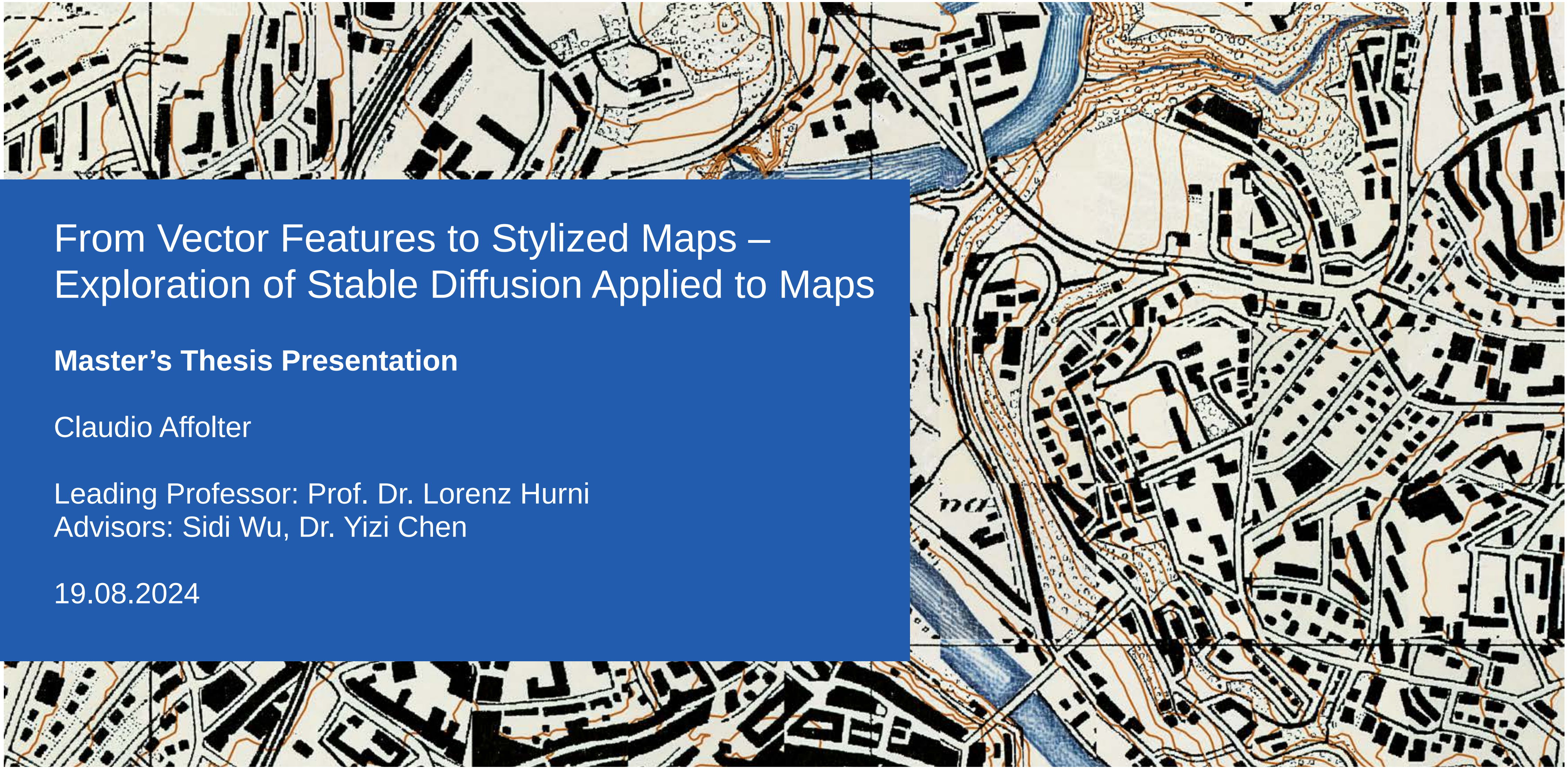


From Vector Features to Stylized Maps – Exploration of Stable Diffusion Applied to Maps

Master’s Thesis Presentation

Claudio Affolter

Leading Professor: Prof. Dr. Lorenz Hurni
Advisors: Sidi Wu, Dr. Yizi Chen

19.08.2024

Table of contents

Introduction and Motivation I

- **In recent years:**
  - Diffusion models have set the state-of-the-art in image generation
  - Principle: Gradual addition of gaussian noise to original image data → model learns to remove noise
  - Allow the creation of images from a text prompt (i.e., a user-given text description)
- **However:**
  - Complex layouts (e.g., maps) are impossible to perfectly describe only with a text prompt
  - Control over the spatial composition and semantic layout is very limited
- **Example:**
  - Prompt: “map of a town with a lake to the northeast, a forest to the south and a small village to the west of the lake”

Introduction and Motivation II

- **We conclude:**
  - For complex layouts such as map tiles, text prompts are not enough to control diffusion models

- **Why would it be useful to generate accurate map tiles using diffusion models?**
  1. Change simulation in a specific map style
  2. Inpainting (e.g., reconstruct damaged or missing map tiles in historical maps)
  3. Map standardization (transform maps from different sources to one common map style)
  4. General synthetic image generation

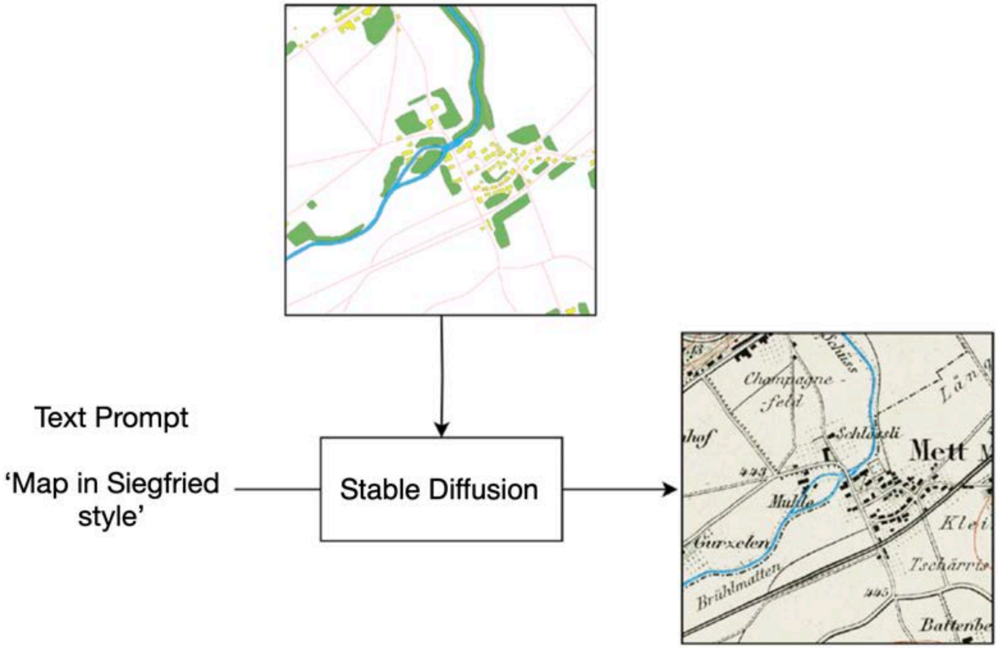
- **How to control diffusion models instead?**
  - Idea: Use **text prompt + vector data**

Goals

1. Explore how image diffusion models, here: *Stable Diffusion*, can be controlled using text prompts in combination with vector data:

   Given input vector data as conditioning controls, the trained model should be able to generate maps or map tiles in various pre-defined styles specified by a text prompt.

2. Development of a web-based application for map tile generation where users can select a specific map style and provide their own vector data to control the layout.



Methods and Workflow I

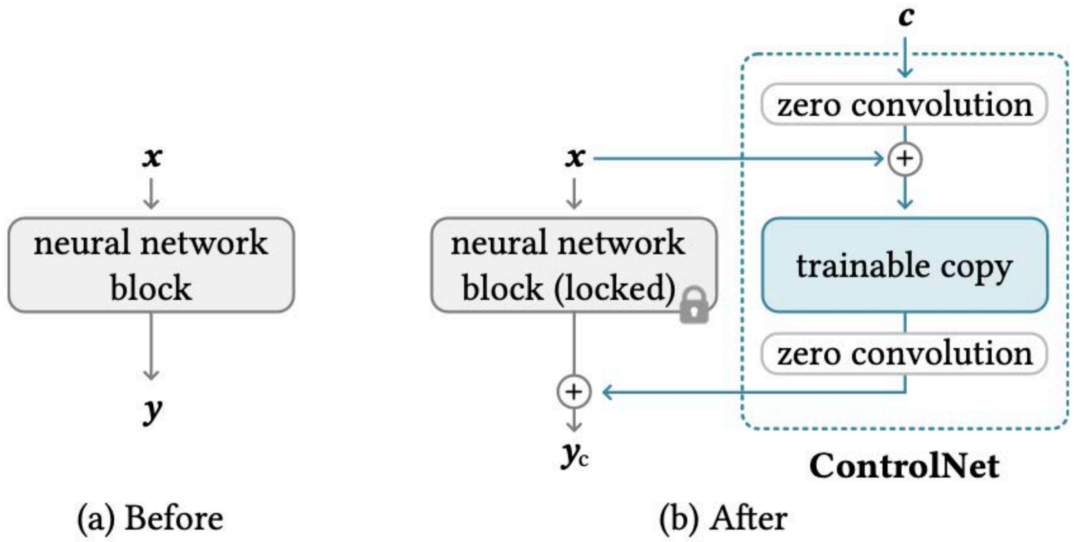
- **Variety of existing approaches for controlling image diffusion models:**
  - Using e.g., bounding boxes, depth images, sketches,…

- **Optimal method for this thesis:**
  - Allows controlling spatial composition (i.e., where exactly map objects are placed)
  - Allows controlling semantic layout (i.e., which map object belongs to which class)
  - Easily able to learn new map styles
  - Allows automation to generate many map tiles

- **Choice for this thesis:**
  - Use *ControlNet* to control the image diffusion model *Stable Diffusion*

Methods and Workflow II

- **Main idea behind ControlNet**
  - Inject additional conditions into the blocks of a neural network.
  - To do so: Parameters of original blocks are locked (frozen) and the blocks are cloned to create a trainable copy that takes external conditioning as input.
  - The locked parameters preserve the capabilities of the pre-trained model.
  - The trainable copy reuses the pre-trained model to create a backbone for handling diverse input conditions.

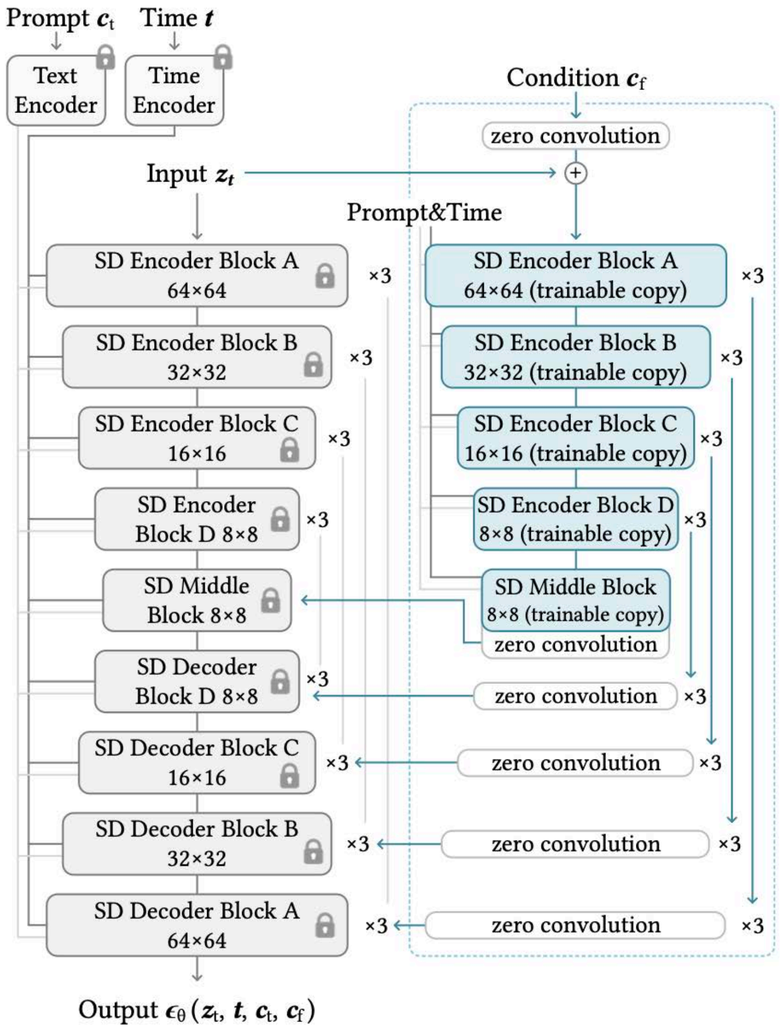


L. Zhang, A. Rao, and M. Agrawala, “Adding conditional control to text- to-image diffusion models”

Methods and Workflow III

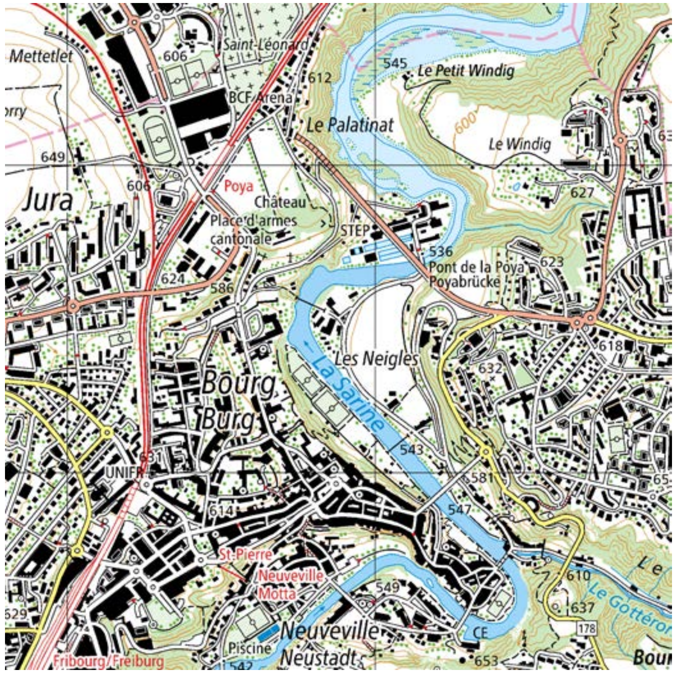
- **ControlNet and Stable Diffusion**
  - Here: ControlNet to add conditional control to Stable Diffusion.
  - Stable Diffusion: U-Net with encoder, middle block and decoder.
  - ControlNet structure is applied to each encoder level of the U-net.
  - Conditioning vector c_f is passed into ControlNet.

L. Zhang, A. Rao, and M. Agrawala, “Adding conditional control to text- to-image diffusion models”

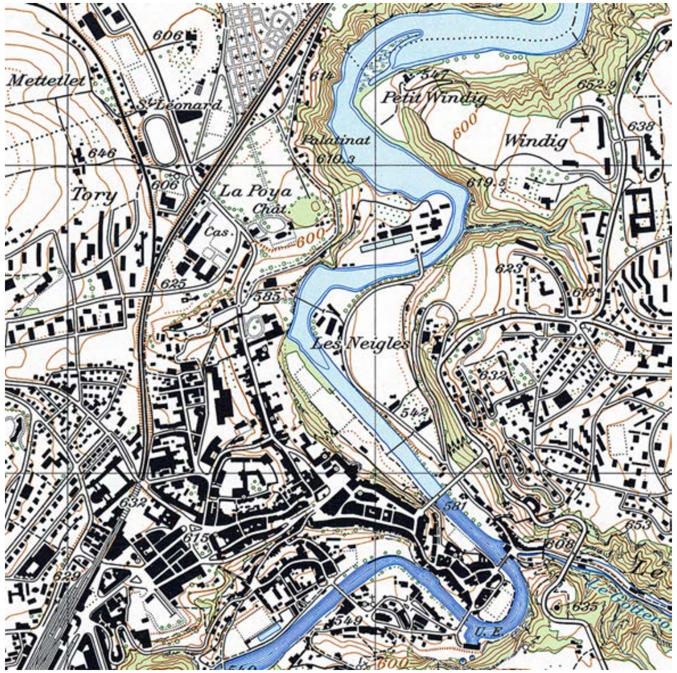


(a) Stable Diffusion (b) ControlNet

Map Styles



Swisstopo Map



Old National Map*
(1952-1979)

Siegfried Map
(1870-1926)

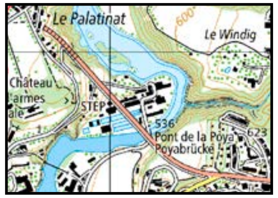
* Version 1:25’000

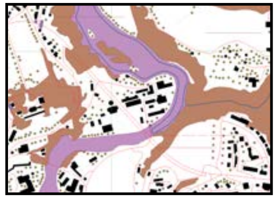
ETH zürich

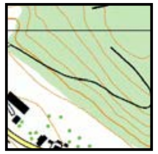
Methods and Workflow IV

- **Workflow Swisstopo:**

Download vector and corresponding raster data from swisstopo → In QGIS: select and symbolize vector data to match the raster data → Create train, validation and test sets consisting of (raster, vector, prompt) triples → Train ControlNet → Evaluate best model on test set using MSE → (back to: In QGIS: select and symbolize vector data to match the raster data)







raster map tile
(512 x 512, 1:5000)

vector map tile
(512 x 512, 1:5000)

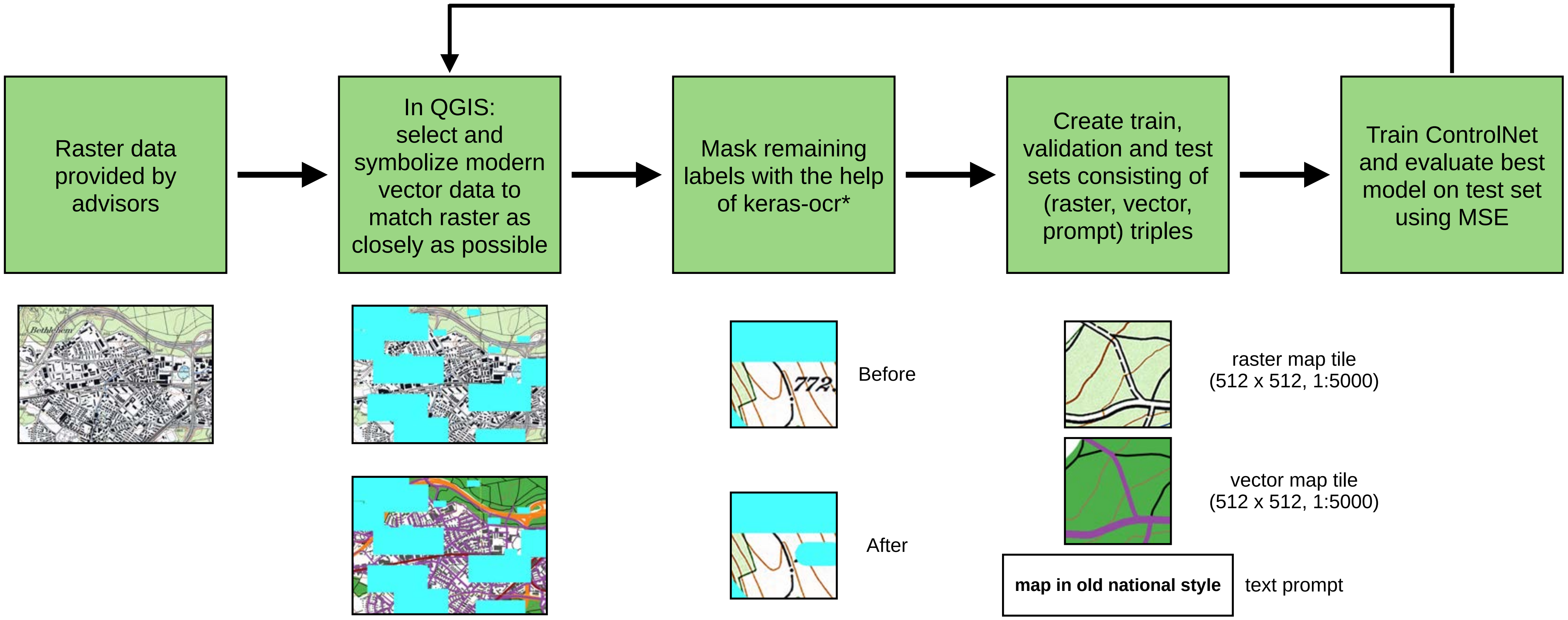
map in swisstopo style

text prompt

ETH zürich

Methods and Workflow V

- **Workflow Old National Map:**

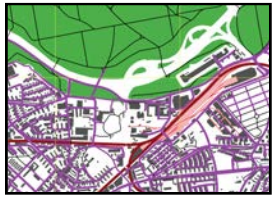


* Fausto Morales. keras-ocr. 2019, https://keras-ocr.readthedocs.io.

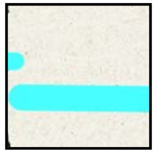
Methods and Workflow VI

- **Workflow Siegfried Map:**

Download raster data from GeoVITe → In QGIS: select and symbolize modern vector data to roughly match raster → Remove contour lines from raster and mask labels with the help of keras-ocr → Create train and test sets consisting of (raster, vector, prompt) triples → Train ControlNet and evaluate model on test set using image segmentation and mask method



Before



After

raster map tile
(512 x 512, 1:5000)

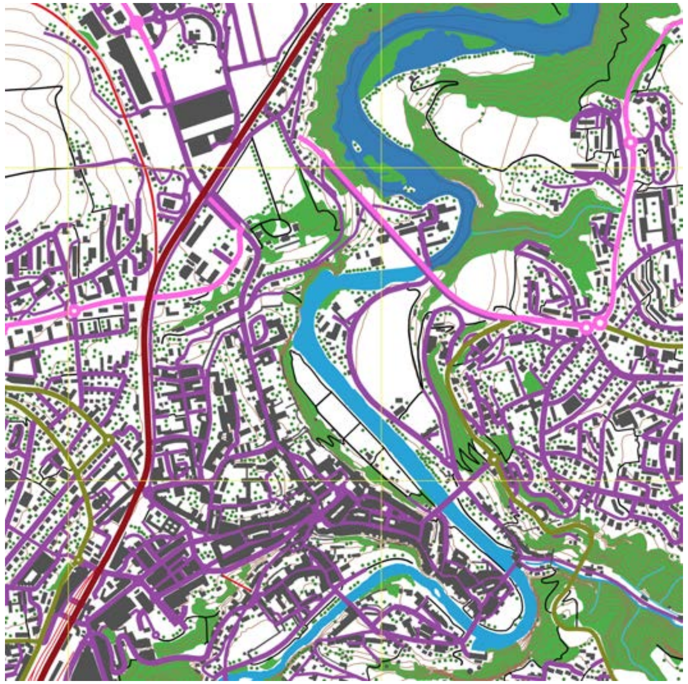
vector map tile
(512 x 512, 1:5000)

map in siegfried style

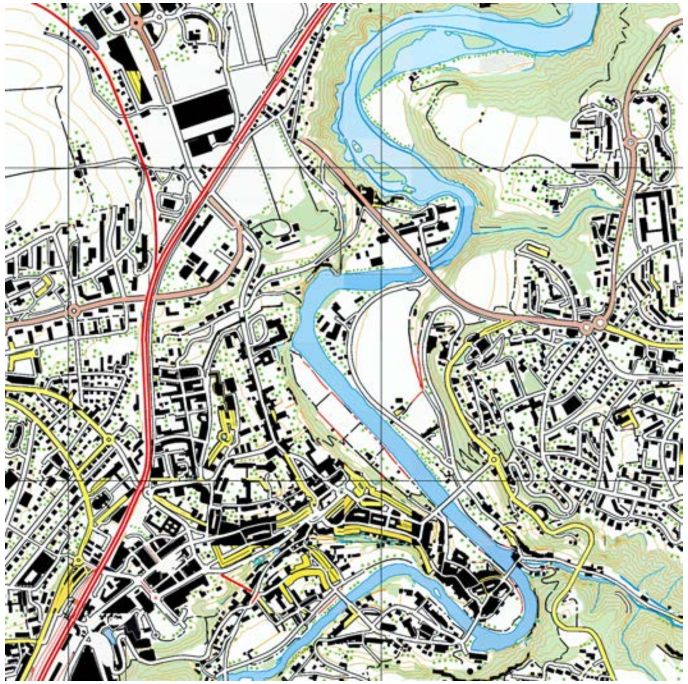
text prompt

Results I

- **Map in swisstopo style (5120 x 5120 pixels, 100 tiles)**

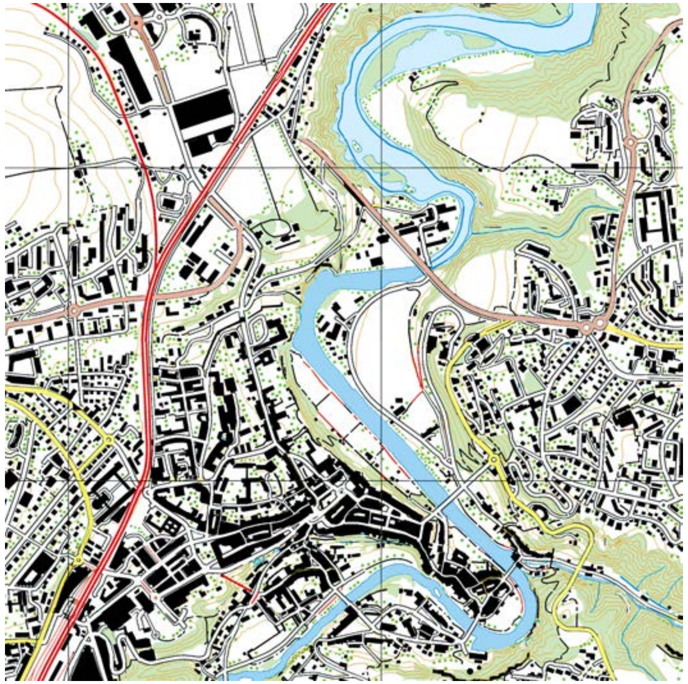


Control Image



Output

MSE: 5647

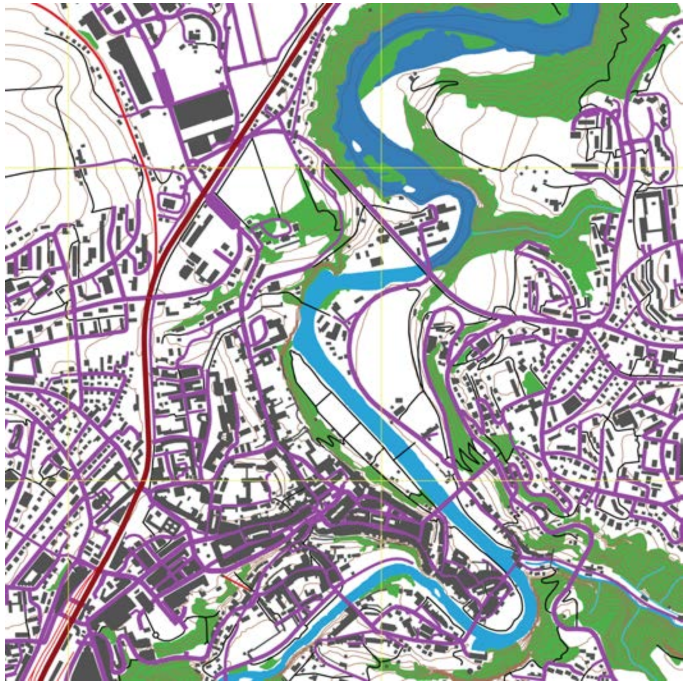


Output (after post-processing)

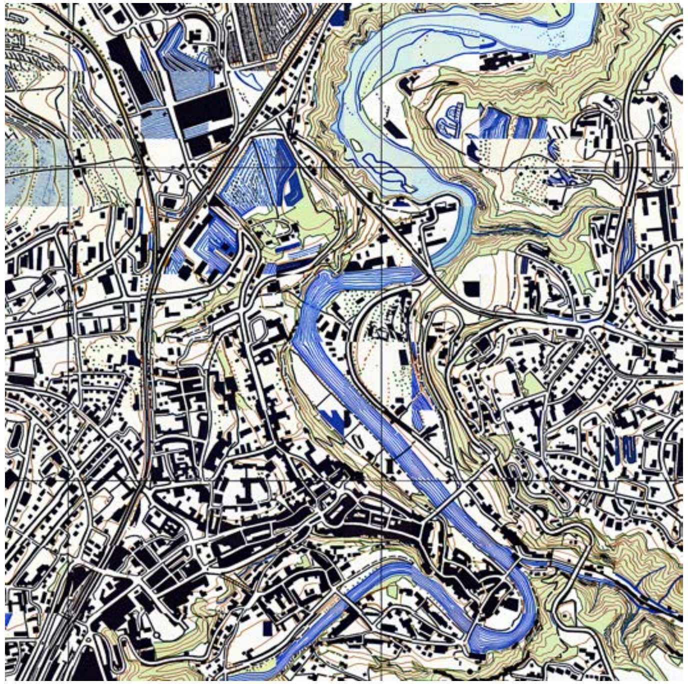
MSE: 4670

Results II

- **Map in Old National style based on modern vector data (5120 x 5120 pixels, 100 tiles)**

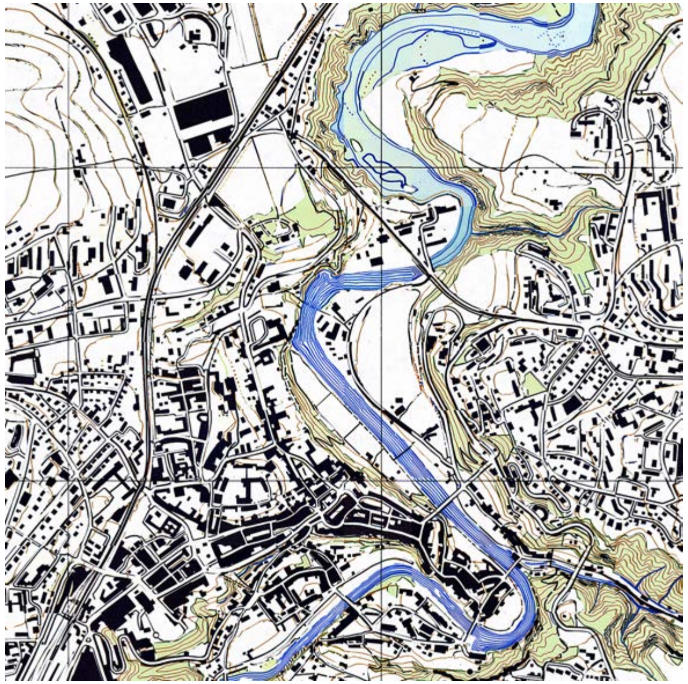


Control Image



Output

MSE: 17459*



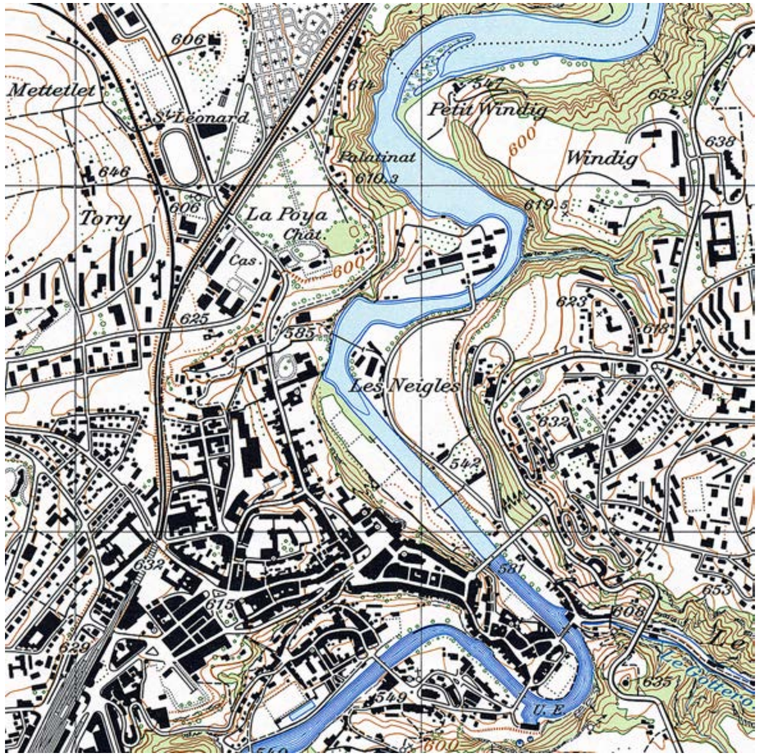
Output (after post-processing)

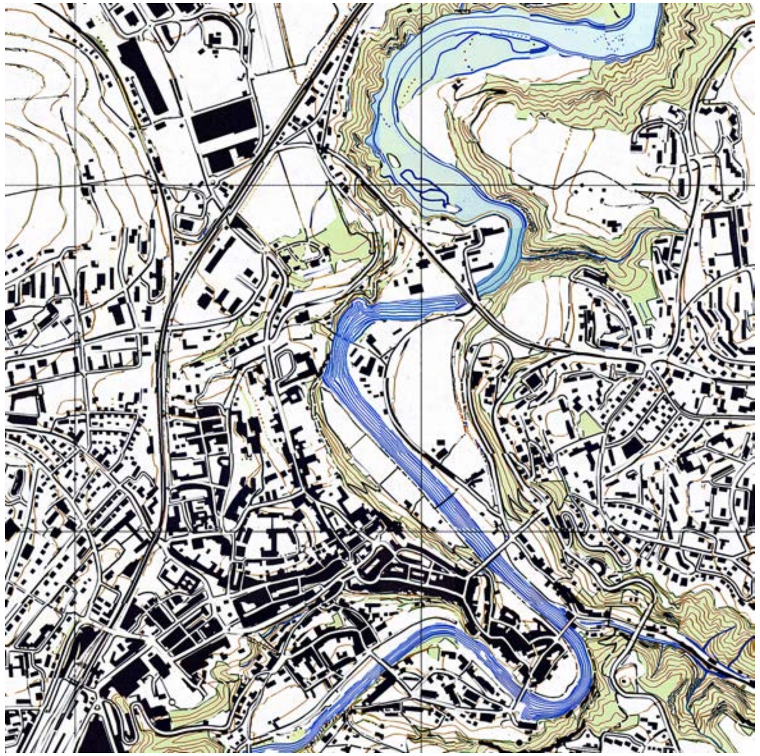
MSE: 15488*

*with respect to the Old National Map raster ≠ true ground-truth

Results III

- **Then vs. Now**





Results IV

- **Map in Siegfried style based on modern vector data (5120 x 5120 pixels, 100 tiles)**

Control Image

Output

Average MIoU: 0.25

Average MSE: 7444

Output (augmented)
Contour lines added afterwards

Average MIoU: 0.33

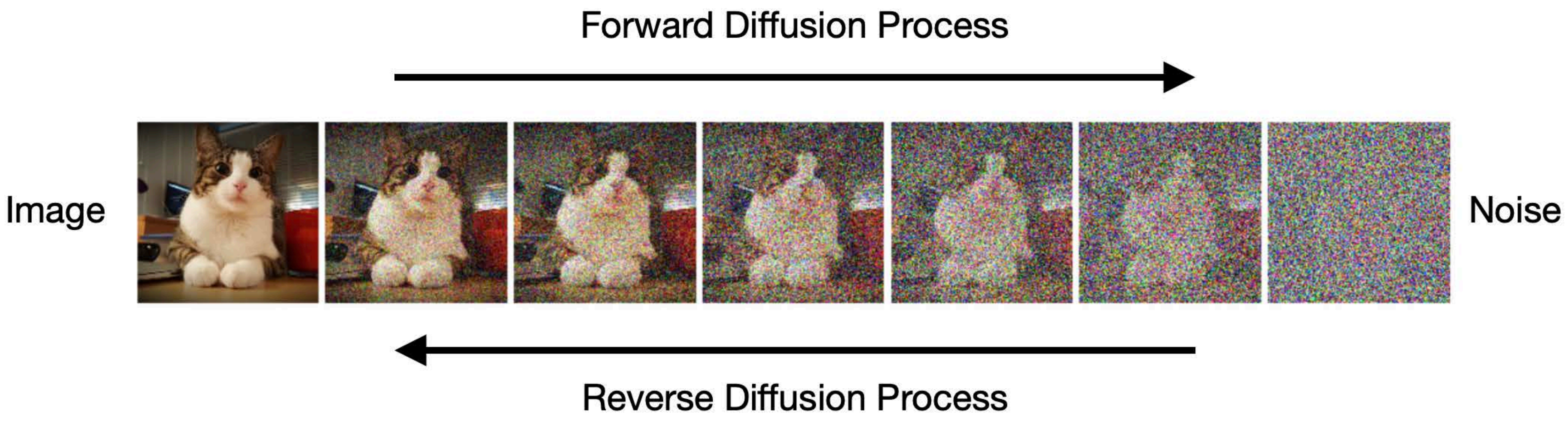
Average MSE: 5307

Results V

- **Then vs. Now**

Live Demo

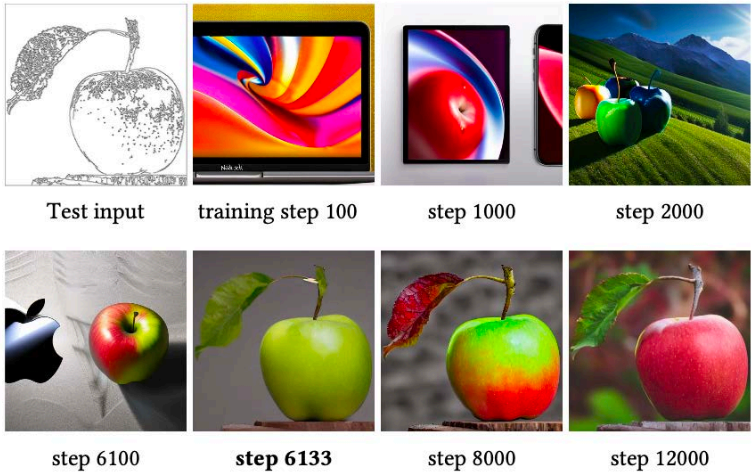
Additional I (Diffusion process)



https://cvpr2022-tutorial-diffusion-models.github.io/

Additional II (ControlNet)

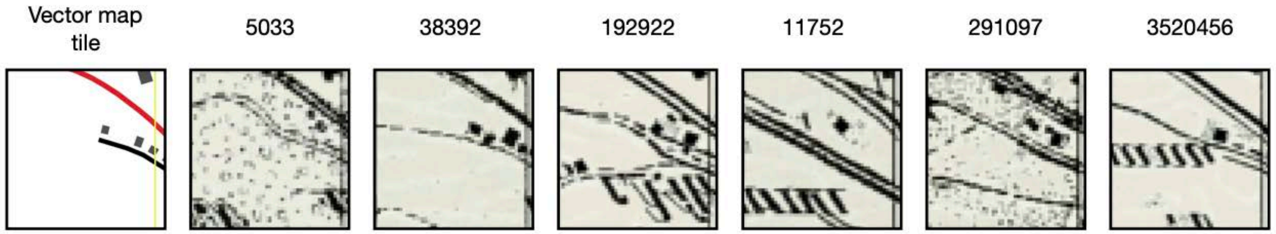
- **Training process of ControlNet**
  - Given a time step t, text prompts c_t, as well as a task-specific condition c_f, image diffusion algorithms learn a network to predict the noise added to the noisy image z_t.
  - Randomly replace 50% of text prompts c_t with empty strings → increases ControlNet's ability to directly recognize semantics in the input conditioning images (e.g., edges, poses, depth, etc.).
  - Zero convolutions do not add noise to the network → always predict high-quality images
  - Model does not gradually learn the control conditions but abruptly succeeds (sudden convergence phenomenon)



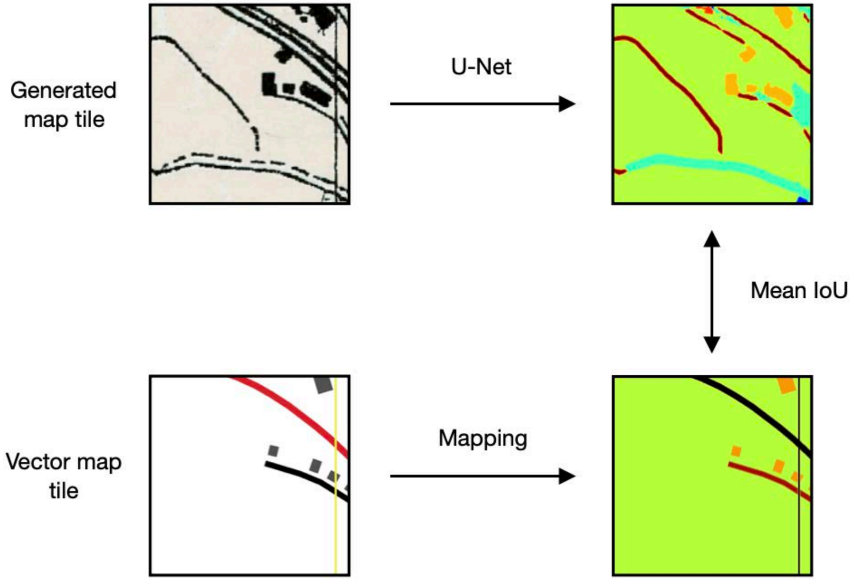
L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text- to-image diffusion models"

Additional III (Siegfried Map evaluation)

- **High influence of seed on resulting map tile**

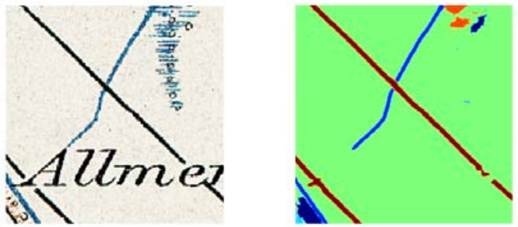


- **How to automatically choose the best Siegfried Map tile?**
  - First method: image segmentation
  - U-Net trained on a small dataset consisting of map sheets and perfectly matching vector data
  - Calculate Mean Intersection over Union (MIoU) score between each generated map tile version and corresponding vector map



Additional IV (Siegfried Map evaluation)

- **Problem with only using image segmentation:**
  - Map labels / fake labels are assigned the ‘background’ class



- **How to automatically check whether map tile is free of labels?**
- Second method: Vector map masks
- For each region of interest evaluate how much it differs from the average Siegfried Map pixel value
- e.g., MSE between generated background areas and the average background color of an original Siegfried Map tile