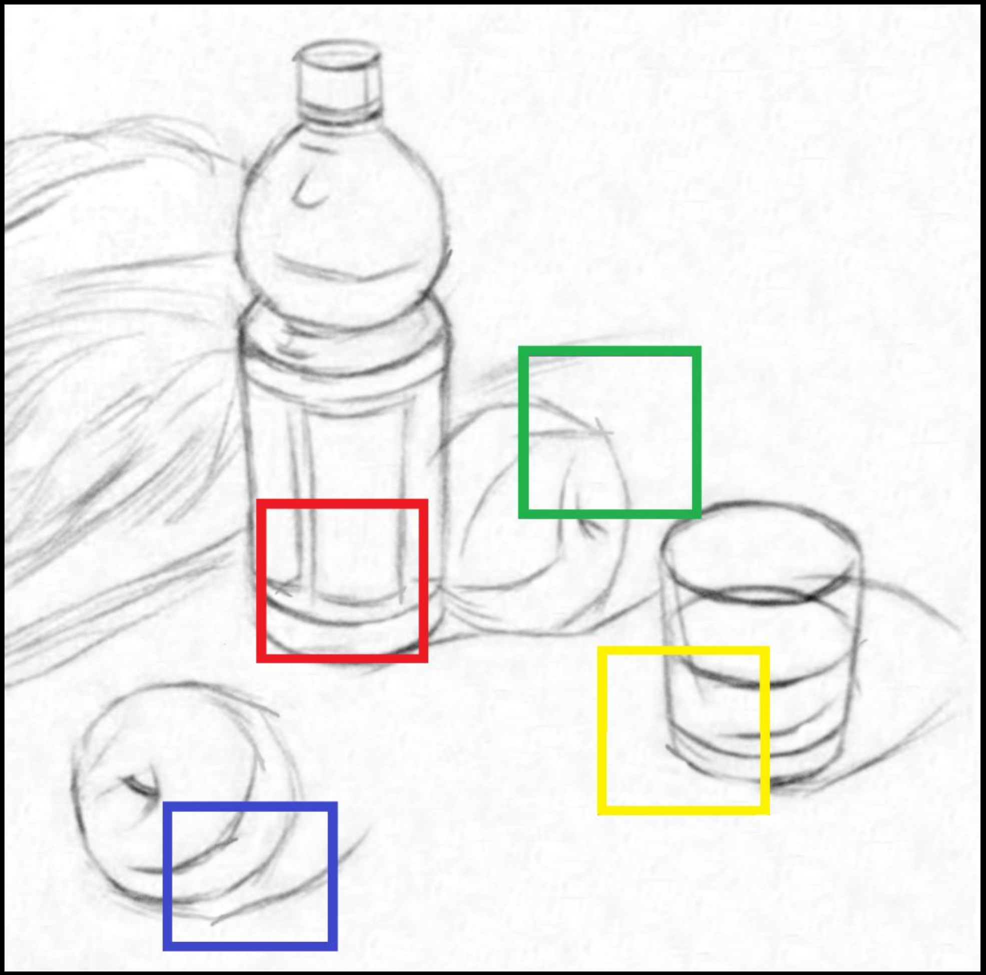
Our Framework (Line Drawing)



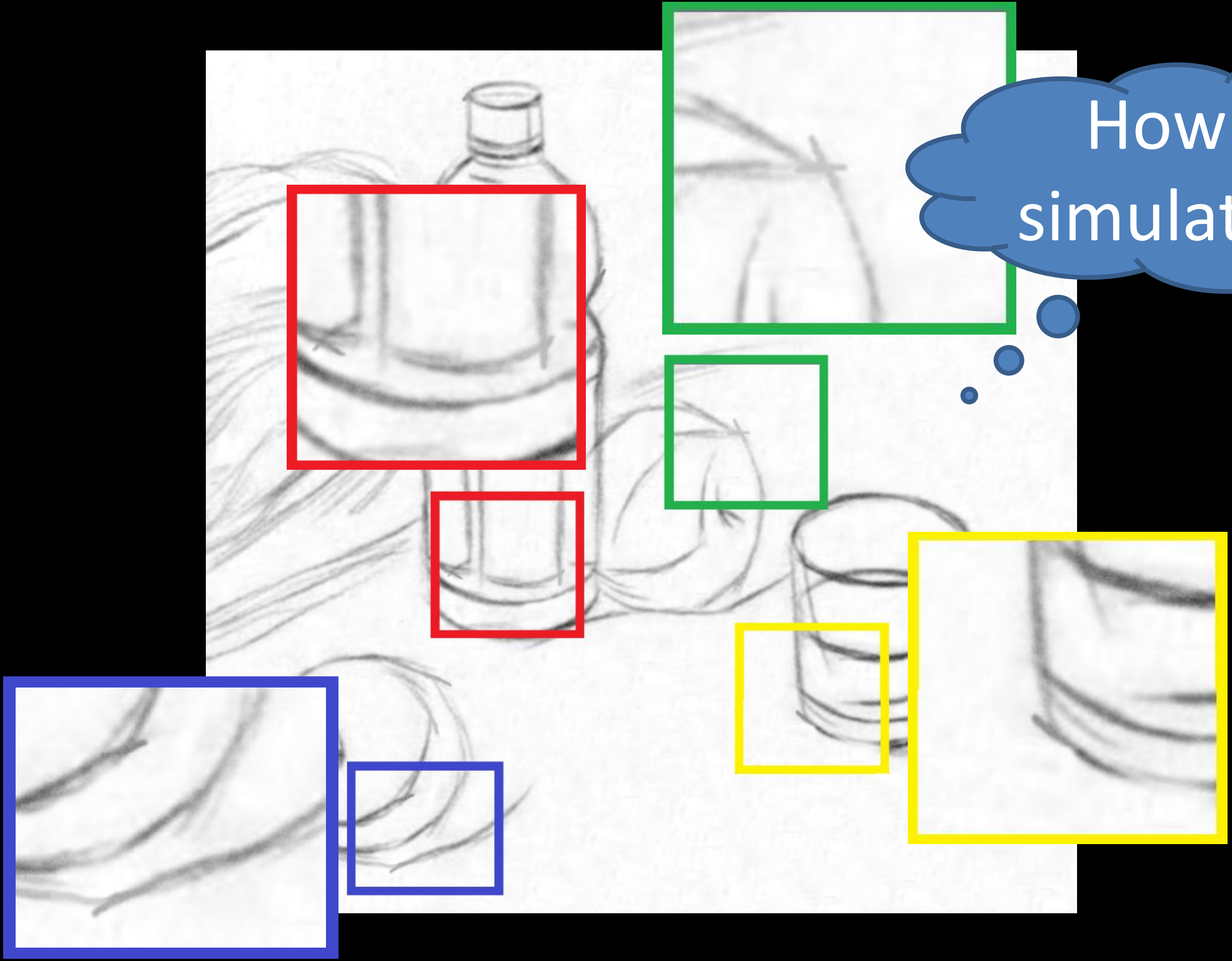
Our Framework (Line Drawing)



Our Framework (Line Drawing)



Our Framework (Line Drawing)



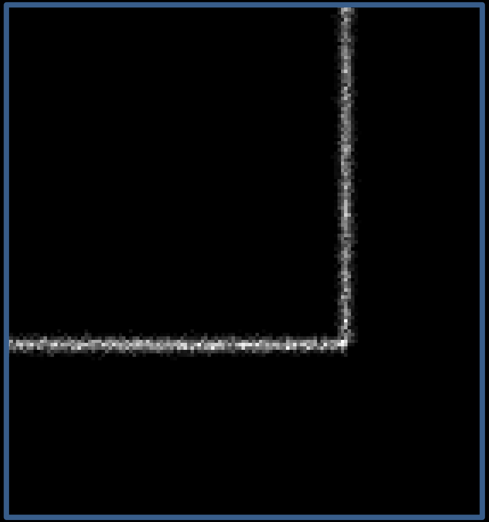
How to simulate it?

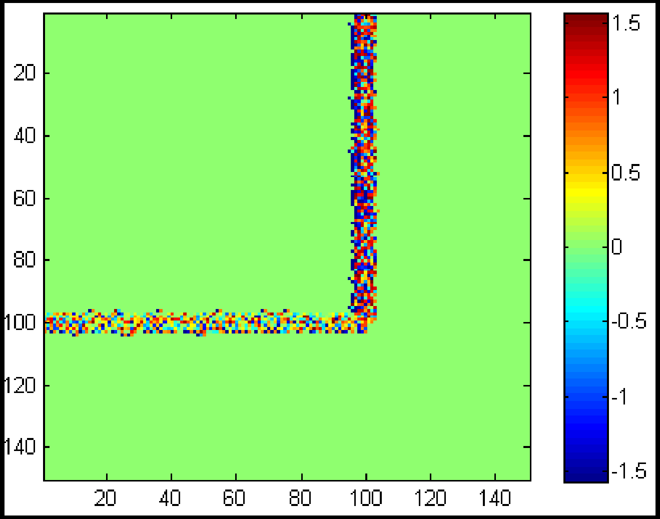
Our Framework (Line Drawing)

- **Sketching cannot be achieved by**
  - **detecting direction of each pixel and**
  - **extending edges along their direction**

Our Framework (Line Drawing)

- **A naive method**

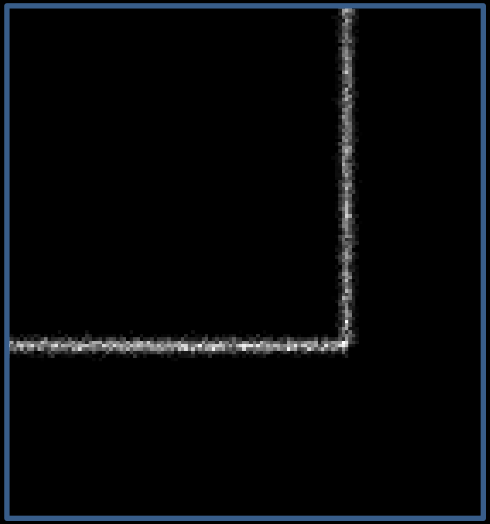


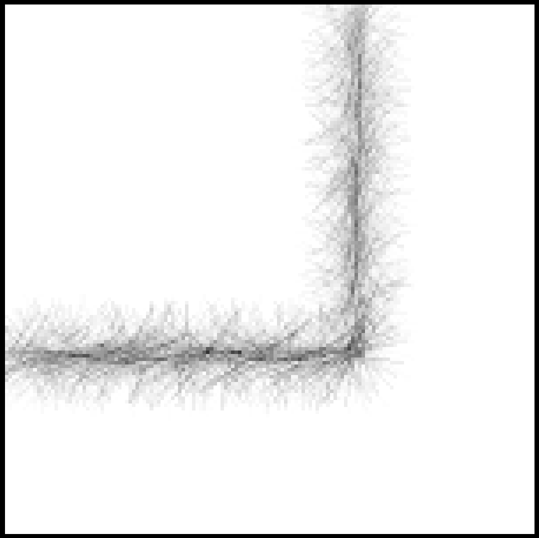


Pixel Direction $\theta = \operatorname{atan}\left(\frac{\partial_y I}{\partial_x I}\right)$

Our Framework (Line Drawing)

- **A naive method**

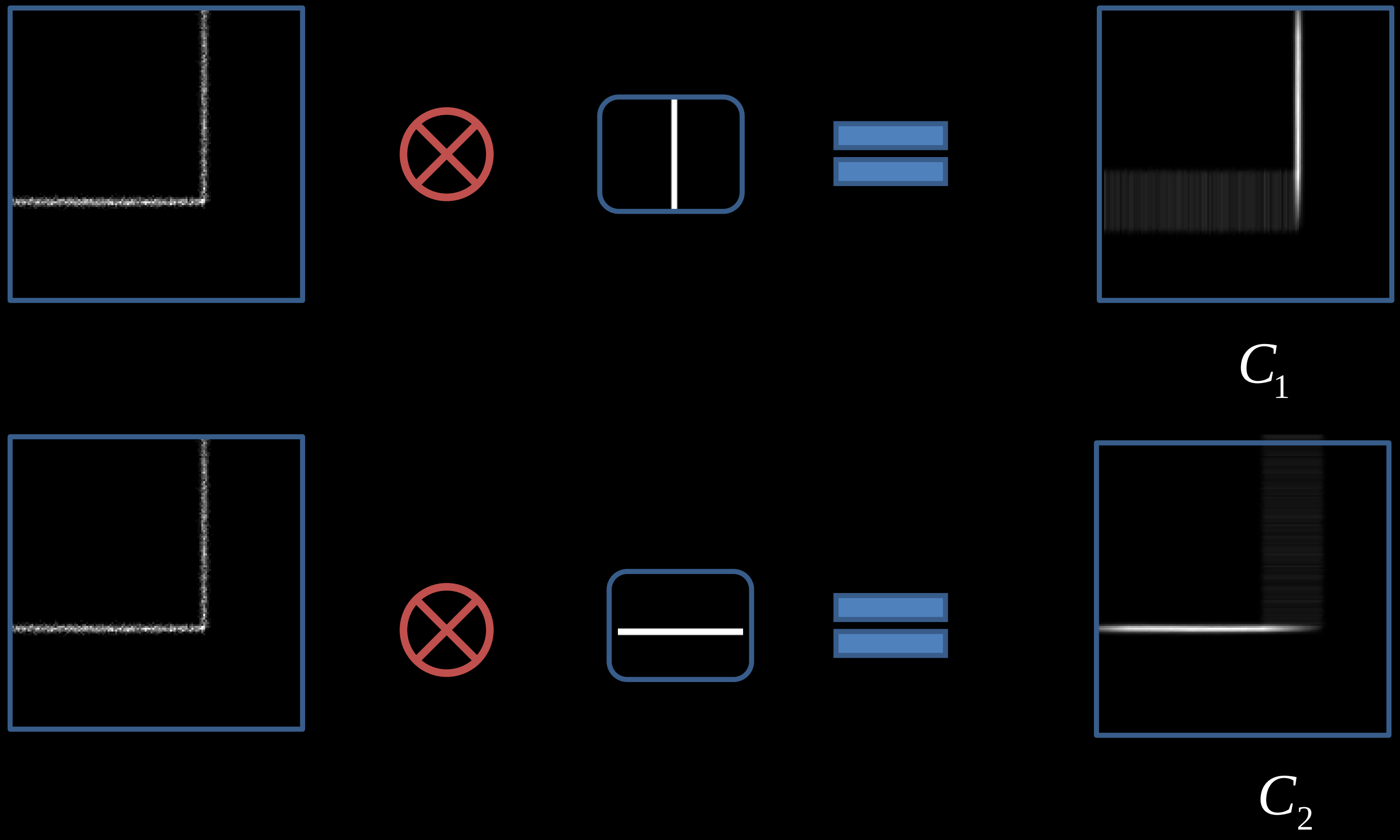




- Extent along their Direction

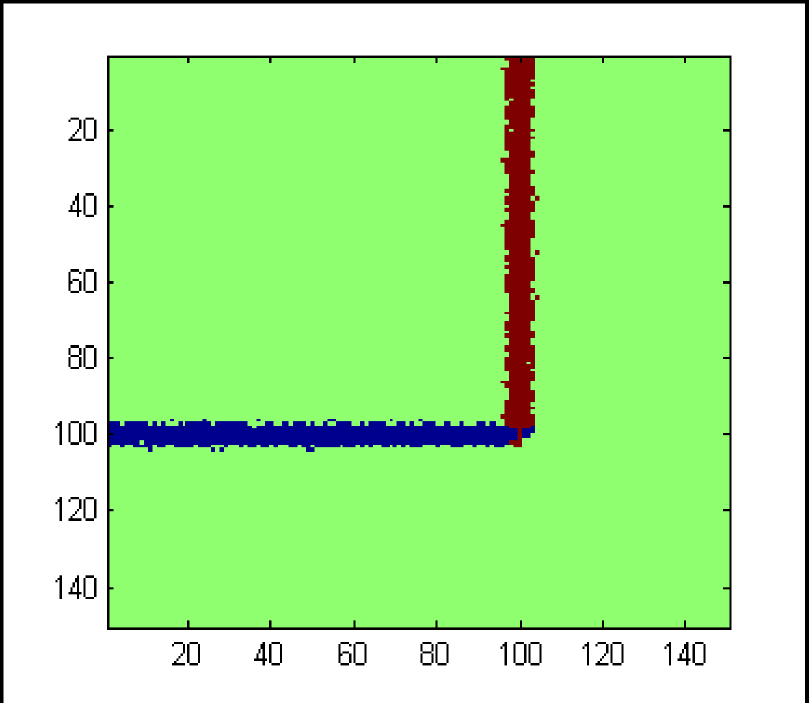
Our Framework (Line Drawing)

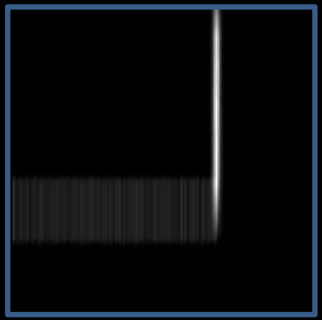
Our Robust Edges Direction Classification



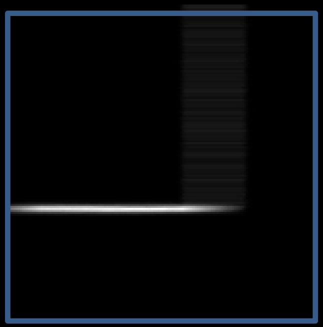
Our Framework (Line Drawing)

Our Robust Edges Direction Classification





C_1



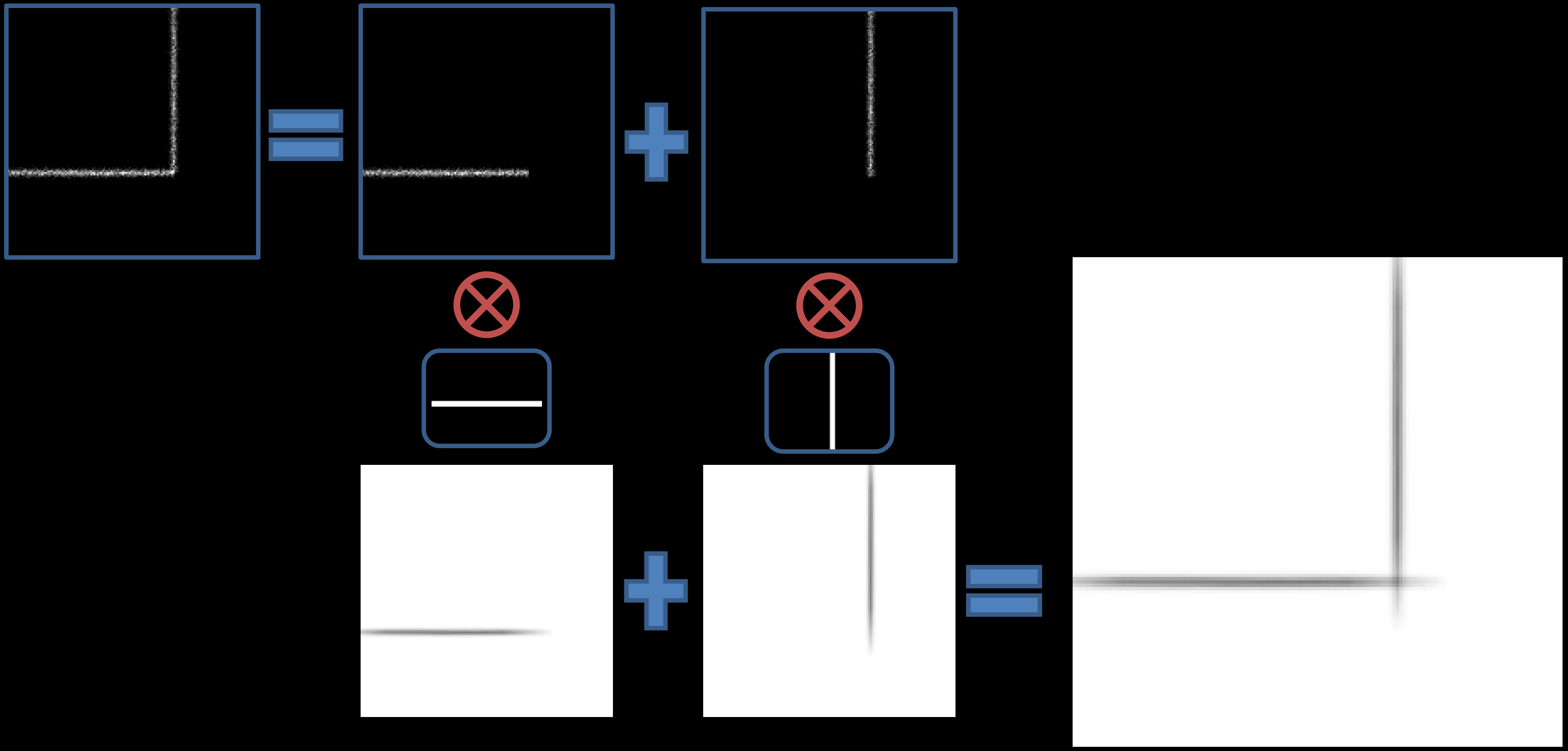
C_2

red $\arg\max\{C_1, C_2\} = 1$

blue $\arg\max\{C_1, C_2\} = 2$

Our Framework (Line Drawing)

- Why it works?



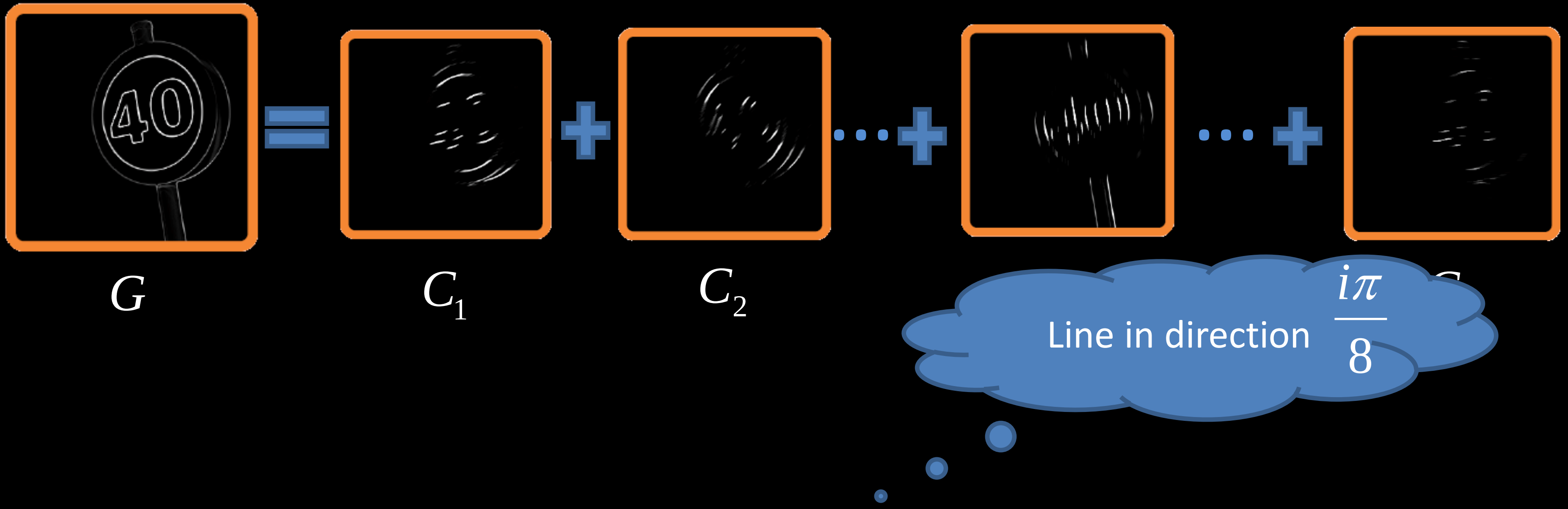
Our Framework (Line Drawing)

Input

Our Framework (Line Drawing)

Gradient

Our Framework (Line Drawing)

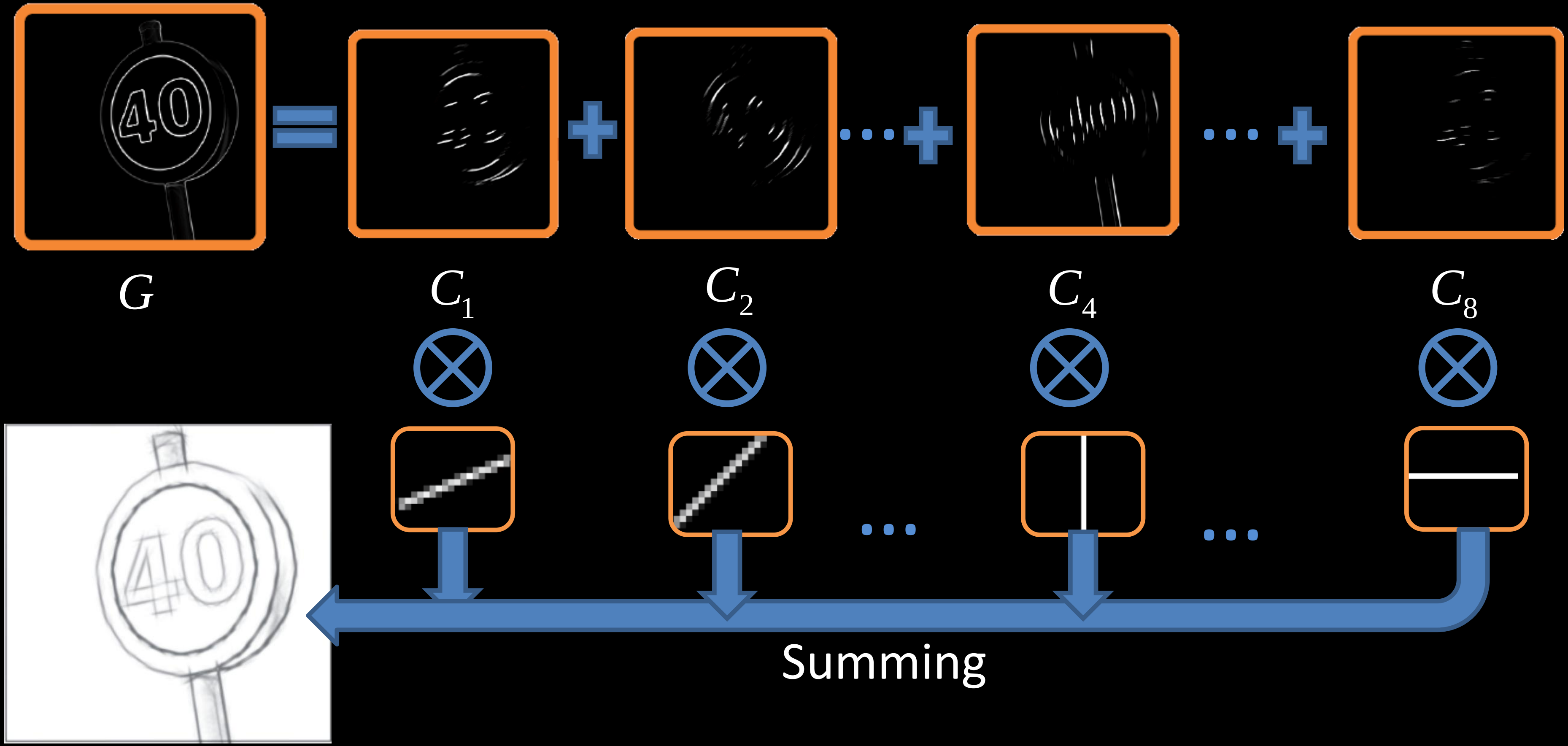


Line in direction $\frac{i\pi}{8}$

$$C_i(p)=\begin{cases} G(p) & \text{if } \operatorname{argmax}\{\psi_i \otimes G\}(p)=i \\ 0 & \text{otherwise} \end{cases}$$

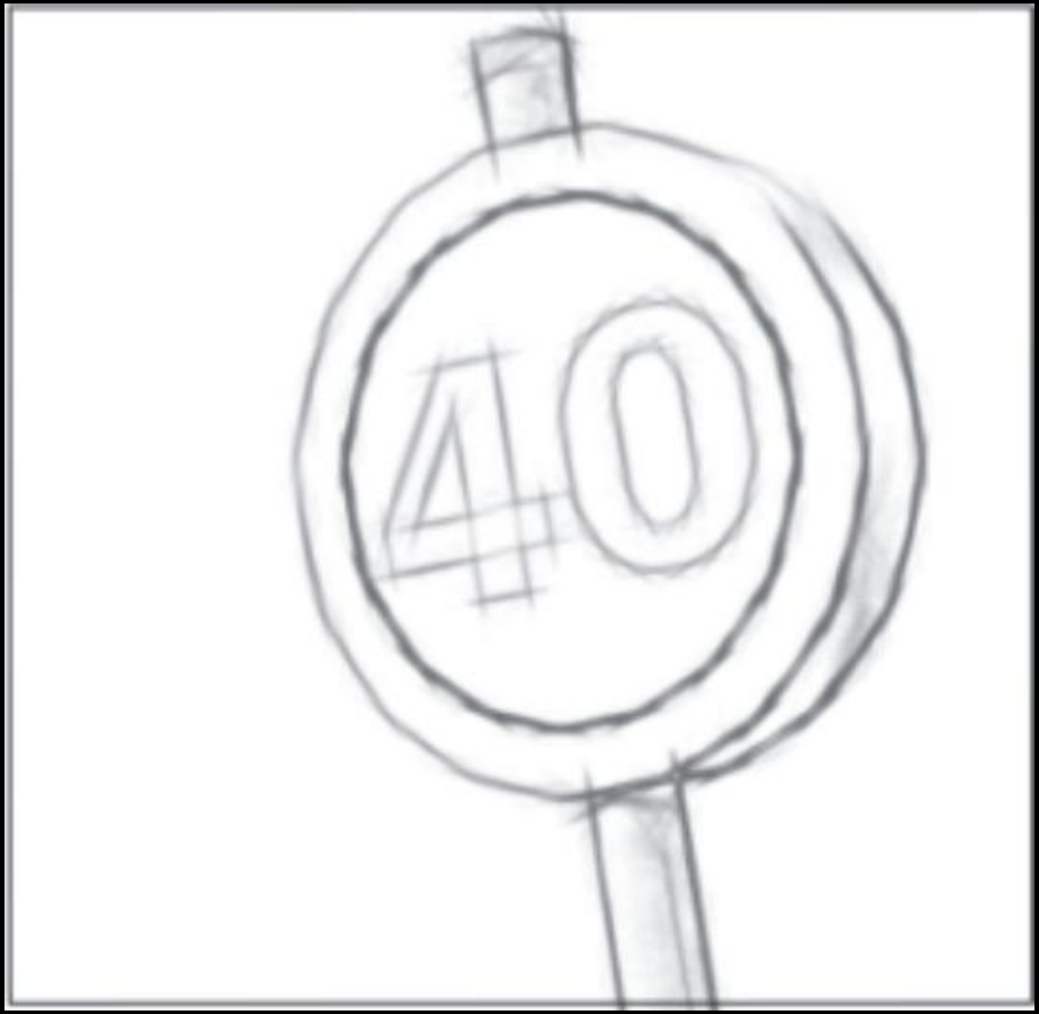
Pixel direction classification (8 directions)

Our Framework (Line Drawing)

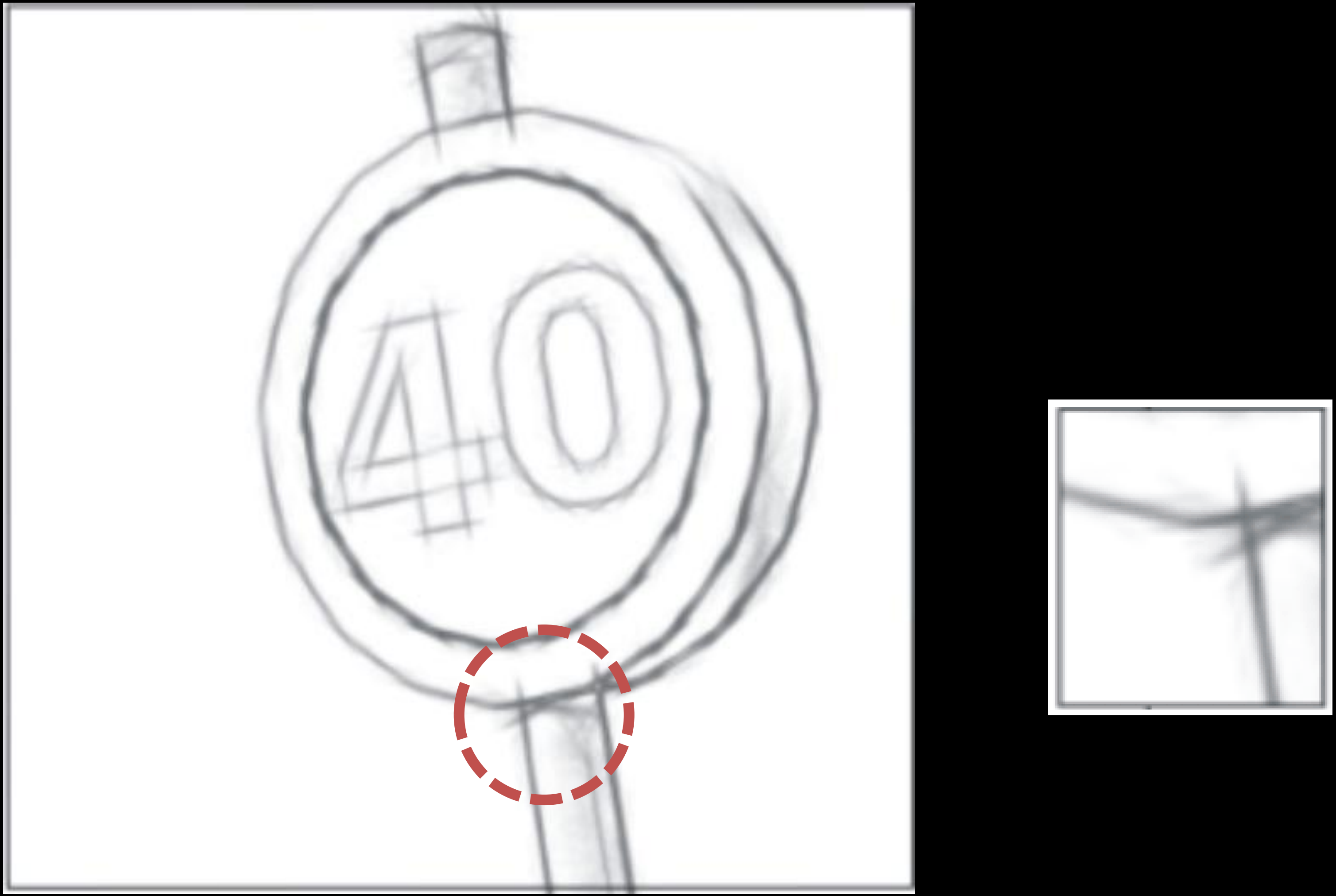


$$S = \sum_{i=1}^{8} \psi_i \otimes C_i$$

Our Framework (Line Drawing)



Our Framework (Line Drawing)

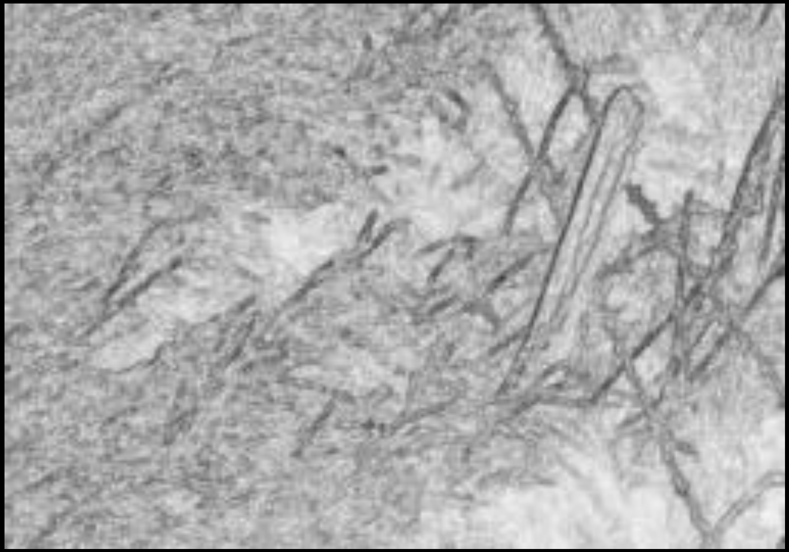


Our Framework (Line Drawing)

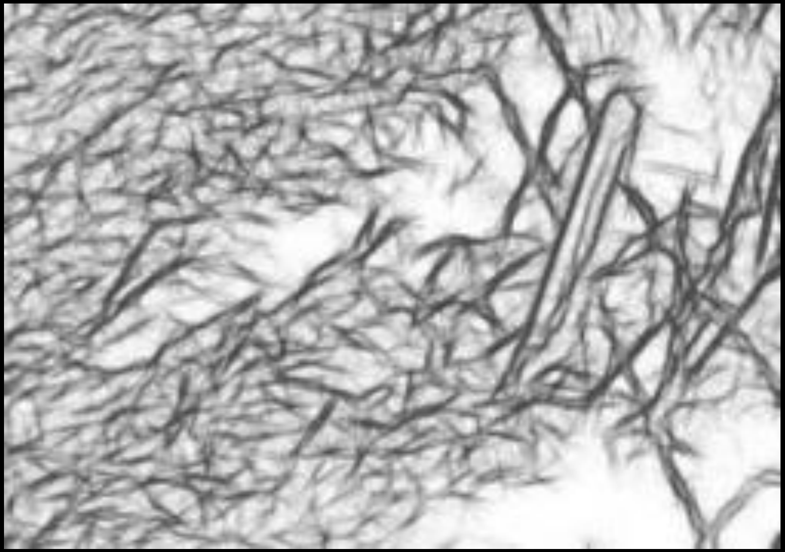
- **Why it works?**
  - **Robust pixel direction classification**
  - **Convolution line shaping**

Our Framework (Line Drawing)

For Texture



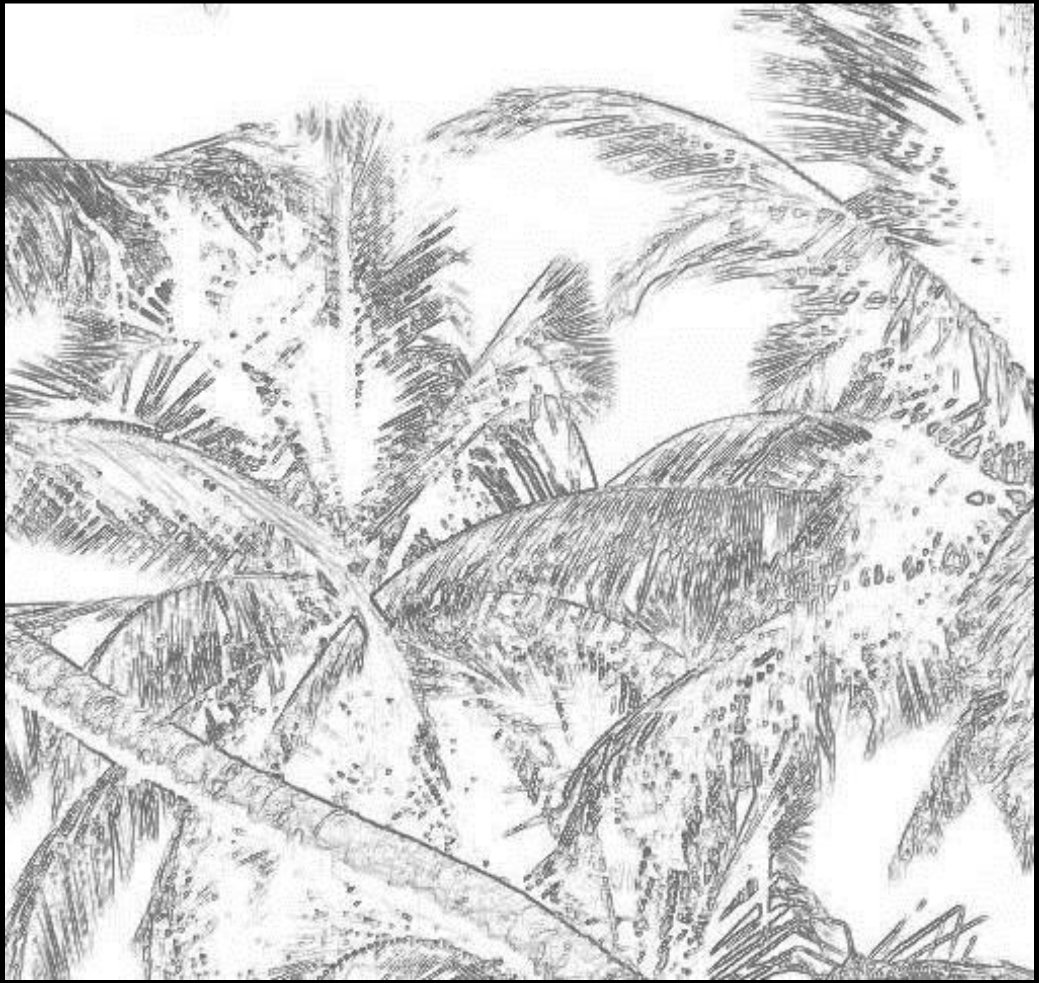
Gradient



Ours

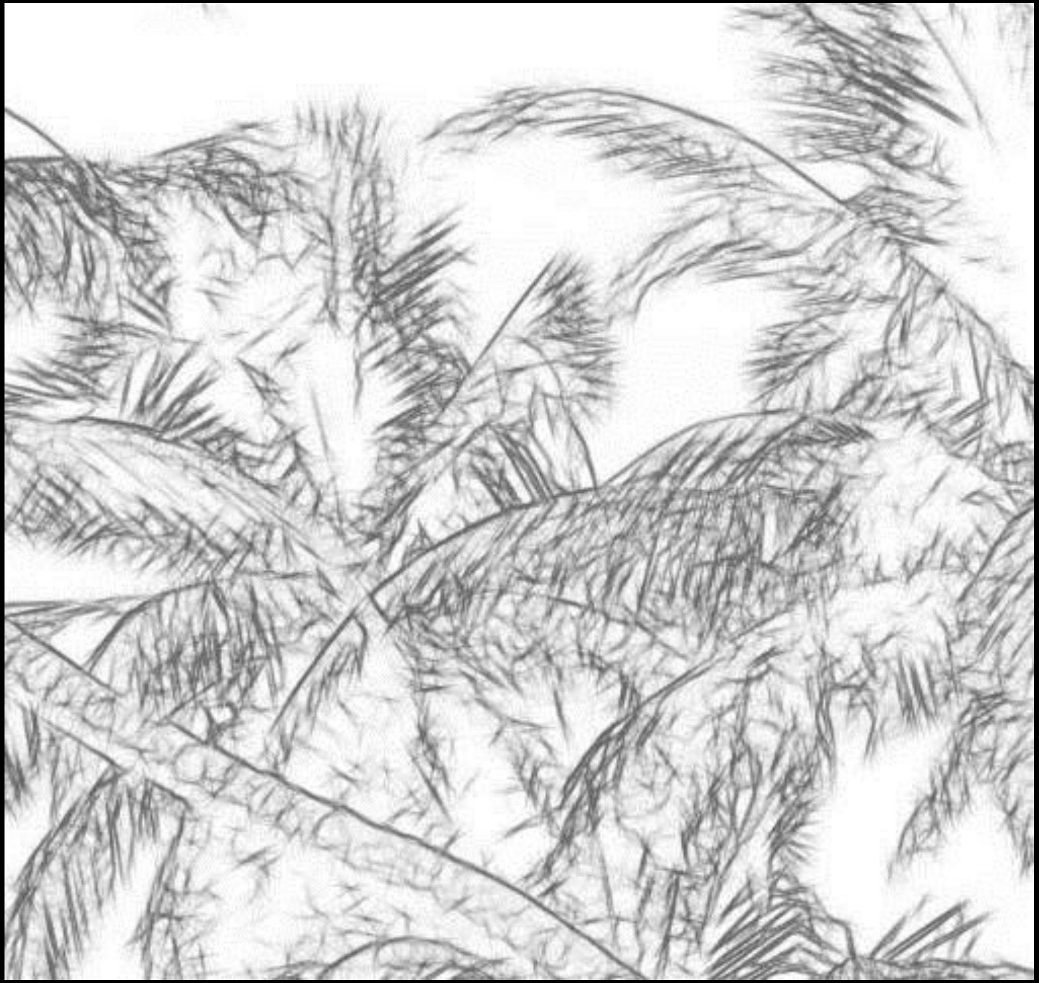
Our Framework (Line Drawing)

Our Framework (Line Drawing)



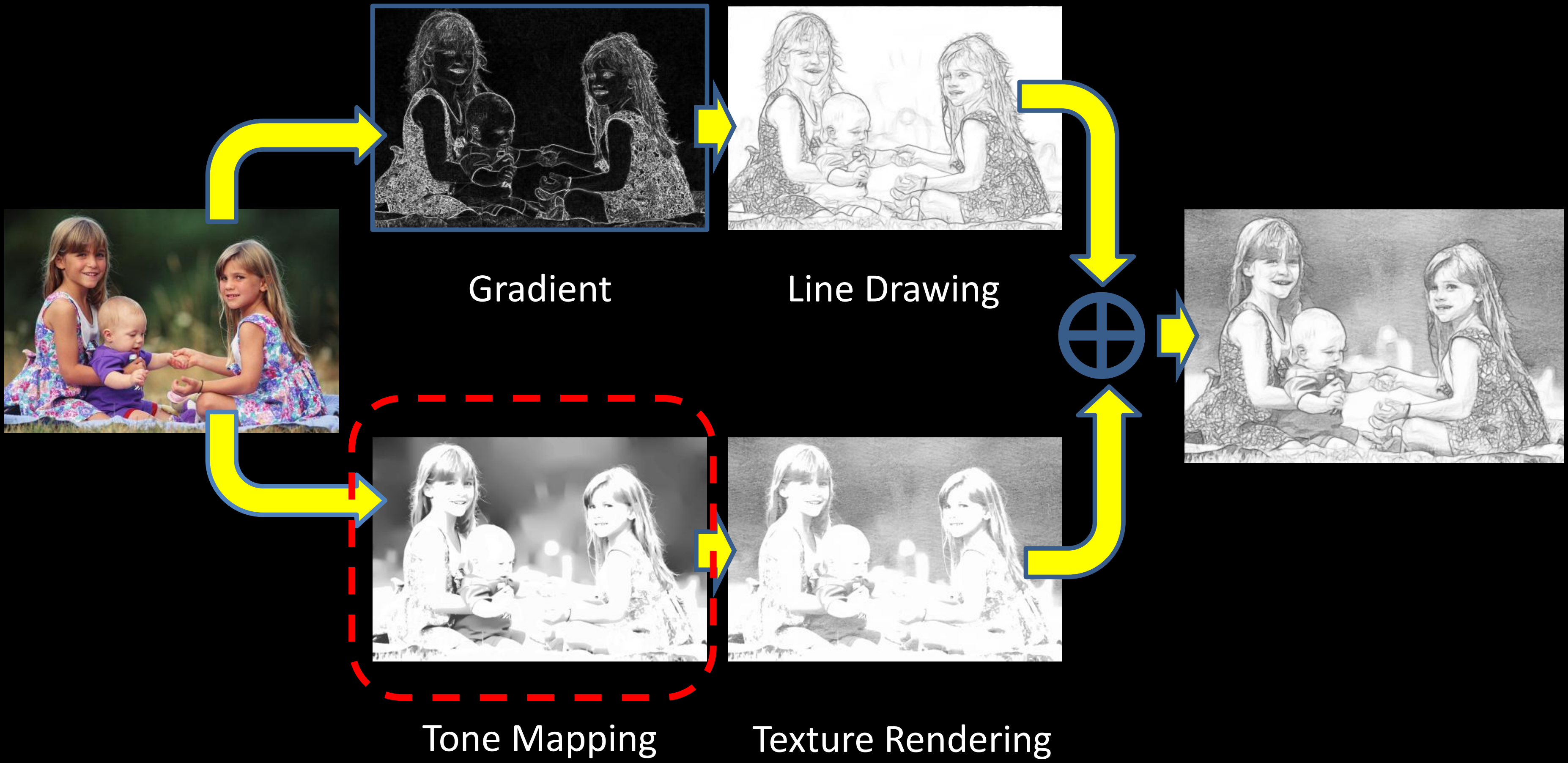
Gradient

Our Framework (Line Drawing)



Ours

Our Framework (Tone Mapping)



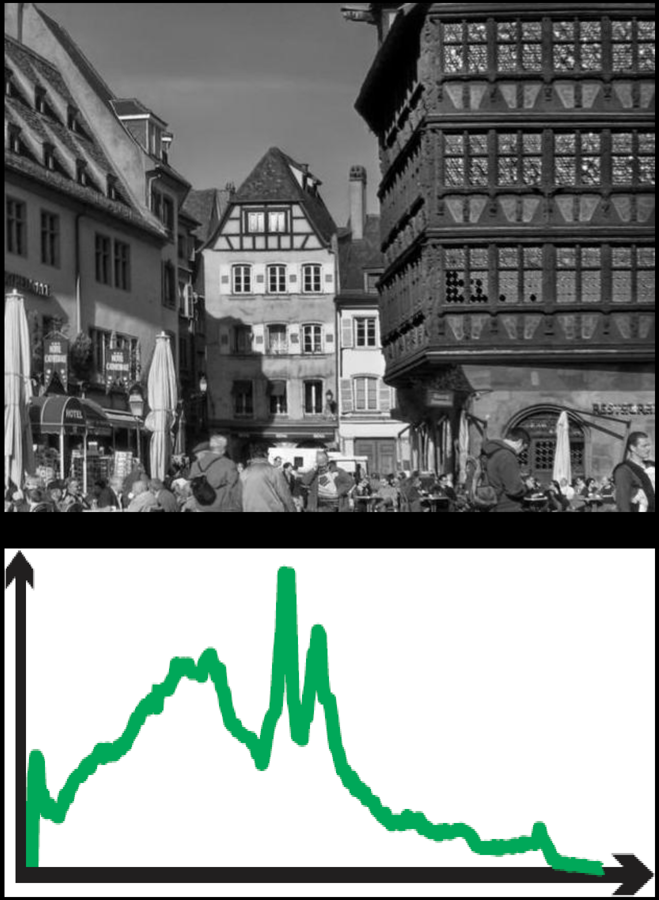
Gradient

Line Drawing

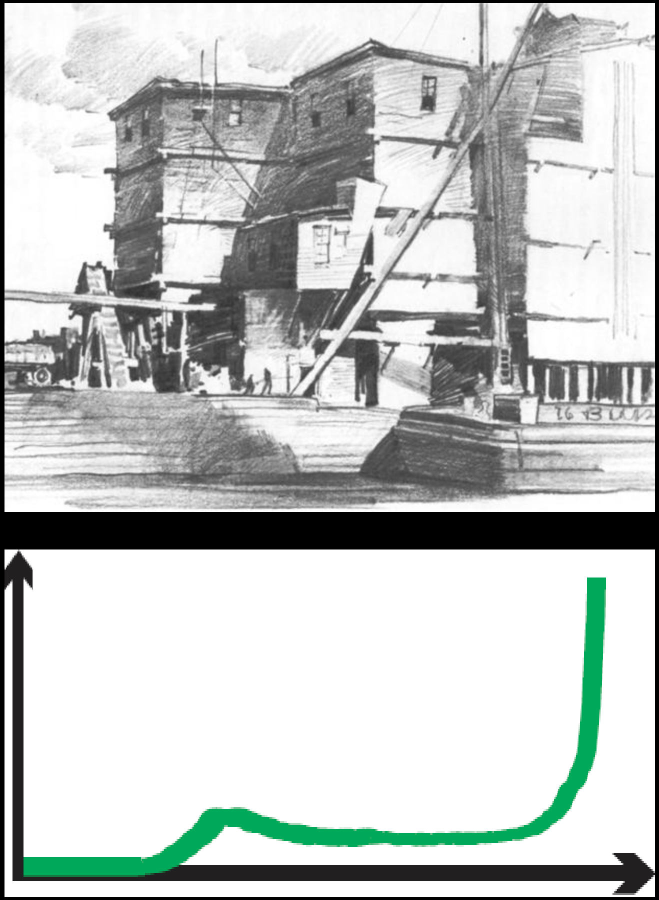
Tone Mapping

Texture Rendering

Our Framework (Tone Mapping)

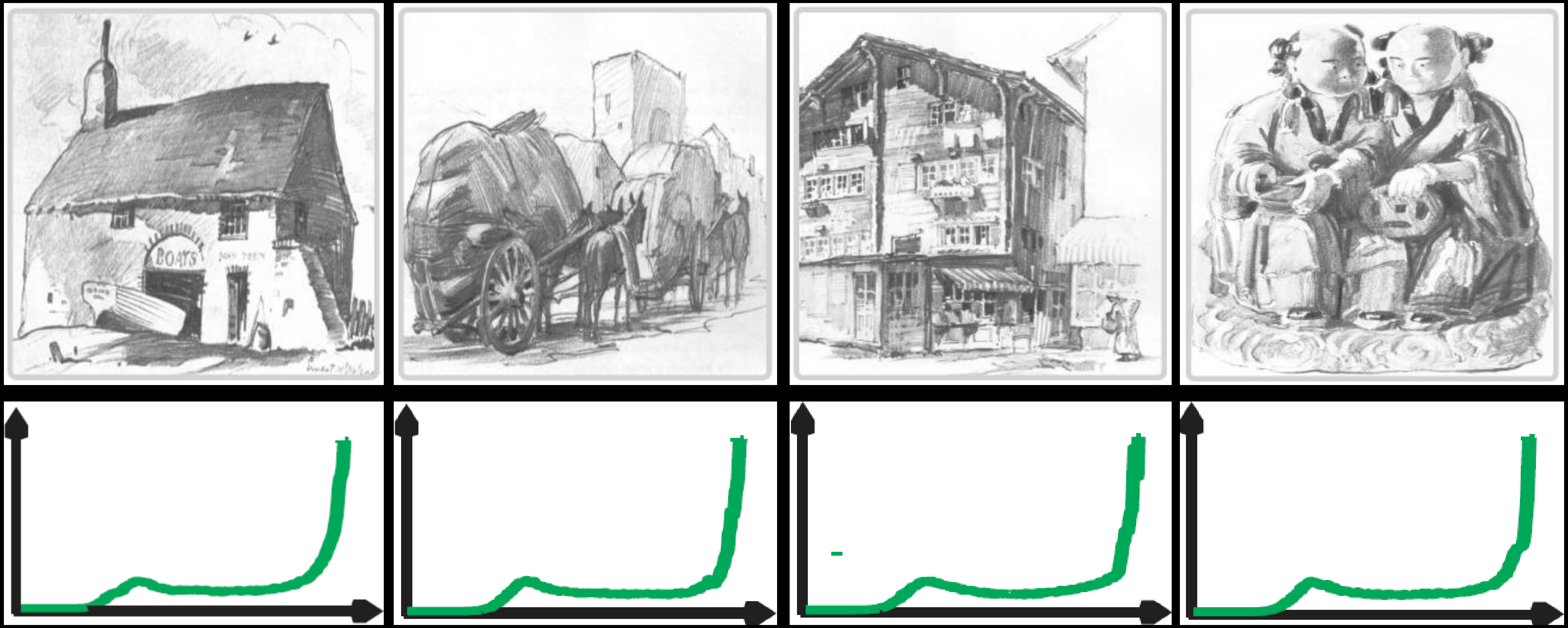


Natural Image



Pencil Drawing

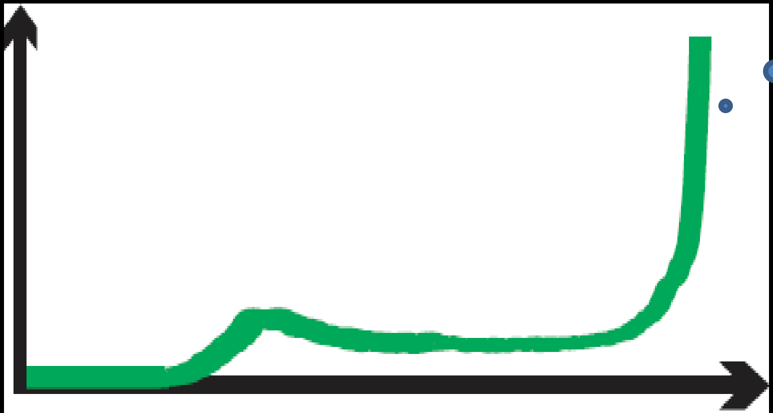
Our Framework (Tone Mapping)



Ernest W. Watson "The art of pencil drawing", 1968

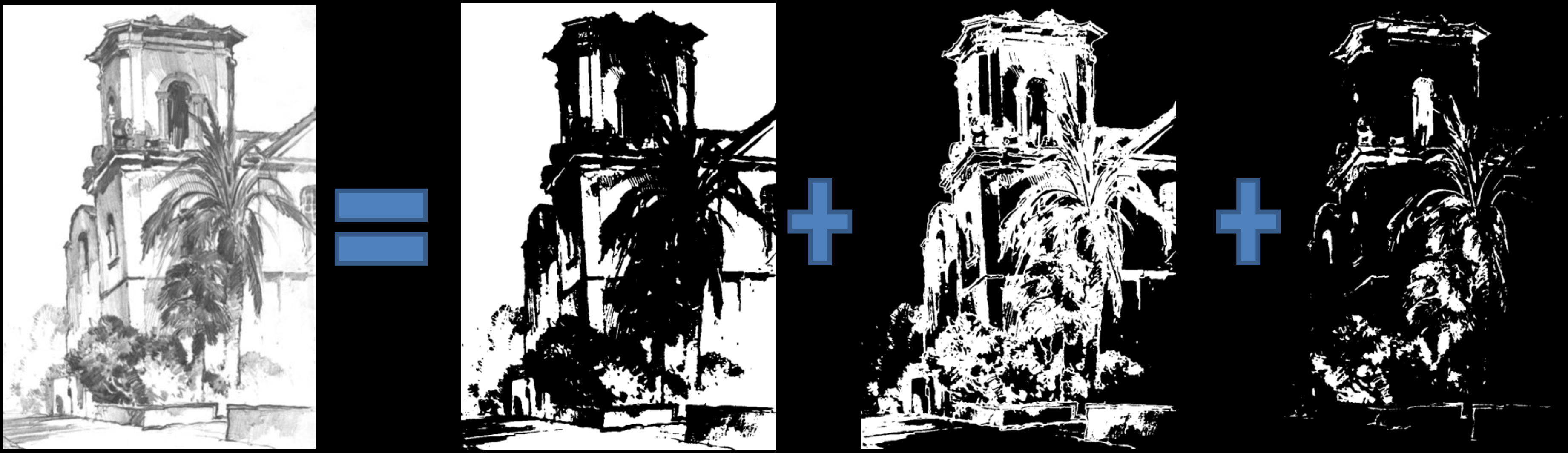
Our Framework (Tone Mapping)

Why is it?



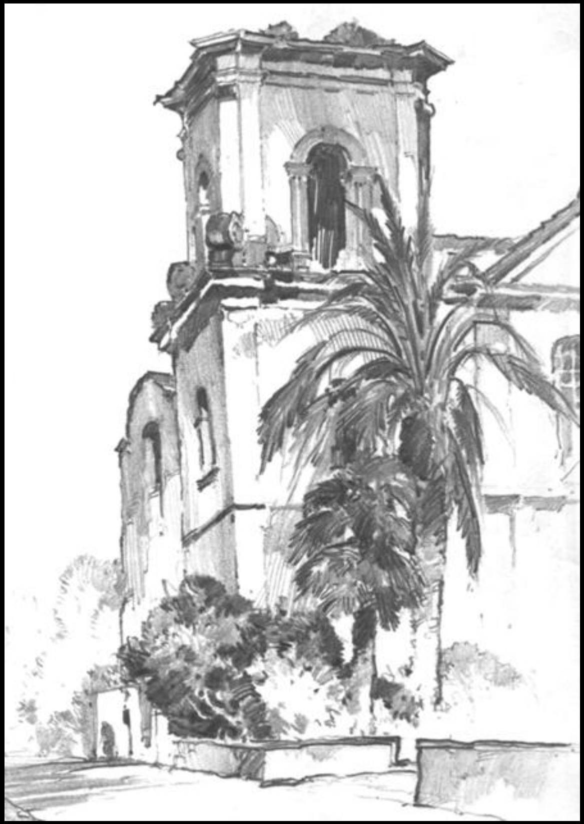
The underling mechanism is complex

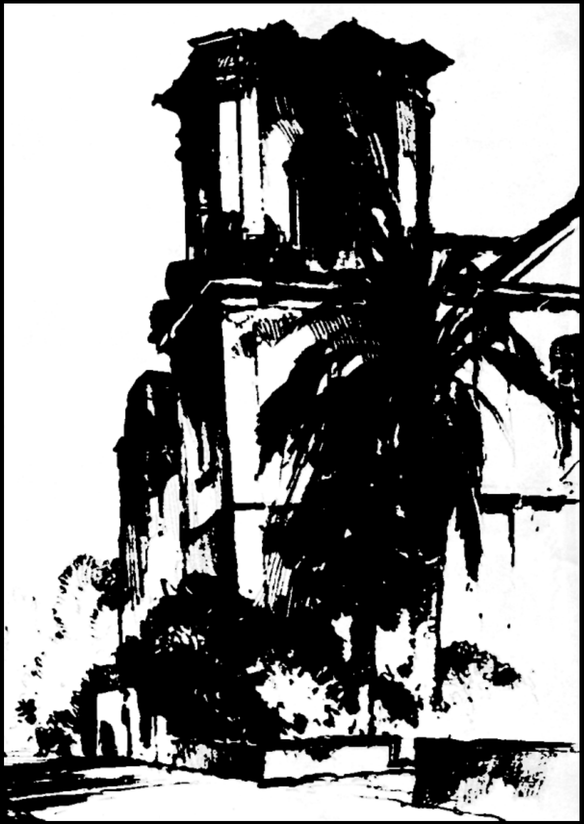
Our Framework (Tone Mapping)



Three-layer structure

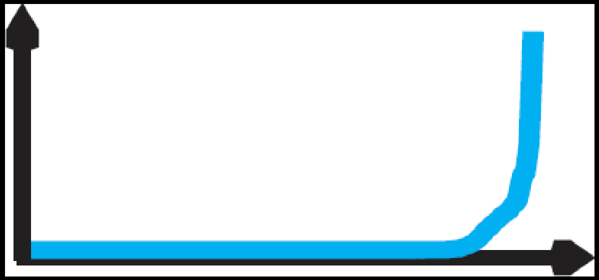
Our Framework (Tone Mapping)





Bright Layer: artists do not draw anything

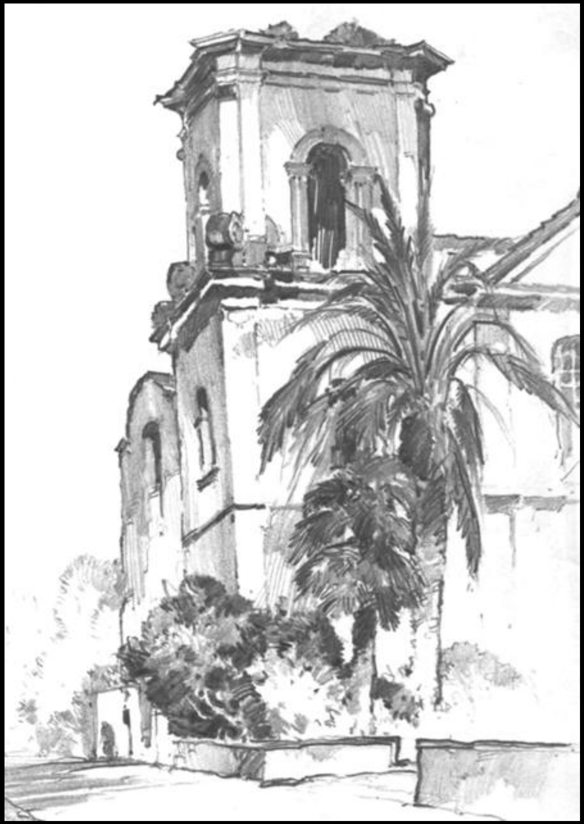
Our Framework (Tone Mapping)

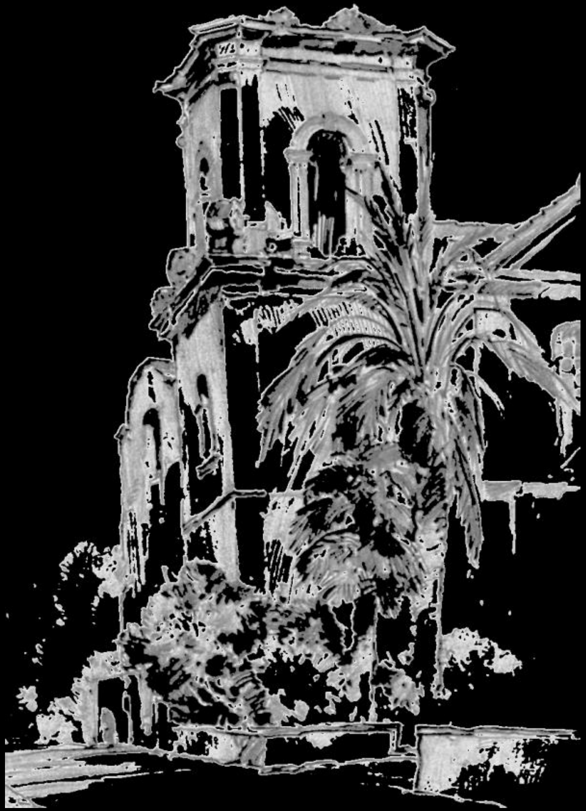


$$P_1(v)=\begin{cases}\frac{1}{\sigma_b}e^{-\frac{1-v}{\sigma_b}} & \text{if } v\le 1\\ 0 & \text{otherwise}\end{cases}$$

Bright Layer: artists do not draw anything

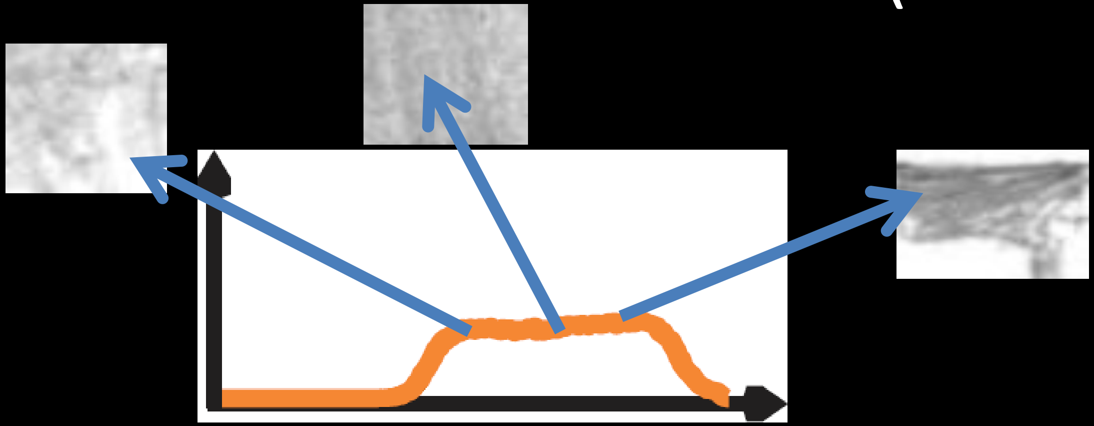
Our Framework (Tone Mapping)



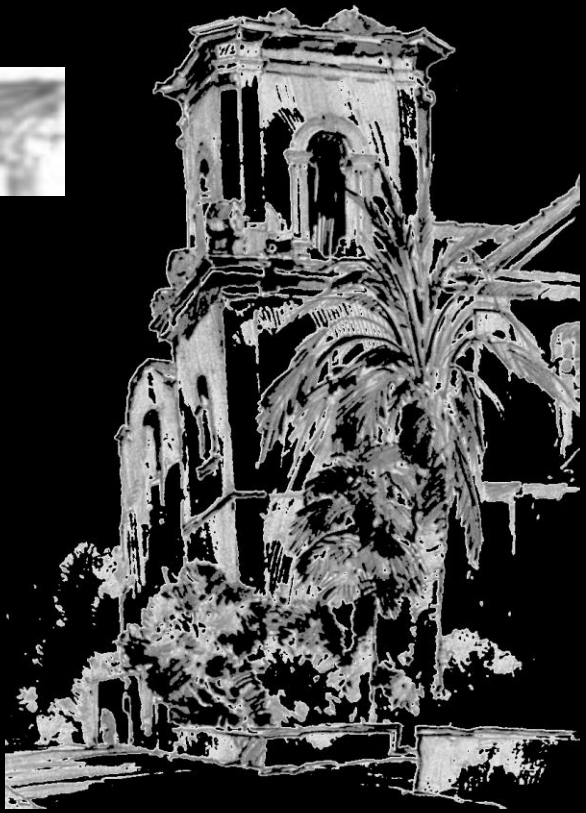


Mid-tone Layer: different pressure

Our Framework (Tone Mapping)

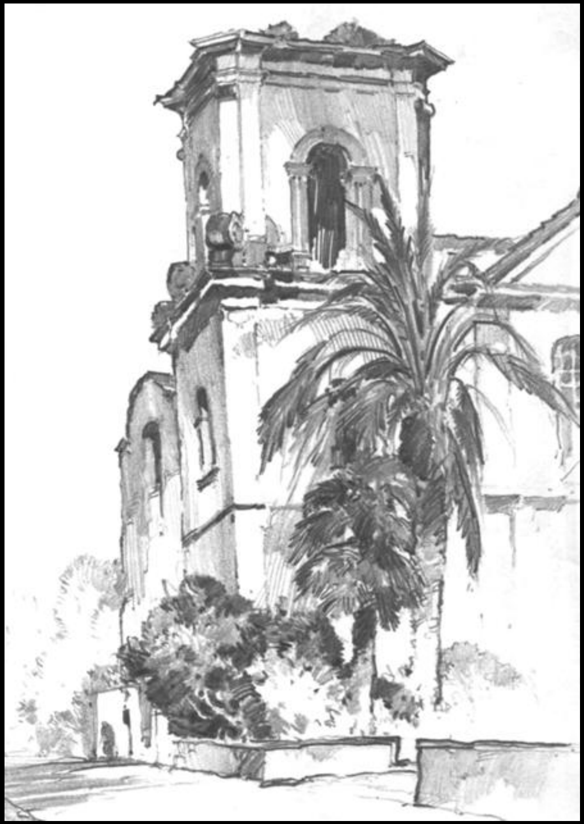


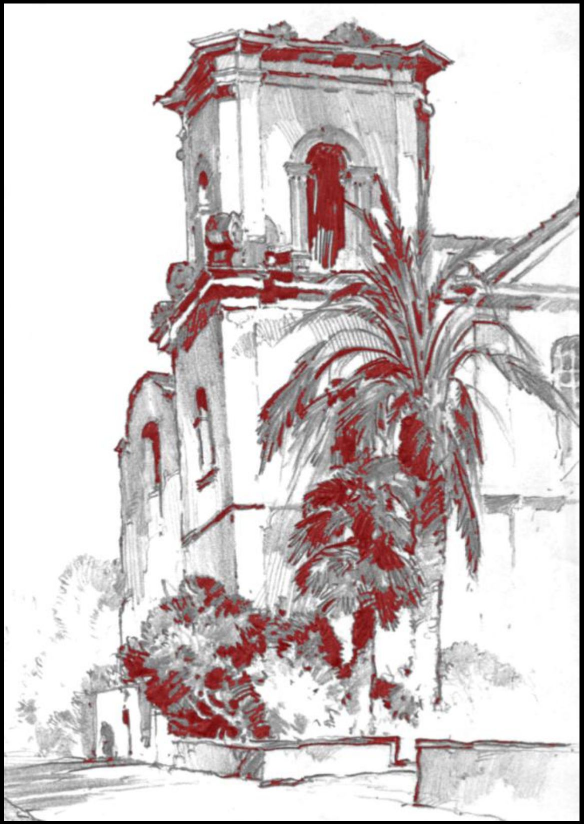
$$P_2(v)=\begin{cases}\dfrac{1}{u_b-u_a} & \text{if } u_a\le v\le u_b\\ 0 & \text{otherwise}\end{cases}$$



Mid-tone Layer: different pressure

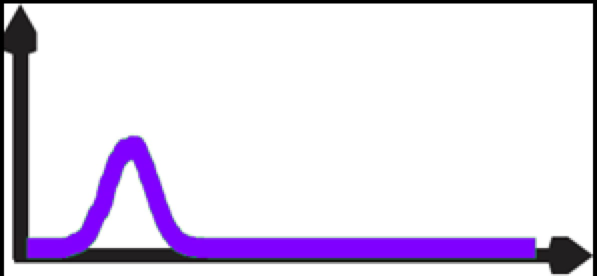
Our Framework (Tone Mapping)



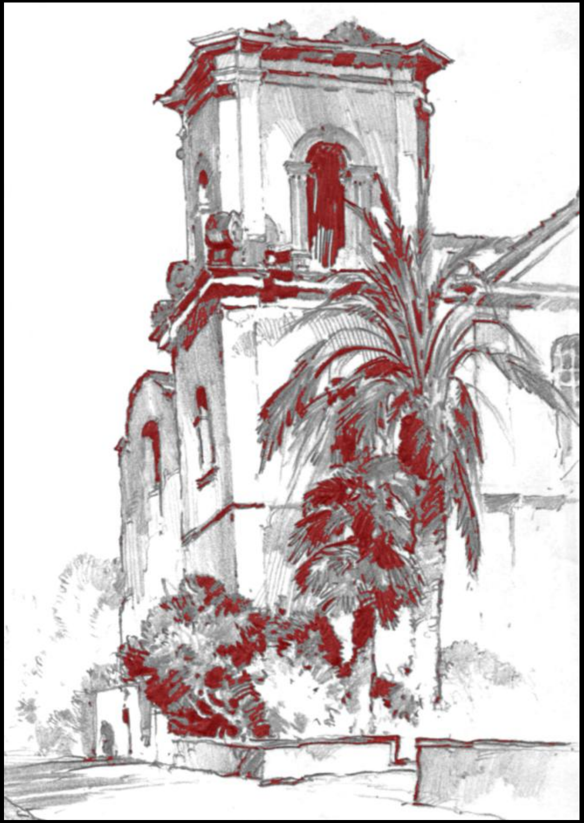


Dark Layer: visually most salient

Our Framework (Tone Mapping)



$$P_3(v) = \frac{1}{\sqrt{2\pi\sigma_d}} e^{-\frac{(v-\mu_d)^2}{2\sigma_d^2}}$$



Dark Layer: visually most salient

Our Framework (Tone Mapping)

$$P(v)=\frac{1}{Z}\left[\omega_1 P_1(v)+\omega_2 P_2(v)+\omega_3 P_3(v)\right]$$

Unknown Parameters

Bright layer	Mild-tone layer	Dark layer
σ_b	u_a u_b	μ_d σ_d

Our Framework (Tone Mapping)

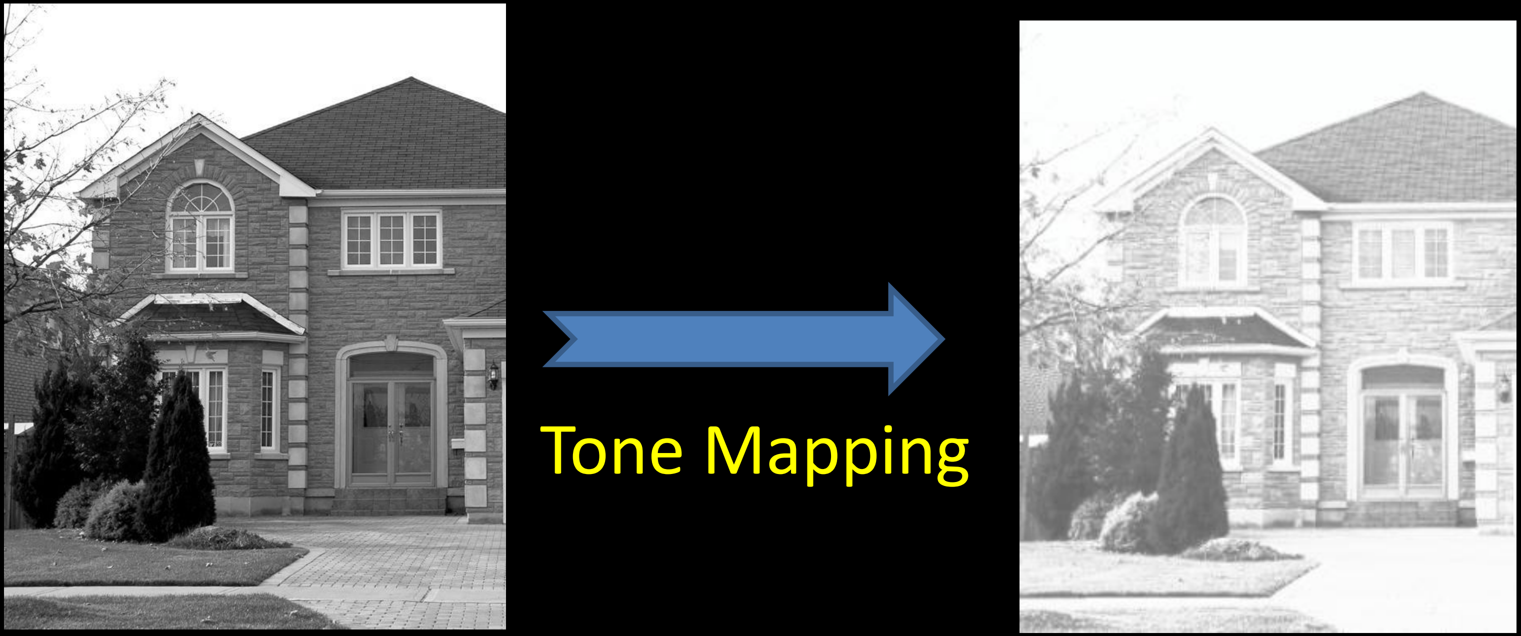
$$P(v) = \frac{1}{Z}\left[\omega_1 P_1(v) + \omega_2 P_2(v) + \omega_3 P_3(v)\right]$$

Unknown Parameters

Bright layer	Mild-tone layer		Dark layer	
σ_b	u_a	u_b	μ_d	σ_d
9	105	225	90	11

Maximum Likelihood Estimation

Our Framework (Tone Mapping)

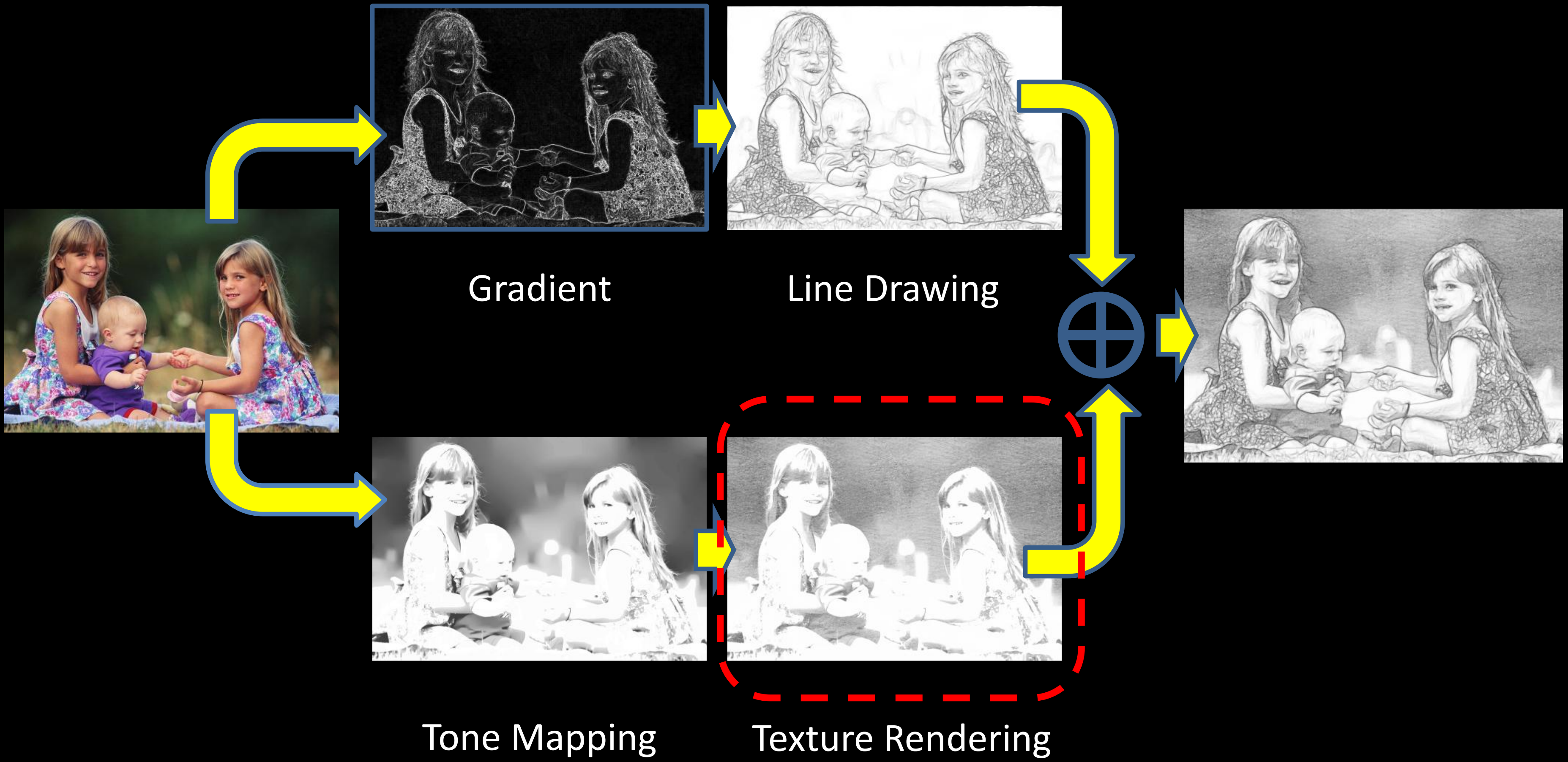


Tone Mapping

$$P(v) = \frac{1}{Z}\left[\omega_1 P_1(v) + \omega_2 P_2(v) + \omega_3 P_3(v)\right]$$

$\omega_1 : \omega_2 : \omega_3$ =29 : 29 : 42

Our Framework (Texture Rendering)



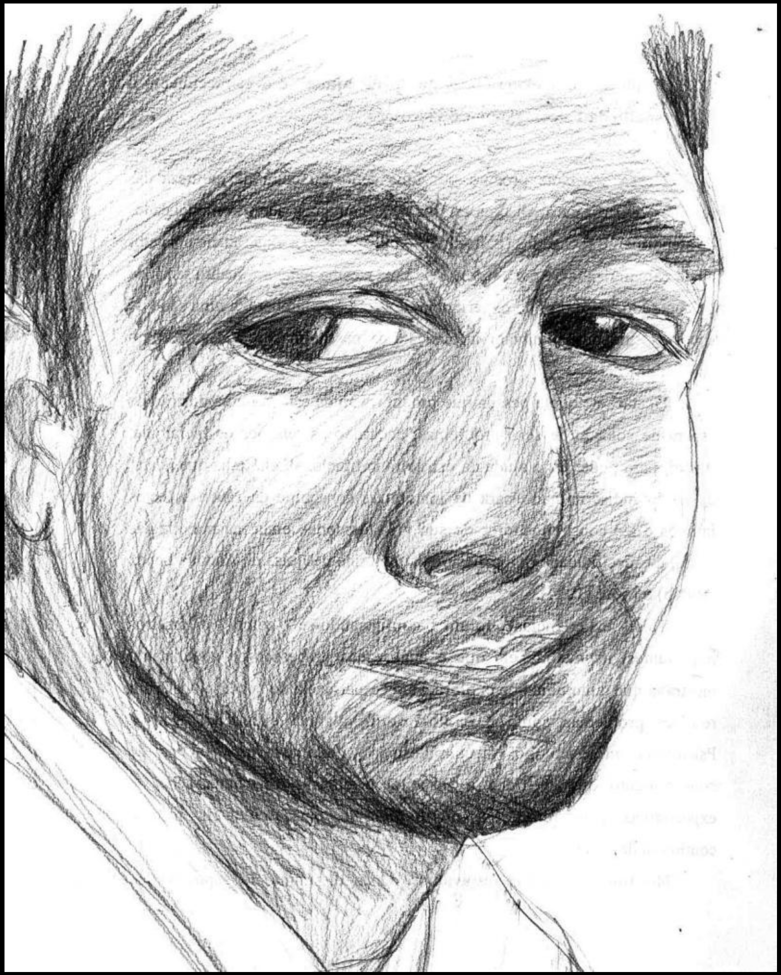
Gradient

Line Drawing

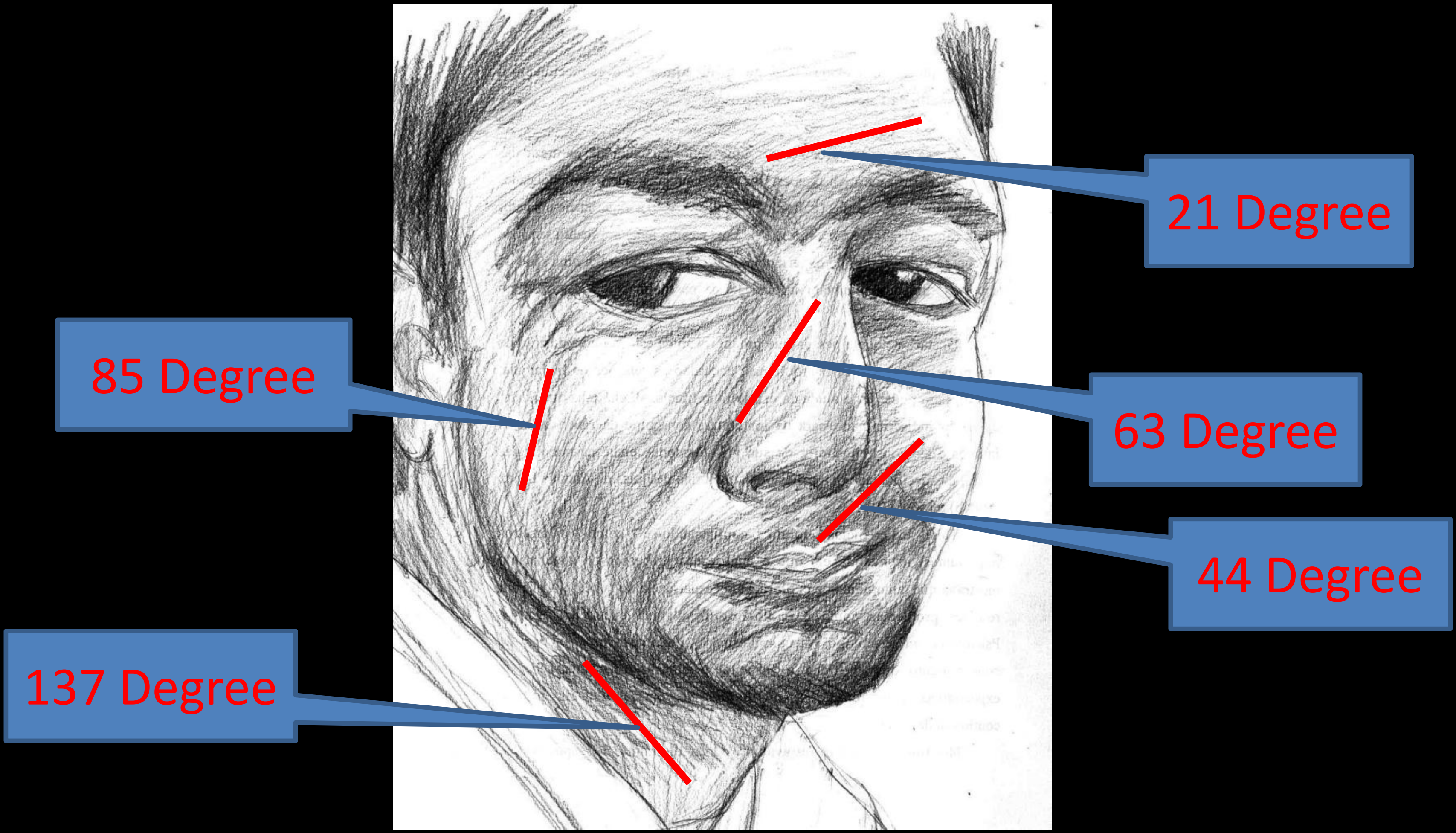
Tone Mapping

Texture Rendering

Our Framework (Texture Rendering)



Our Framework (Texture Rendering)



21 Degree

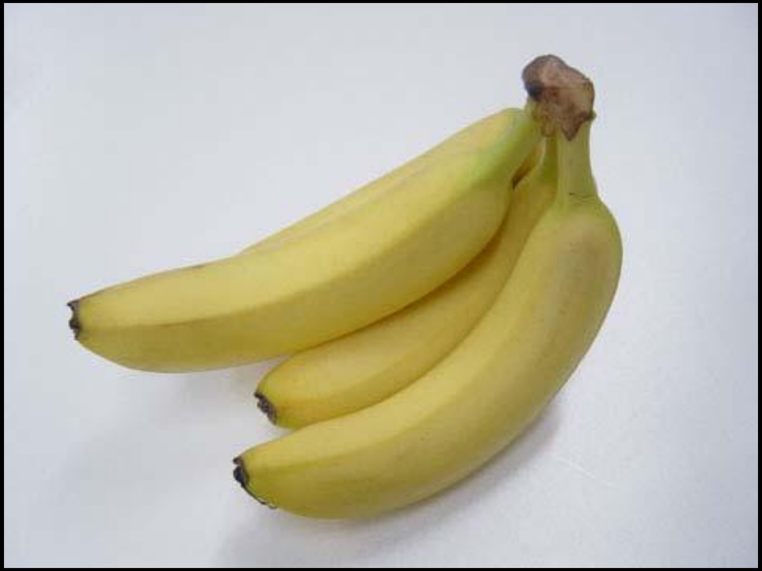
85 Degree

63 Degree

44 Degree

137 Degree

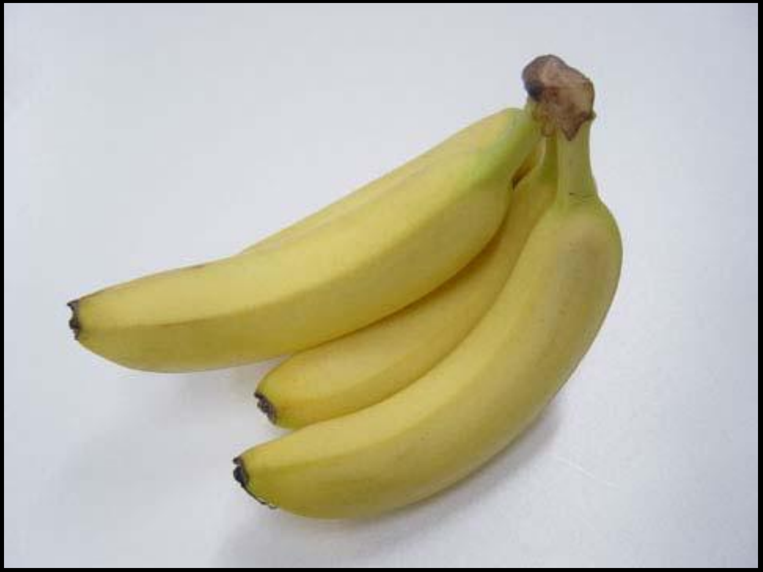
Our Framework (Texture Rendering)



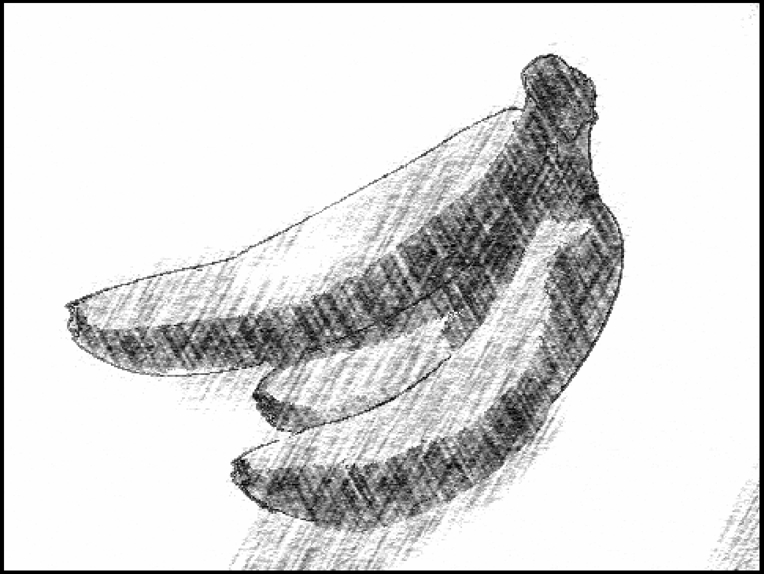
Input

Guess Hatch Direction

Our Framework (Texture Rendering)

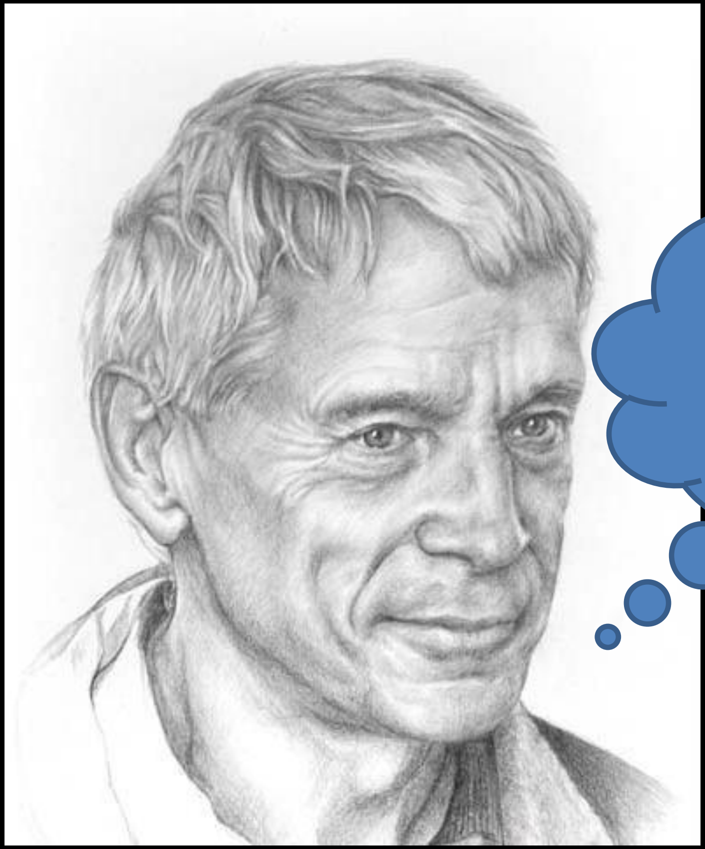


Input



Guess Hatch Direction

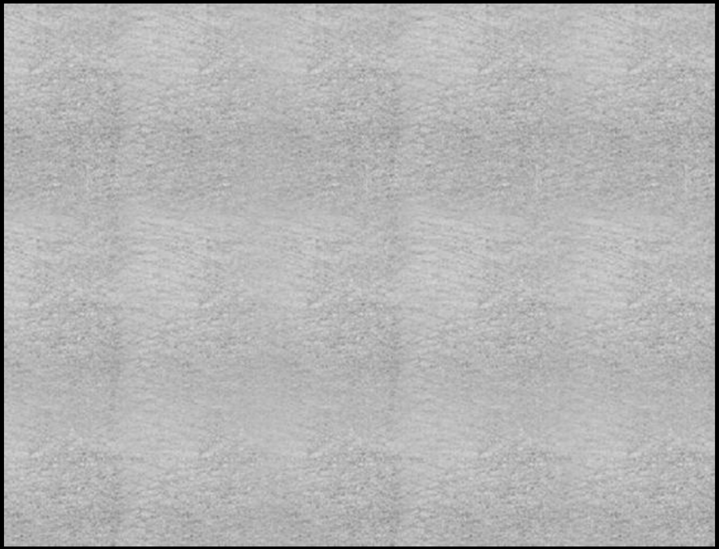
Our Framework (Texture Rendering)



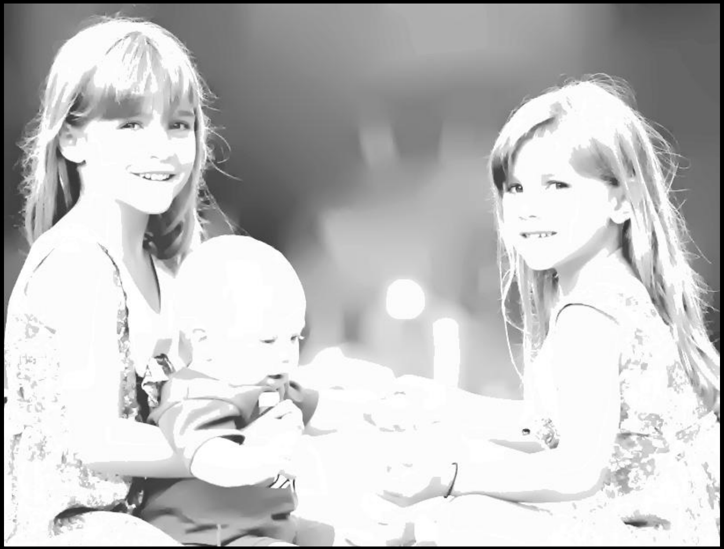
No Guess!
So it is robust to
all cases!

Tonal texture case
Drawn by Katja, 2008

Our Framework (Texture Rendering)



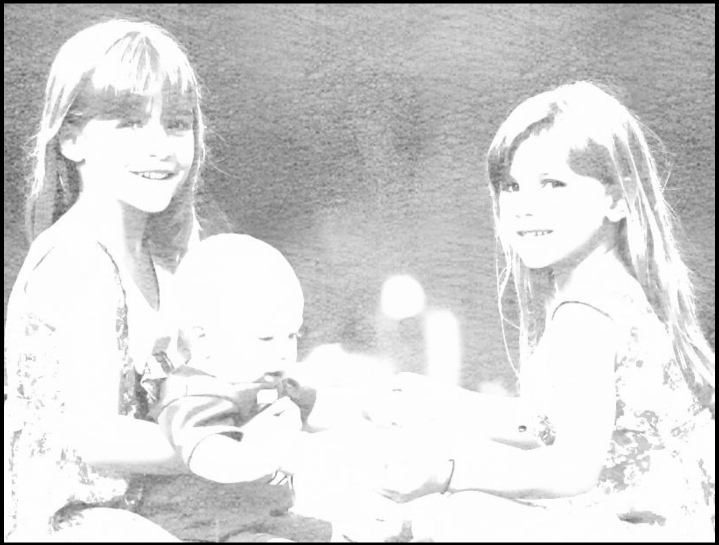
Tonal texture



Pencil Tone

Pencil Texture Transfer

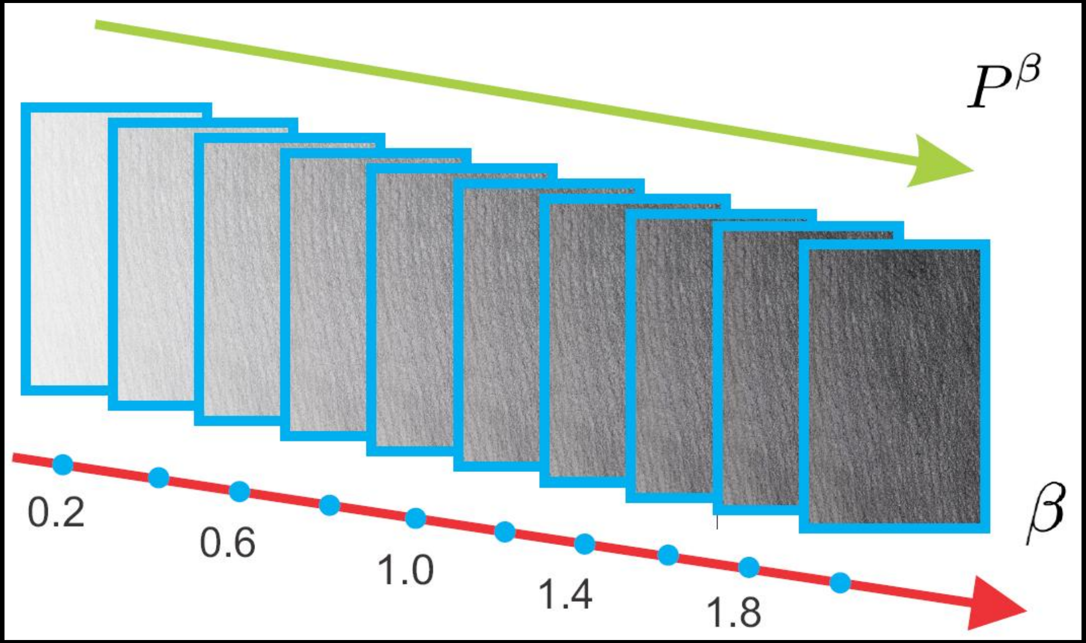
Our Framework (Texture Rendering)



Pencil Texture Transfer

Our Framework (Texture Rendering)

P



Multiplication Model

Our Framework (Texture Rendering)

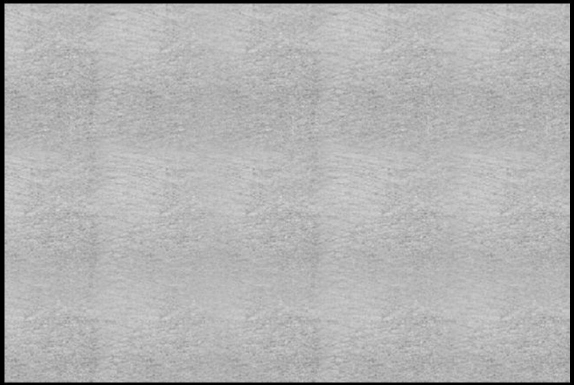
- In multiplication model

Repeated drawing

$$P(x)^{\beta(x)} \approx J(x)$$

Tone Mapping result

Tonal Texture



Our Framework (Texture Rendering)

- In multiplication model, we require:

$$\beta(x)\log\left[P(x)\right]\approx\log\left[J(x)\right]$$

Our Framework (Texture Rendering)

- In multiplication model, we require:

$$\min_{\beta} \left\| \beta \log[P] - \log[J] \right\|_2^2$$

Our Framework (Texture Rendering)

- In multiplication model, we require:

$$\min_{\beta} \left\| \beta \log[P] - \log[J] \right\|_2^2$$

- We also require: Local Smoothness

$$\min_{\beta} \left\| \nabla \beta \right\|_2^2$$

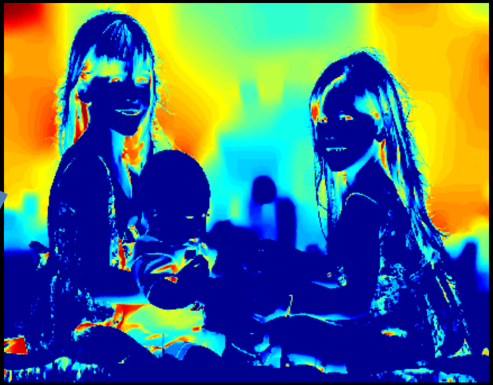
Our Framework (Texture Rendering)

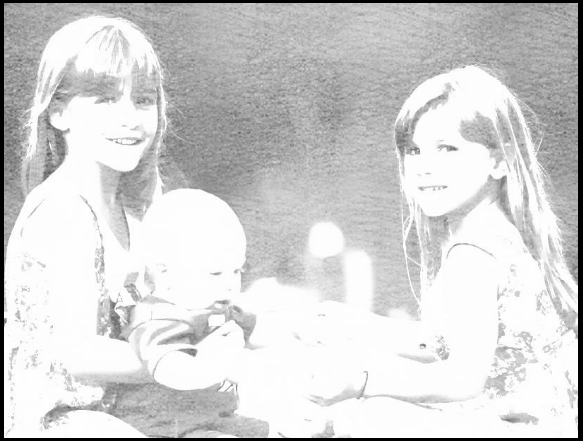
Objective function

$$\beta^* = \arg\min_{\beta} \left\{ \left\| \beta \log[P] - \log[J] \right\|_2^2 + \lambda \left\| \nabla \beta \right\|_2^2 \right\}$$

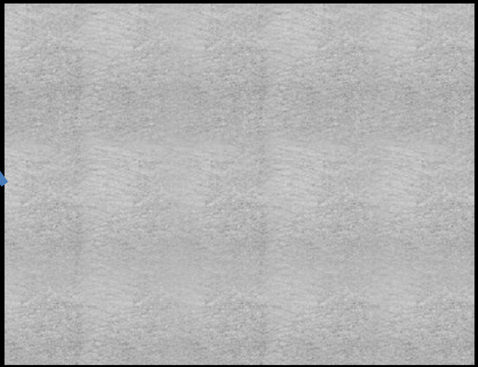
Our Framework (Texture Rendering)

- Rendering Result

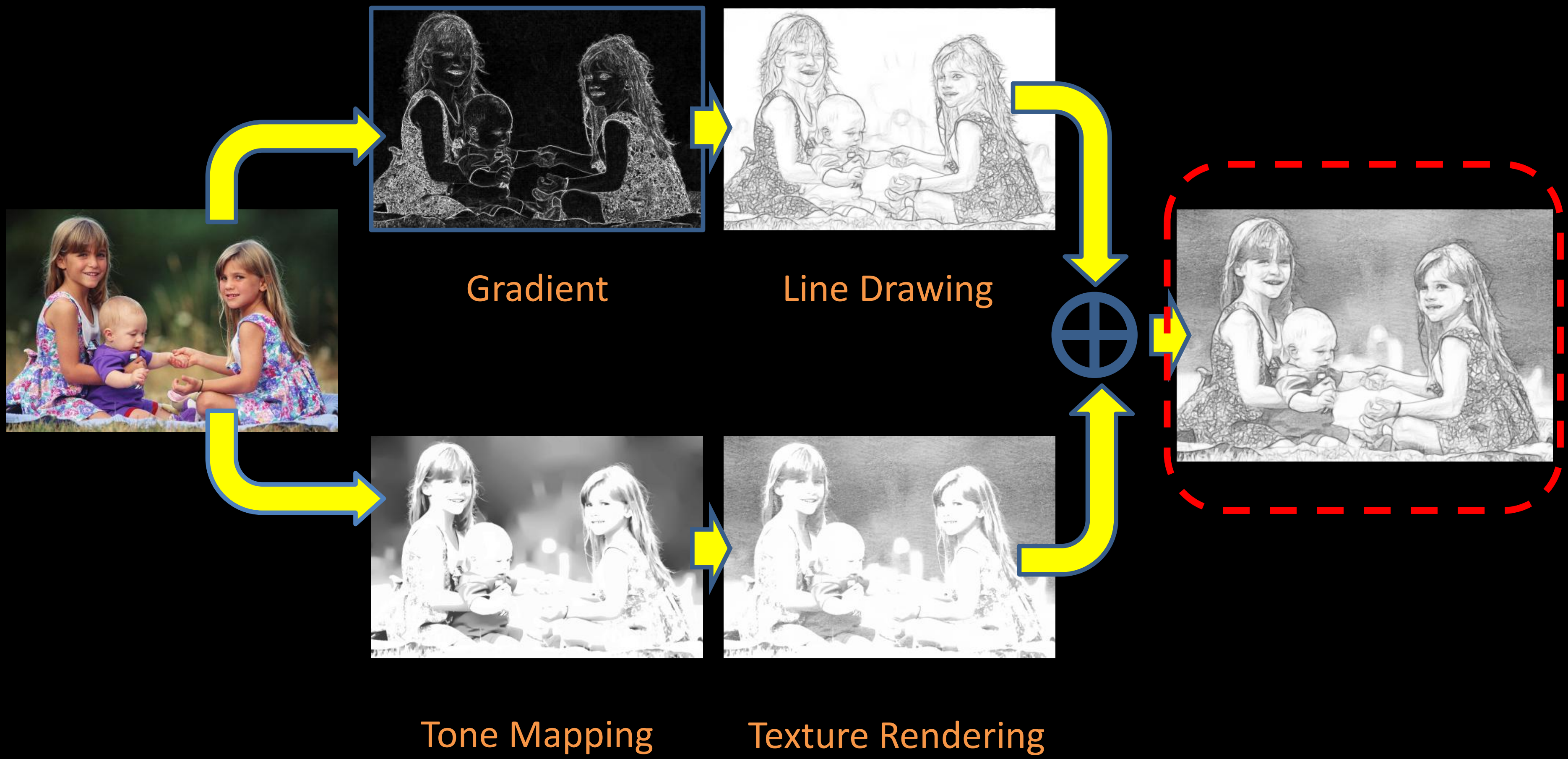




$$S = P^{\beta^*}$$



Our Framework (Tone Mapping)



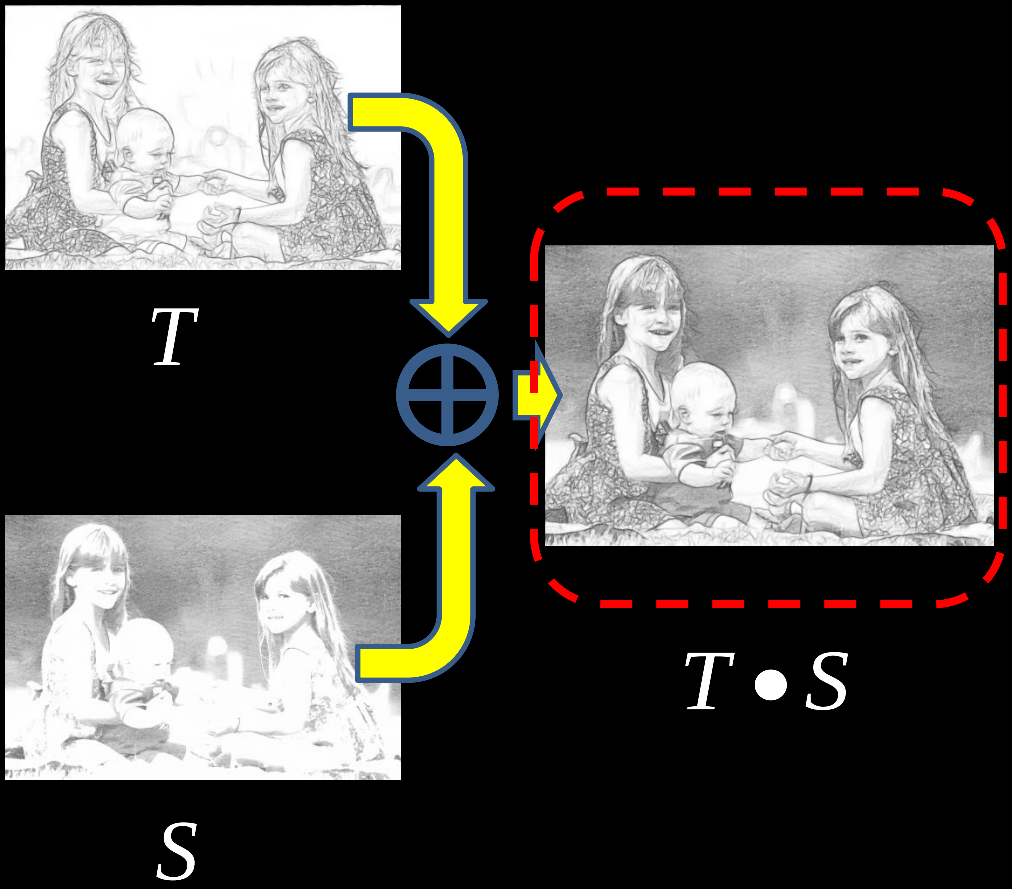
Gradient

Line Drawing

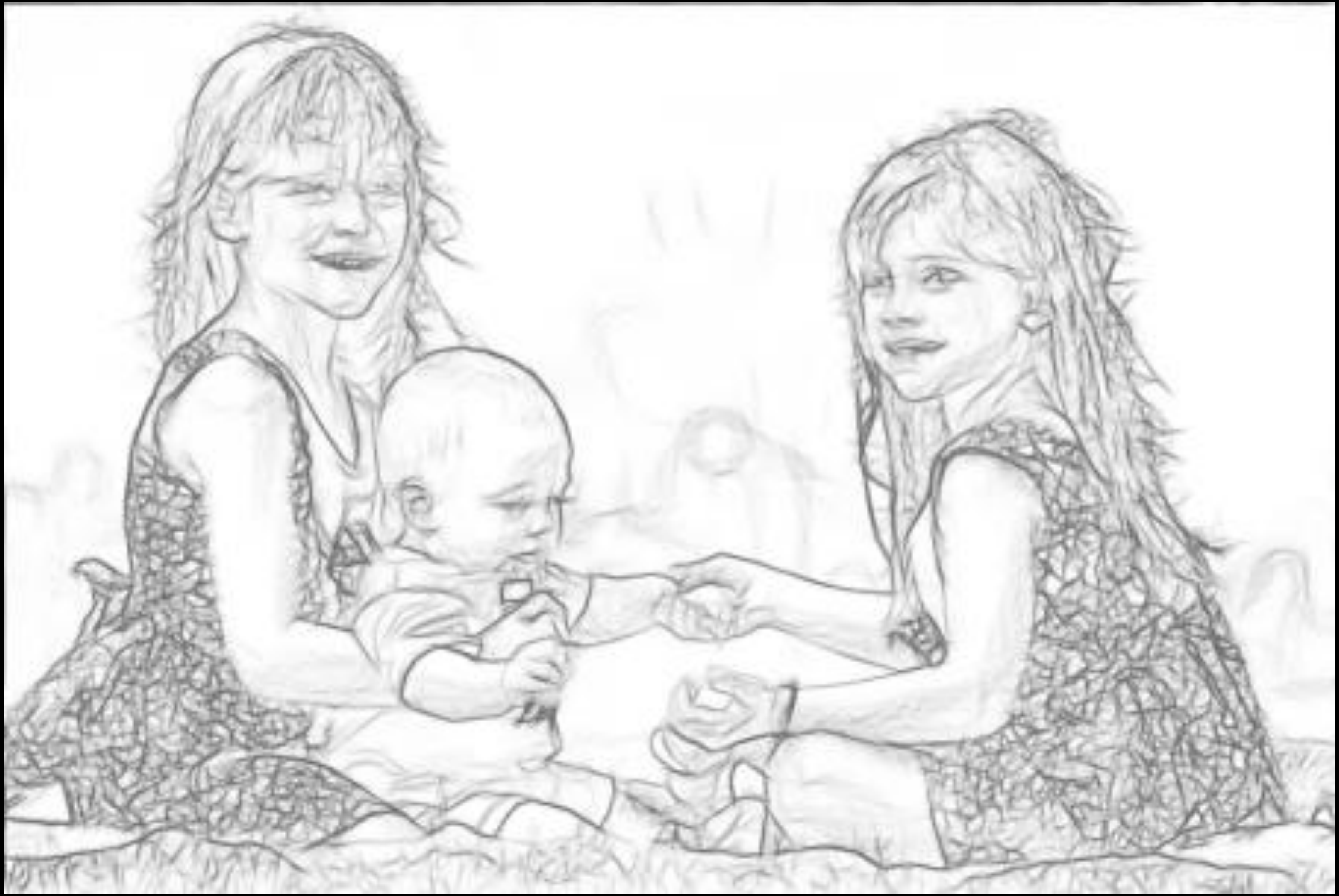
Tone Mapping

Texture Rendering

Our Framework (Combination)



Our Framework (Combination)

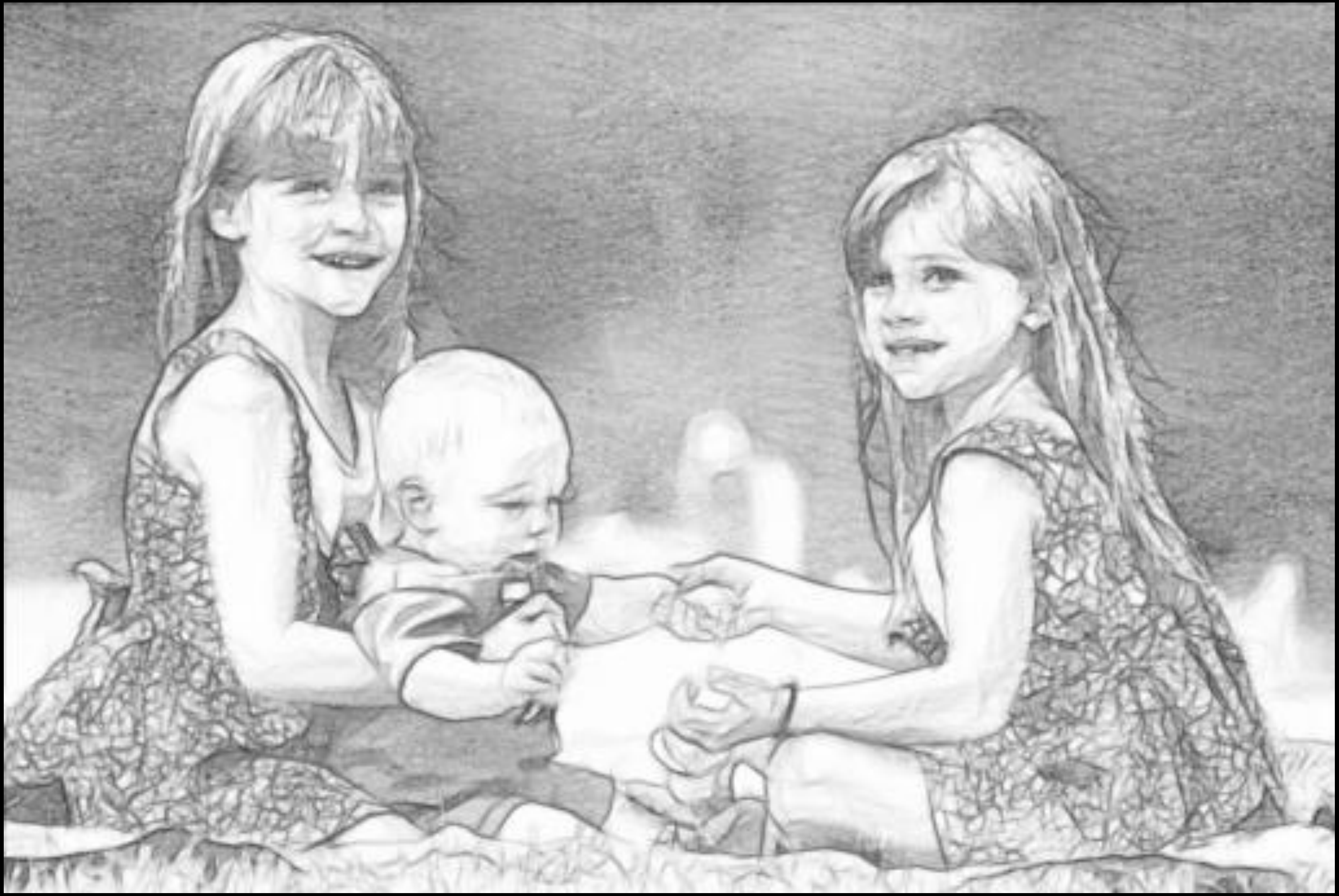


T

Our Framework (Combination)

S

Our Framework (Combination)

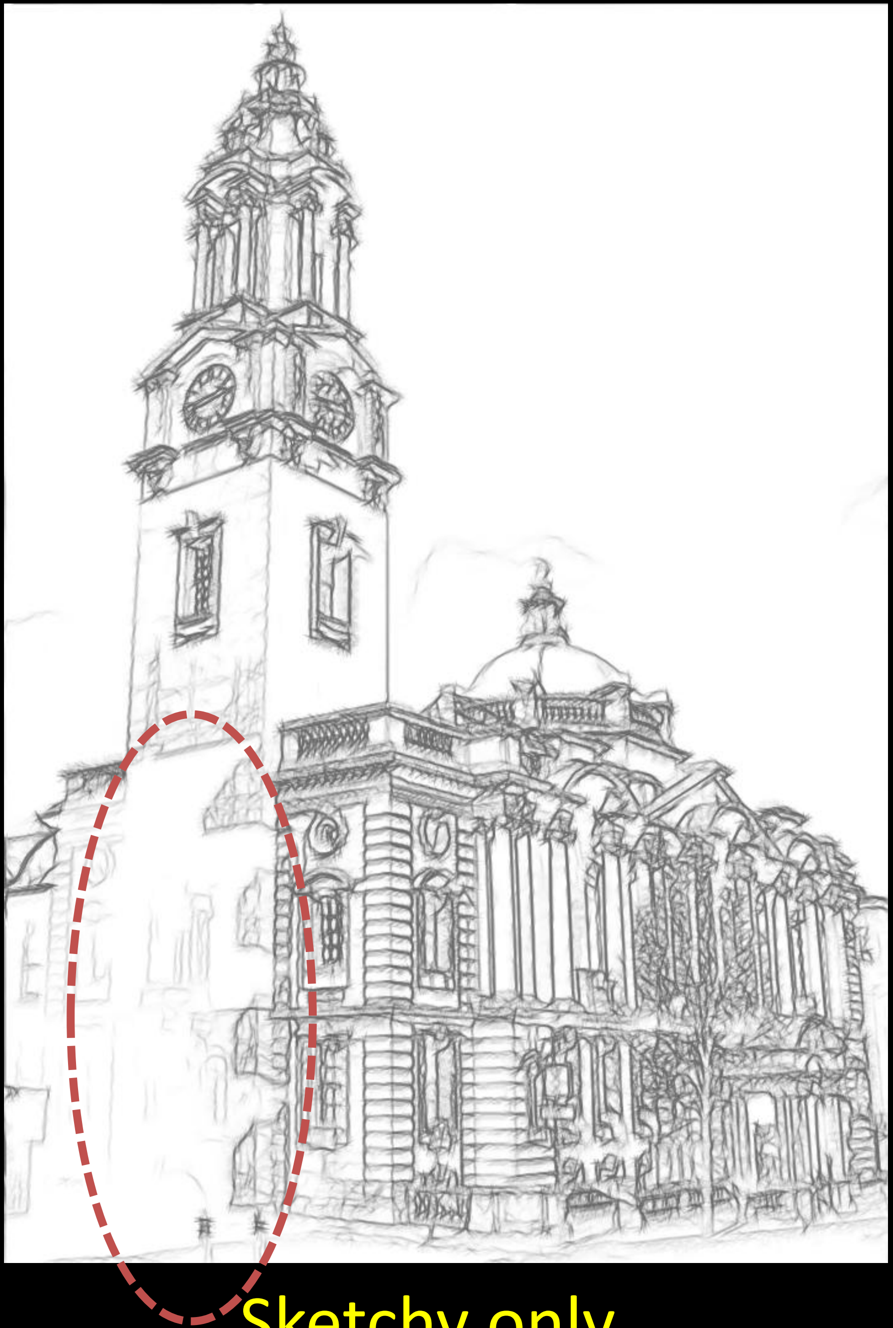


$T \bullet S$

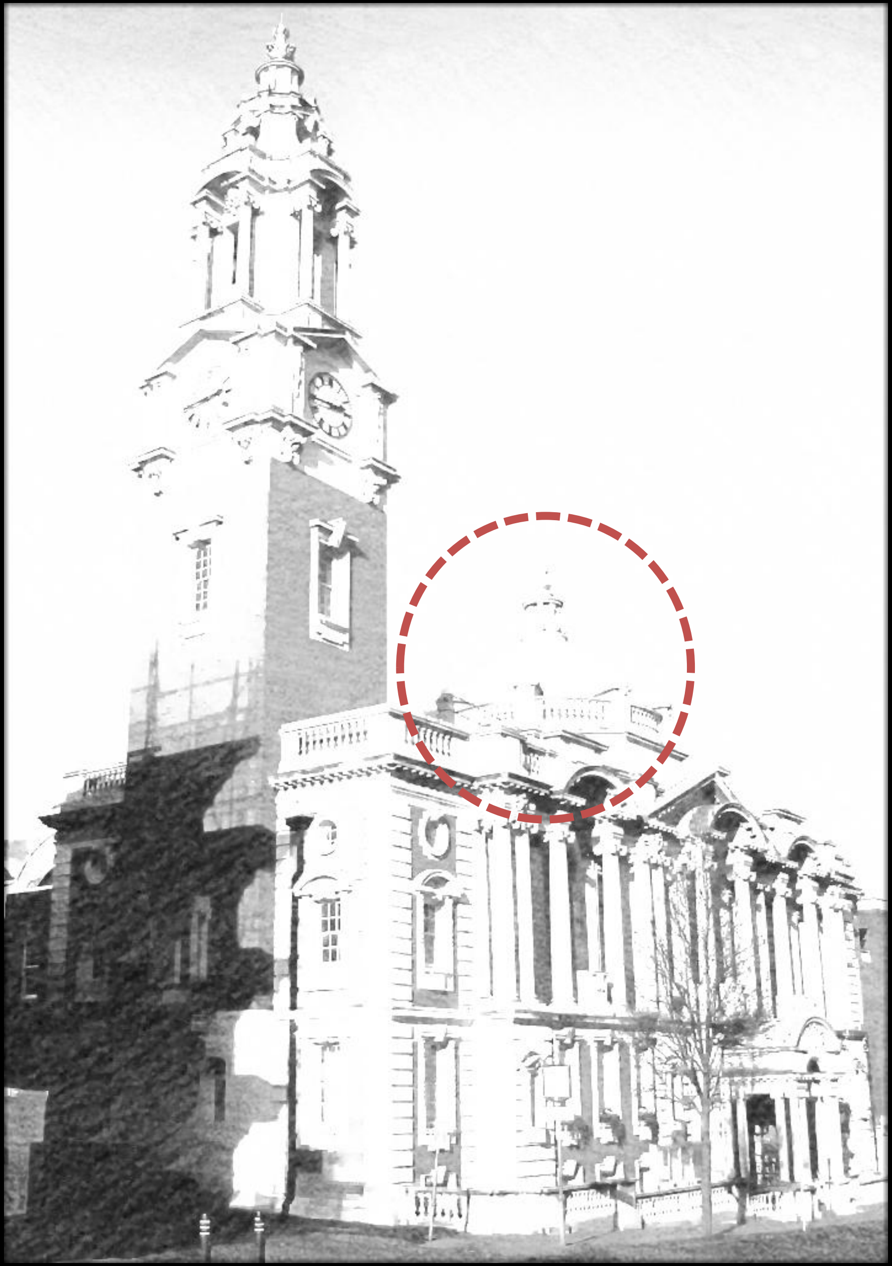
Outline

- Previous work
- Observation (Two stages)
- Framework (Pencil drawing prior including)
- **Discussion**
- Comparison and Results
- Conclusion

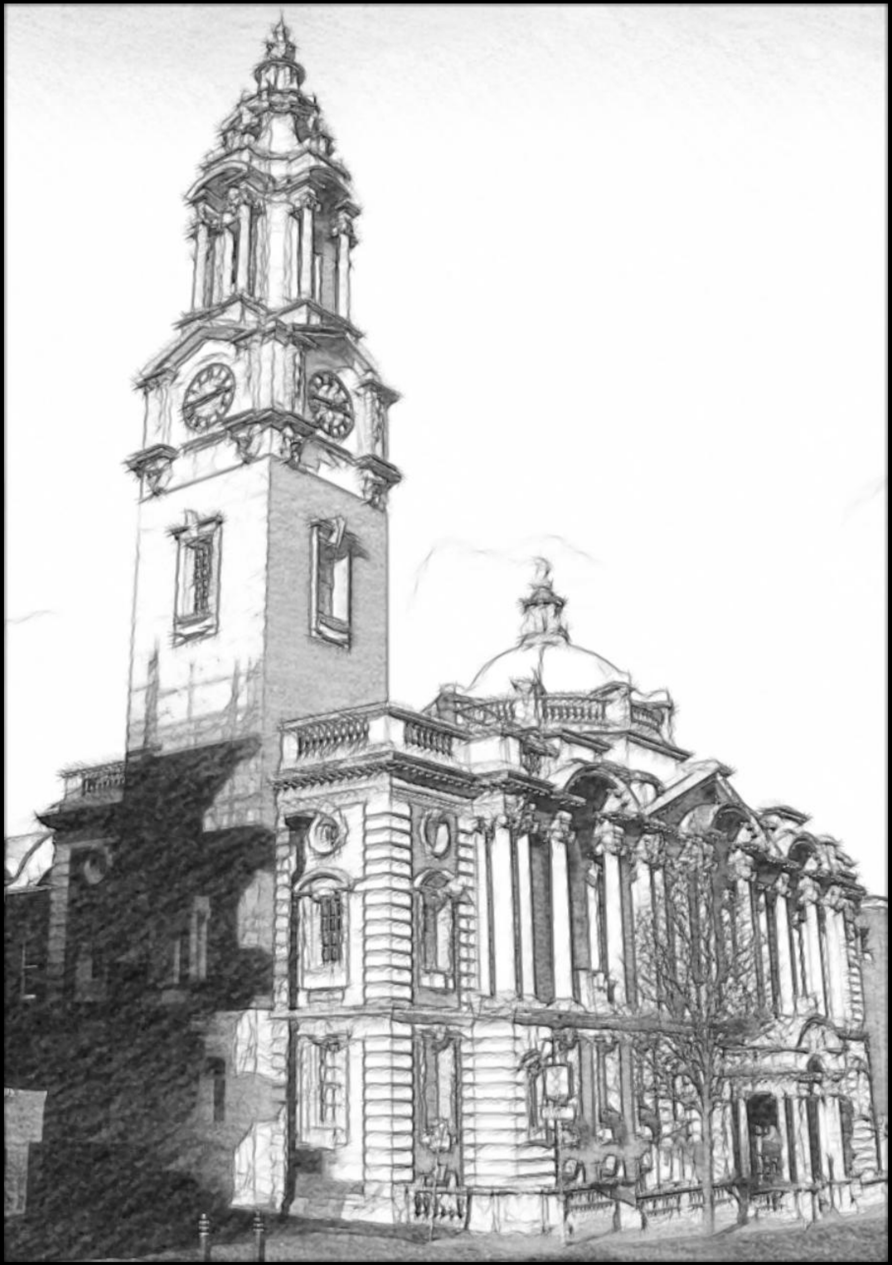
MARKET STREET



Sketchy only



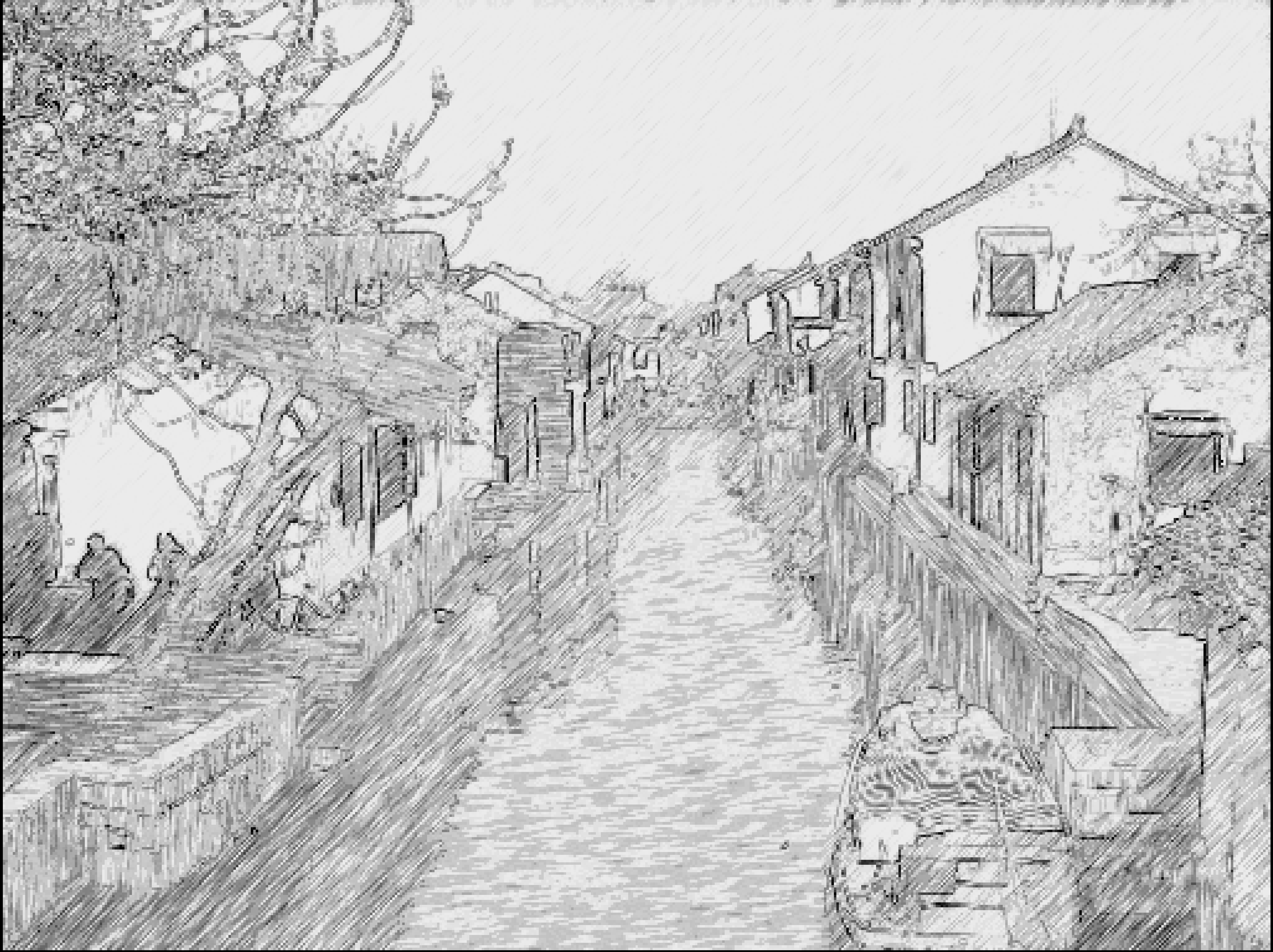
Tone only



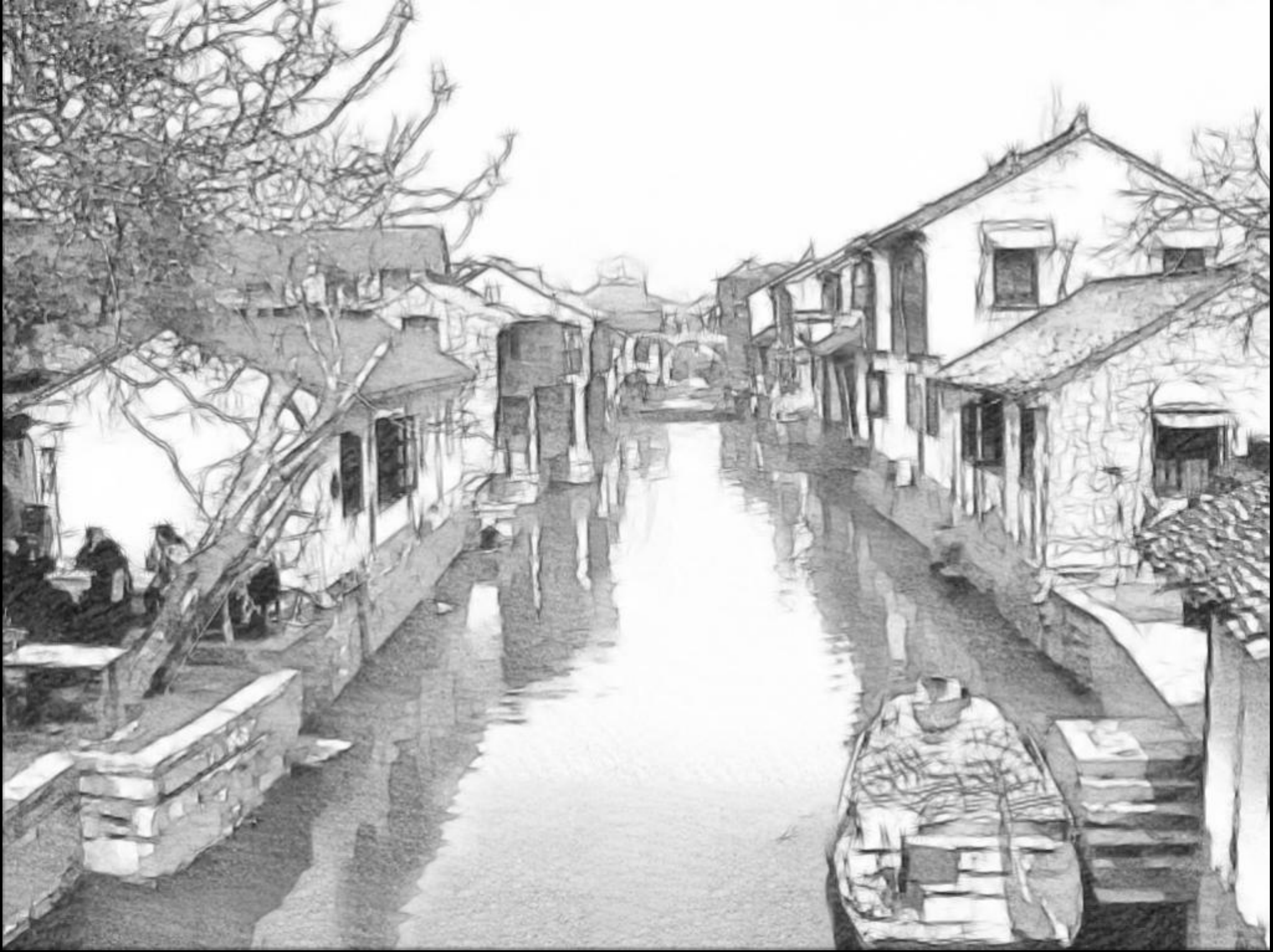
Combination

Outline

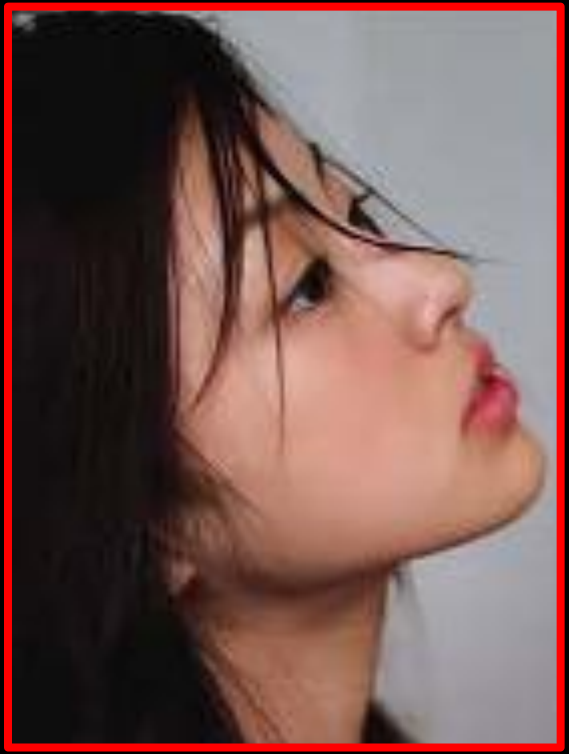
- Previous work
- Observation (Two stages)
- Framework (Pencil drawing prior including)
- Discussion
- **Comparison and Results**
- Conclusion



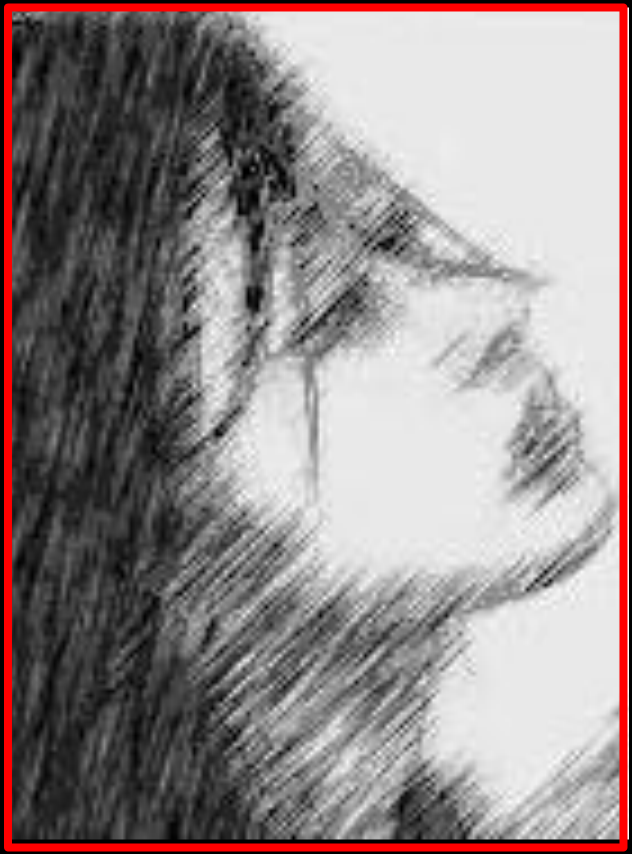
Sun et al, Efficient Region-Based Pencil Drawing,2007



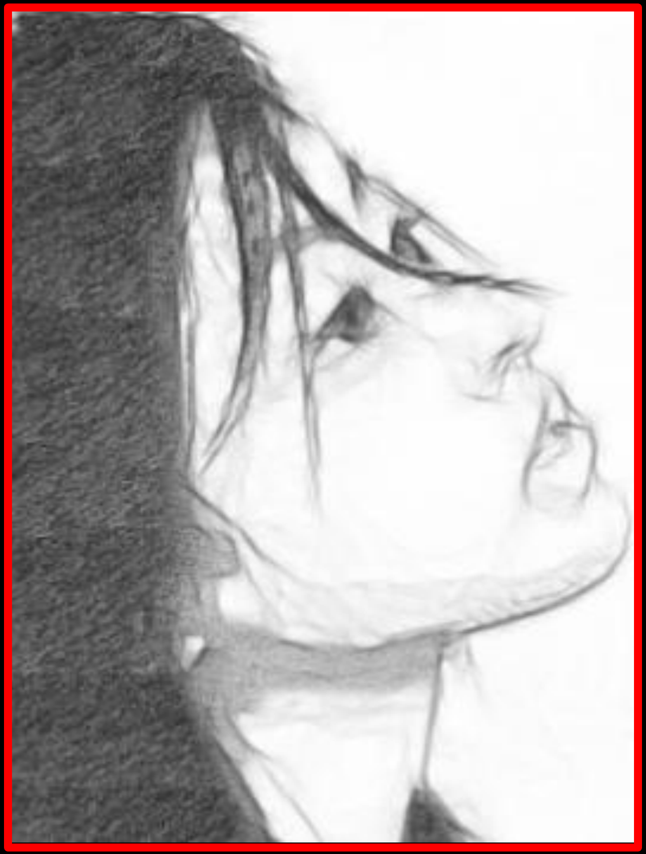
Ours



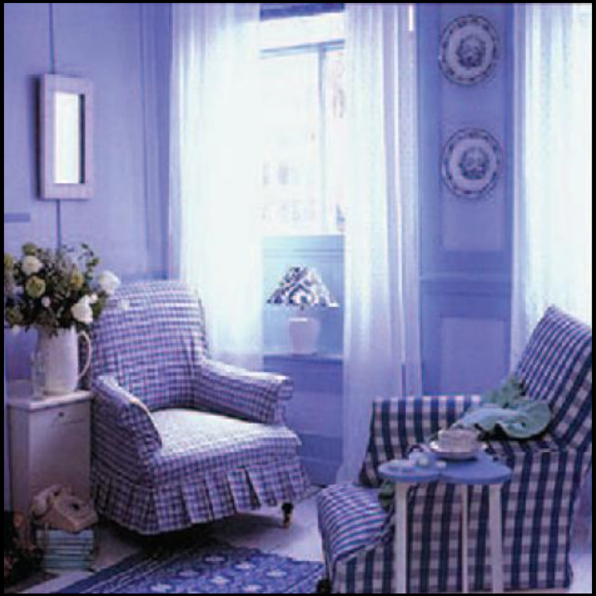
Input



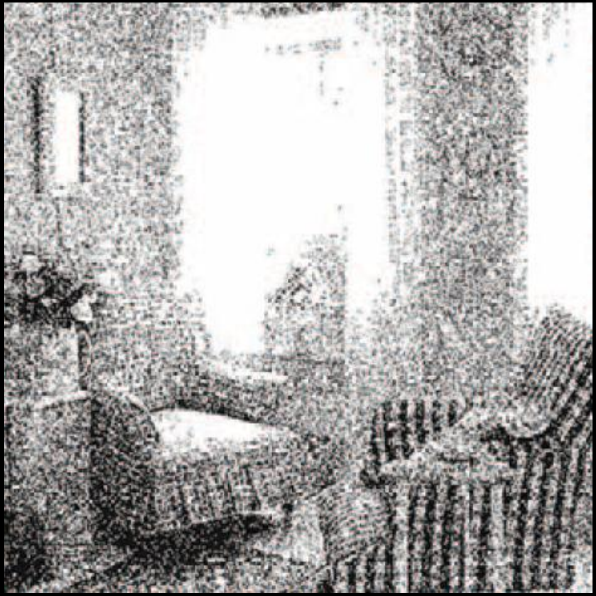
Sun et al, 2007



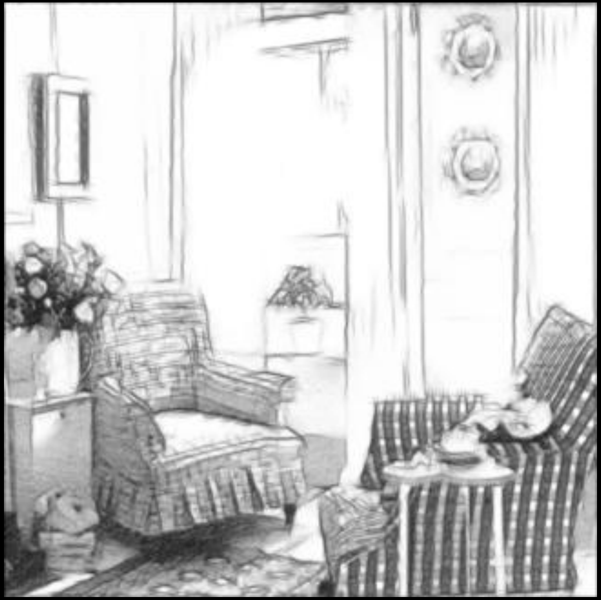
Ours



Input



Gao et al, 2010

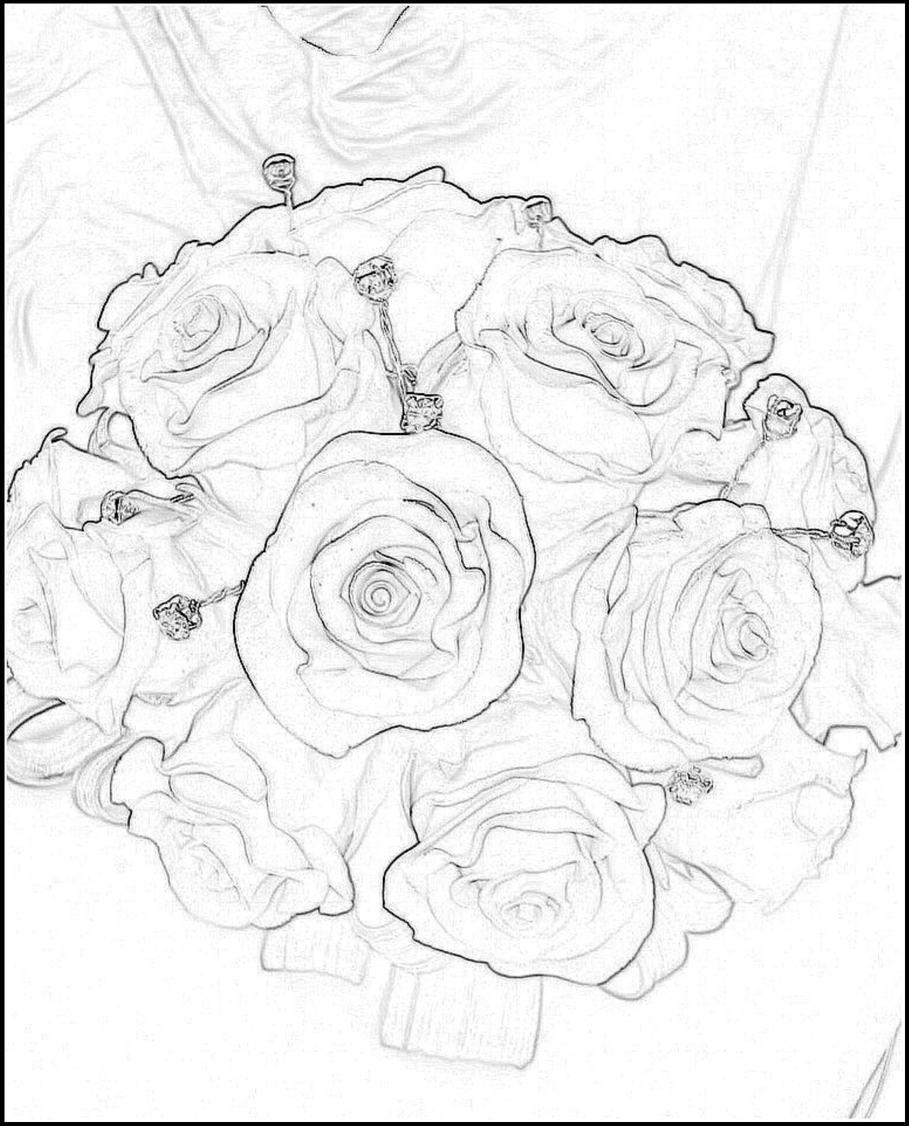


Ours

Gao et al, Automatic Generation of Pencil Sketch for 2D Images, 2010

Comparison with Software

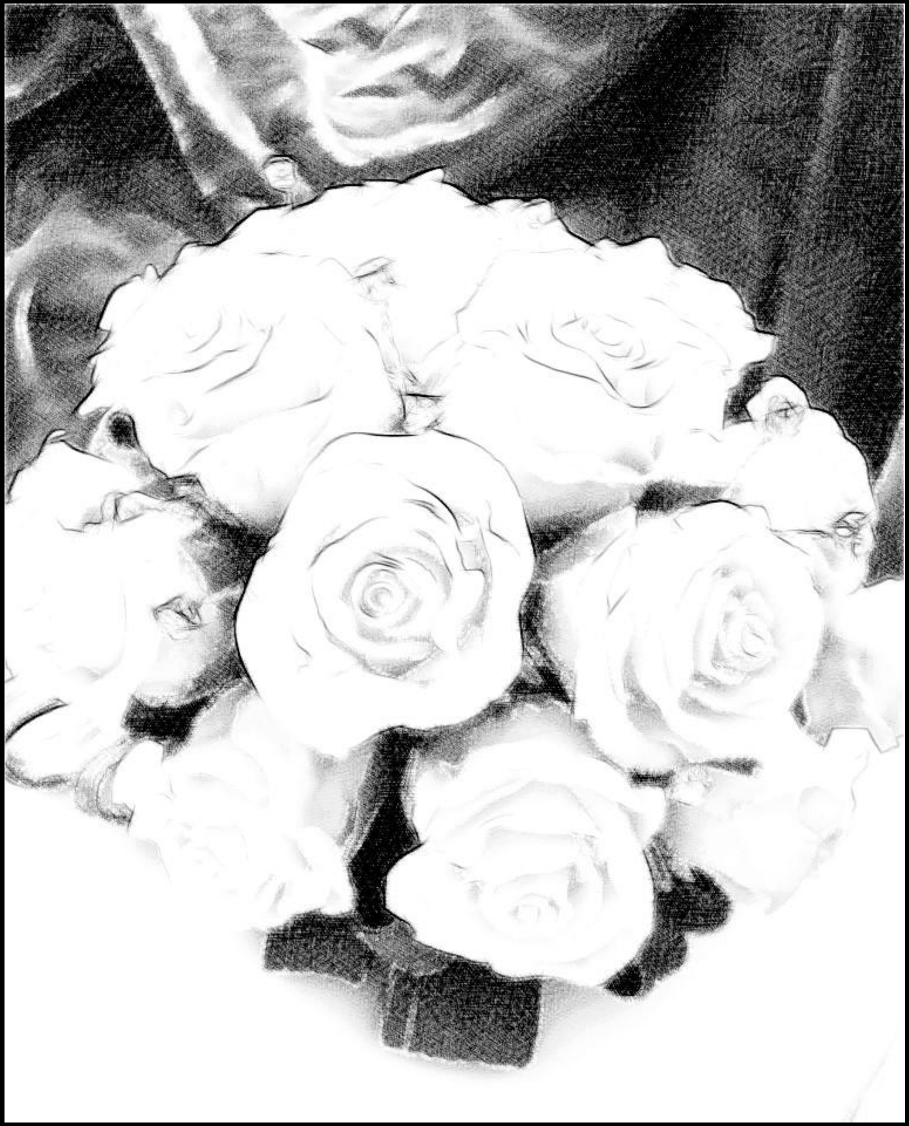
Input



PhototoSketch v3.5

http://download.cnet.com/Photo-to-Sketch/

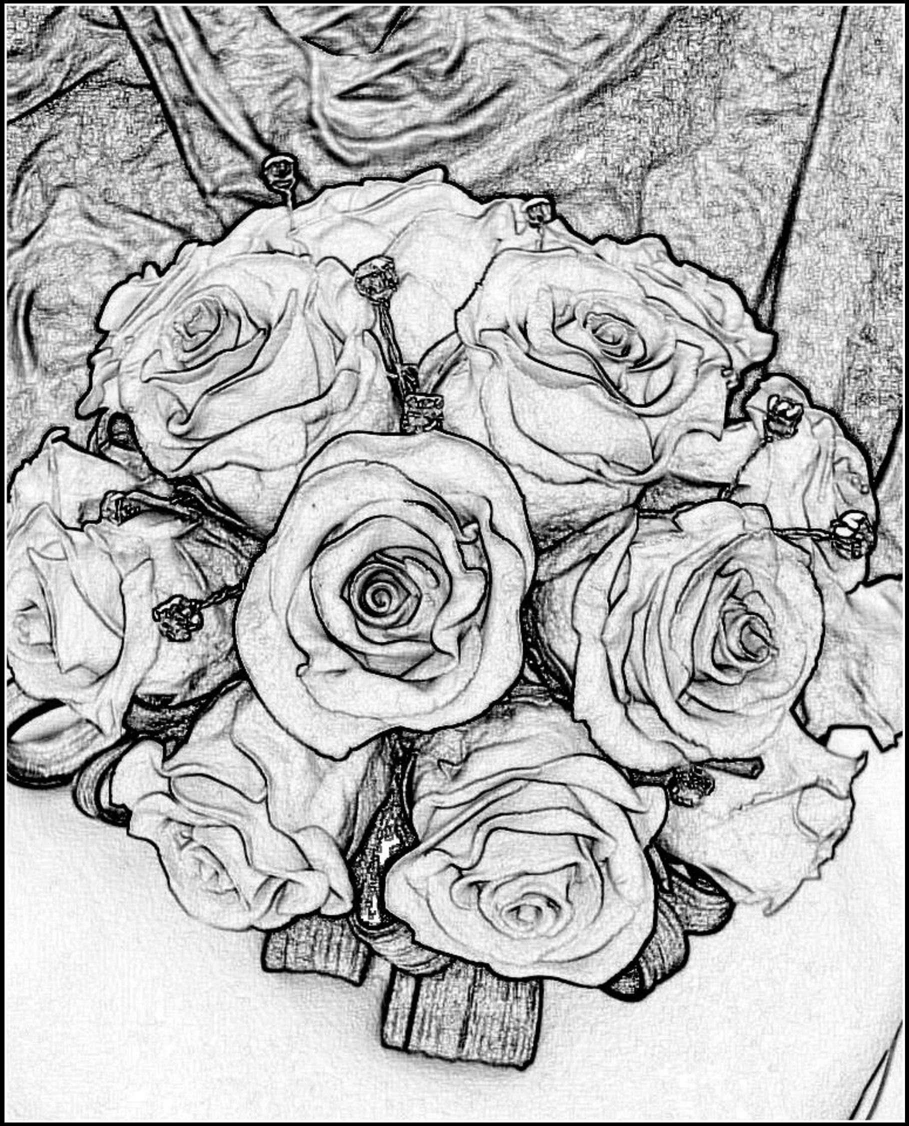
Input



FotoSketcher

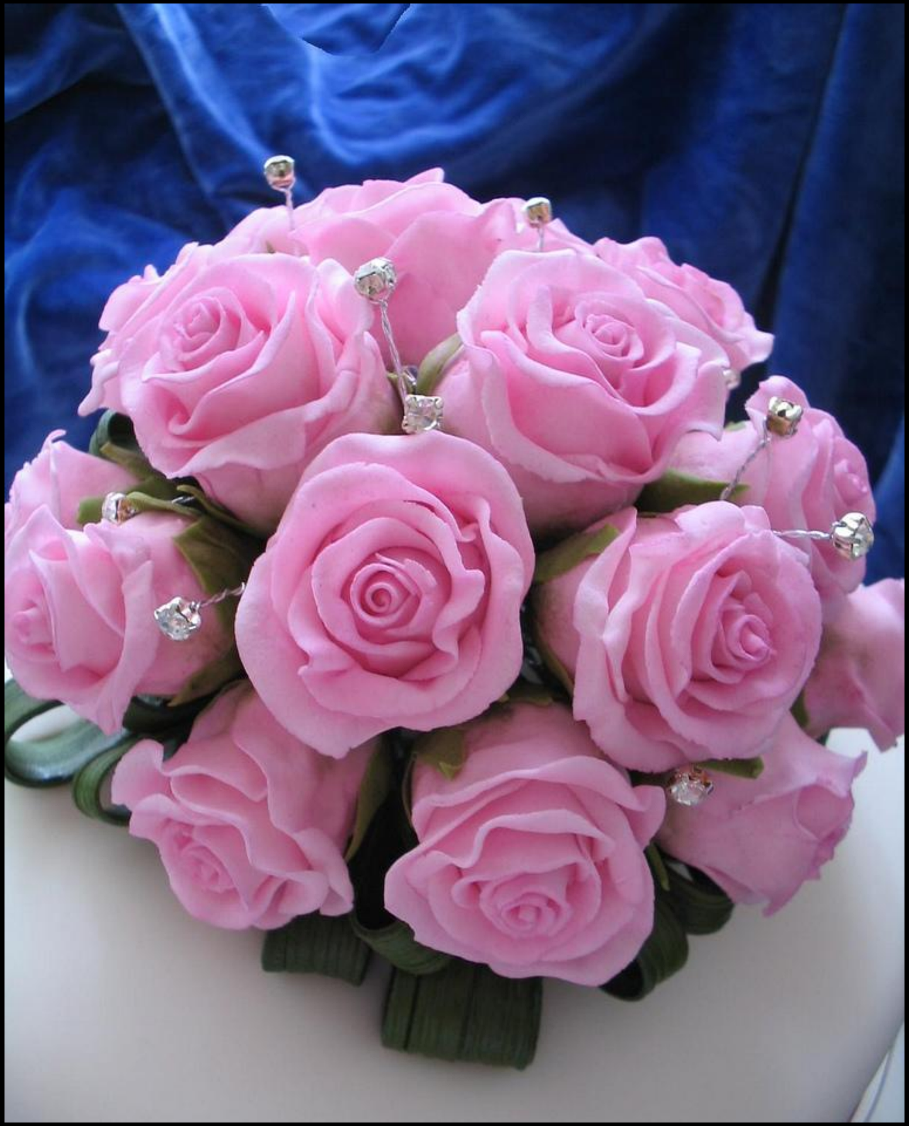
http://www.fotosketcher.com/FotoSketcher.exe

Input

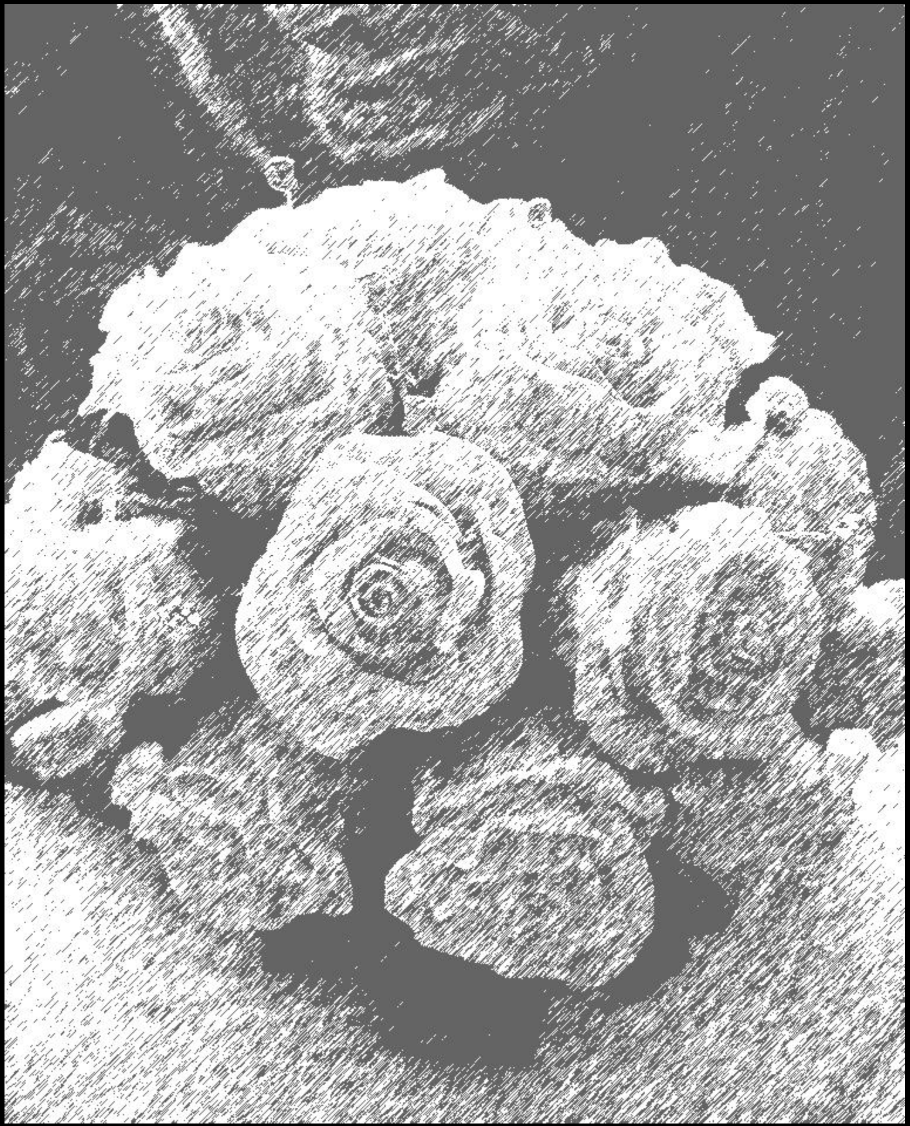


Instant

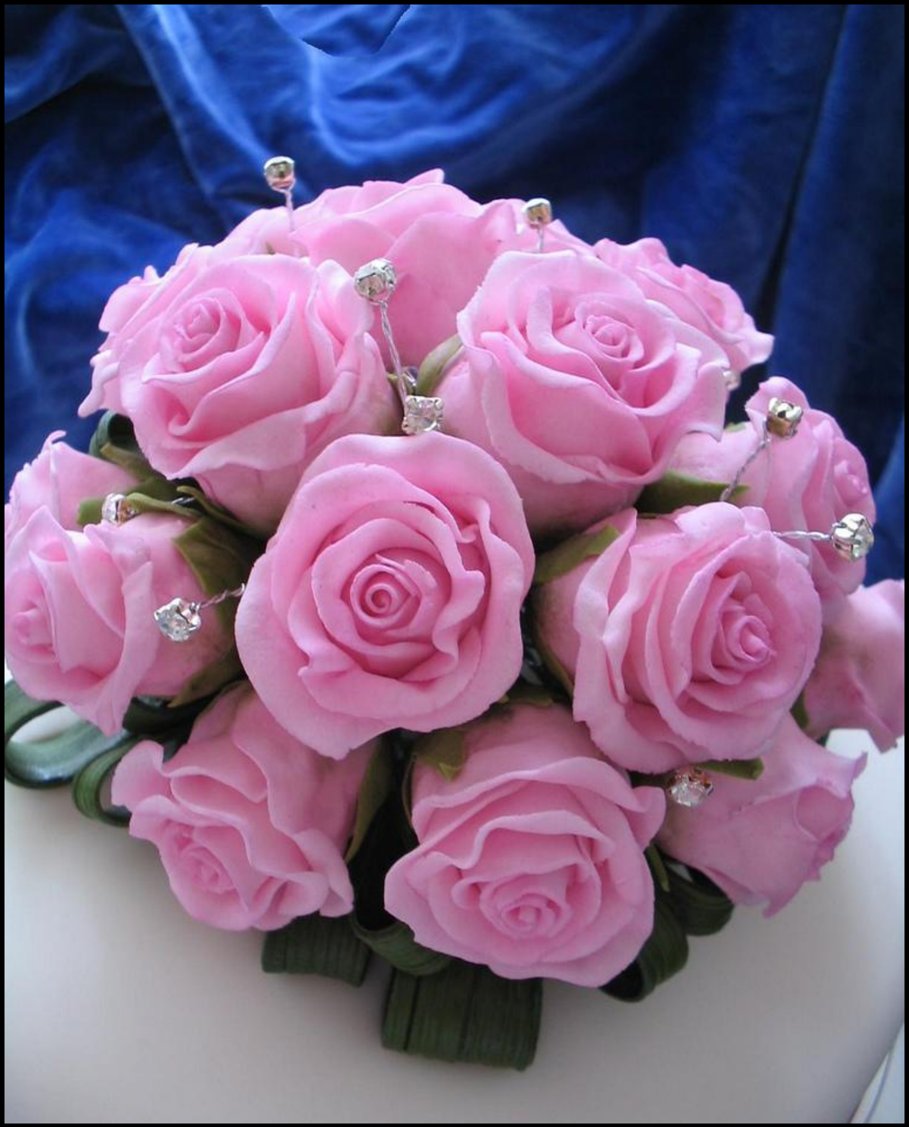
http://clipping-path-studio.com/downloads/InstantPhotoSketchSetup.exe



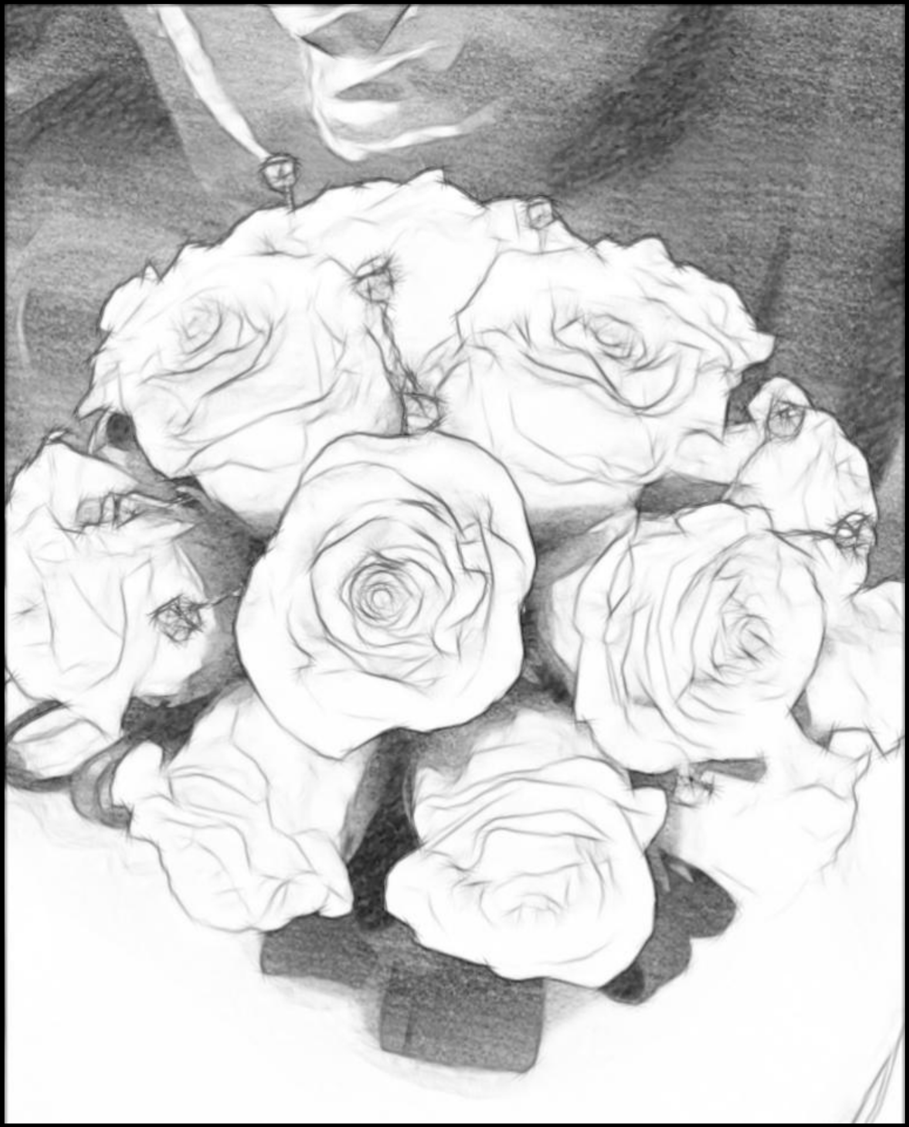
Input



Photoshop CS5

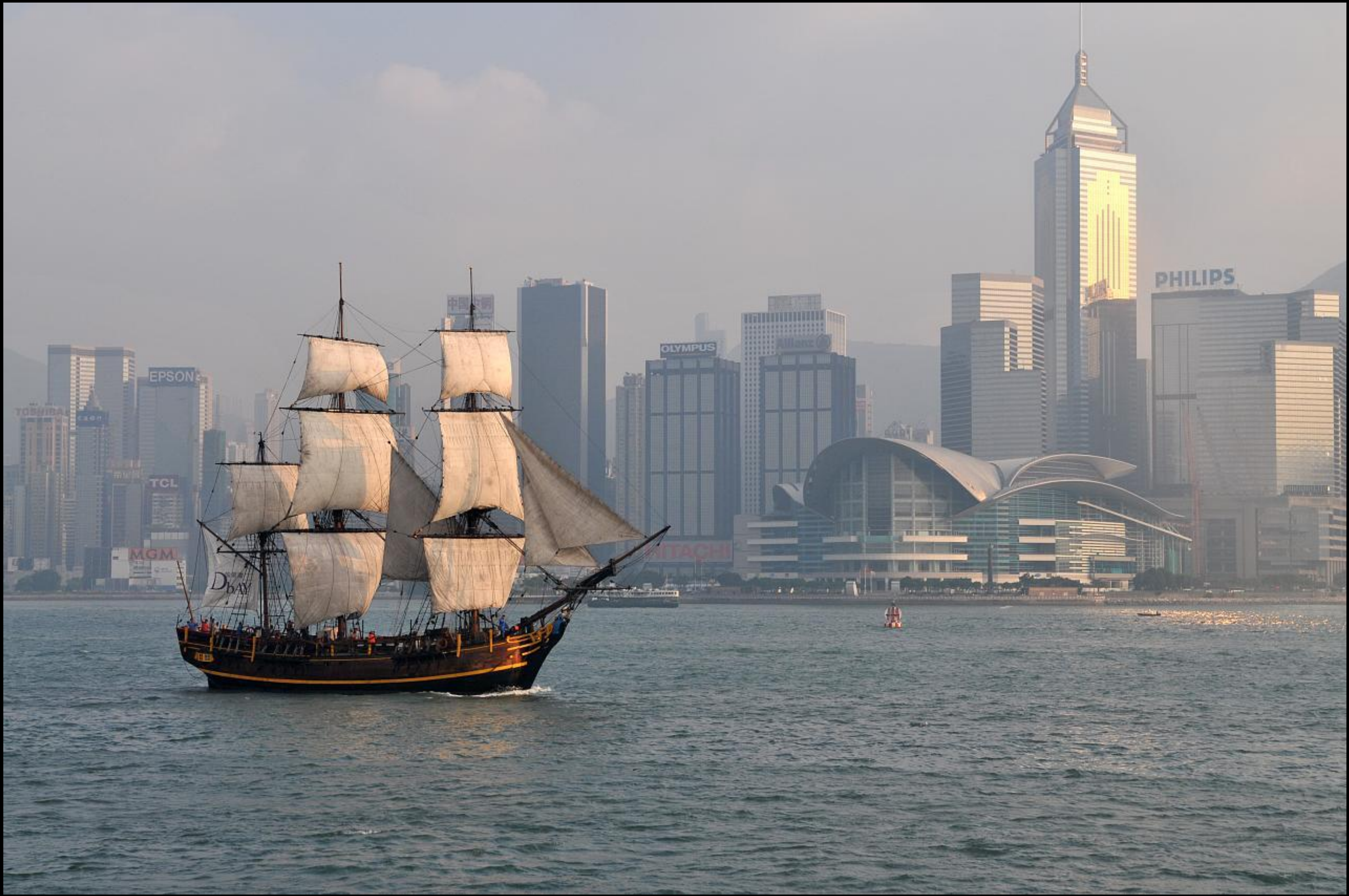


Input

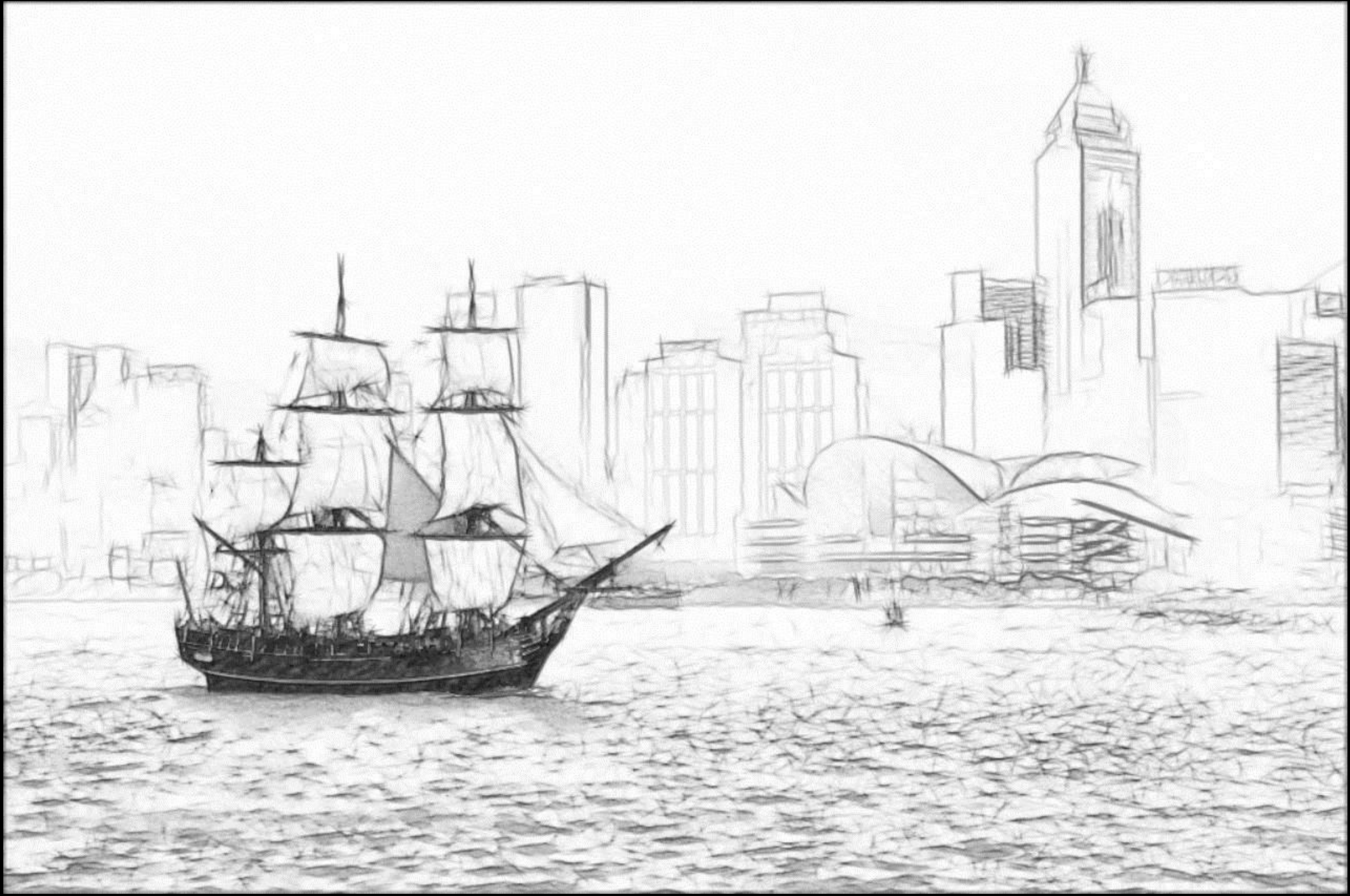


Ours

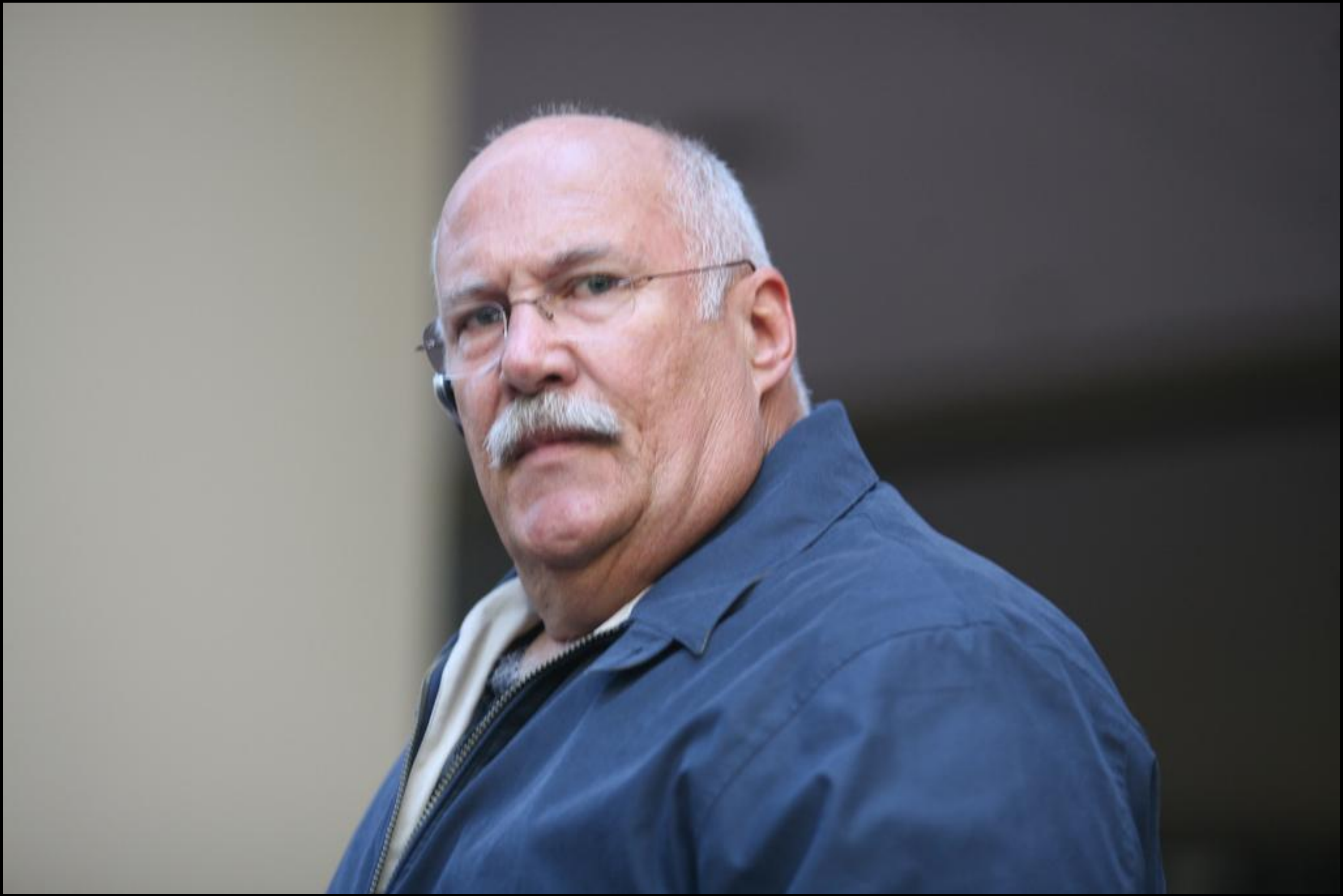
More Results



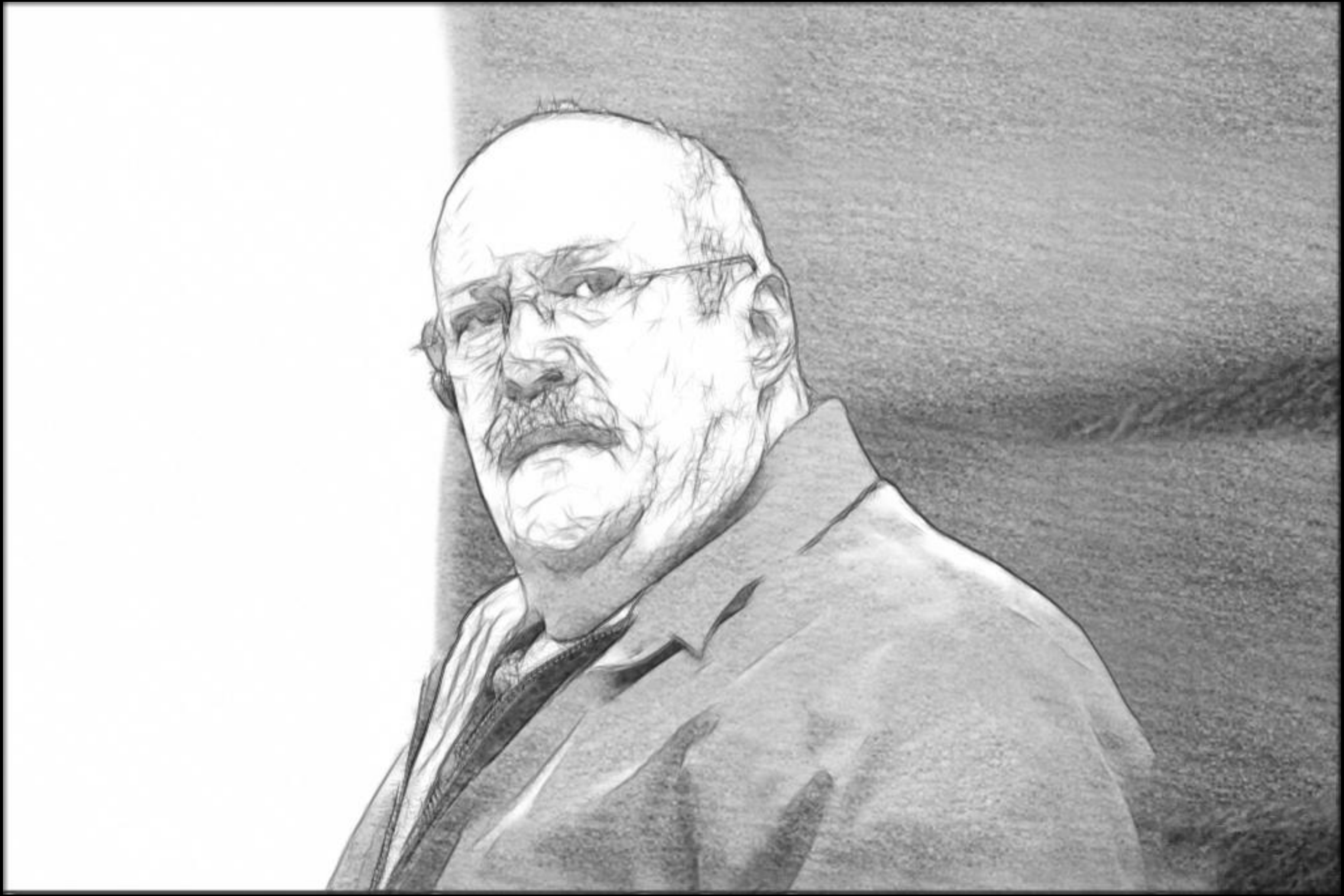
Input



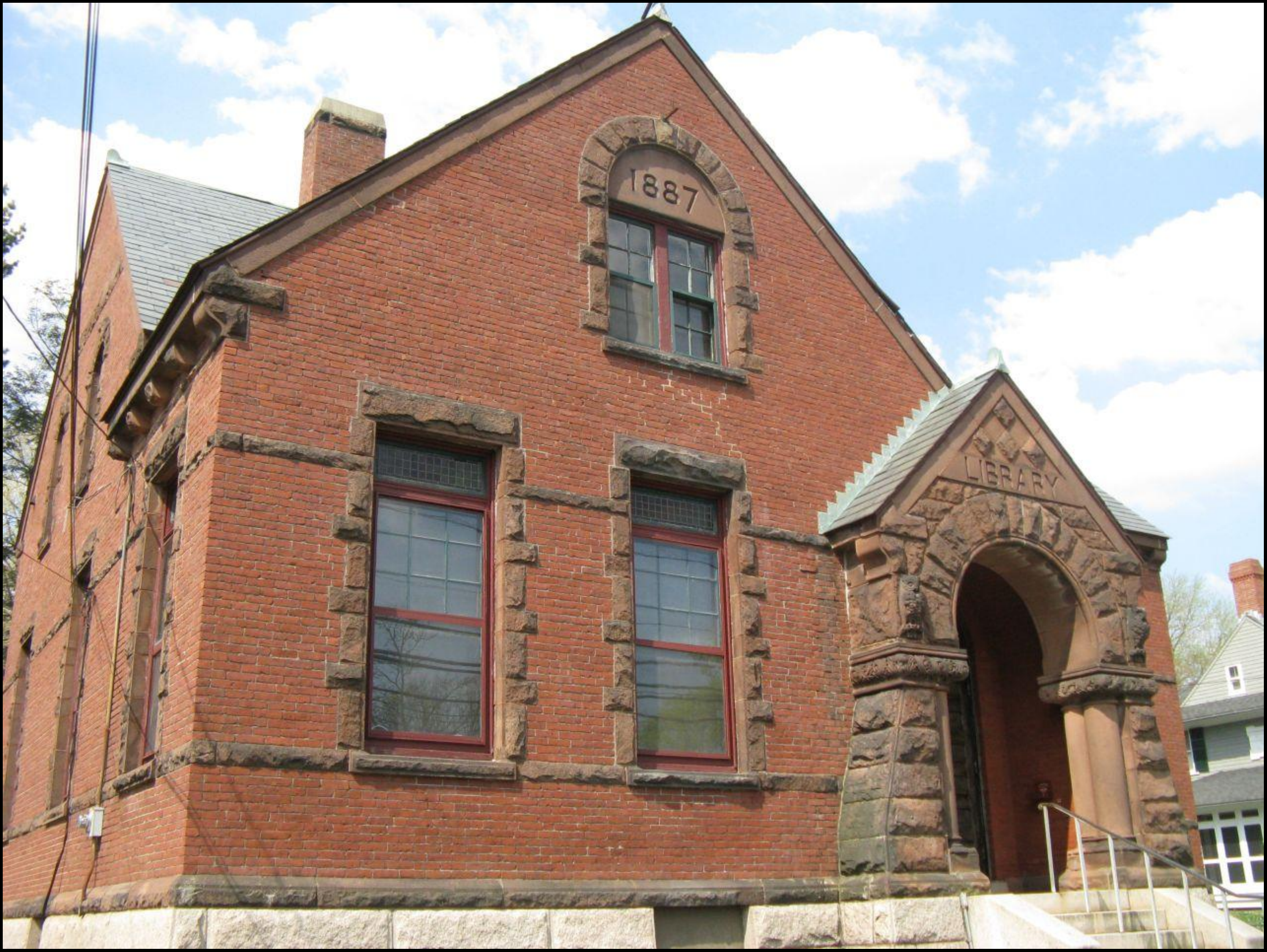
Output



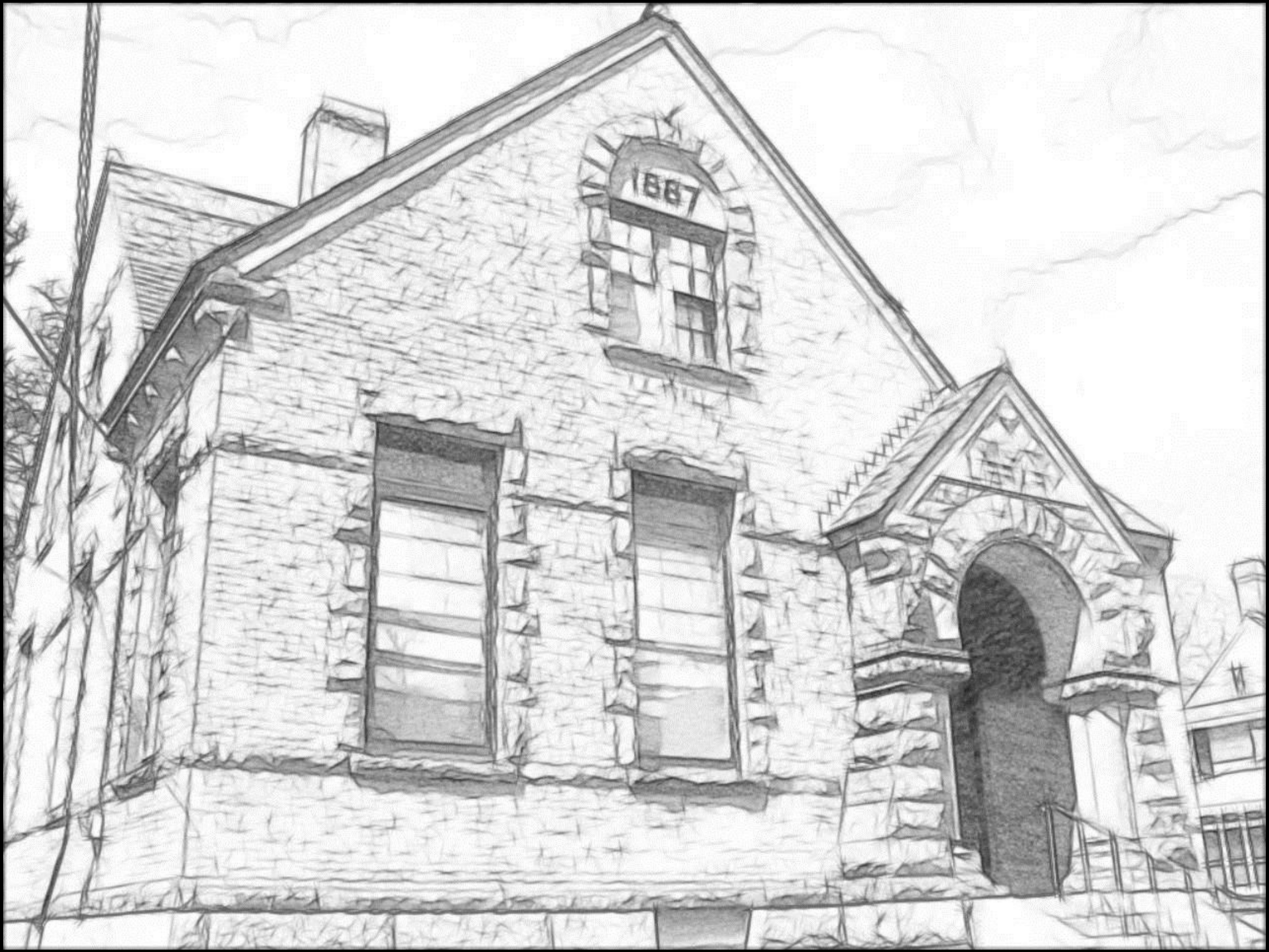
Input



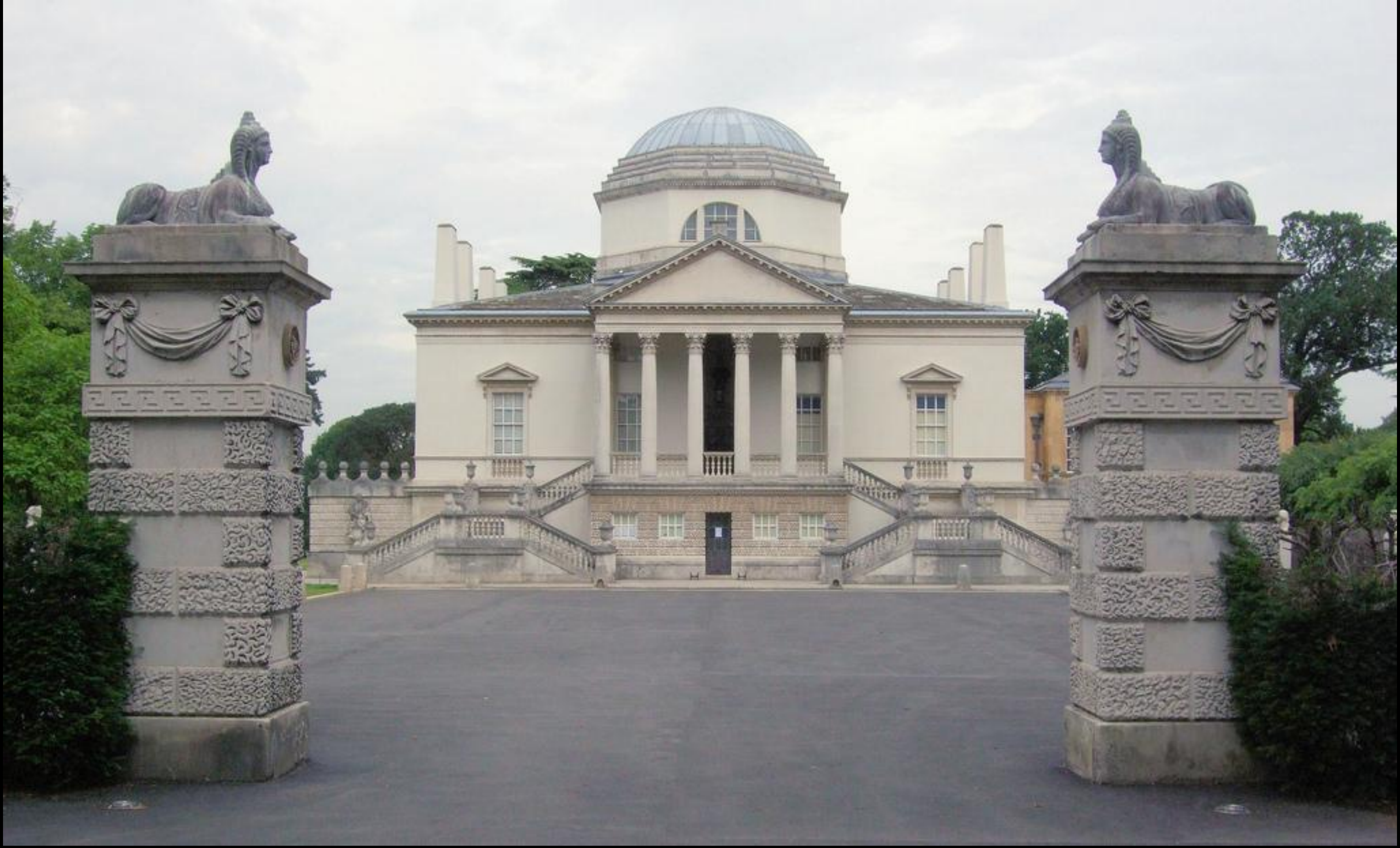
Output



Input

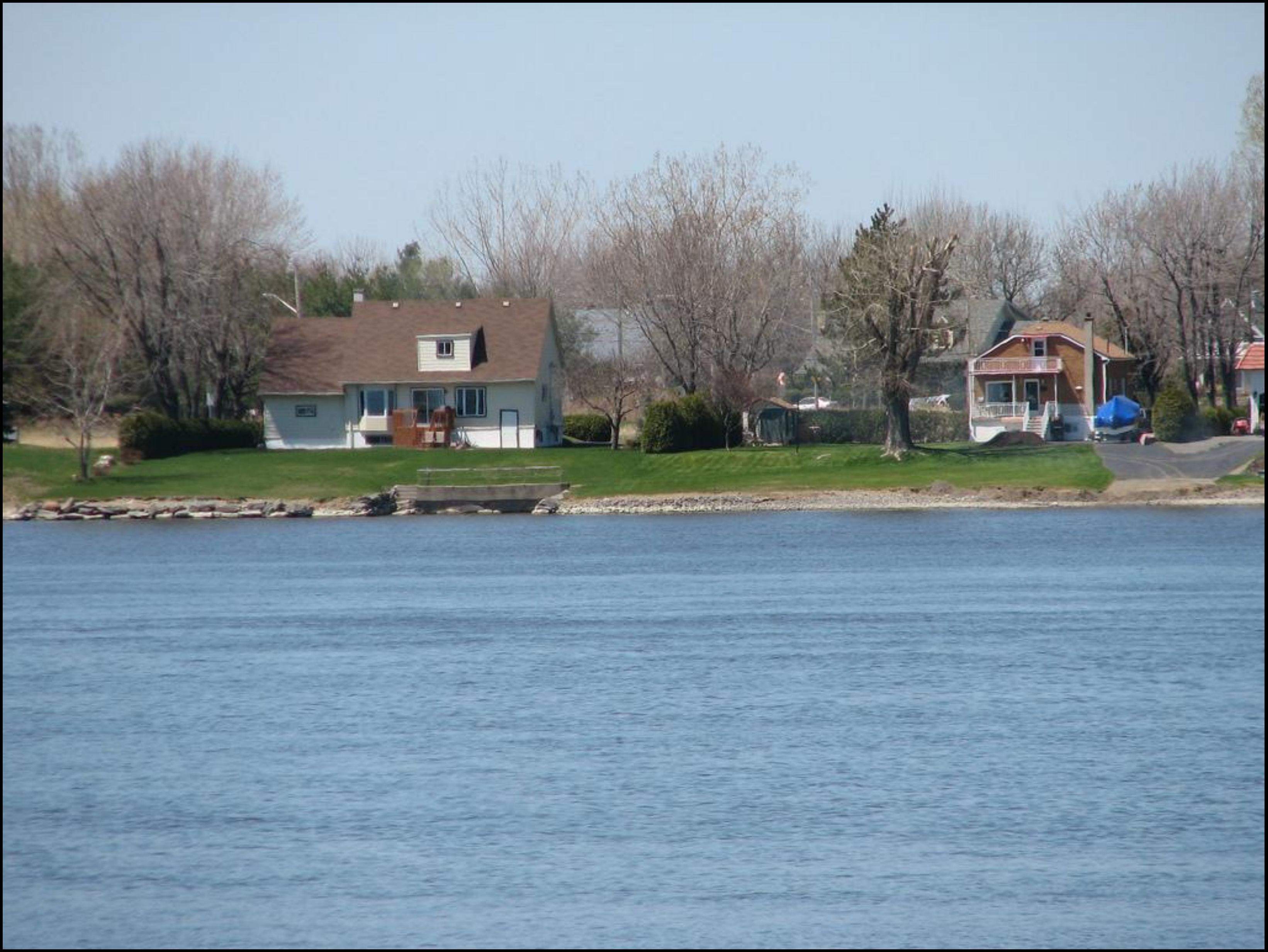


Output

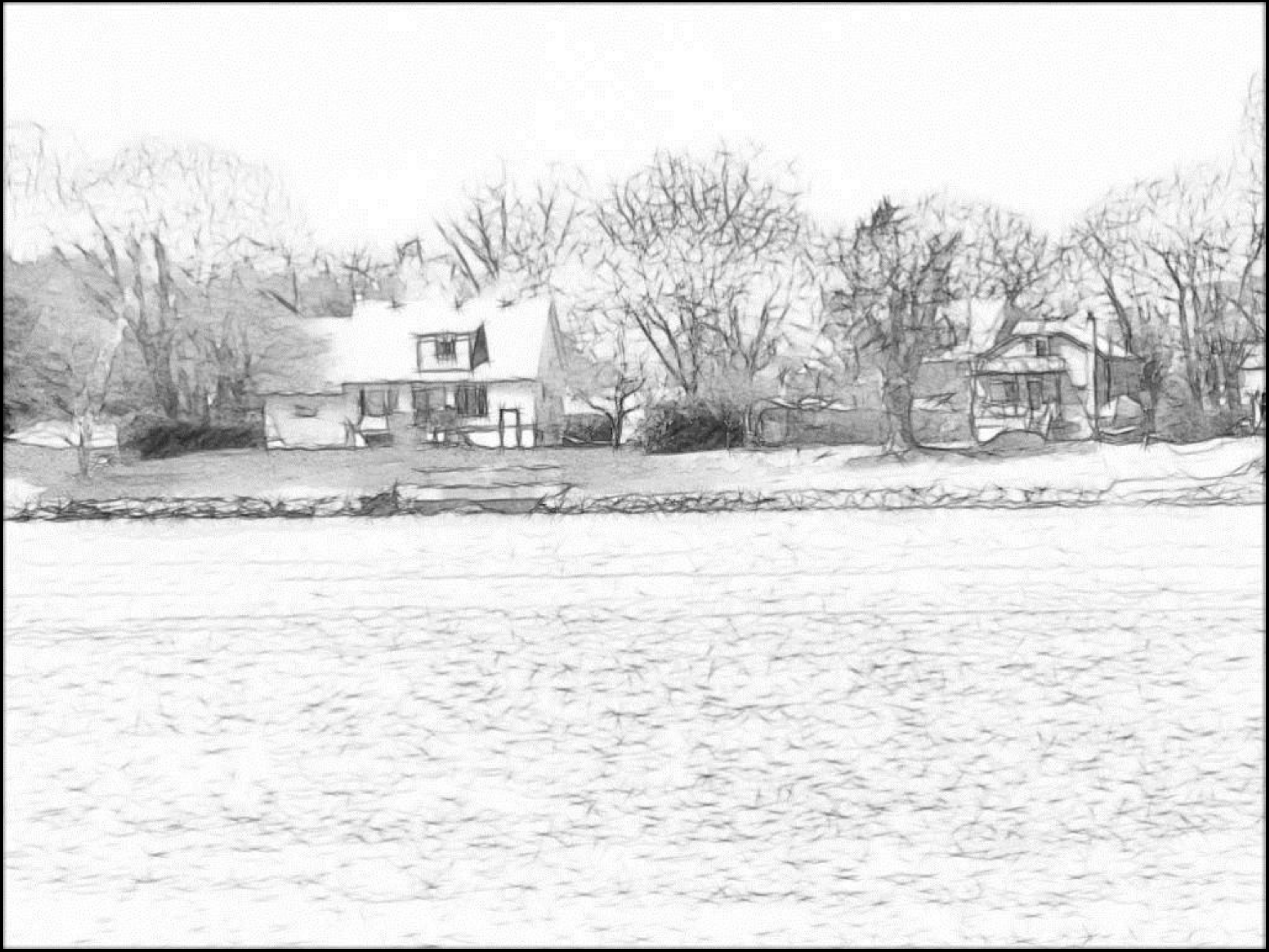


Input

Output

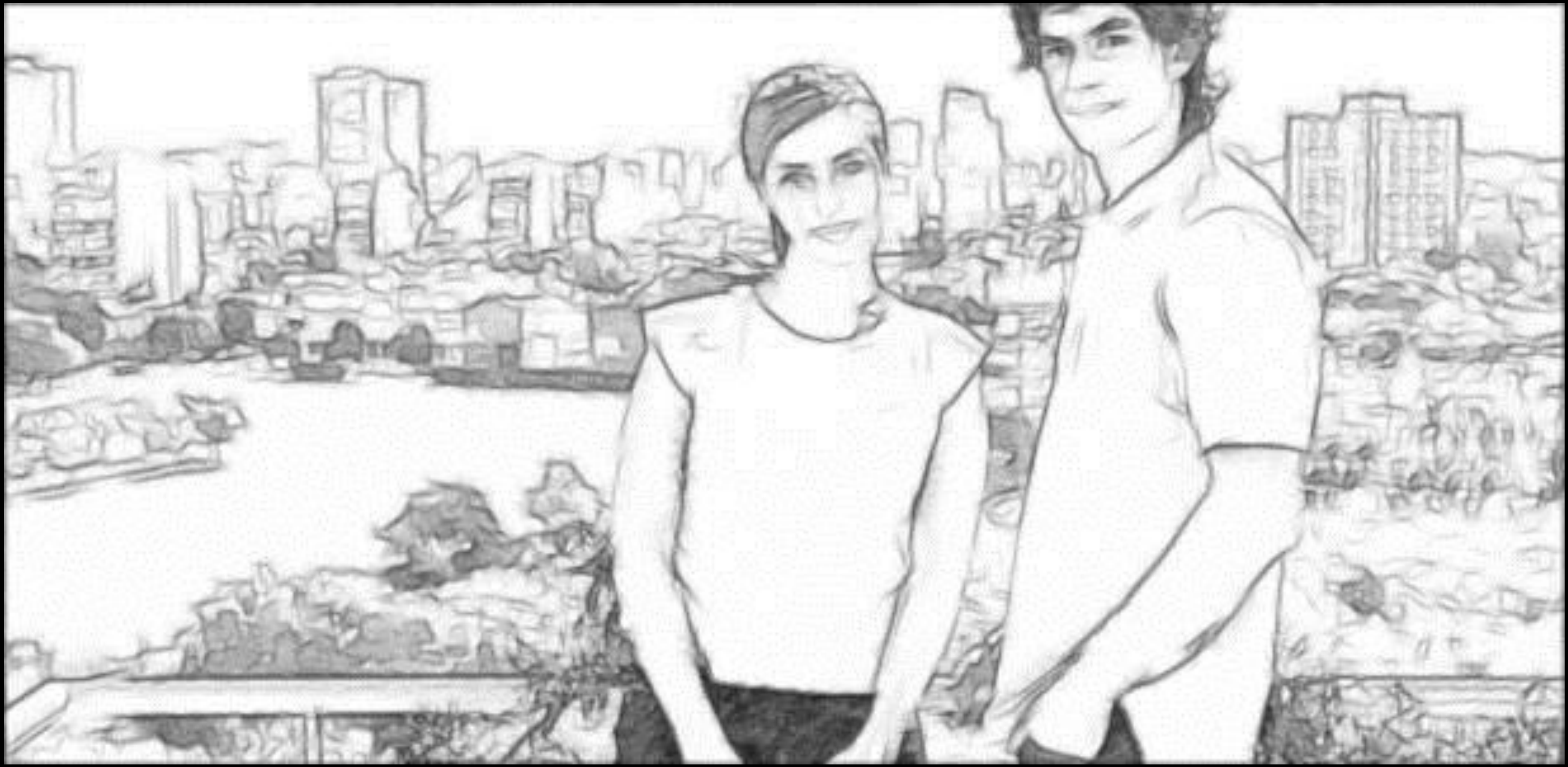


Input



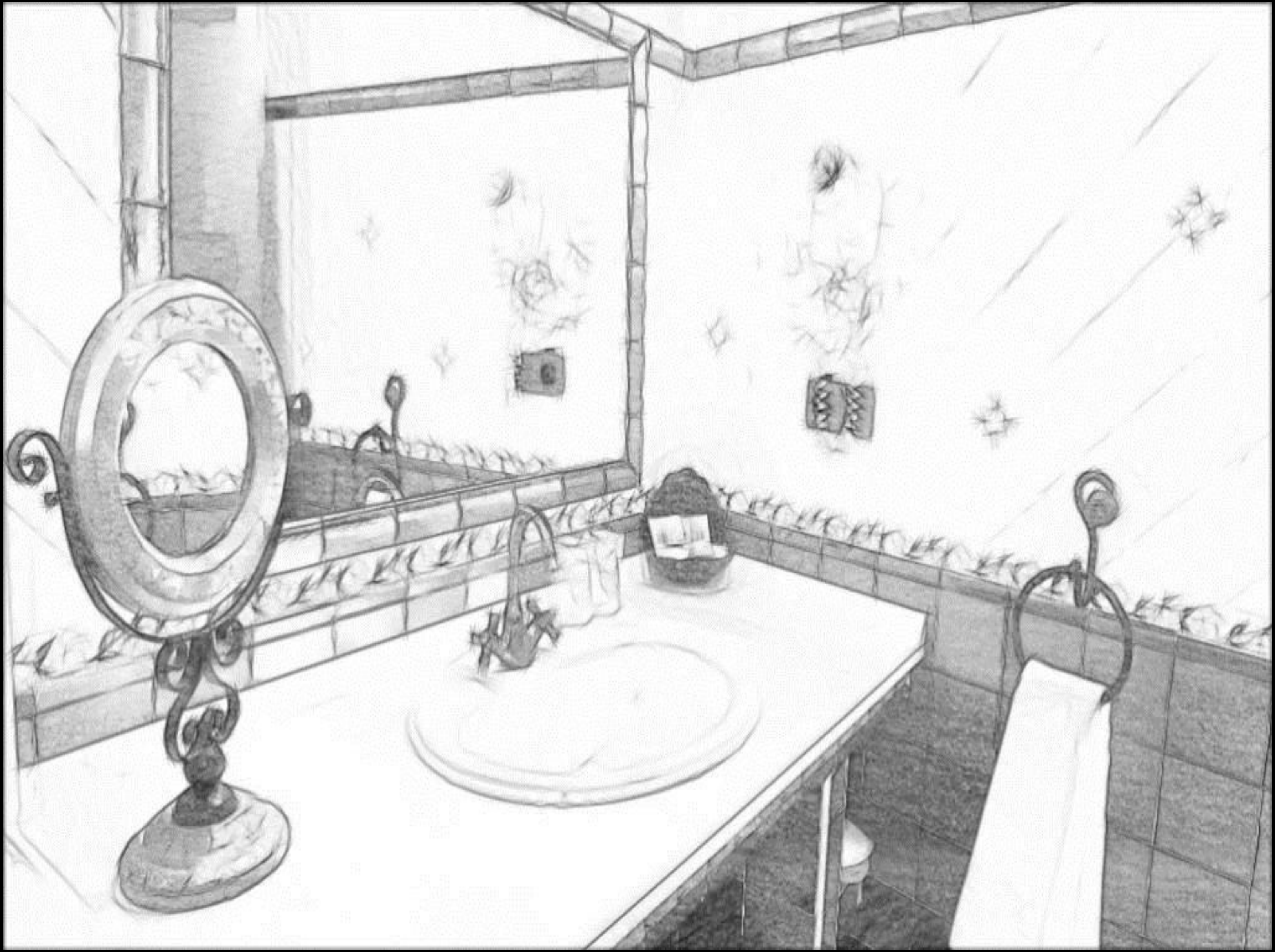
Output

Input

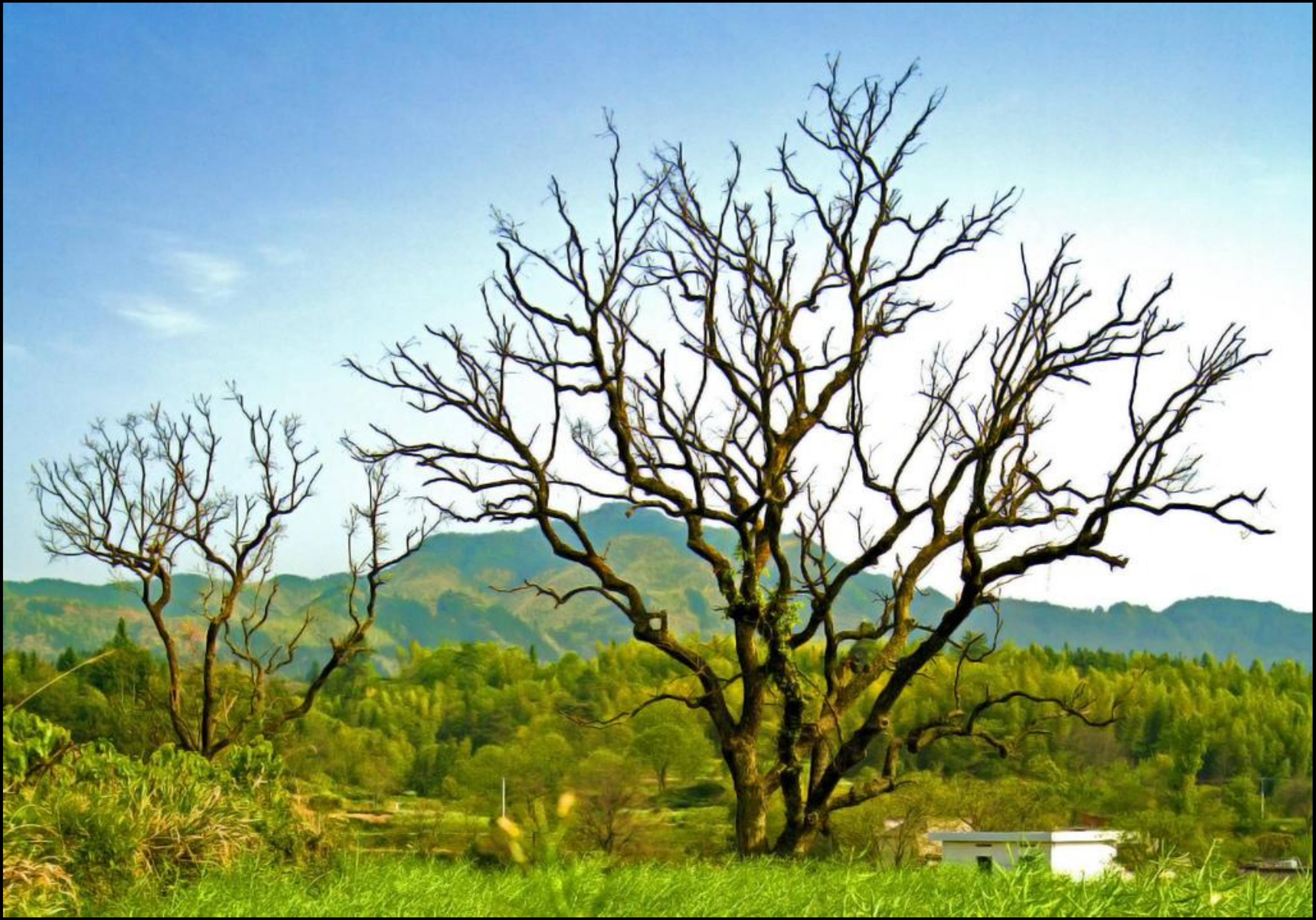


Output

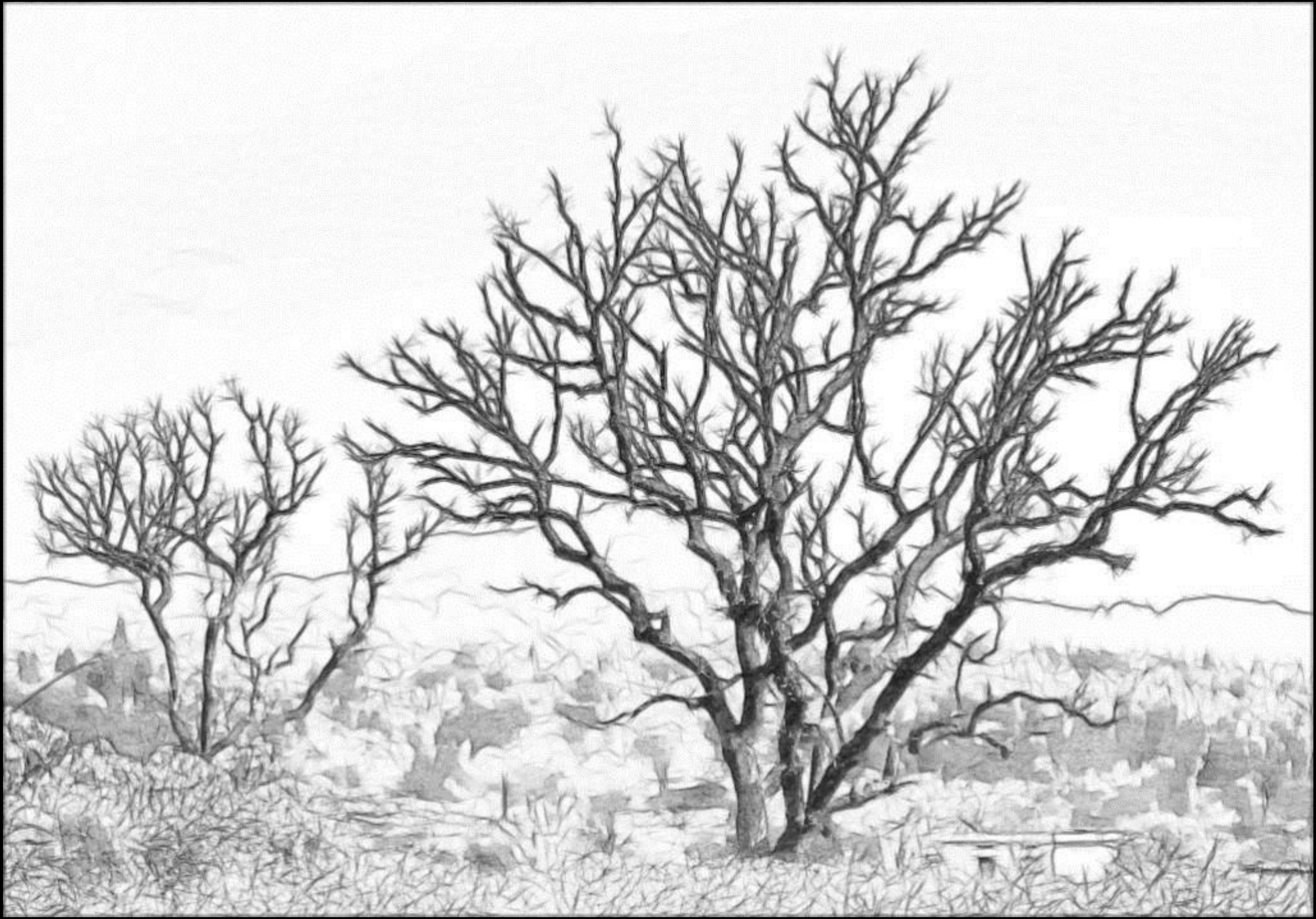
Input



Output

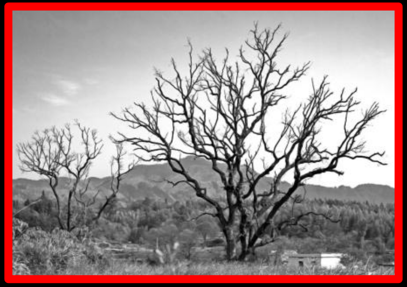


Input

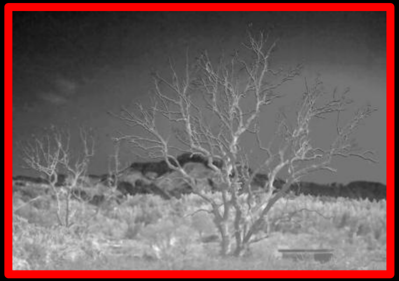


Output

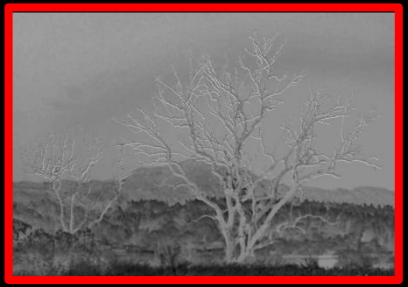
- Color Pencil Drawing



Y Channel

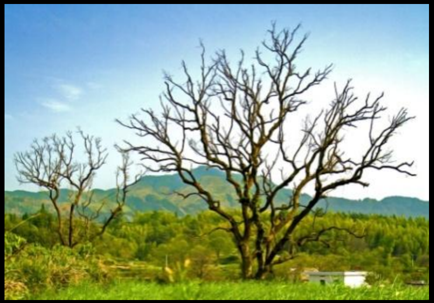


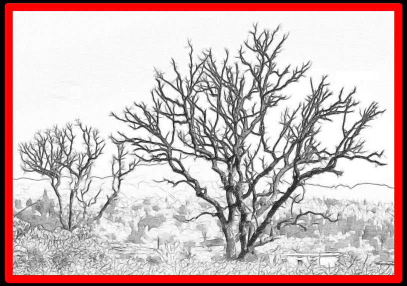
U Channel



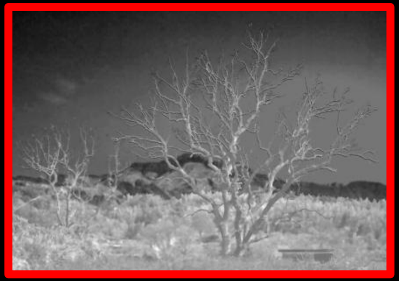
V Channel

- Color Pencil Drawing

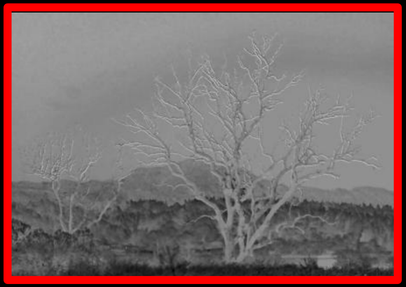




Pencil Drawing

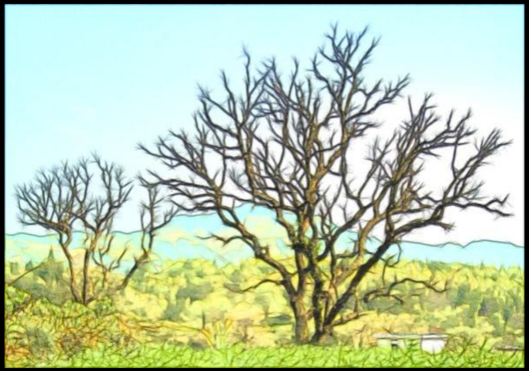


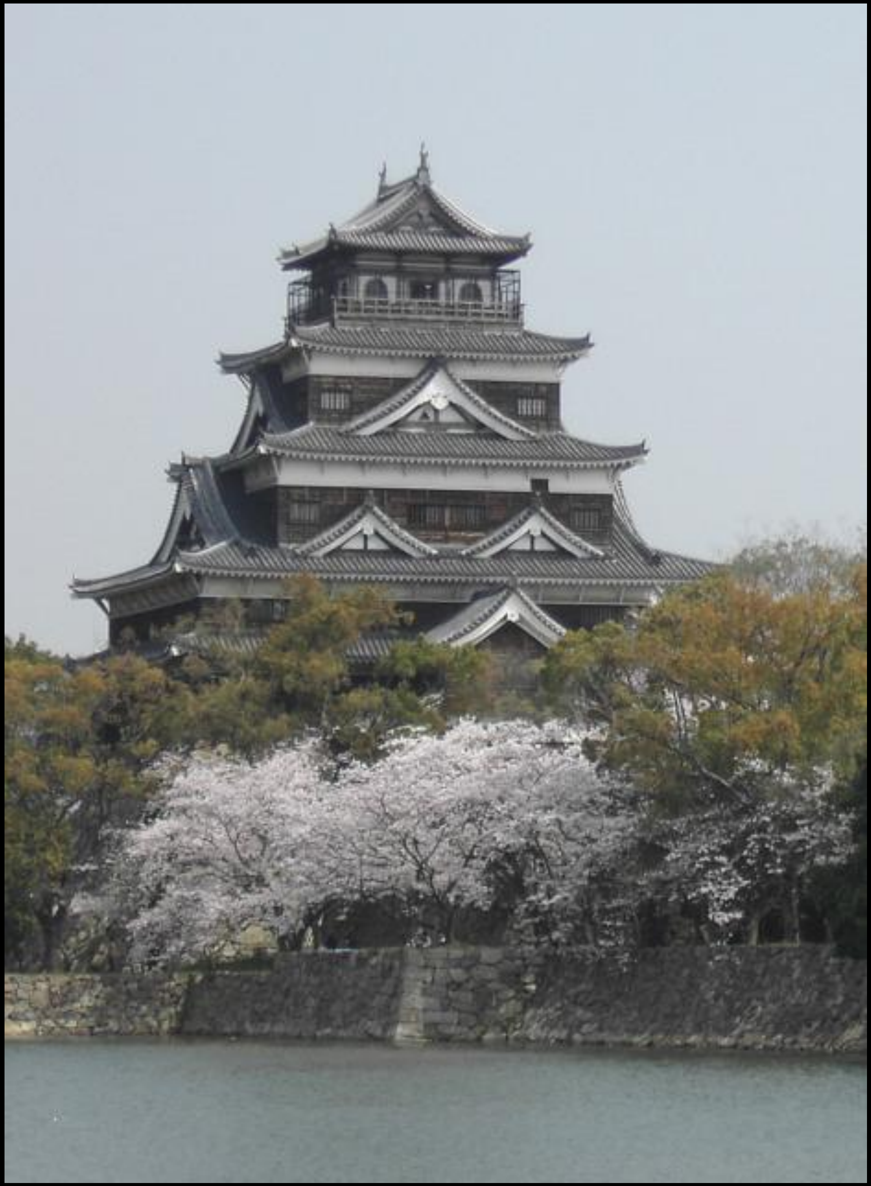
U Channel



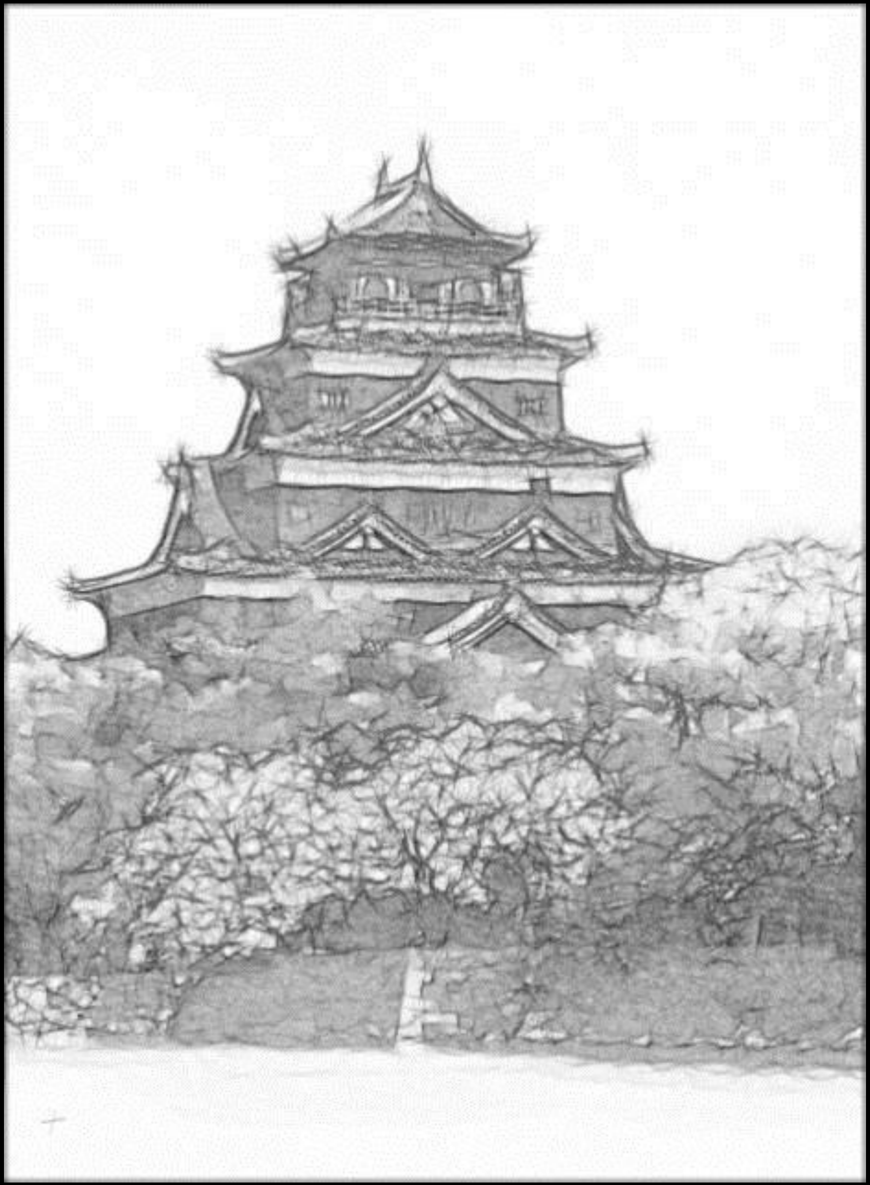
V Channel

YUV->RGB

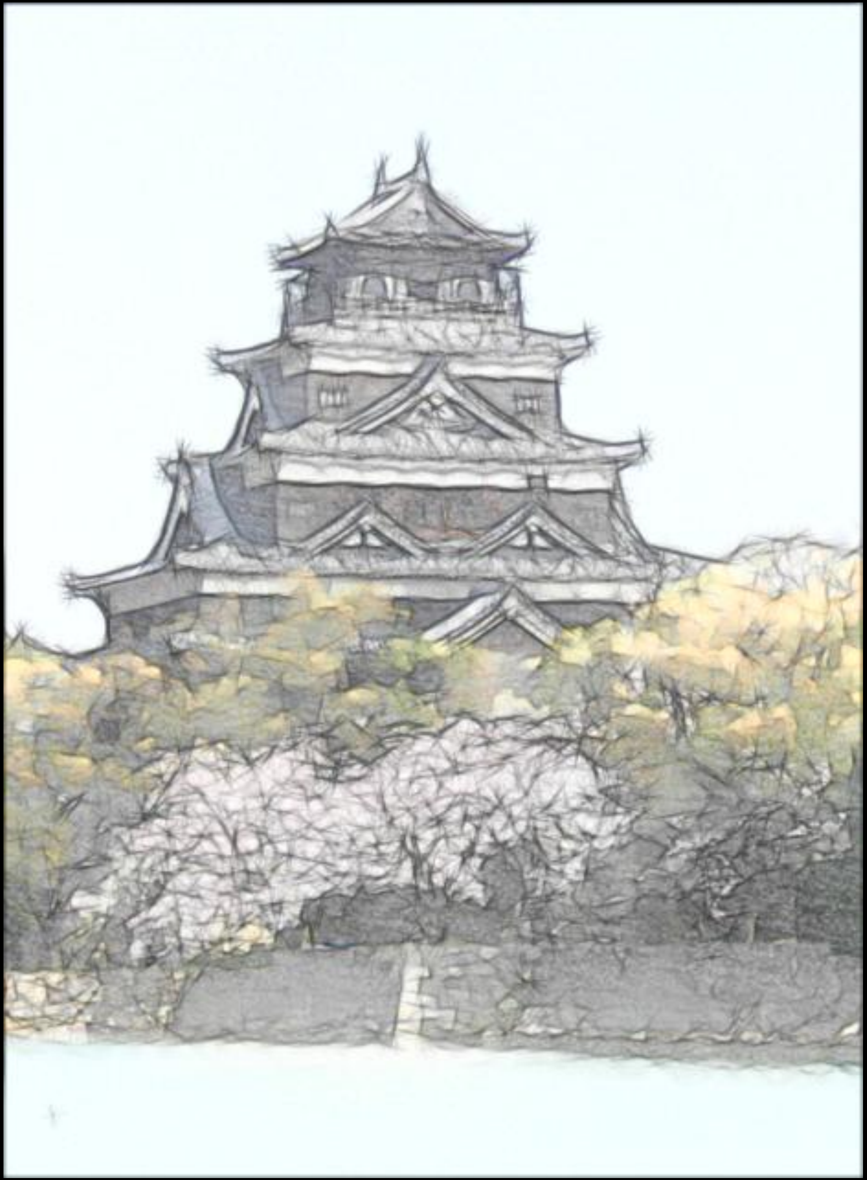




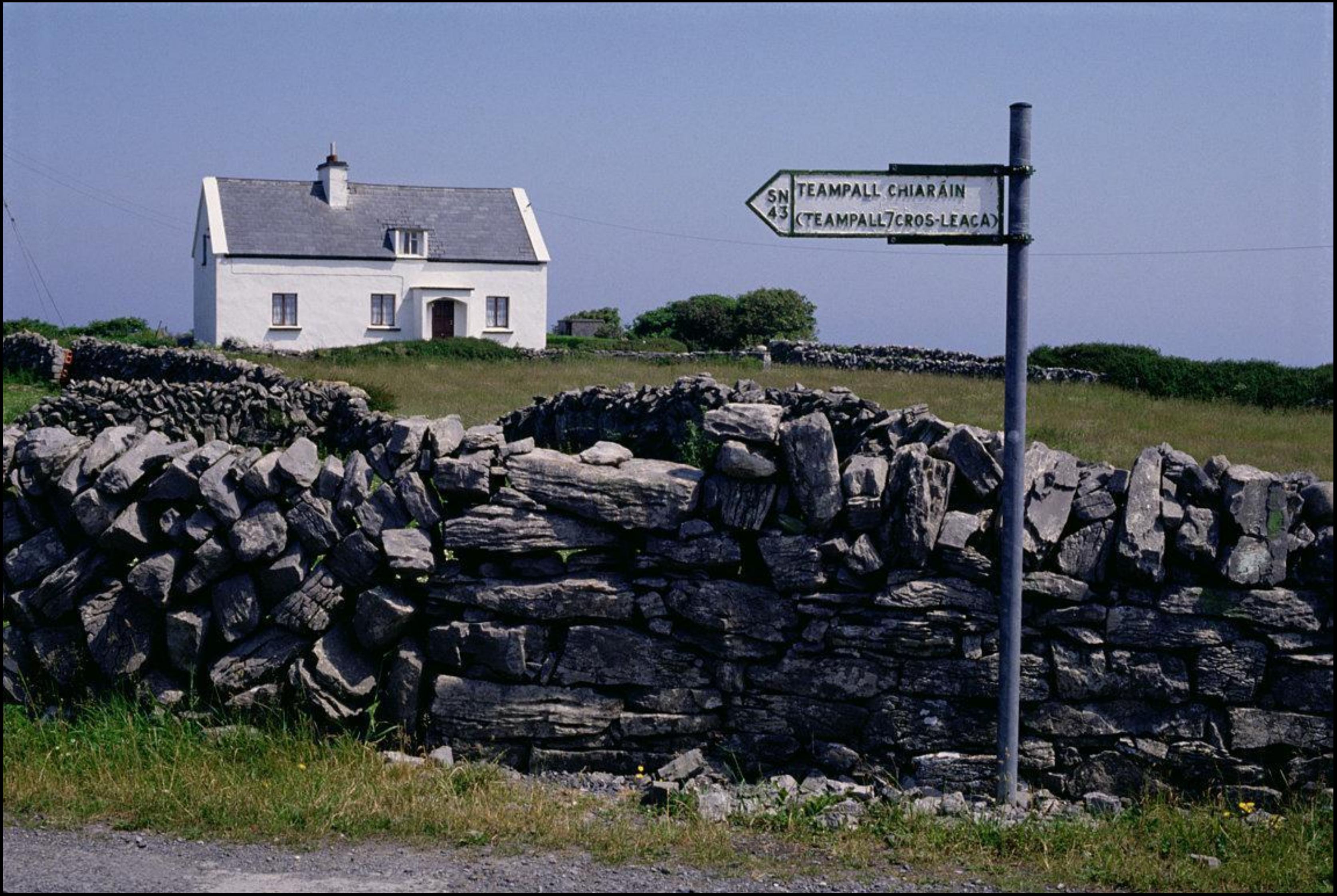
Input



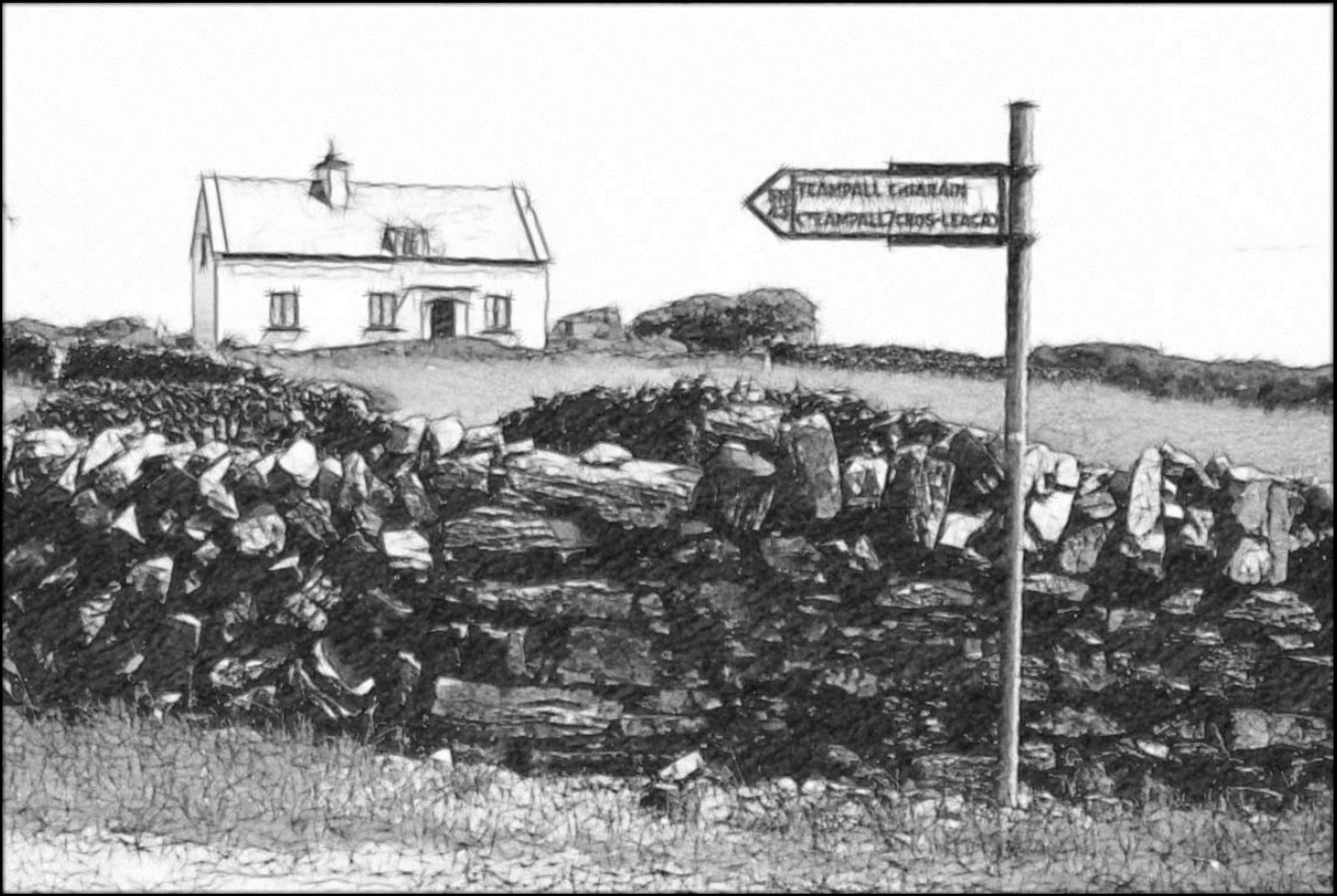
Output



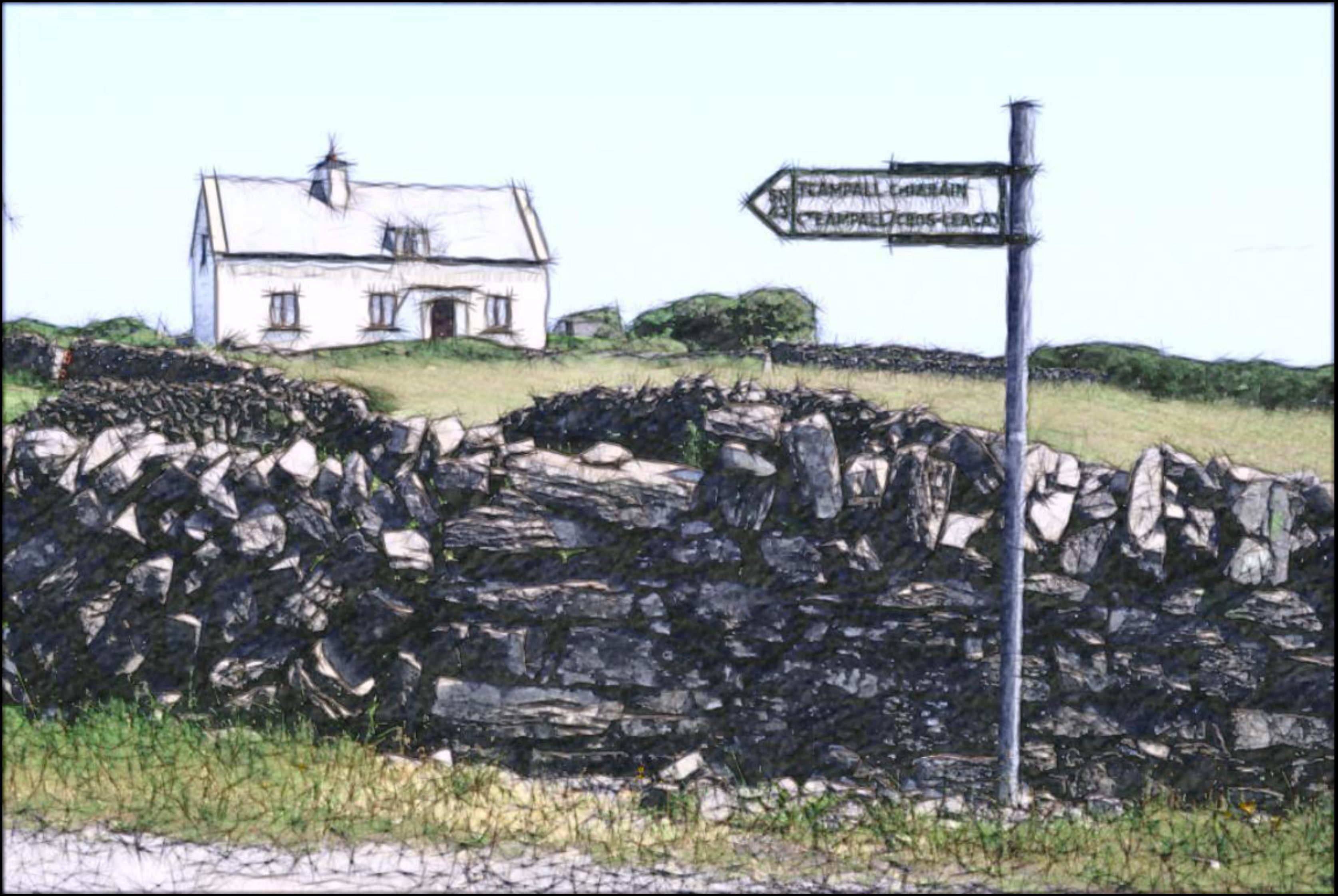
Output (color)



Input

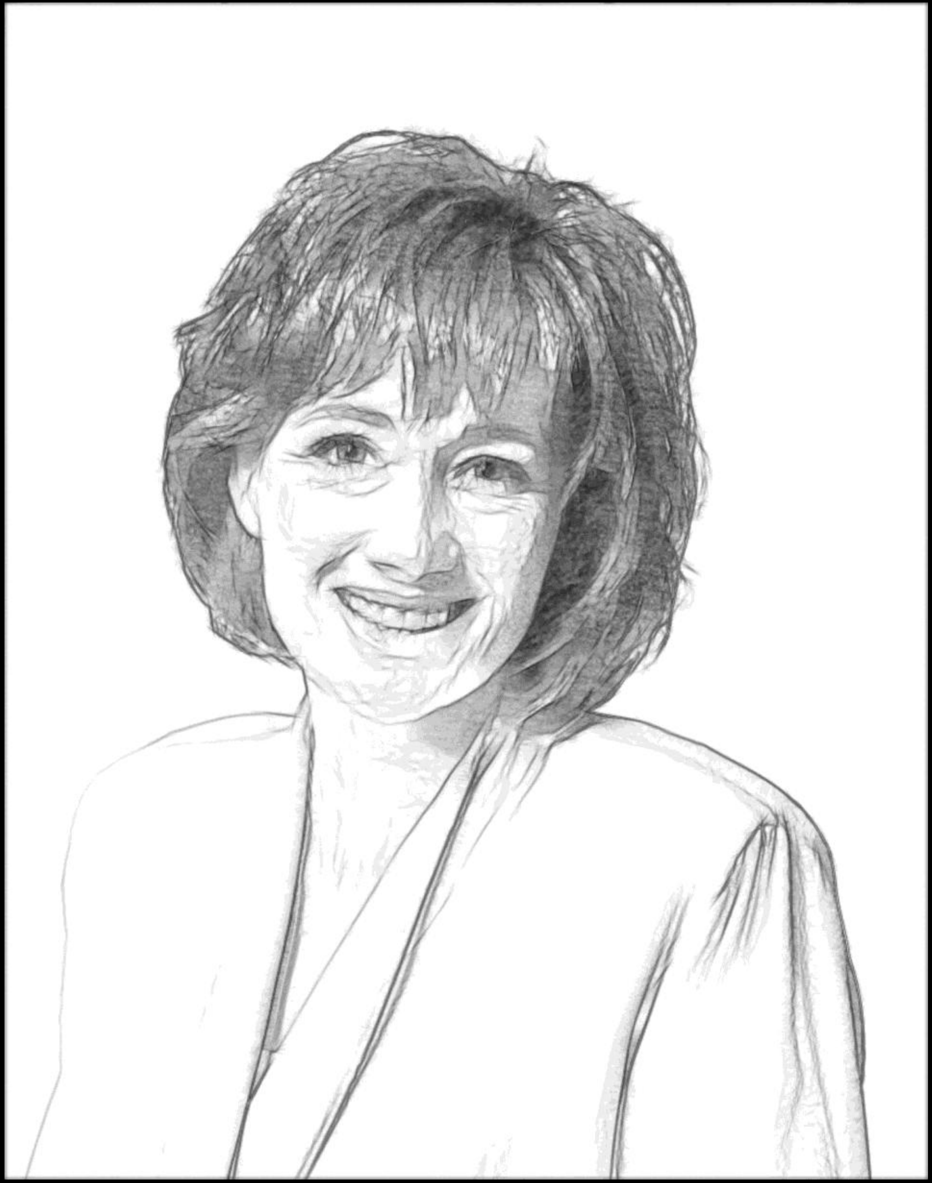


Output

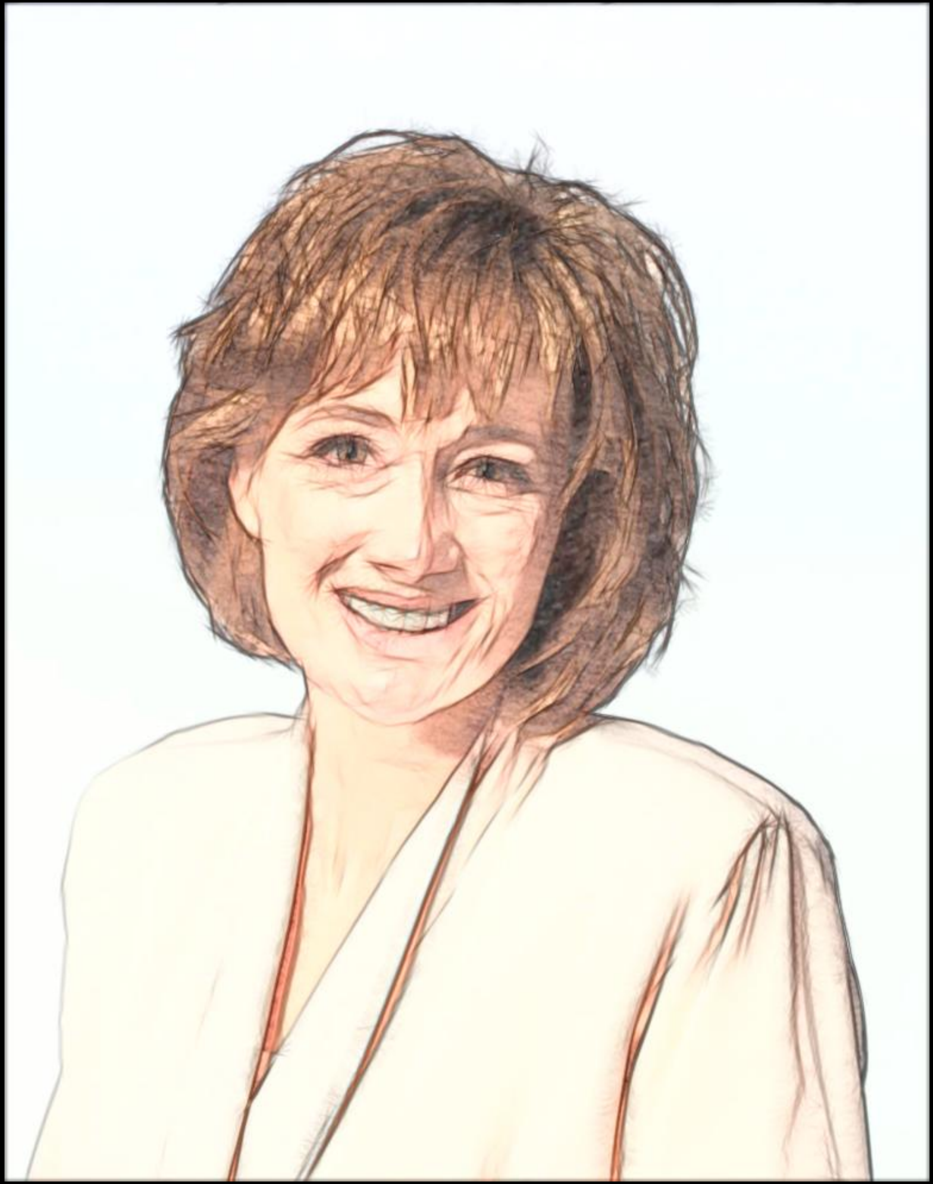


Output (color)

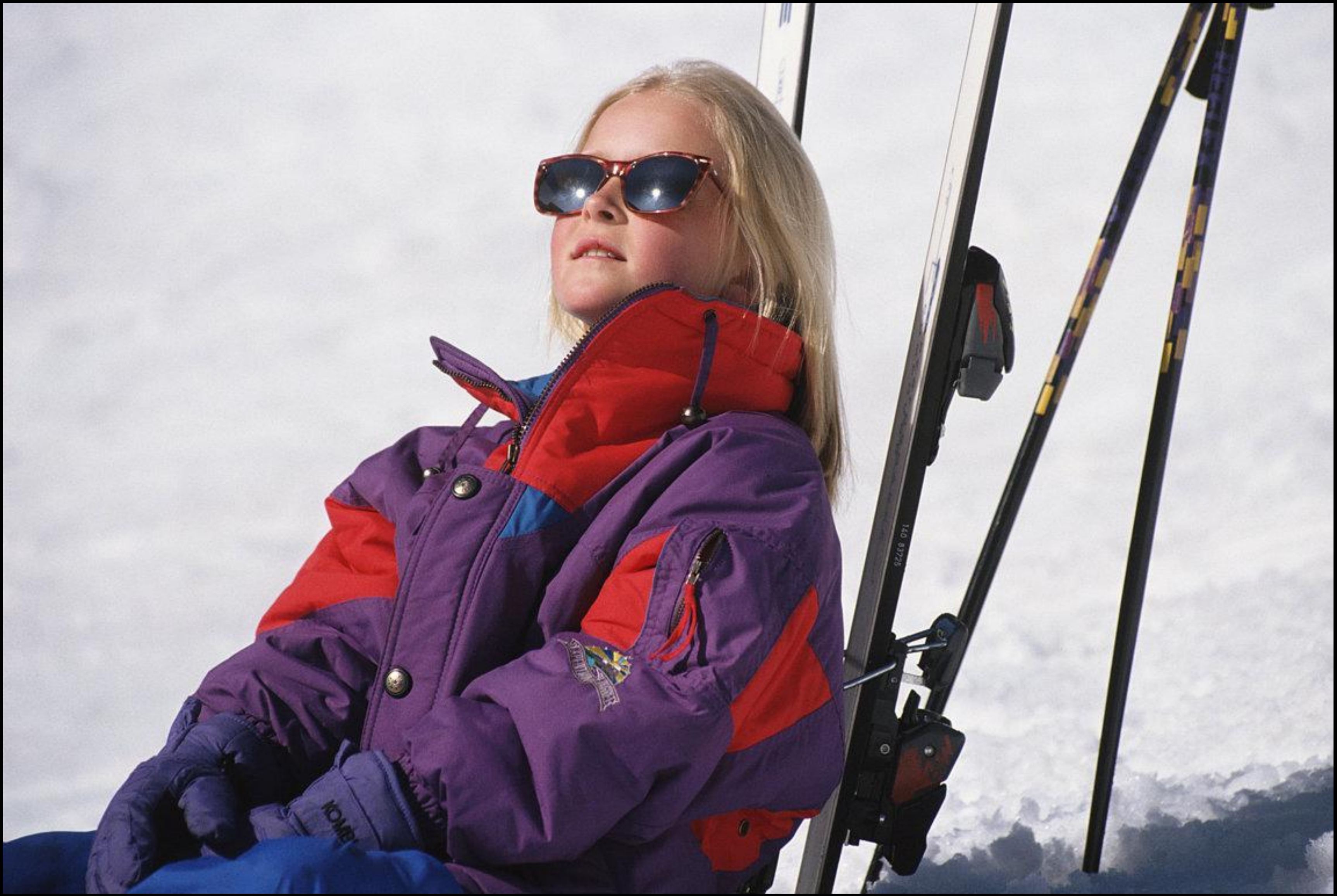
Input



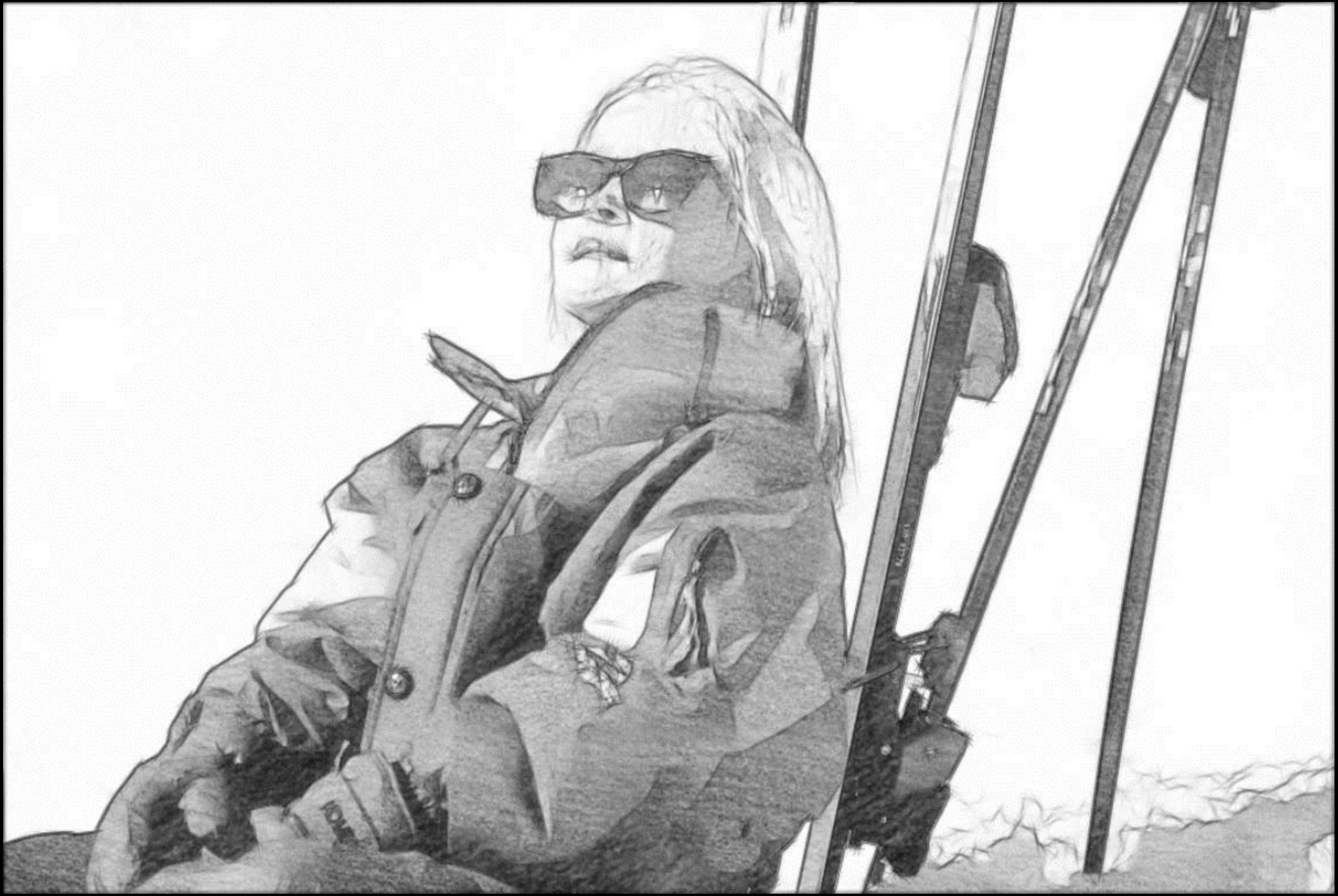
Output



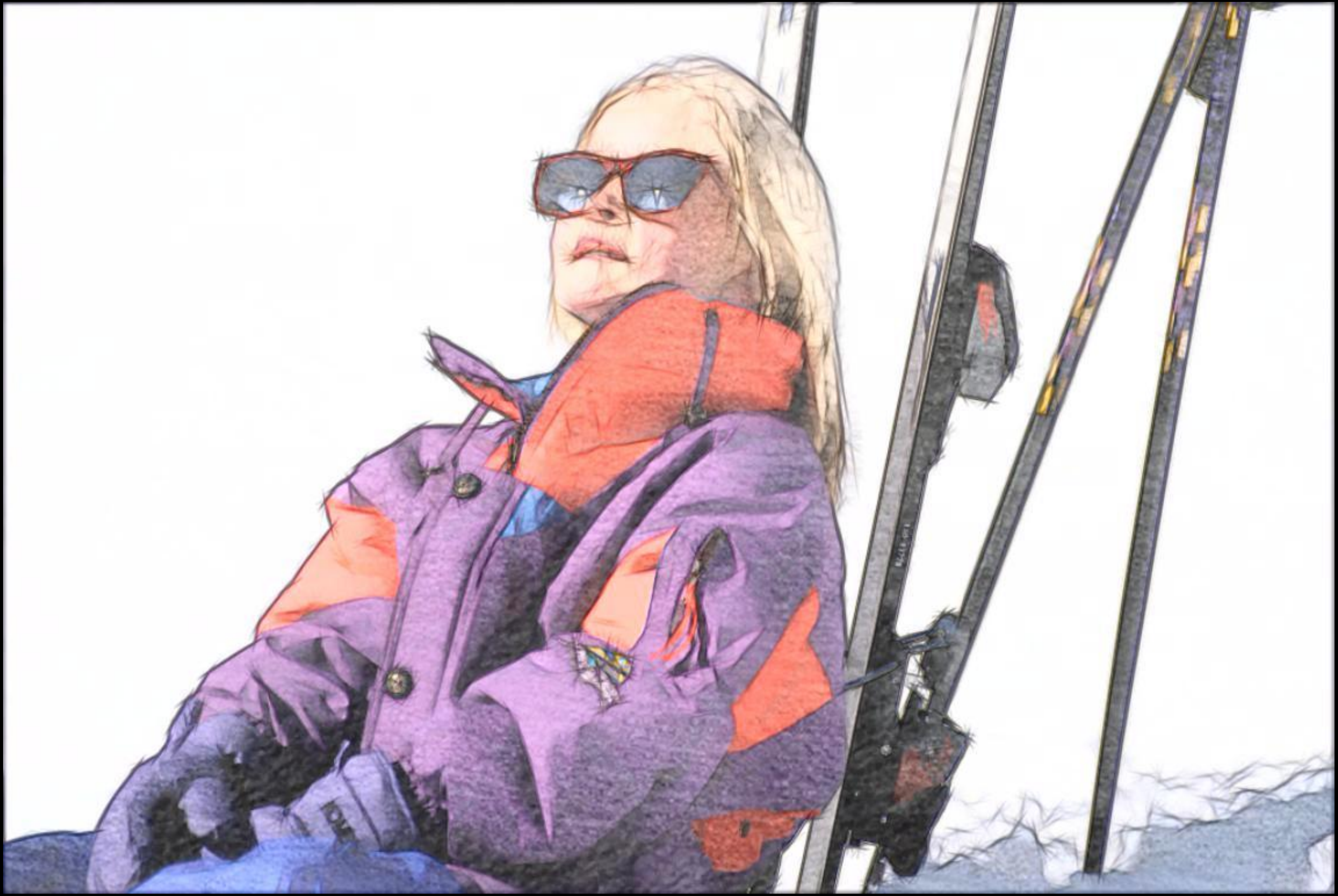
Output (color)



Input

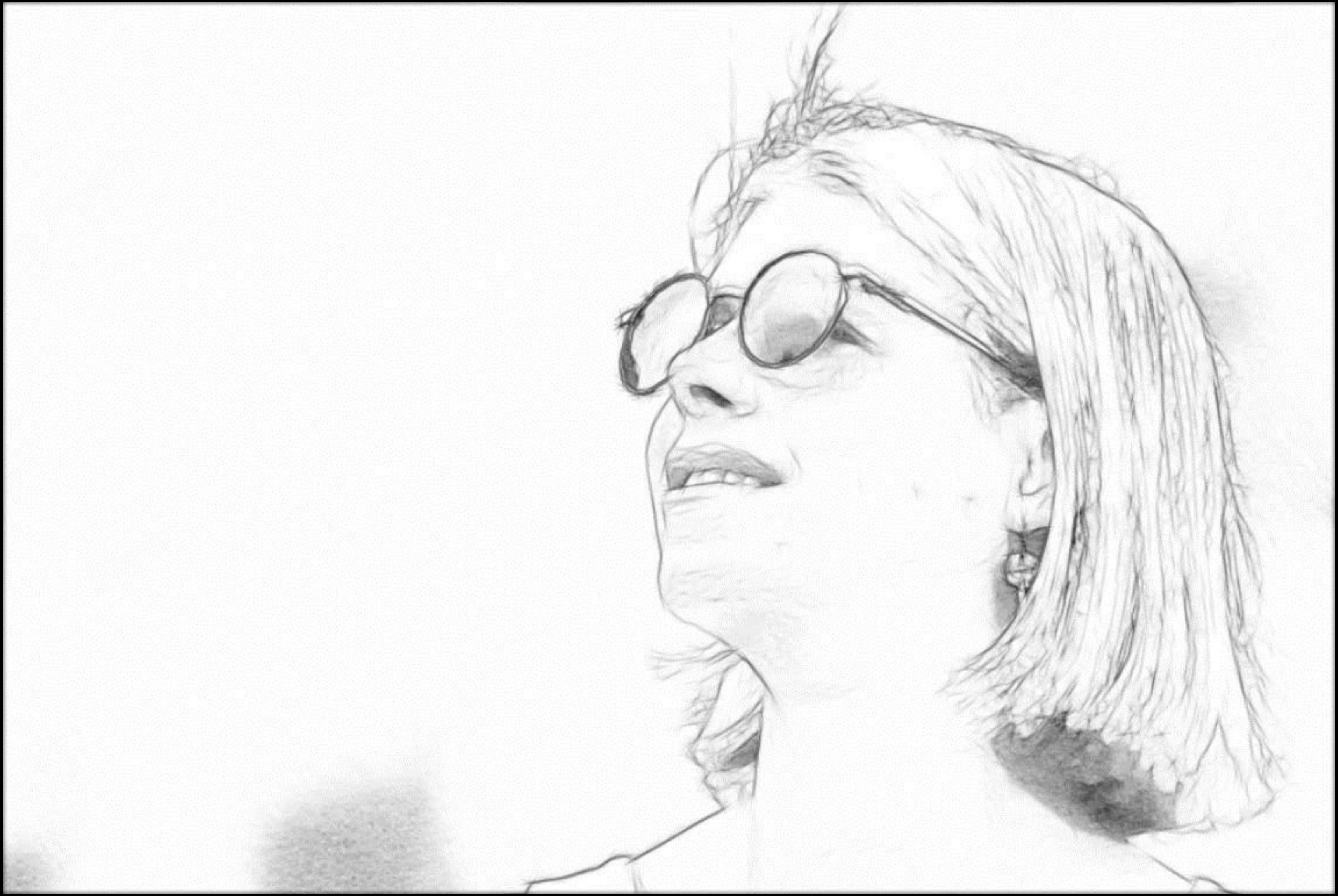


Output

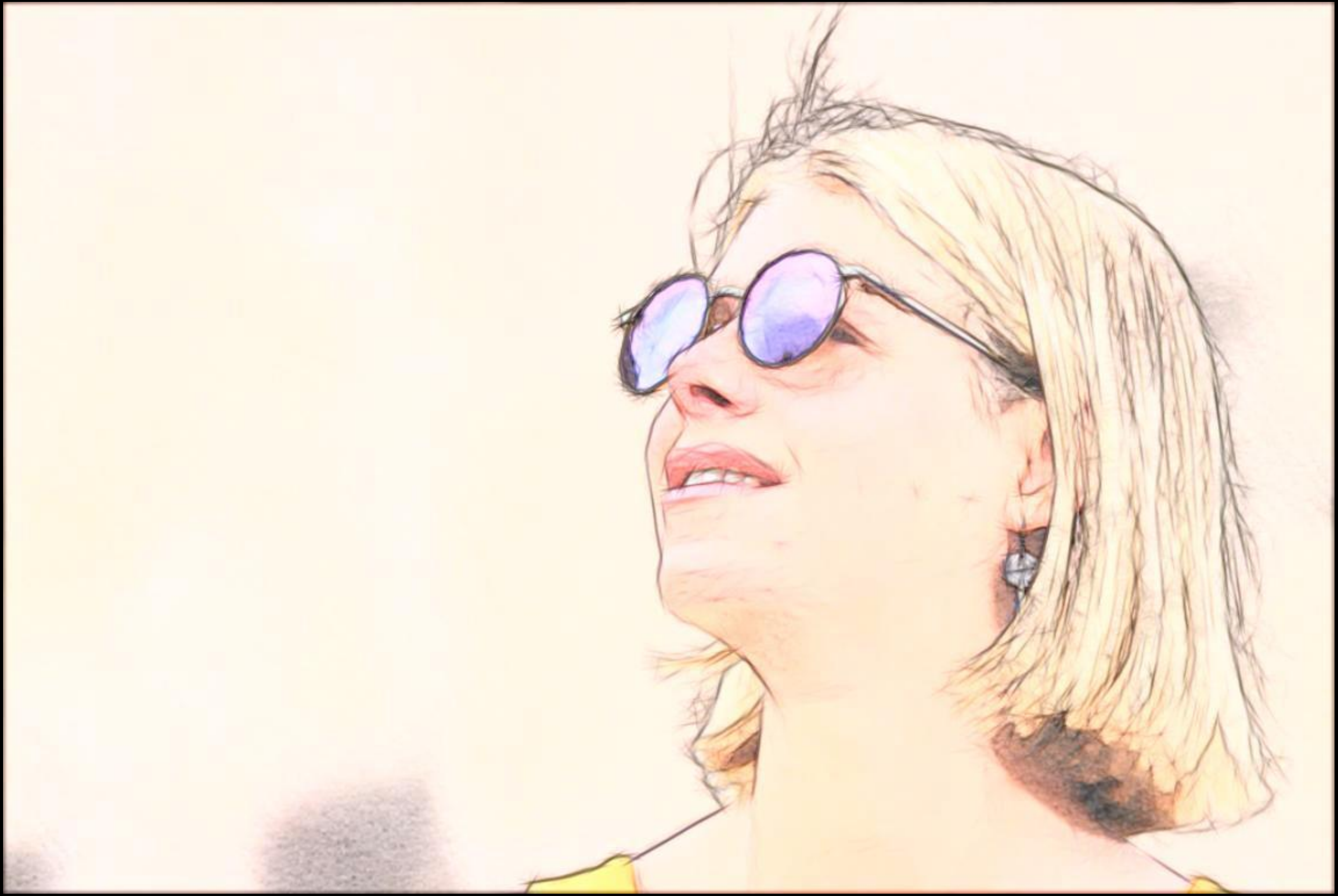


Output (color)

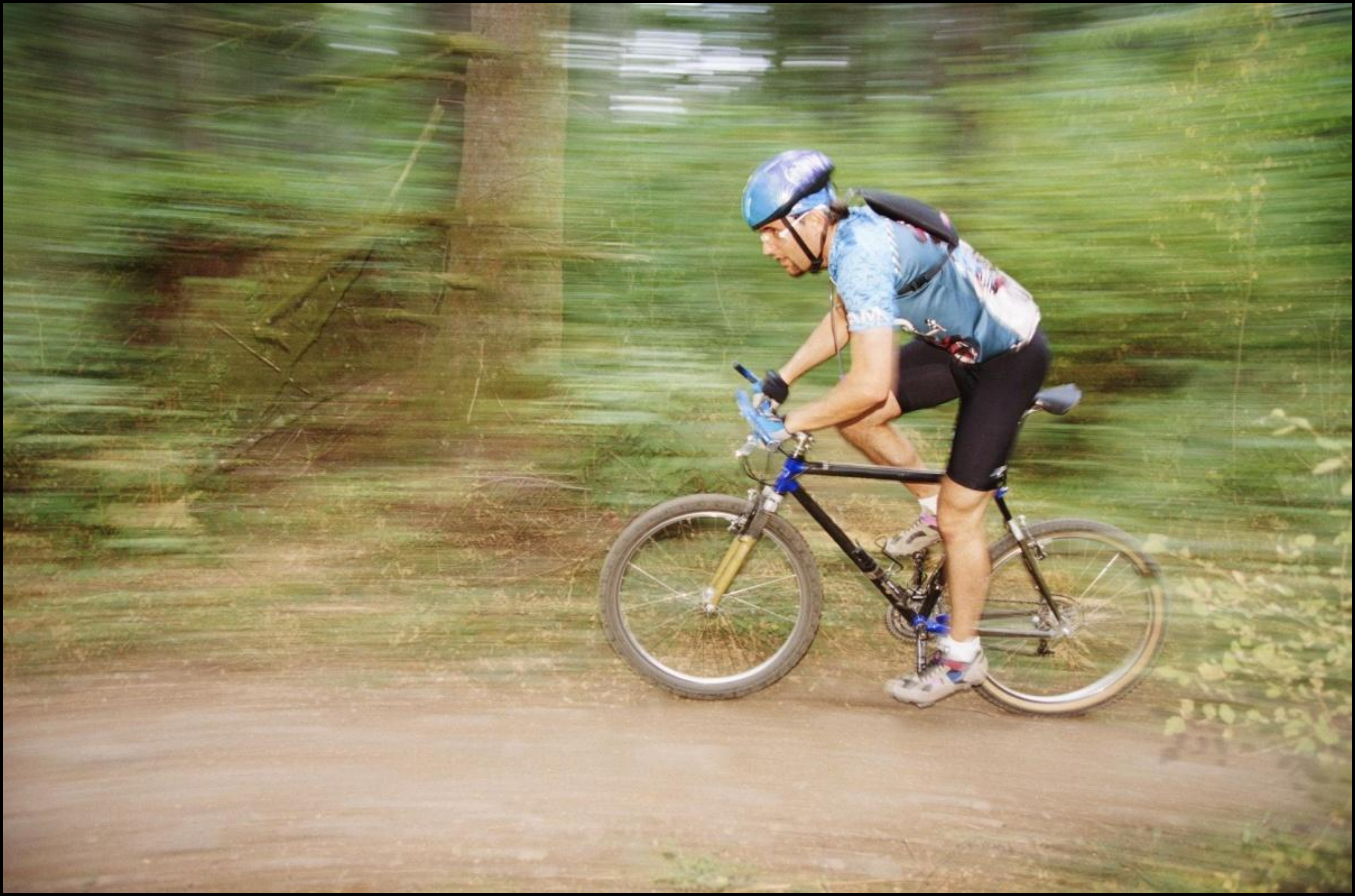
Input



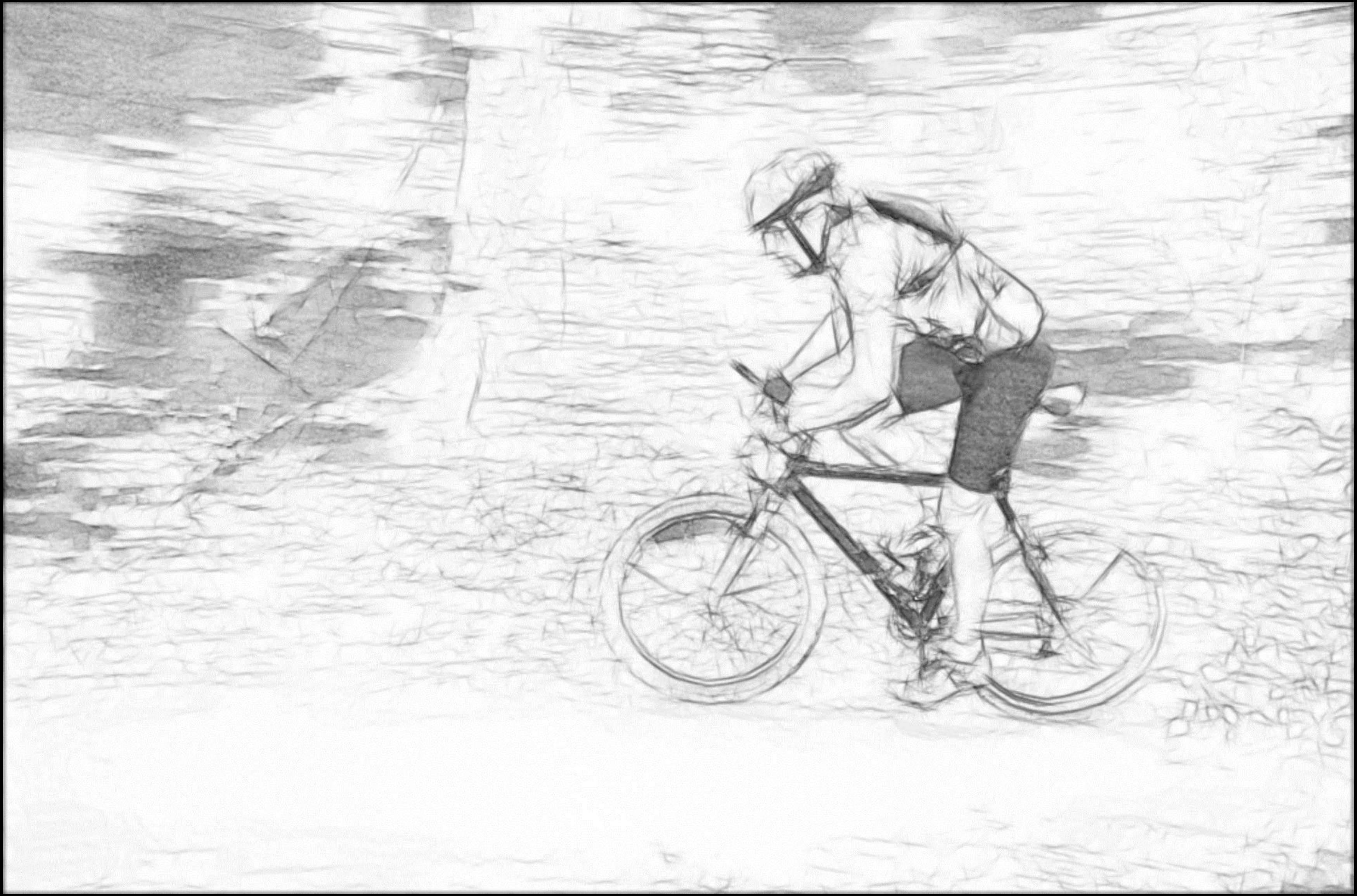
Output



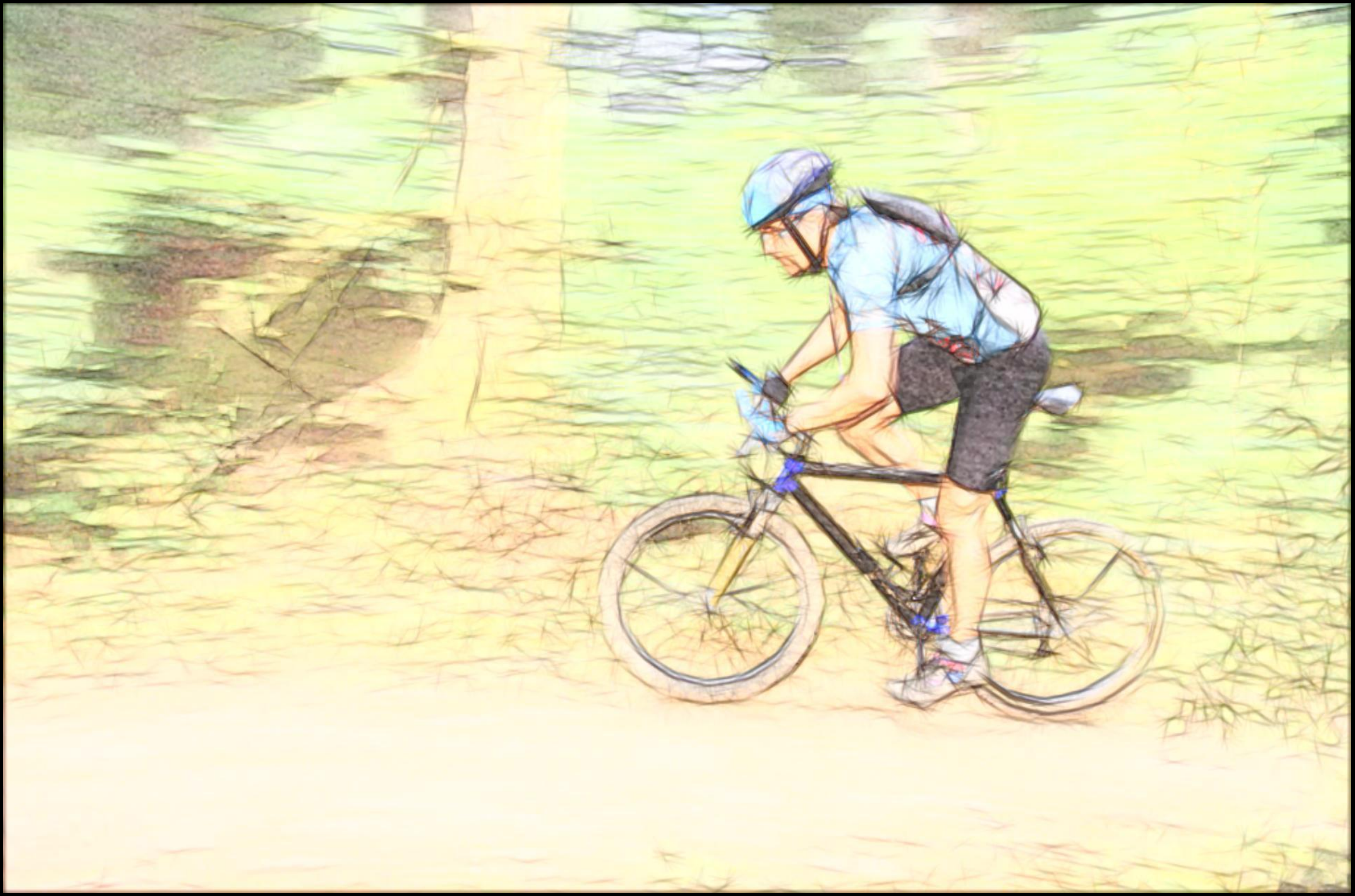
Output (color)



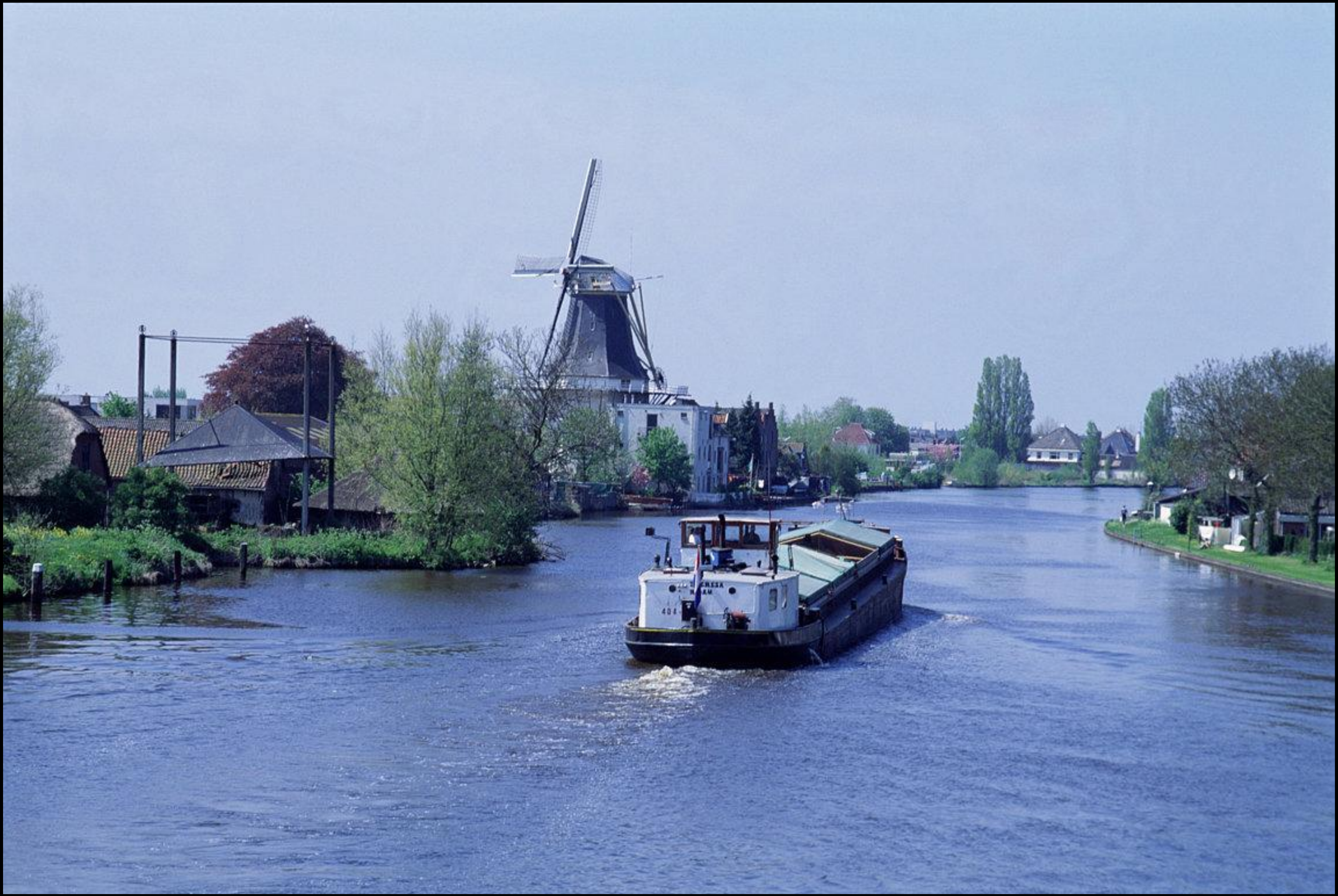
Input



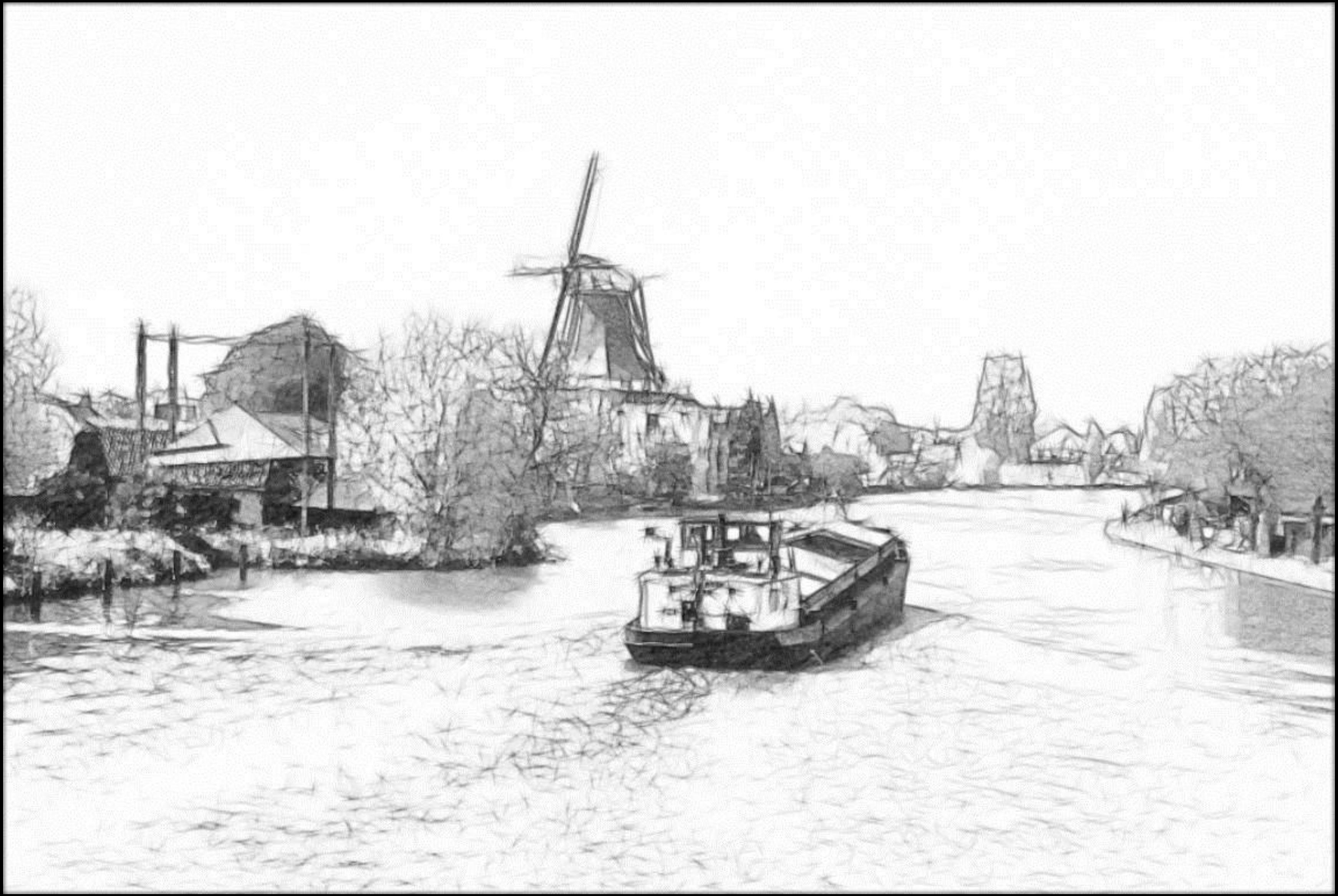
Output



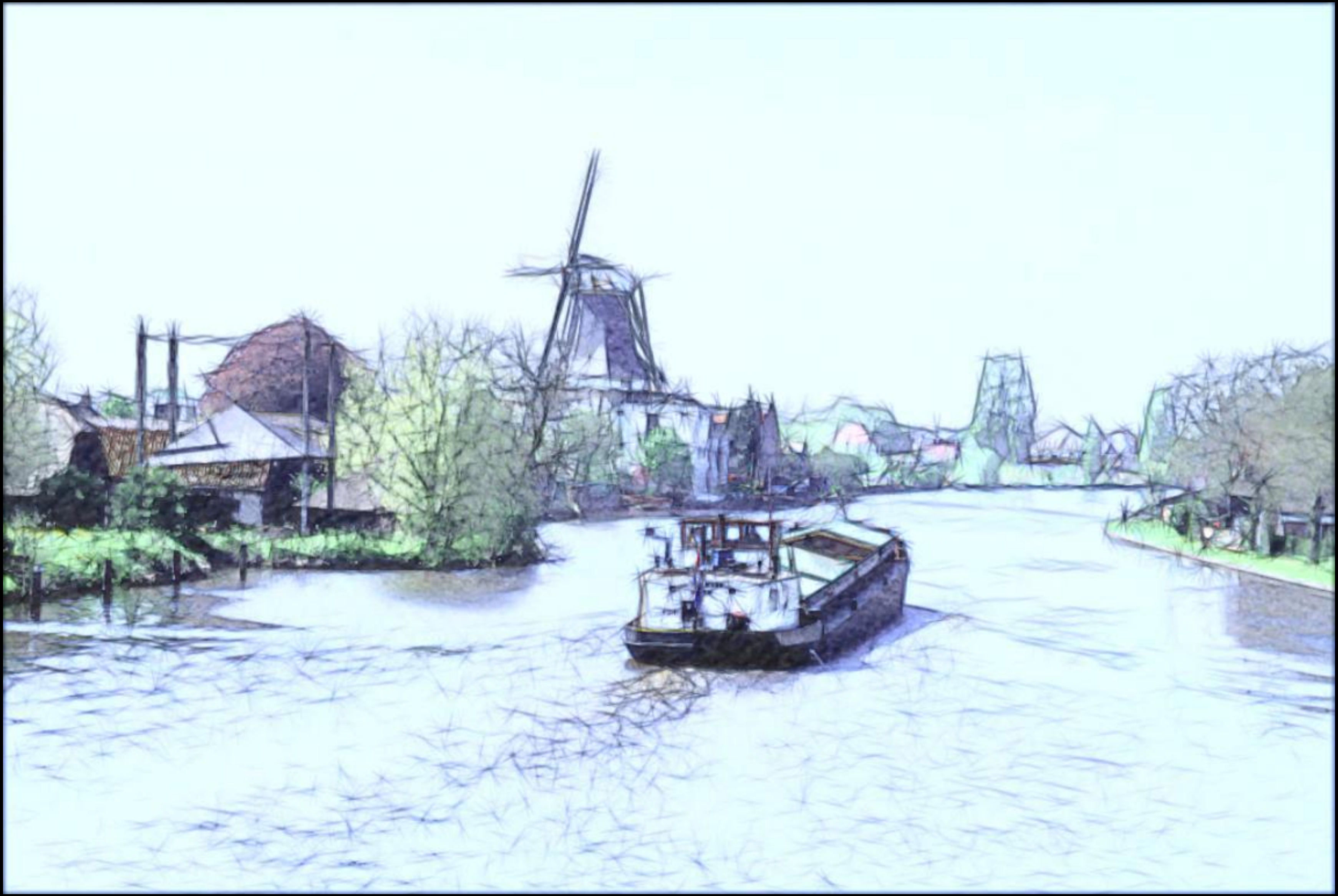
Output (color)



Input

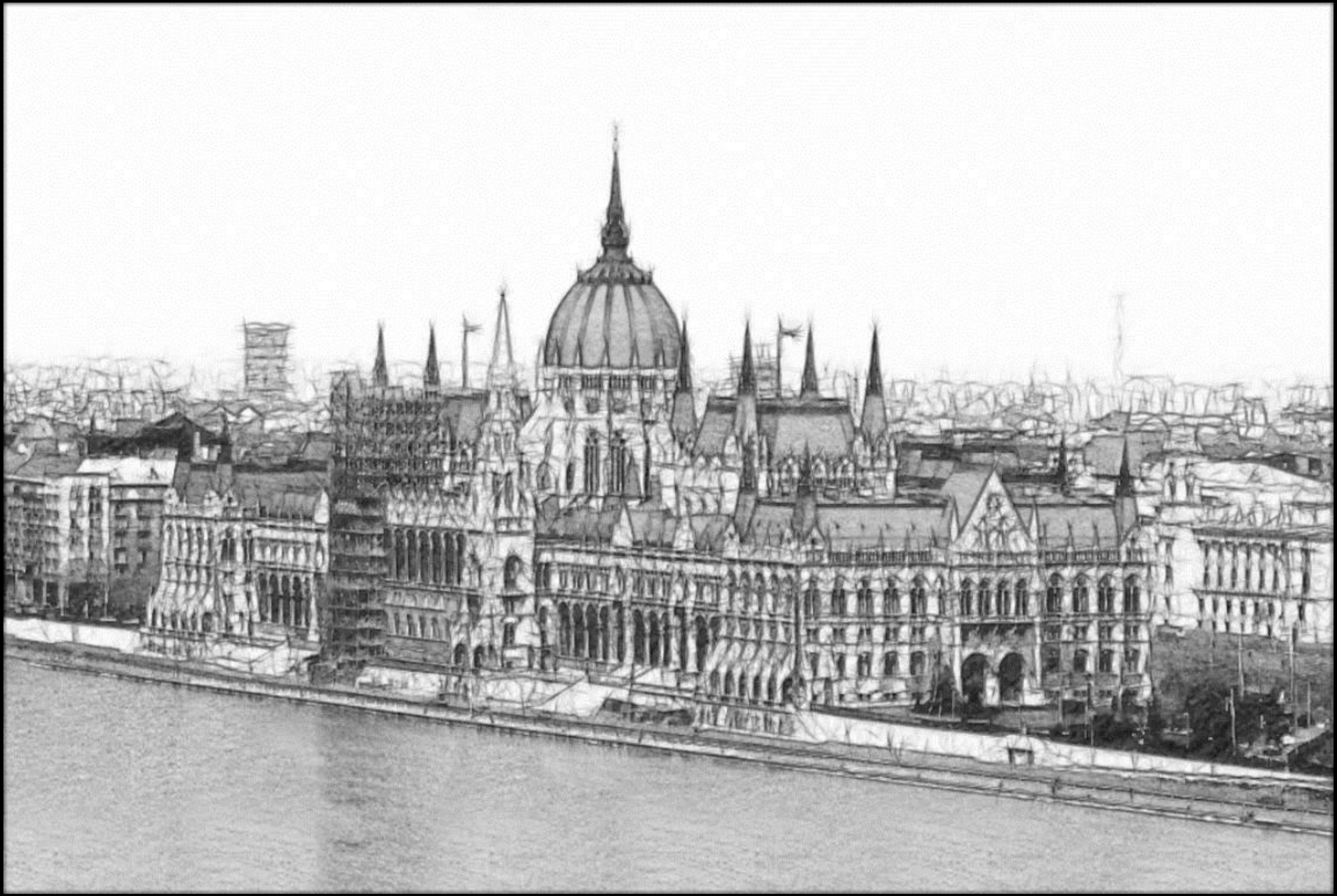


Output

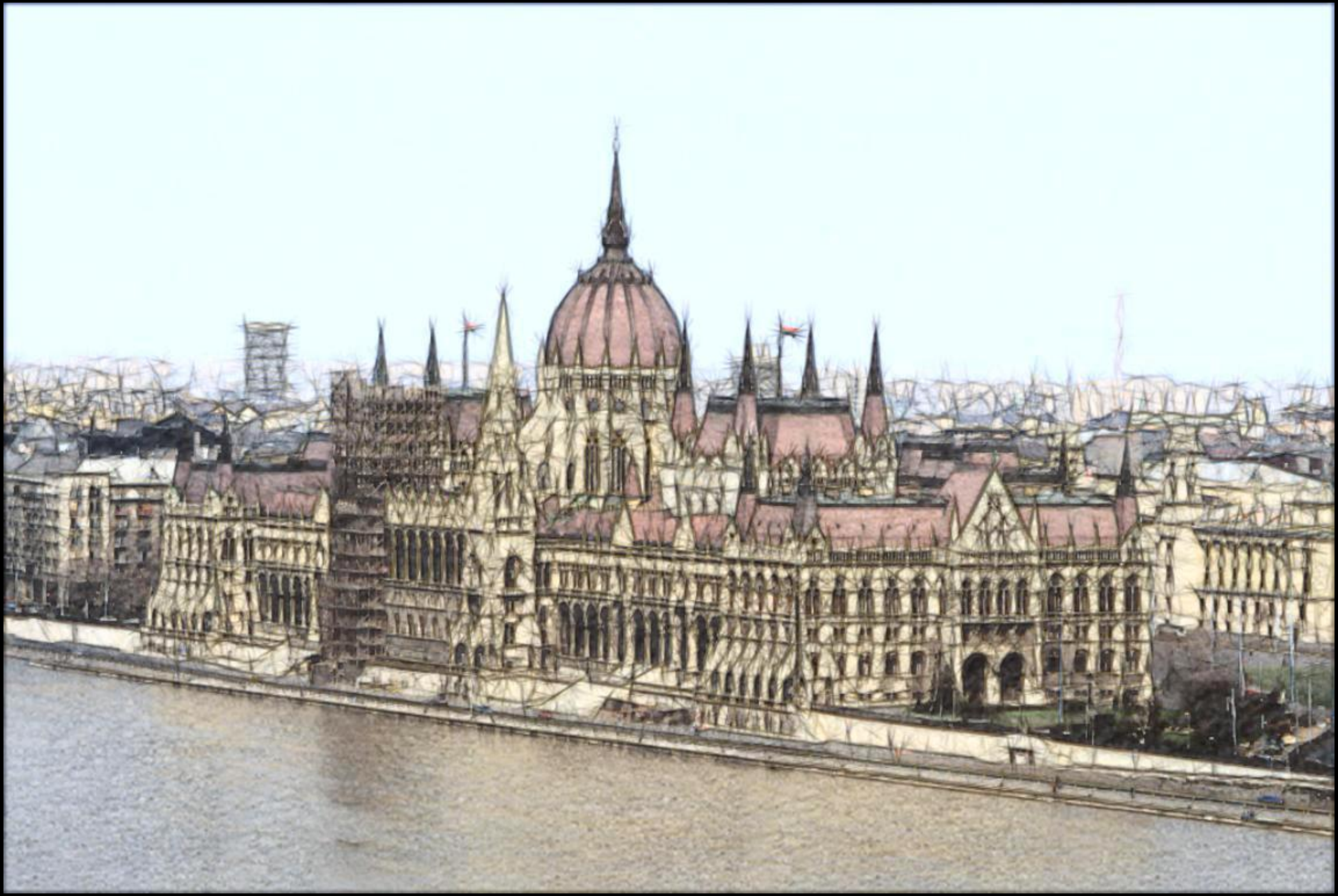


Output (color)

Input



Output



Output (color)

Conclusion

- An effective and robust image-based pencil drawing system

- Pencil drawing priors: crossing junction, pencil tone transfer, tonal pencil texture rending

Thank you

Annecy, France