Registration by ANN

ZOIsem 04/2022

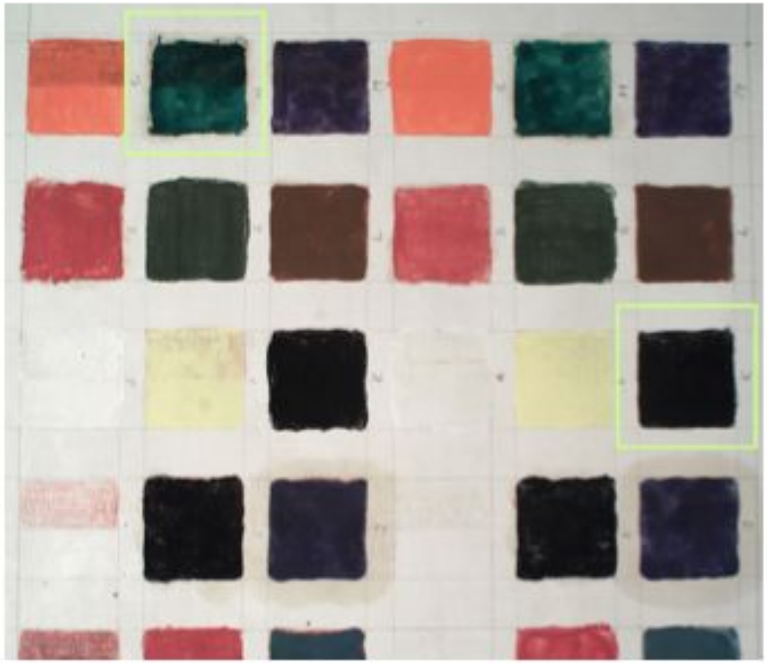
Postulating the problem

- Two different modalities
  - E.g. visible + near infra-red
  - with non-trivial information in common
  - Pixel in one modality is defined by several intensity values (RGB or narrow wavebands)
    - The intensity bands in one modality are registered (**assumption**!?)
- Modality datasets are slightly misregistered
  - E.g. capturing by the same lens causes different spherical distortion due to different index of refraction of different wavelengths.

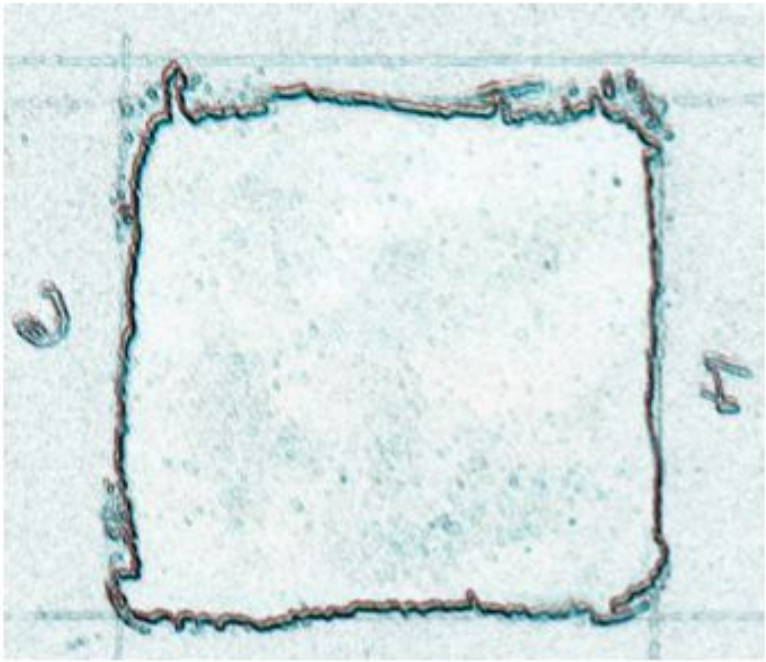
Shift ±50px
Scale ±10%
Rotation ±4°

Our goal:

- **Find out transformation parameters** for precise registration of both modalities.
- Use this parameters to improve input dataset to a registered one.

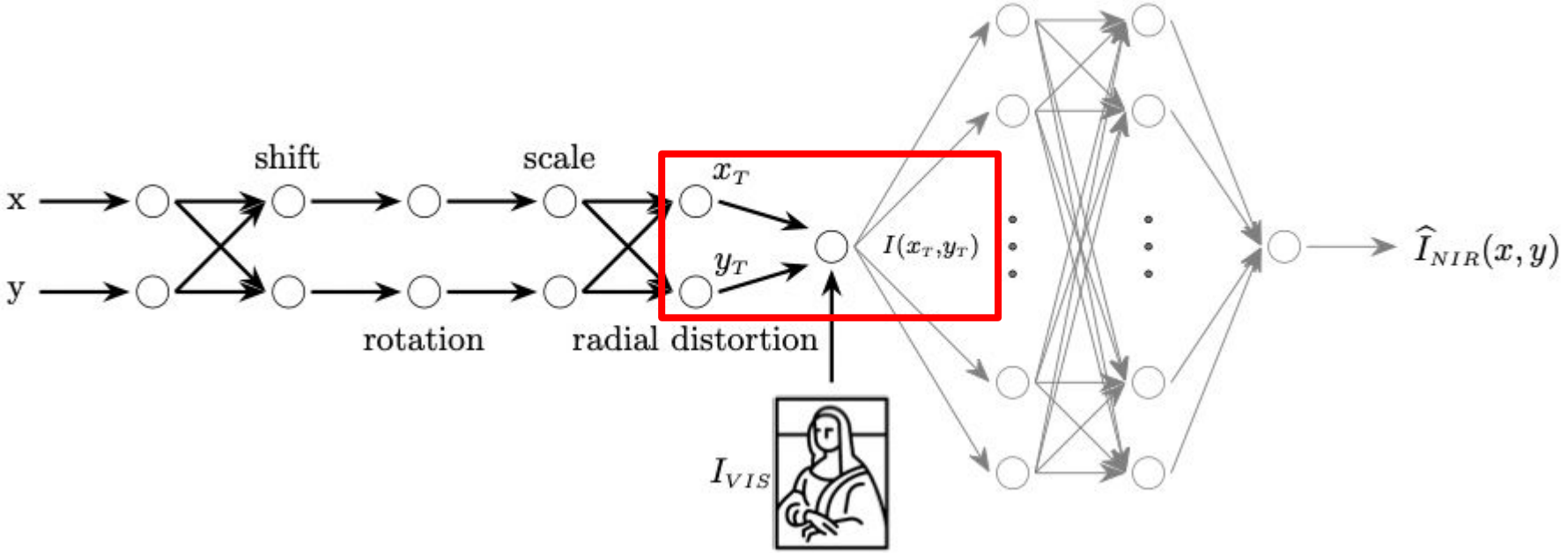


(a) Origin of cuts (original size was $3156 \times 4752px$). Green rectangle indicates zoomed areas.



(b) **Left top part**. Misplacement of NIR and VIS contour is approx. 6 pixels to the right

The schema


x
y
shift
rotation
scale
radial distortion
x_T
y_T
$I(x_T, y_T)$
I_{VIS}
$\hat{I}_{NIR}(x, y)$

Idx2px layer for Tensorflow

Forward:

- Interpolation of out of grid values
- In our case bilinear/bicubic interpolation of pixel intensities

Backward:

- Gradient in x and y
  - For bilinear interpolation is undefined for integer coordinates !!
  - Otherwise defined as intensity gradient in requested direction.

Idx2px layer and its useful possible variations

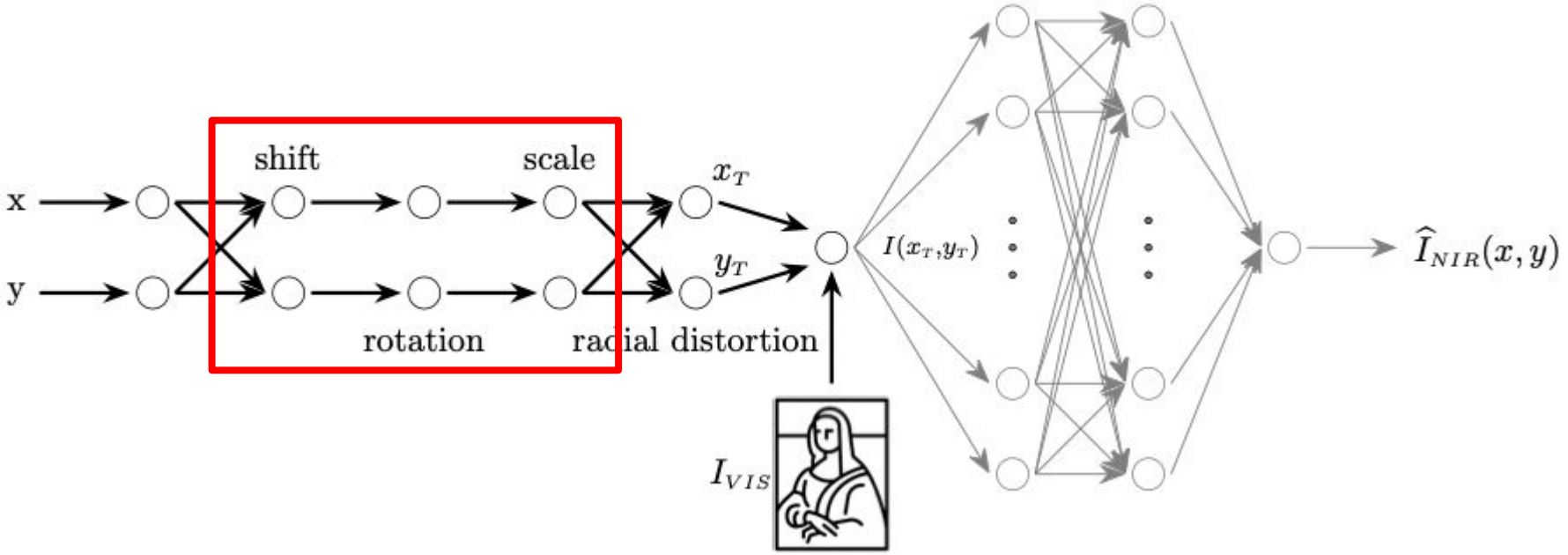
Forward:

- Not limited to one pixel (can be a patch)
  - Enables CNN approach
- Can be extended by **dropout** (selects random subset of pixel intensity vector)
  - More robust to "**assumption**"
- Etc...
  - Trainable parameters
    - VIS intensity correction (denoising, gamma correction, contrast enhancement)
    - Introduces problems with gradient => requires gradient in ∂p_i for each parameter p_i

Backward:

- For effective training is good to have gradients:
  - Smooth
  - Nonzero (the higher absolute value the better)
- **Smoothness is a problem of parameters varying in effect over the image!**
  - Scale
  - Rotation
  - Spherical distortion
- "Nonzero" requires variance in data

The schema


x
y
shift
scale
rotation
radial distortion
x_T
y_T
$I(x_T, y_T)$
I_{VIS}
$\hat{I}_{NIR}(x, y)$

Transformation layers

- Our naive approach:
  - train part of affine transform matrix
  - Split into 3 separate stages for experimenting
    - Shift
      - without any issue in convergence
      - Up to float precision limit
    - Rotation & shift
      - Problematic - intensity gradient does not work well because of varying shift based on pixel [x,y]
- How to improve?
  - Adapt gradient computation to obtain smooth and nonzero gradient function
    - E.g. trainset from annulus for rotation training, lowpass/highpass filtering for scale?

Q/A?

... and discussion