



## Being an Art Restorer

A. Kleefeld

L. Bundrock, Y. Cao, T. Dombrovskij, T. Emunds, A. Kristof,  
S. Tafazoli

Aachen, June 15th 2019



RWTH AACHEN  
UNIVERSITY

# Background

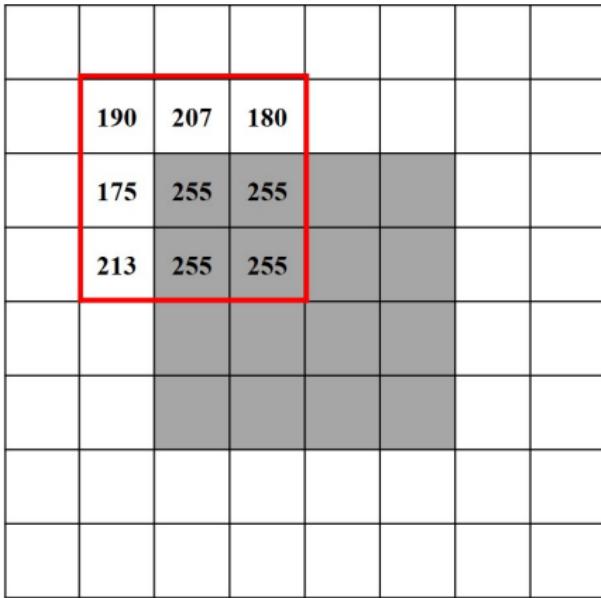
Professional art restorer's MO:

- Using global image data to fill the missing parts
- Continuing structure of missing parts by neighboring structures, e.g. contour lines
- Matching colors of missing parts
- Adding texture

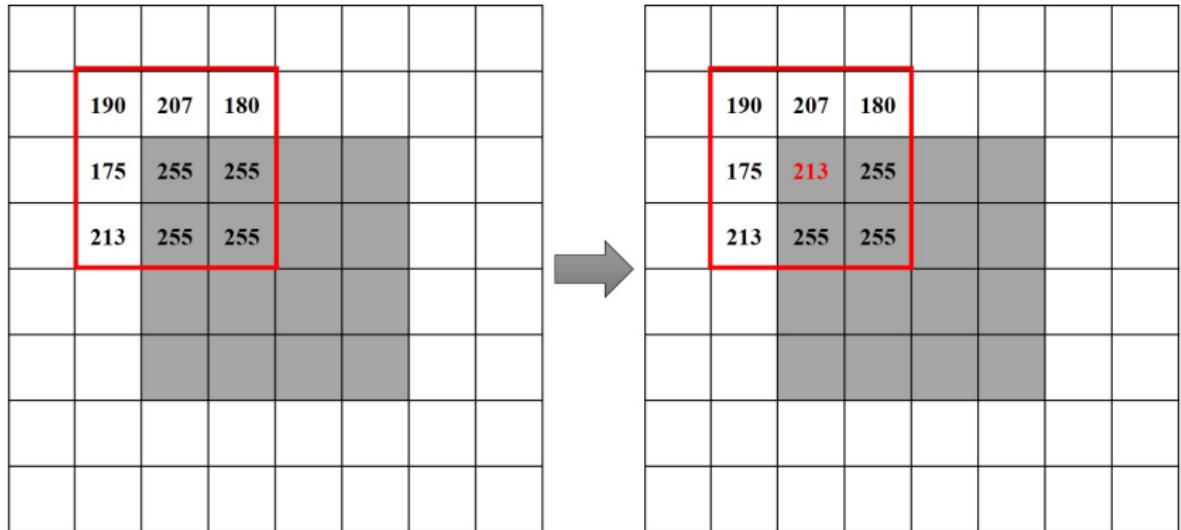


# Median Filter

- The gray region is corrupted. The red  $3 \times 3$  window is filter

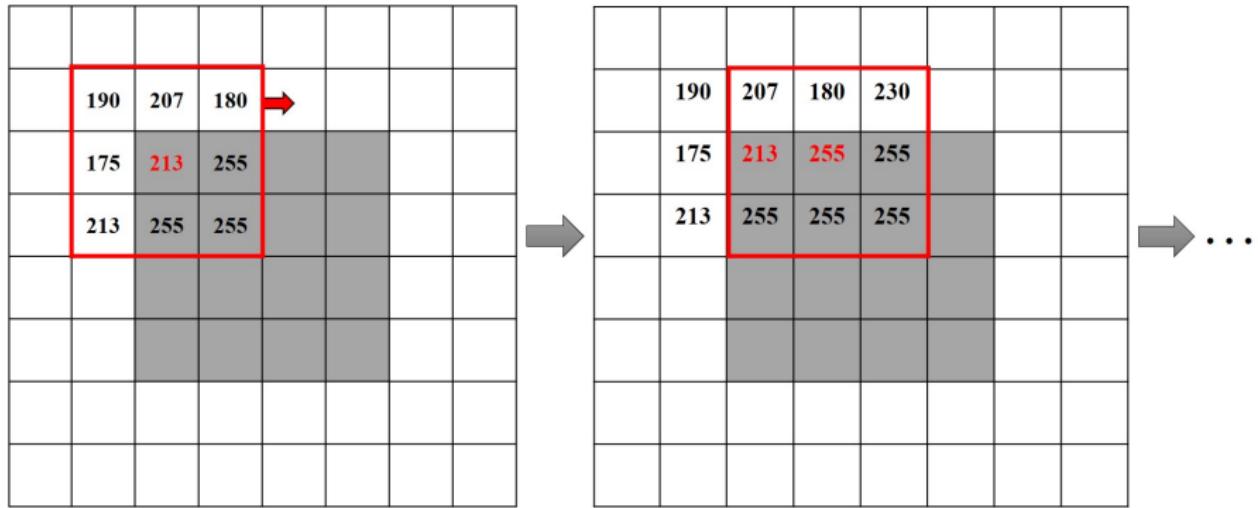


# Median Filter



$$\text{median}\{175, 180, 190, 207, 213, 255, 255, 255, 255\} = 213$$

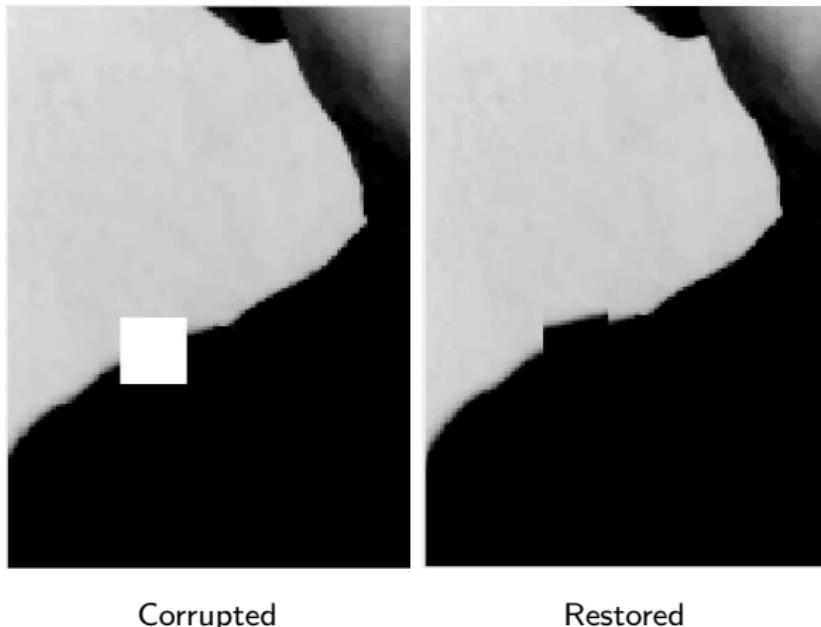
# Median Filter



$$\text{median}\{180, 207, 213, 230, 255, 255, 255, 255, 255\} = 255$$

# Median Filter

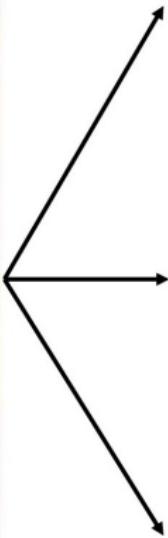
- Trivial solution is to increase filter size iteratively, but still does not fix the problem



Corrupted

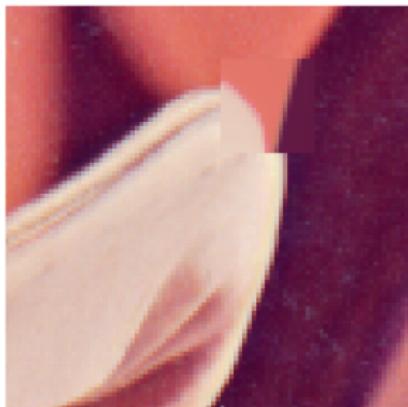
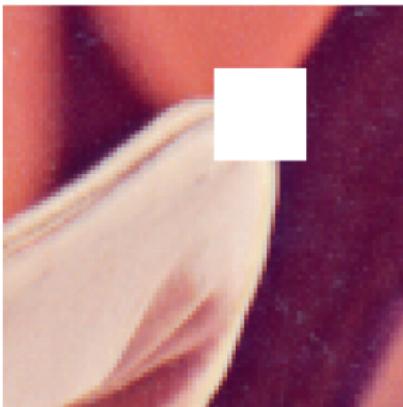
Restored

# Restoring Color Images



# Restoring Color Images

- Leads to color shift problem in corrupted region



- Necessary to constrain with contours

# Extraction of Contours

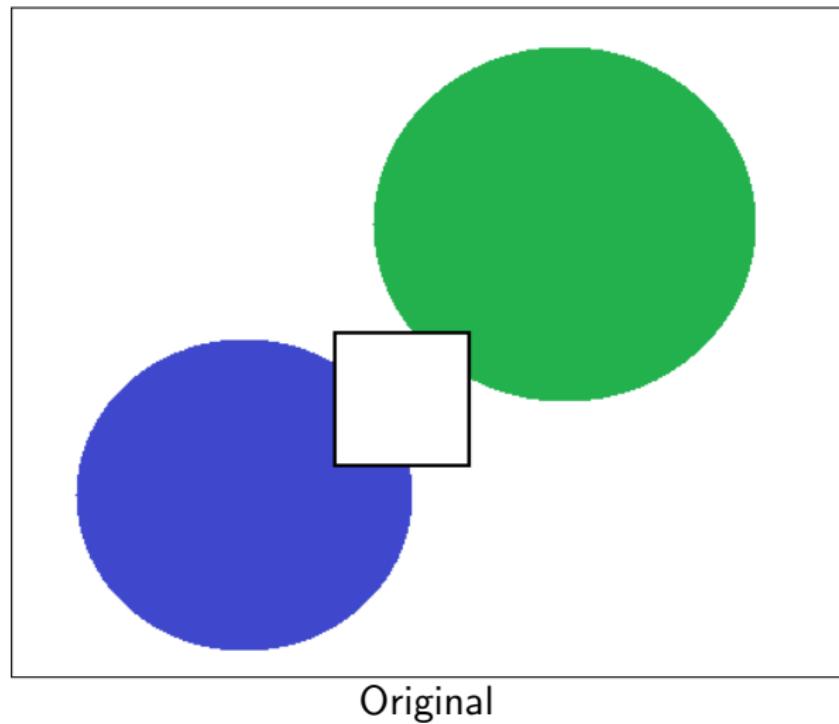


Laplacian

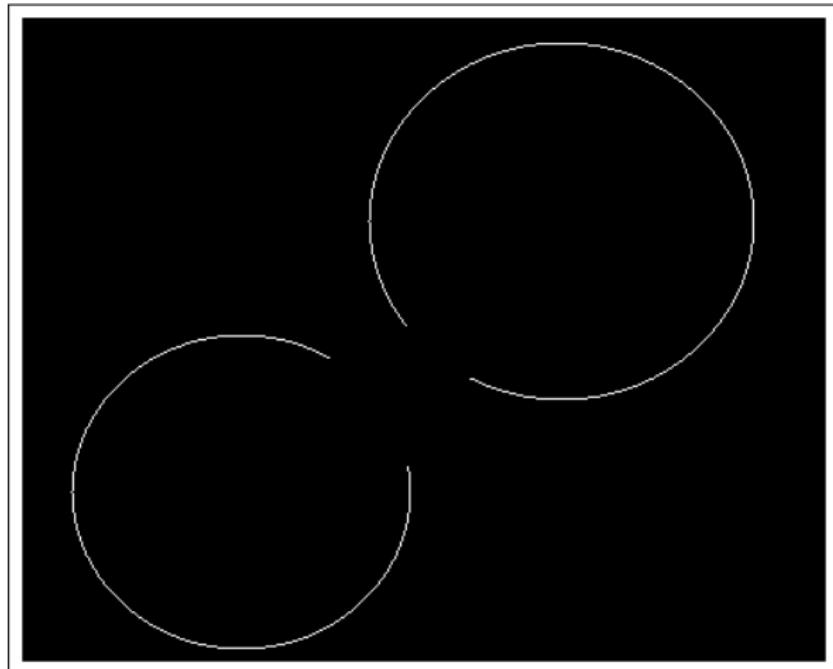


Sobel

# Recognizing Contourlines

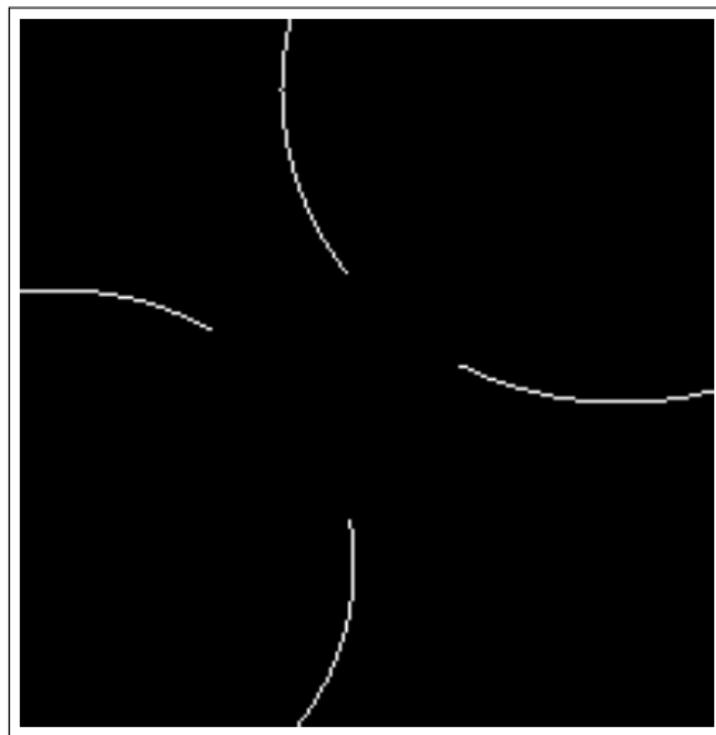


# Recognizing Contourlines

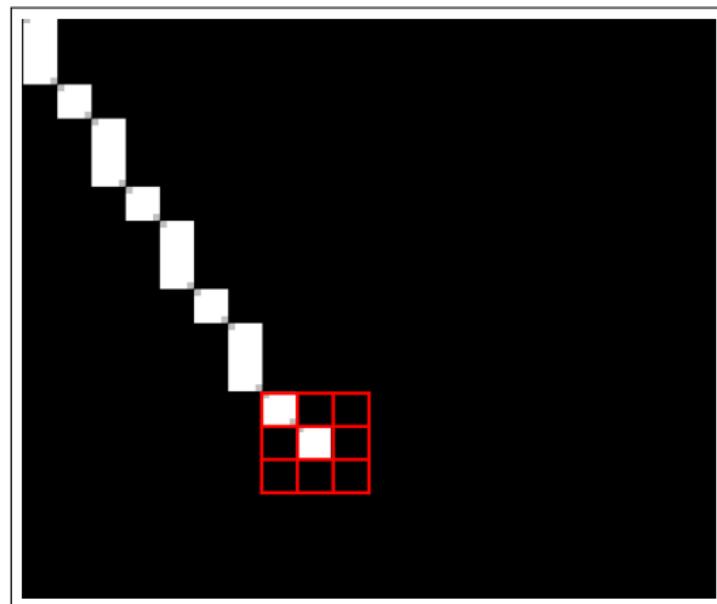


Correction of misleading error box

# Recognizing Contourlines

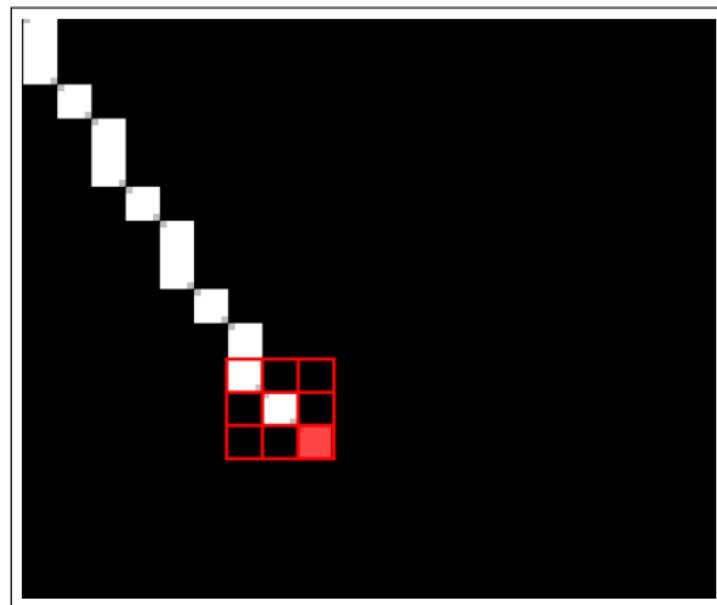


# Recognizing Contourlines



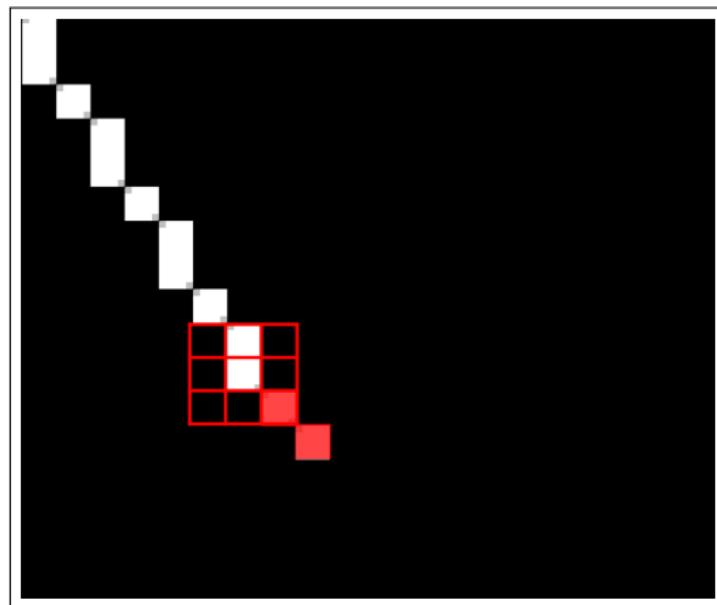
Step 1

# Recognizing Contourlines



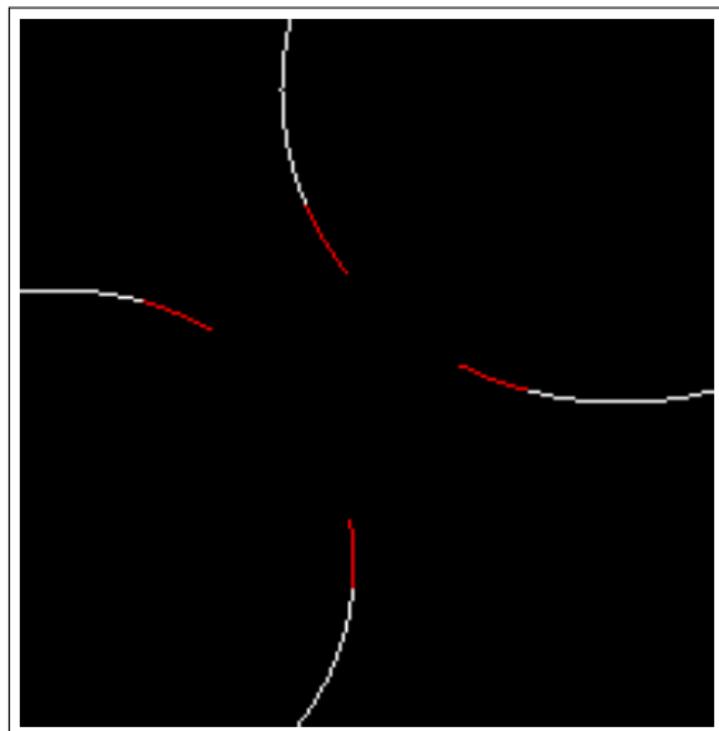
Step 2

# Recognizing Contourlines



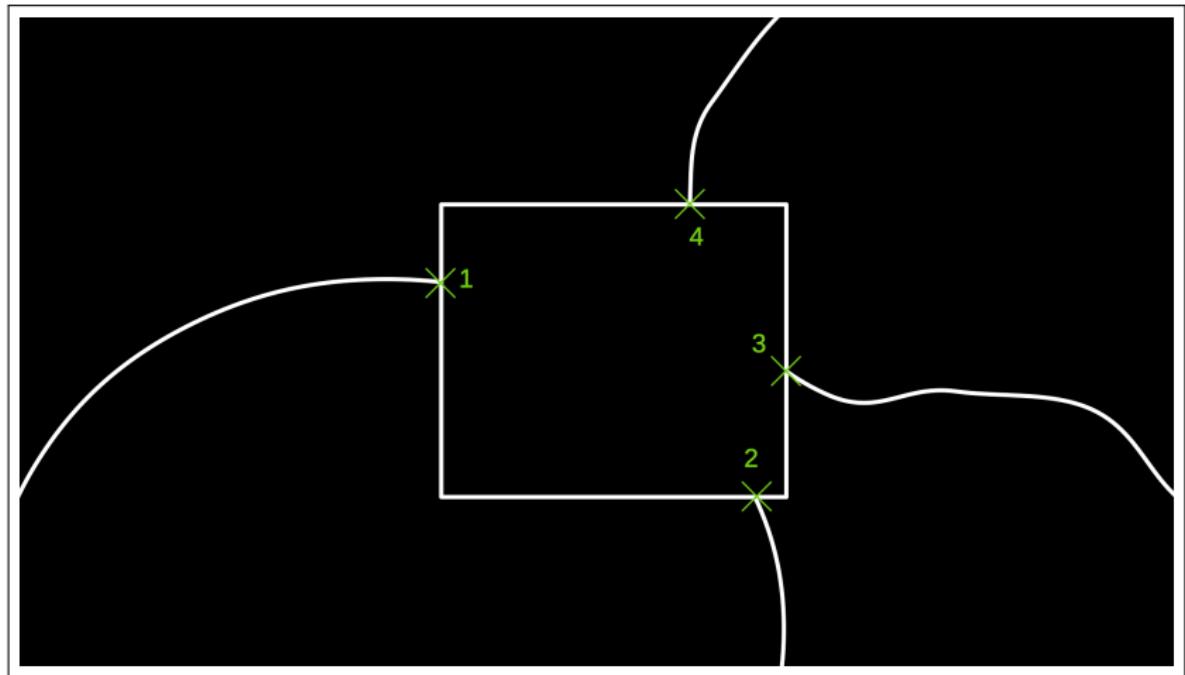
Step 3

# Recognizing Contourlines

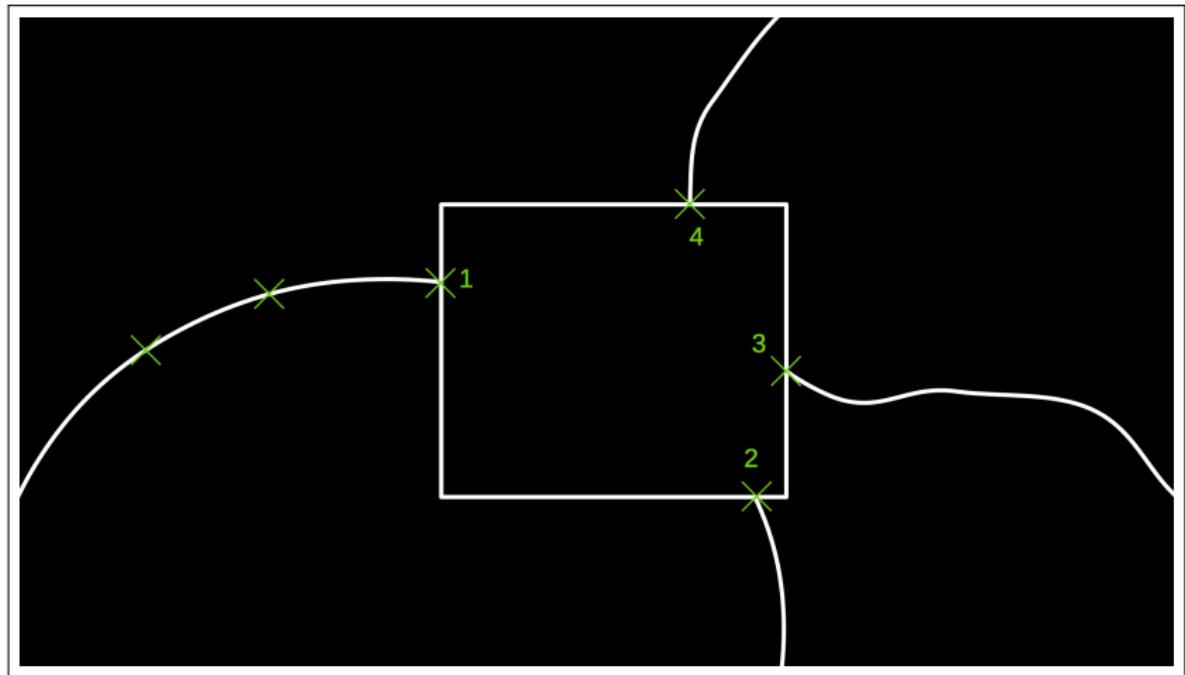


Detection of contourlines

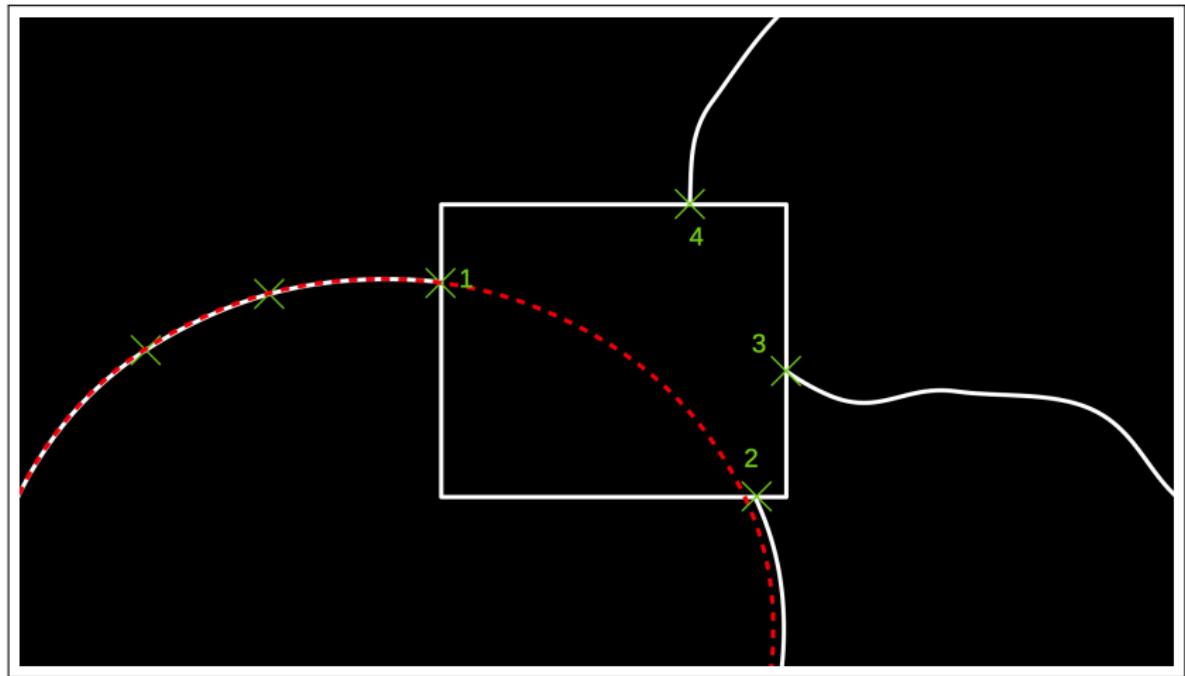
# Matching the Contours



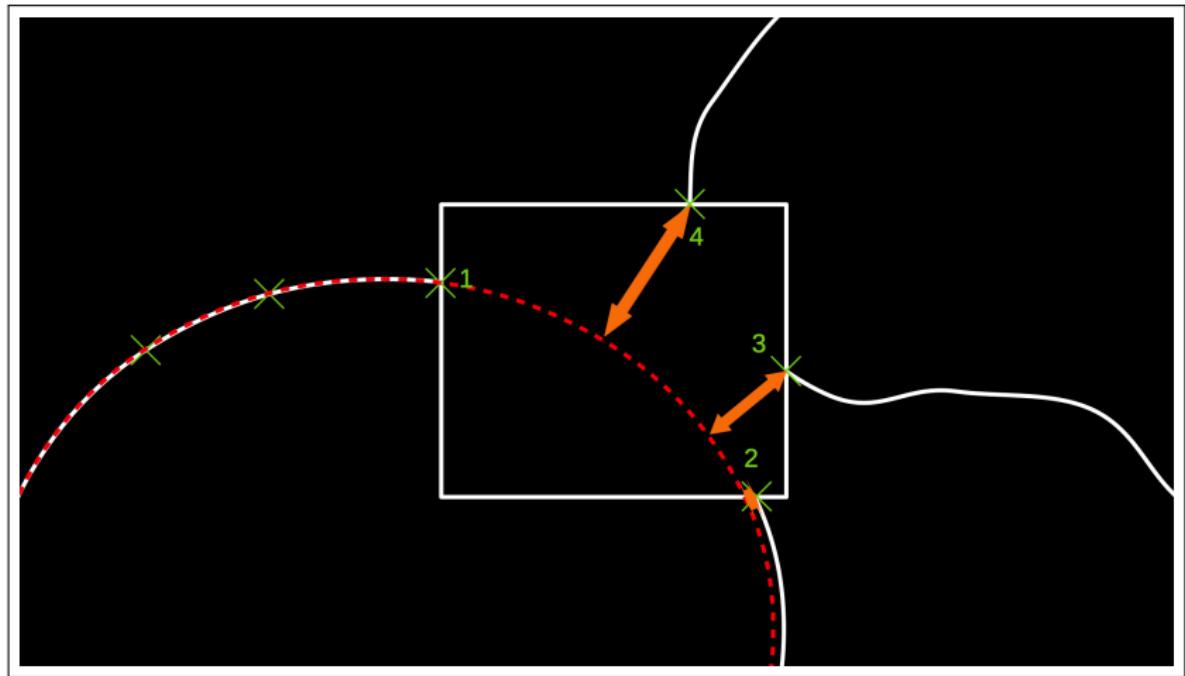
# Matching the Contours



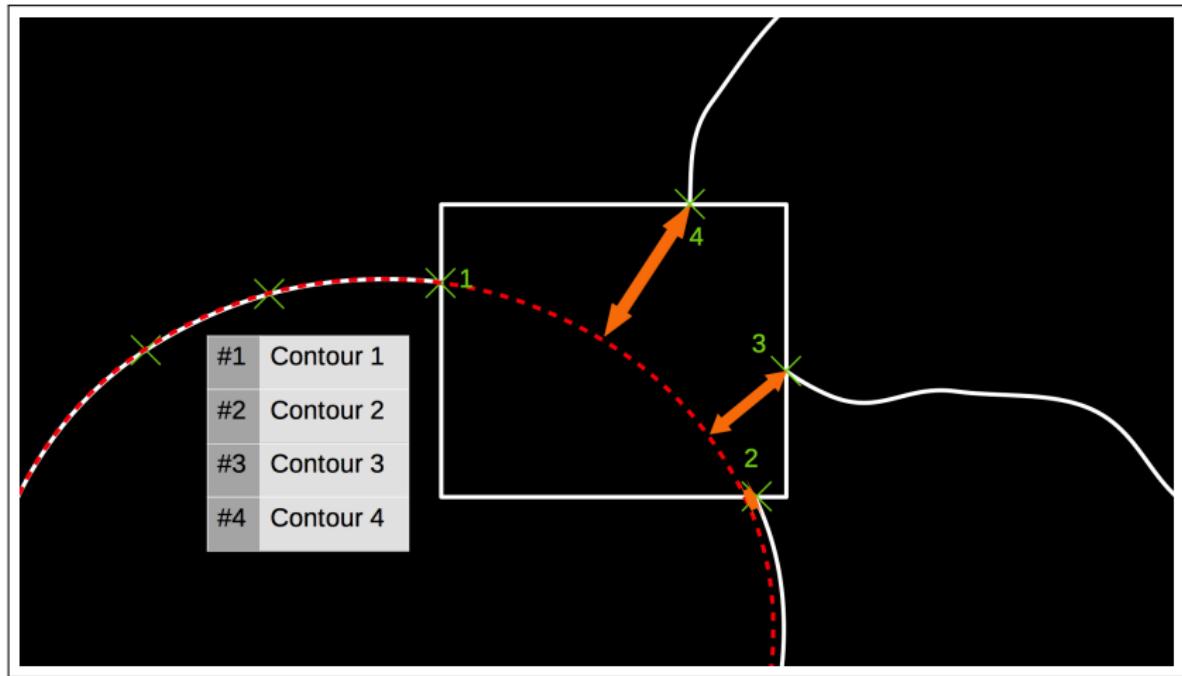
# Matching the Contours



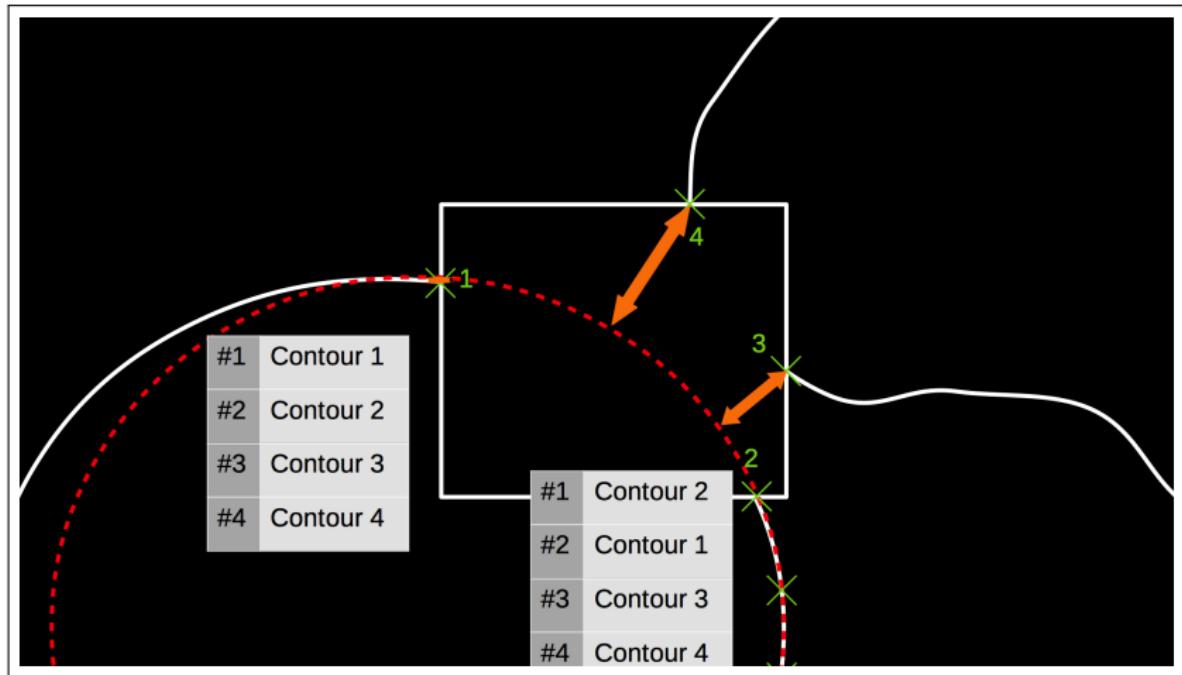
# Matching the Contours



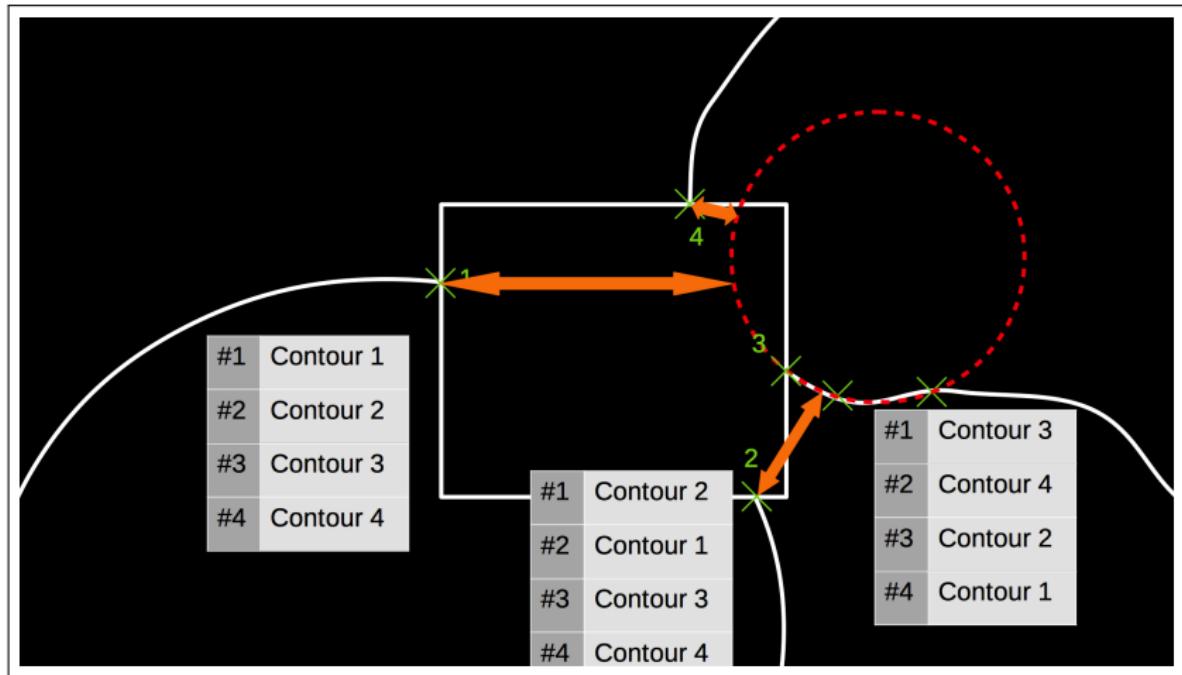
# Matching the Contours



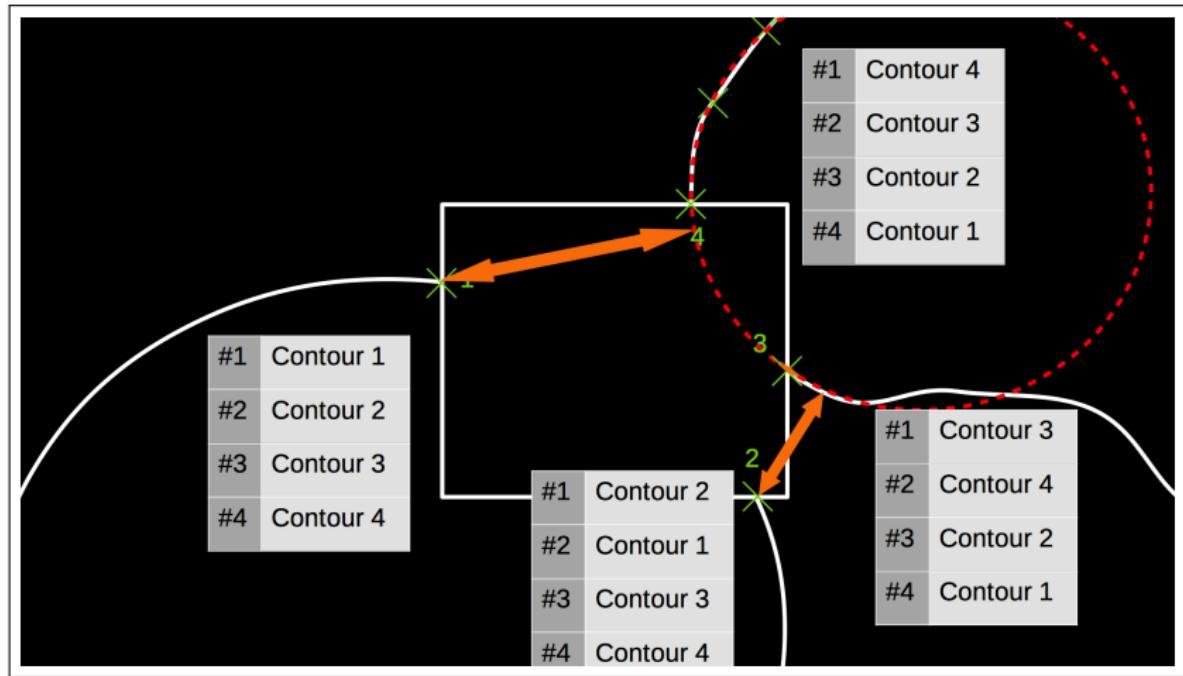
# Matching the Contours



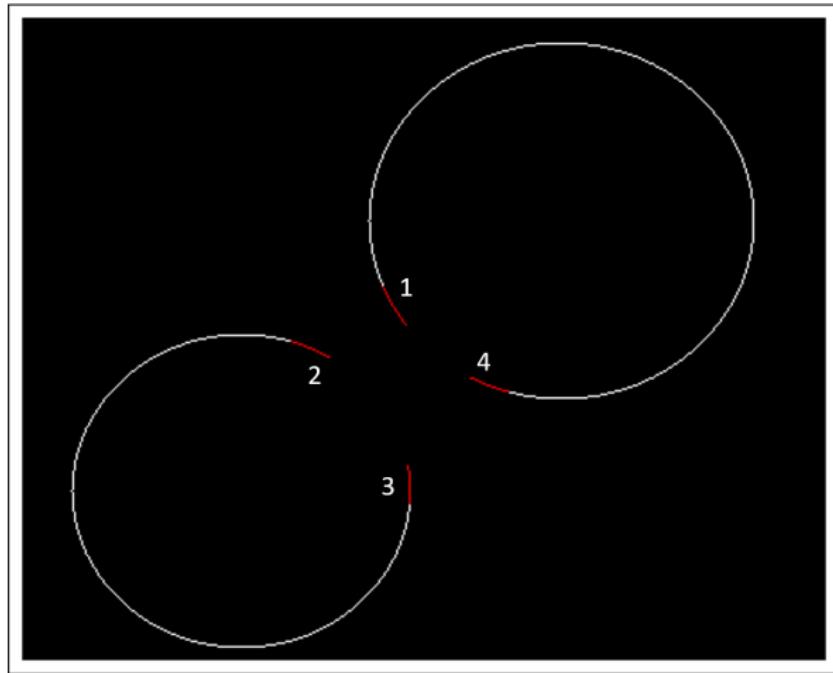
# Matching the Contours



# Matching the Contours



# Matching the Contours



# Connecting Contours

## ① Curve-Parametrization:

$$\Phi : \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}, \quad k \mapsto \begin{pmatrix} x(k) \\ y(k) \end{pmatrix}$$

# Connecting Contours

- ① Curve-Parametrization:

$$\Phi : \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}, \quad k \mapsto \begin{pmatrix} x(k) \\ y(k) \end{pmatrix}$$

- ② Componentwise interpolation with red marked sample points

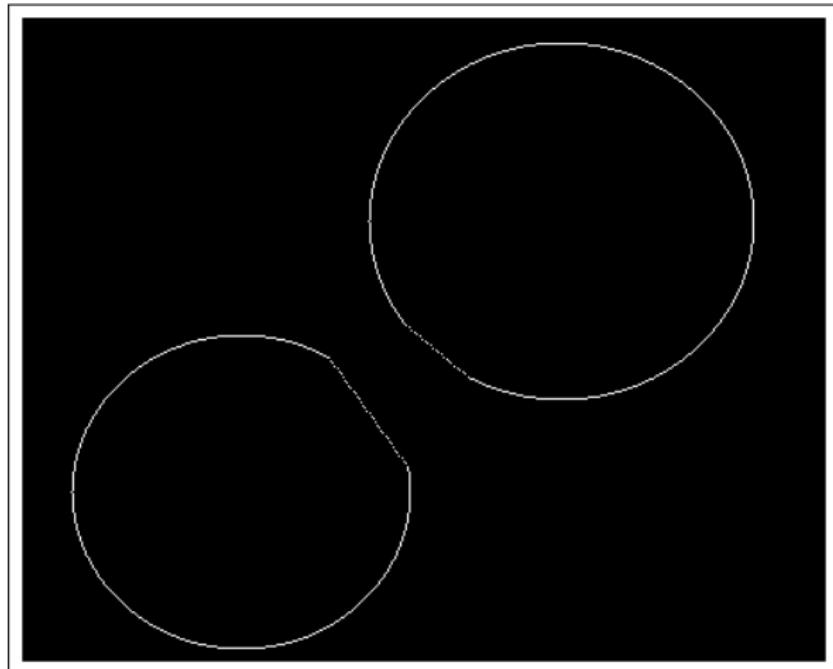
# Connecting Contours

- ① Curve-Parametrization:

$$\Phi : \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}, \quad k \mapsto \begin{pmatrix} x(k) \\ y(k) \end{pmatrix}$$

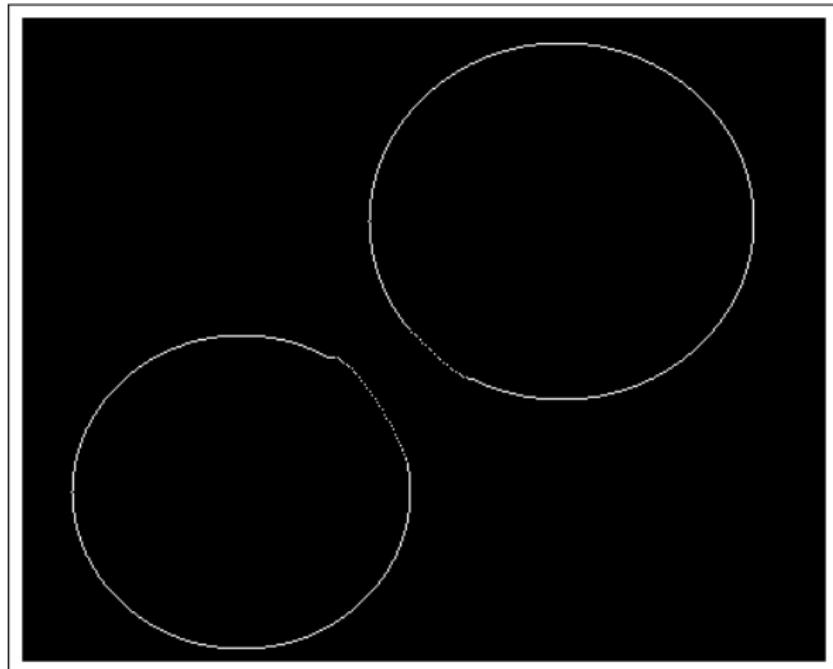
- ② Componentwise interpolation with red marked sample points
- ③ Mapping to pixel grid

# Connecting Contours



Linear interpolation

# Connecting Contours

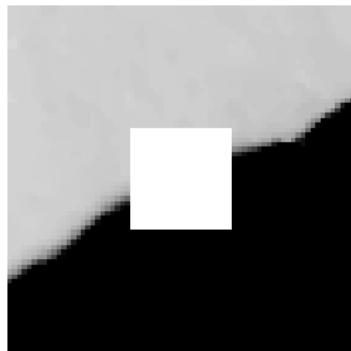


Cubic spline interpolation

# Grayscale Test Case

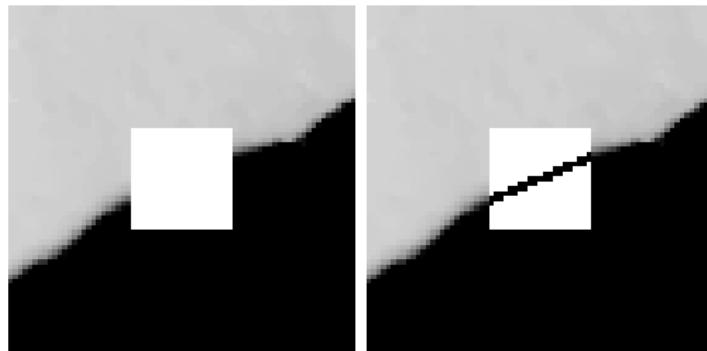


# Results



Missing area

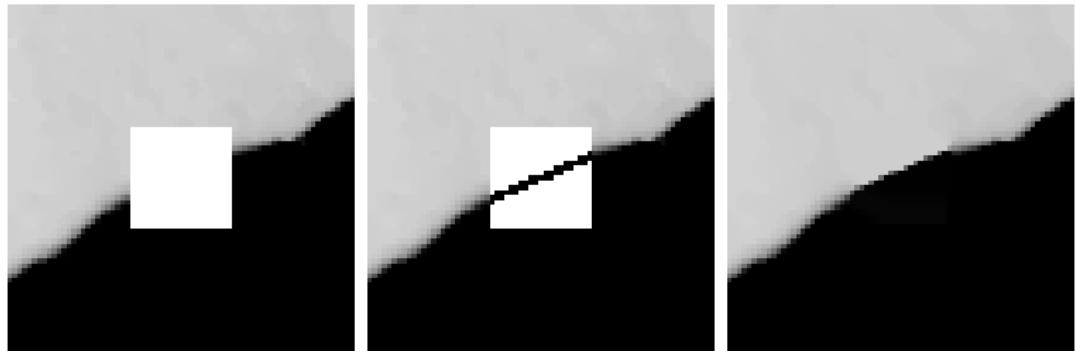
# Results



Missing area

Linear interpolation

# Results



Missing area

Linear interpolation

Color blending

# Results

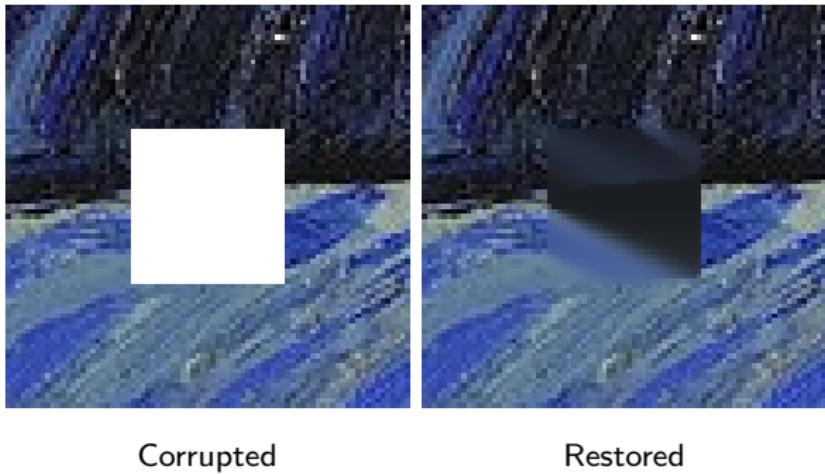


Corrupted



Restored

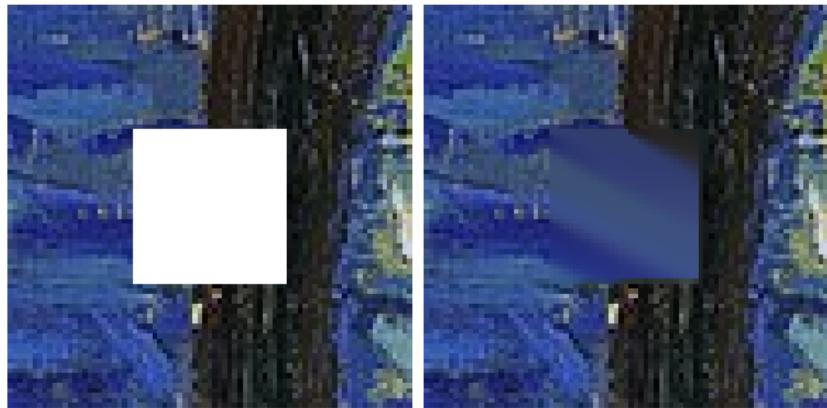
# Results



Corrupted

Restored

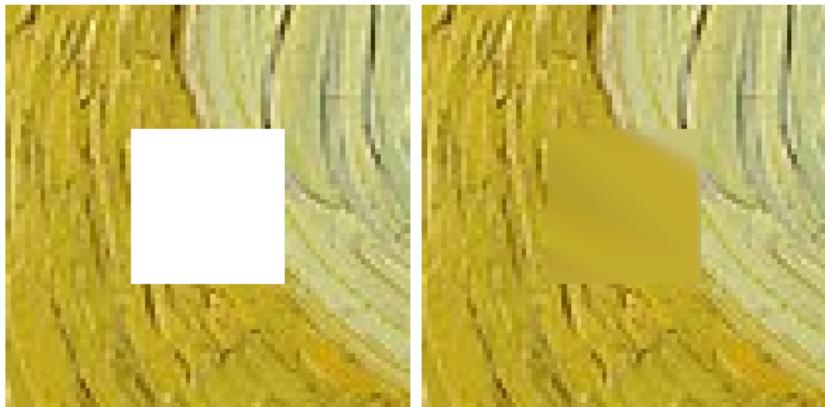
# Results



Corrupted

Restored

# Results



Corrupted

Restored

# Results



Restored Van Gogh "The Starry Night"

## Further Suggestions

- Find a better solution for contour extraction, e.g. curvelet transform
- Other sophisticated algorithms based on PDEs
- Including texture with help of Machine Learning methods, e.g. texture synthesis
- Using auto-encoder for larger and more detailed missing areas