Self-Supervised Learning for Stereo Reconstruction on Aerial Images

Patrick Knöbelreiter, Christoph Vogel, Thomas Pock

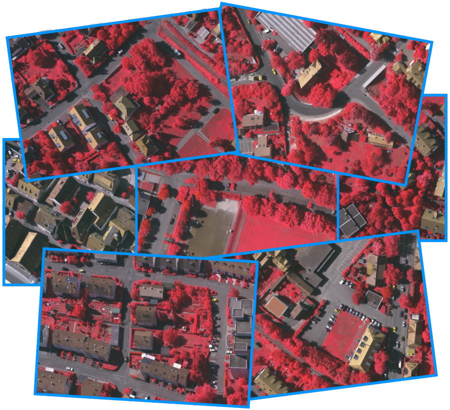
Institute for Computer Graphics and Vision
Graz University of Technology

Center for Vision, Automation & Control,
AIT Austrian Institute of Technology GmbH

July 20, 2018

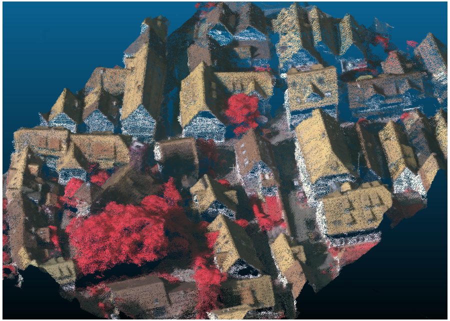
Introduction

Introduction



[Vaihingen Dataset]

⇒

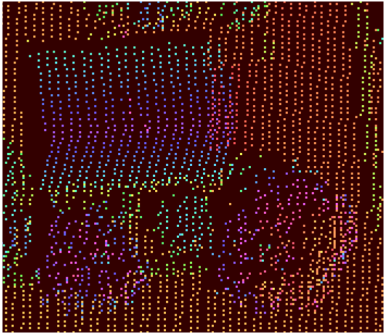


3D point cloud

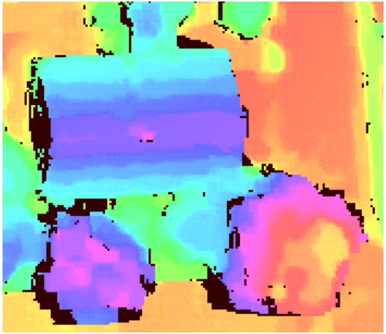
- Compute a dense 3D point cloud from high resolution overlapping images

Aerial 3D - Approaches

	LiDAR	Dense image matching
Flight height[1]	500 m	1800 m
Density[1]	6.7 pt/m^2	39.1 pt/m^2
Reliability	high	medium - high



LiDAR (Light detection and Ranging)



Dense image matching

[1]Vaihingen Dataset

Dense Image Matching

- Efficient algorithms exist
  - Plane Sweep Stereo [Collins]
  - Semi-global Matching (SGM) [Hirschmüller]
- Recent developments in deep learning lead to considerable performance improvements
  - MC-CNN: Learned matching + Post-processing [Zbontar *et al.*]
  - Content CNN: Learned features + Post-processing [Luo *et al.*]
  - CNN-CRF: Principled End-to-End Approach without Post-processing [Knöbelreiter *et al.*]

Utilize a modern deep learning based approach without the need of huge amount labeled of training data

- We are going to use the CNN-CRF approach in a self-supervised learning setting

Preliminaries

What is Self-Supervised Learning?

Central Assumption

"If we are highly confident about our prediction, we assume that it is correct."

What is Self-Supervised Learning?

Central Assumption

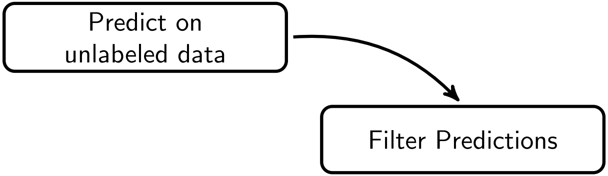
"If we are highly confident about our prediction, we assume that it is correct."

Predict on
unlabeled data

What is Self-Supervised Learning?

Central Assumption

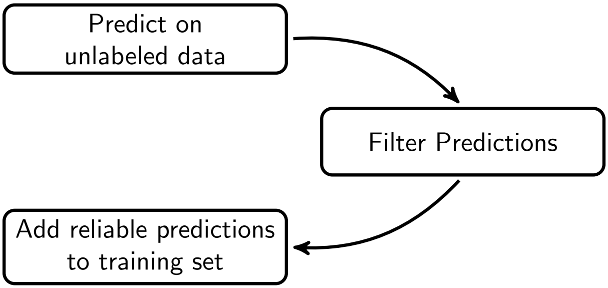
"If we are highly confident about our prediction, we assume that it is correct."



What is Self-Supervised Learning?

Central Assumption

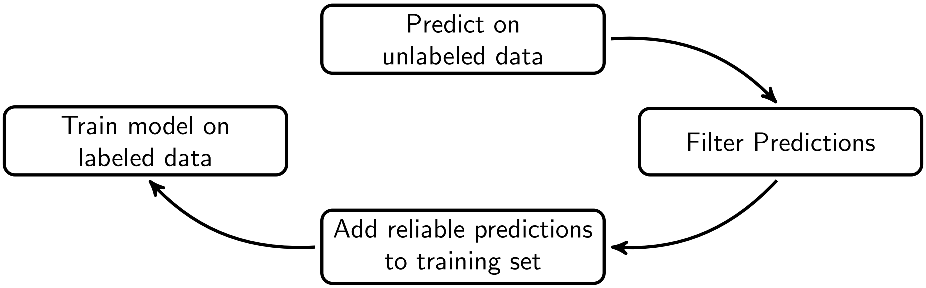
"If we are highly confident about our prediction, we assume that it is correct."



What is Self-Supervised Learning?

Central Assumption

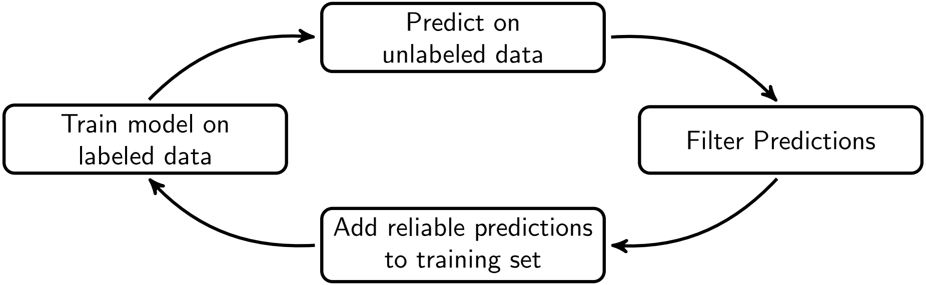
"If we are highly confident about our prediction, we assume that it is correct."



What is Self-Supervised Learning?

Central Assumption

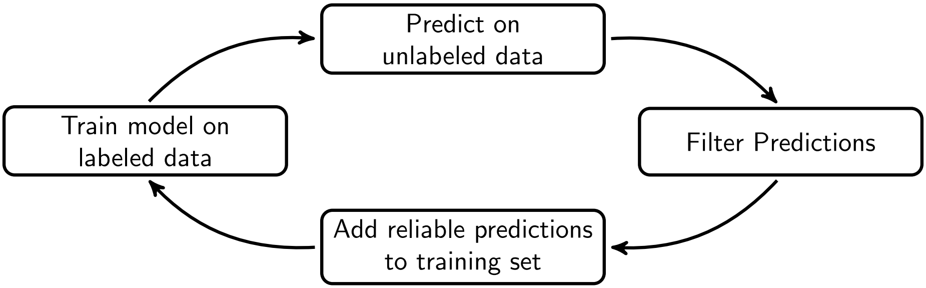
"If we are highly confident about our prediction, we assume that it is correct."



What is Self-Supervised Learning?

Central Assumption

"If we are highly confident about our prediction, we assume that it is correct."



- Key: We need a good criterion to distinguish between good and bad predictions

Self-Supervised Learning - Algorithm

Input: (Pre-trained) learner $f^{(0)}$, unlabeled data $X_u = \{x_u^{(i)}\}_{i=1}^M$
Result: Learner $f^{(N)}$

$X_l = \{\}$
$Y_l = \{\}$
for $n \leftarrow 1$ **to** N **do**

```
// 1. Predict on unlabeled data
```

$Y_p = \{\}$
for $i \leftarrow 1$ **to** M **do**

$y_p^{(i)} = f(x_u^{(i)})$
$Y_p = Y_p \cup y_p^{(i)}$

end

```
// 2. Filter predictions
```

$(X_f, Y_f) = \{(x_u^{(i)}, y_p^{(i)}) : c(y_p^{(i)}) > \tau\}_{i=1}^M$

```
// 3. Add reliable predictions to training set
```

$X_l = X_l \cup X_f$
$Y_l = Y_l \cup Y_f$

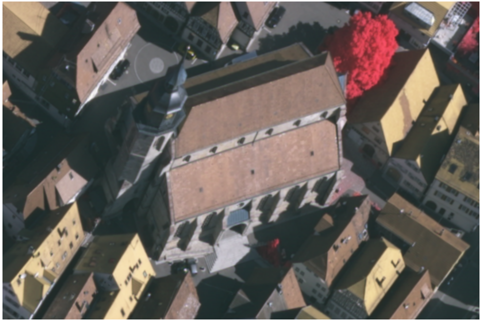
```
// 4. Train model on labeled data
```

$f^{n+1} \leftarrow$ train f^n on (X_l, Y_l)

end

Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows

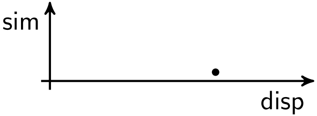


Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows

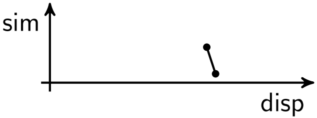
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



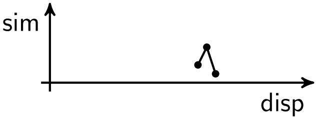
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



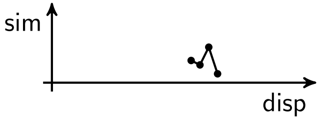
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



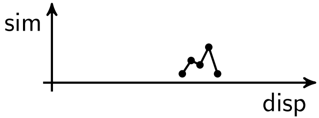
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



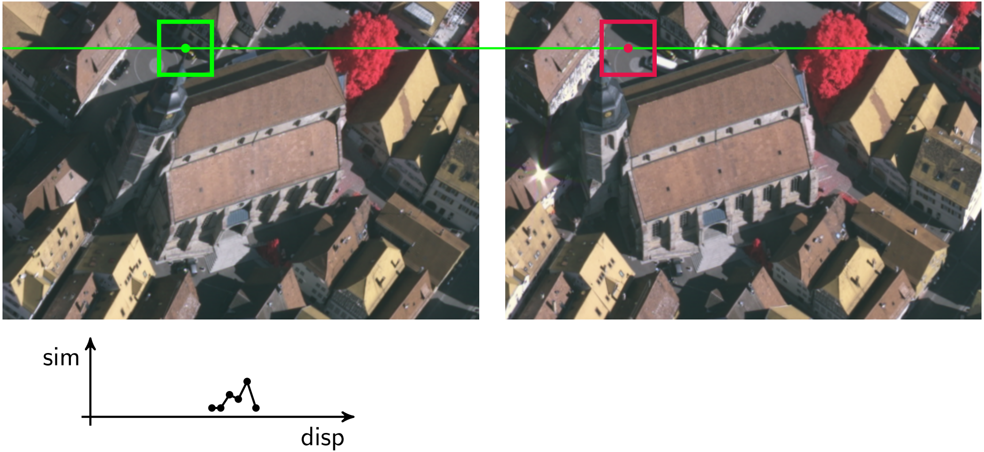
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



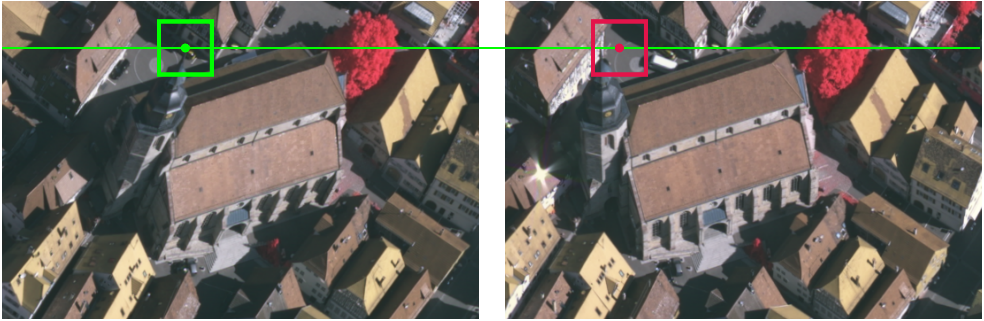
Dense Image Matching (Stereo)

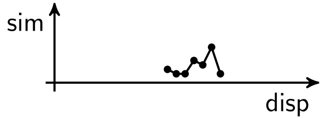
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



Dense Image Matching (Stereo)

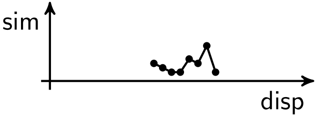
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image





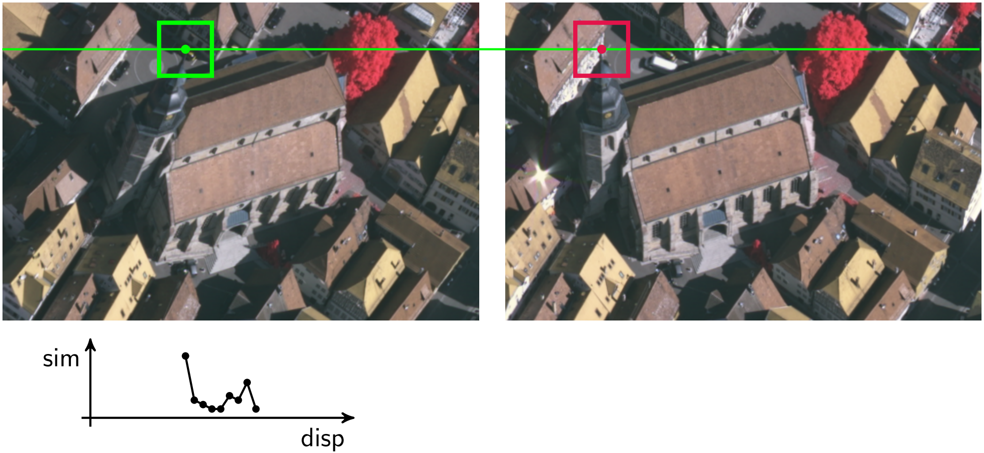
Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



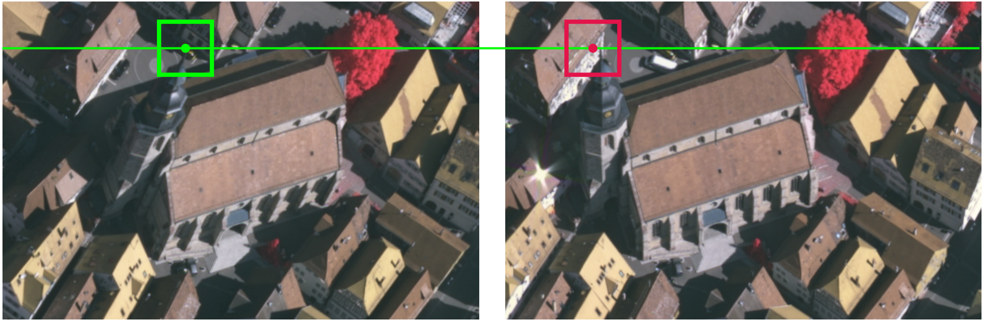
Dense Image Matching (Stereo)

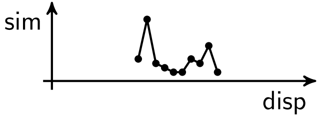
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



Dense Image Matching (Stereo)

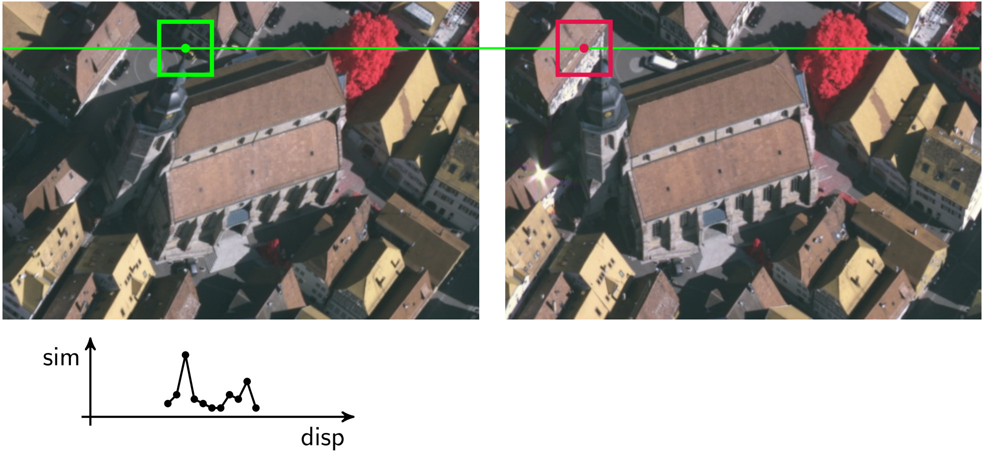
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image





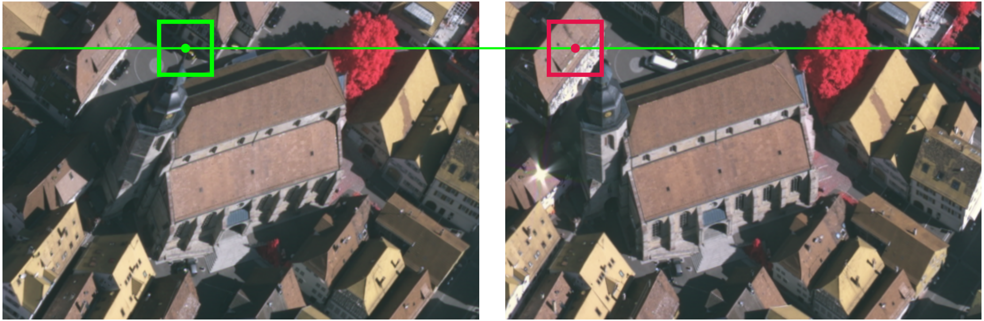
Dense Image Matching (Stereo)

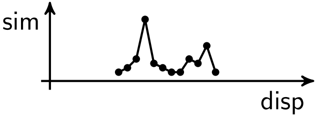
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



Dense Image Matching (Stereo)

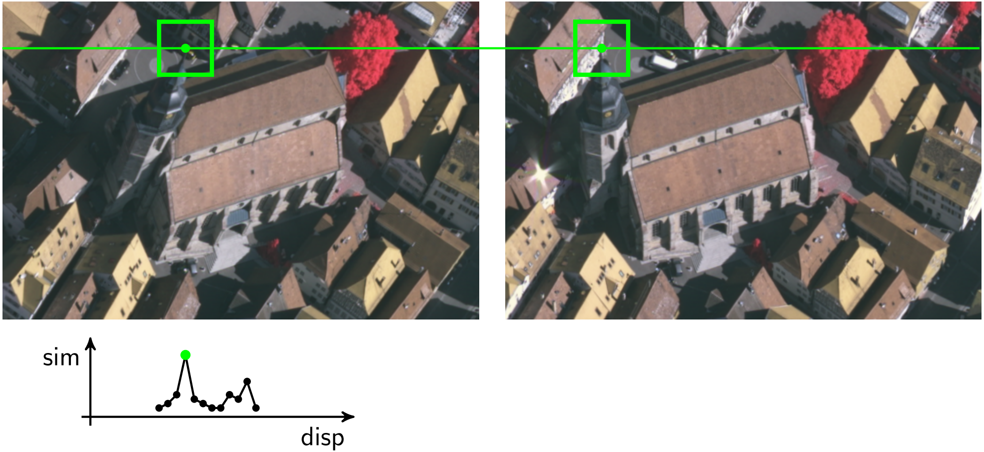
- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image





Dense Image Matching (Stereo)

- **Input**: Two images from a calibrated camera pair
  $\Rightarrow$ Epipolar lines correspond to image rows
- **Taks**: For each pixel in the left image find the corresponding pixel in the right image



CNN-CRF Model

I_0

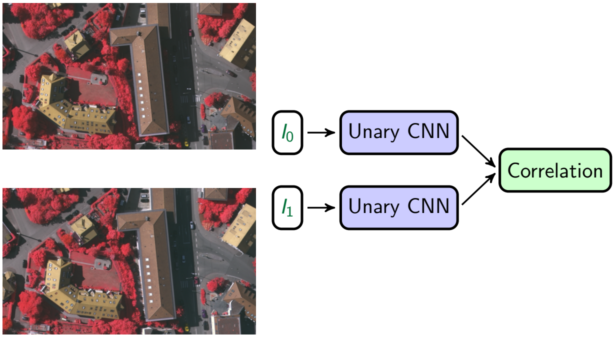
I_1

CNN-CRF Model

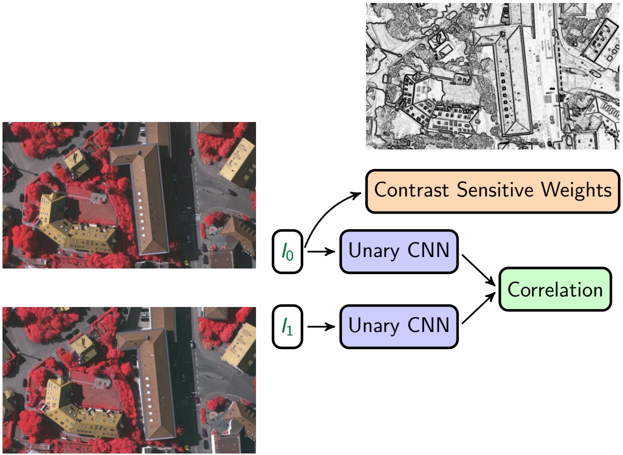
$I_0 \rightarrow$ Unary CNN

$I_1 \rightarrow$ Unary CNN

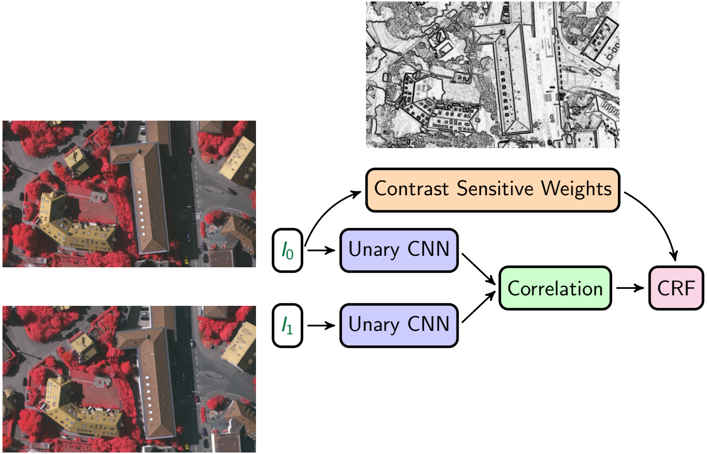
CNN-CRF Model



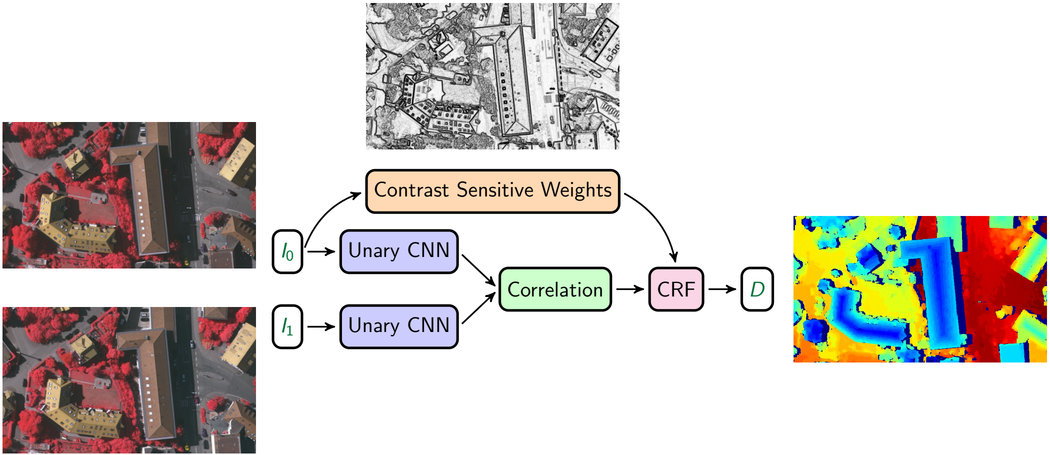
CNN-CRF Model



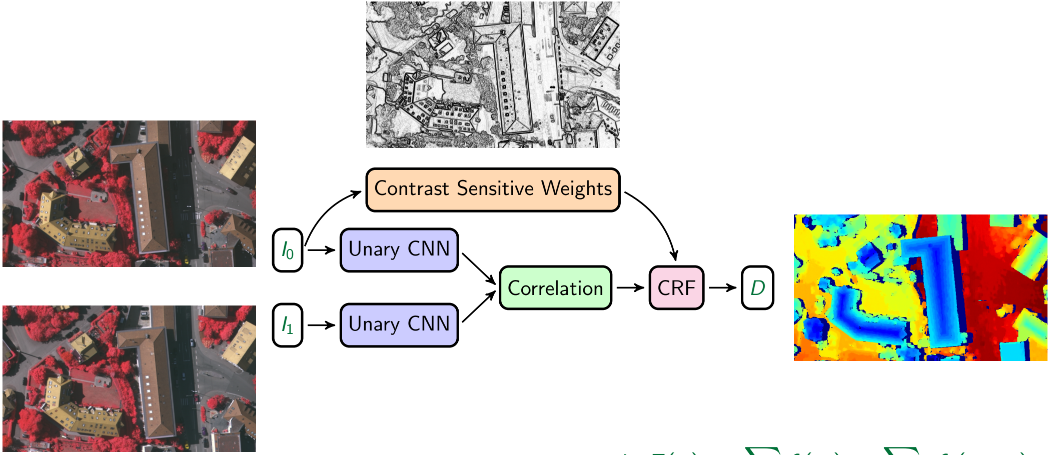
CNN-CRF Model



CNN-CRF Model



CNN-CRF Model



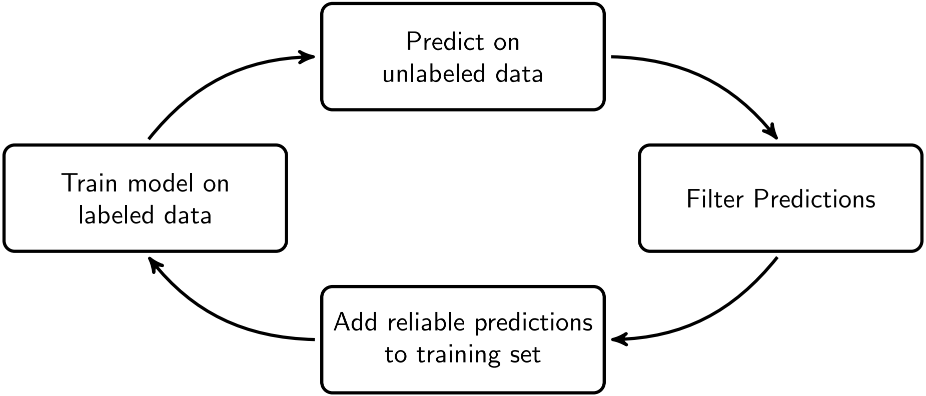
- Minimize total energy consisting of data term and smoothness term

$$\min_{x \in \mathcal{L}} E(x) := \underbrace{\sum_{i \in \mathcal{V}} f_i(x_i)}_{\text{Data term}} + \underbrace{\sum_{i \sim j \in \mathcal{E}} f_{ij}(x_i, x_j)}_{\text{Smoothness term}}$$

Self-Supervised Learning for Stereo

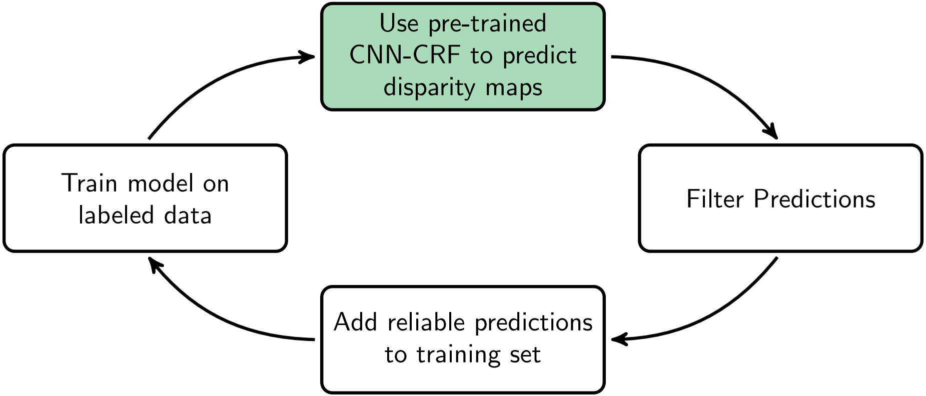
Self-Supervised Learning for Stereo

- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



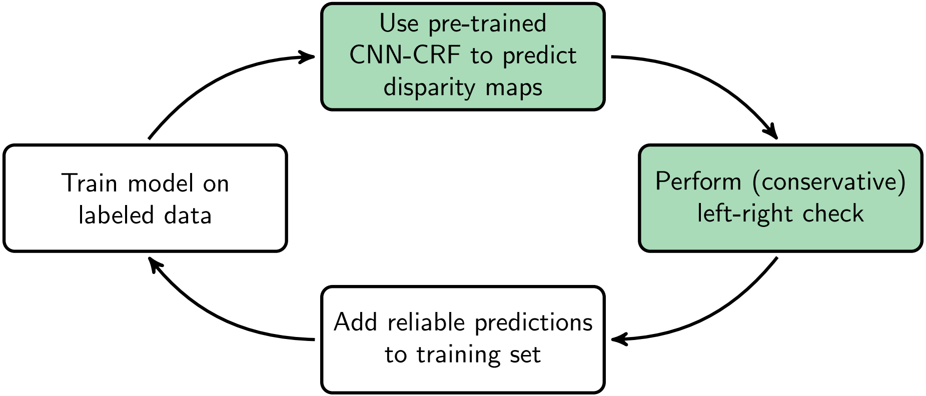
Self-Supervised Learning for Stereo

- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



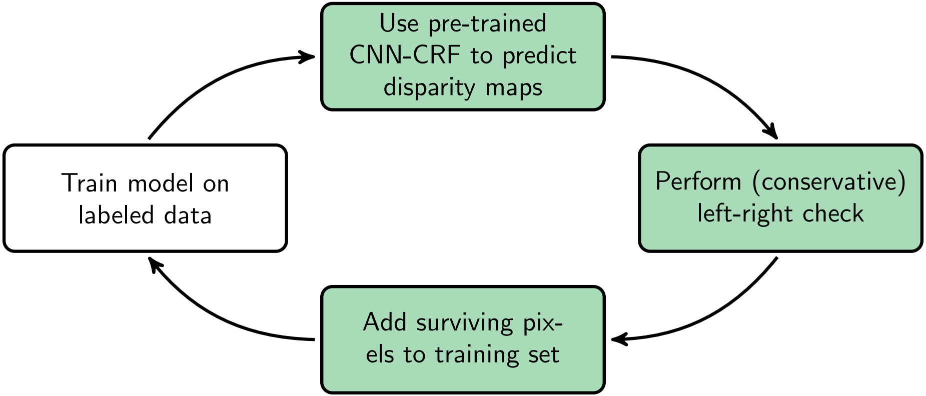
Self-Supervised Learning for Stereo

- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



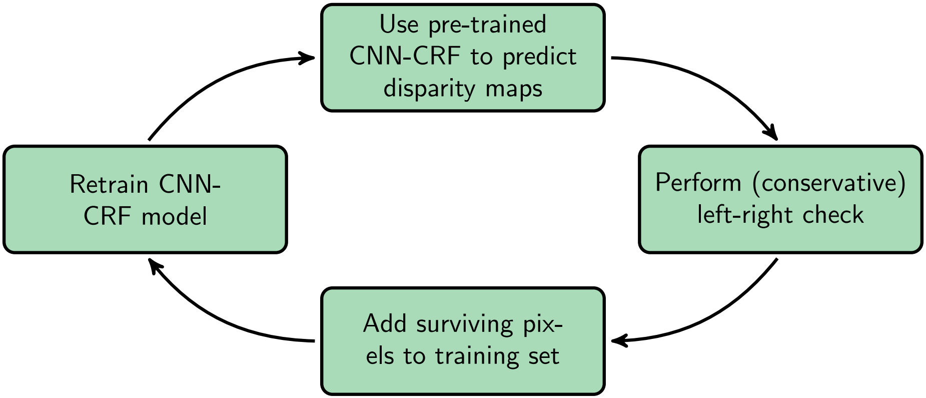
Self-Supervised Learning for Stereo

- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



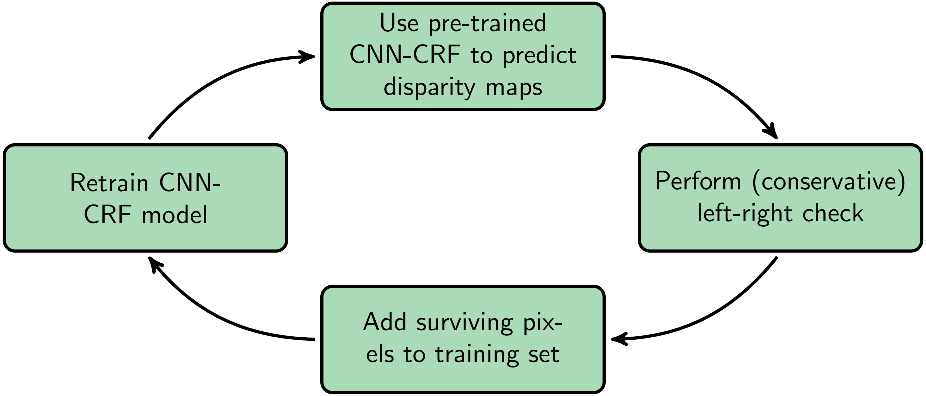
Self-Supervised Learning for Stereo

- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



Self-Supervised Learning for Stereo

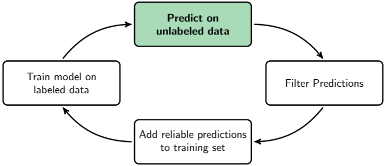
- Recall the *Self-Supervised Learning Circle*
- Transform the general circle to stereo



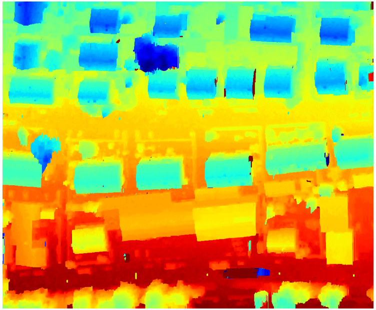
- Note: The process can be repeated

Self-Learning Approach

- Use CNN-CRF pre-trained on Middlebury
- Predict on Vaihingen dataset
- Note: We expect some outliers due to a completely different domain



Left/right input

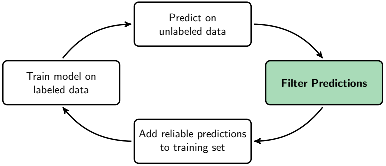


Unfiltered CNN-CRF Prediction

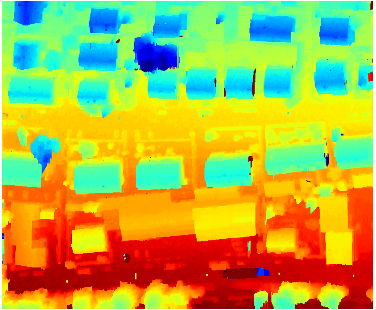
Self-Learning Approach

- Perform (conservative) left-right consistency check
- A pixel x survives the left-right consistency check if

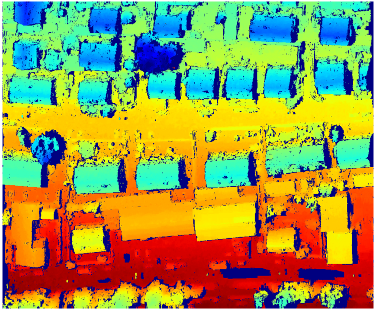
$$|d_l(x) + d_r(x + d_l(x))| < \epsilon$$



- Occluded pixels and wrong matches are filtered out



Unfiltered



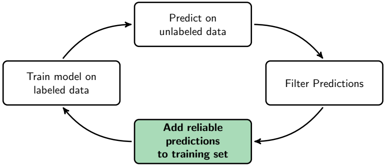
Filtered

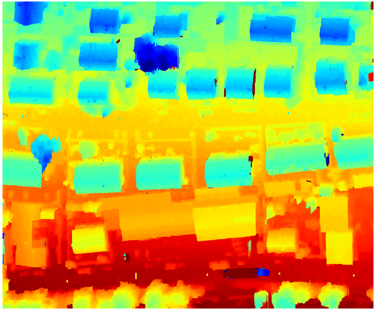
Self-Learning Approach

- Perform (conservative) left-right consistency check
- A pixel x survives the left-right consistency check if

$$|d_l(x) + d_r(x + d_l(x))| < \epsilon$$

- Occluded pixels and wrong matches are filtered out

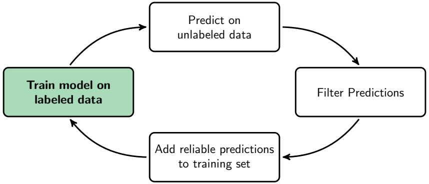




Unfiltered

Mask

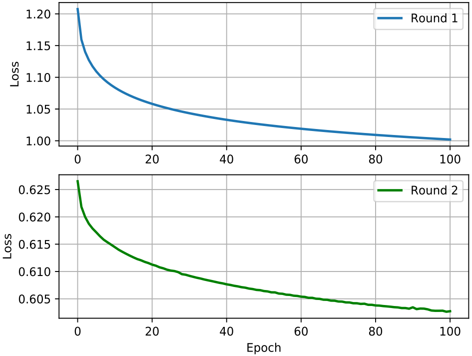
Self-Learning Approach



- Train unary term of CNN-CRF model
- Use the filtered disparity maps as ground-truth
- Maximum-likelihood training

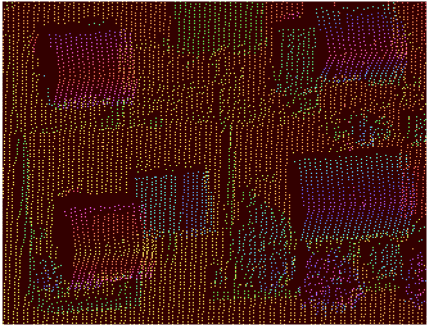
$$\min_{\theta} L(f(\theta), f^*) := -\sum_{i \in \Omega} \log f_{i,d^*}(\theta)$$

- Adam Optimizer
  - Learn-rate is 10^{-4}
  - 100 epochs
- Two rounds of self-learning

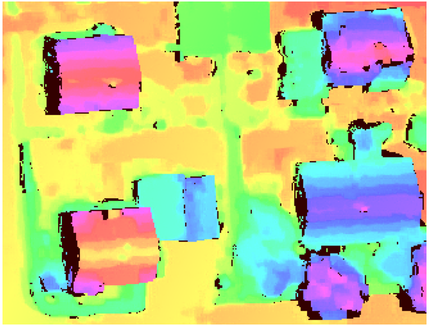


Evaluation

- ISPRS Vaihingen dataset
  - 20 aerial images with resolution 7680×13824
  - 250 million reconstructed points
  - 3D laser data is reference data
- Compare the predicted height with the laser height



Mapped laser reference data



Predicted height

Evaluation

- **Recall**: Percentage of points reconstructed with our approach relative to the points reconstructed by the laser

$$\mathrm{Rec} = \frac{|\mathcal{P}_L \cap \mathcal{P}_S|}{|\mathcal{P}_L|}$$

$\Rightarrow$ can be interpreted as completeness ratio

Evaluation

- **Recall**: Percentage of points reconstructed with our approach relative to the points reconstructed by the laser

$$\mathrm{Rec} = \frac{|\mathcal{P}_L \cap \mathcal{P}_S|}{|\mathcal{P}_L|}$$

$\Rightarrow$ can be interpreted as completeness ratio

- **Accuracy**: Percentage of points within a defined 3D distance d between $\mathcal{P}_S$ and $\mathcal{P}_L$

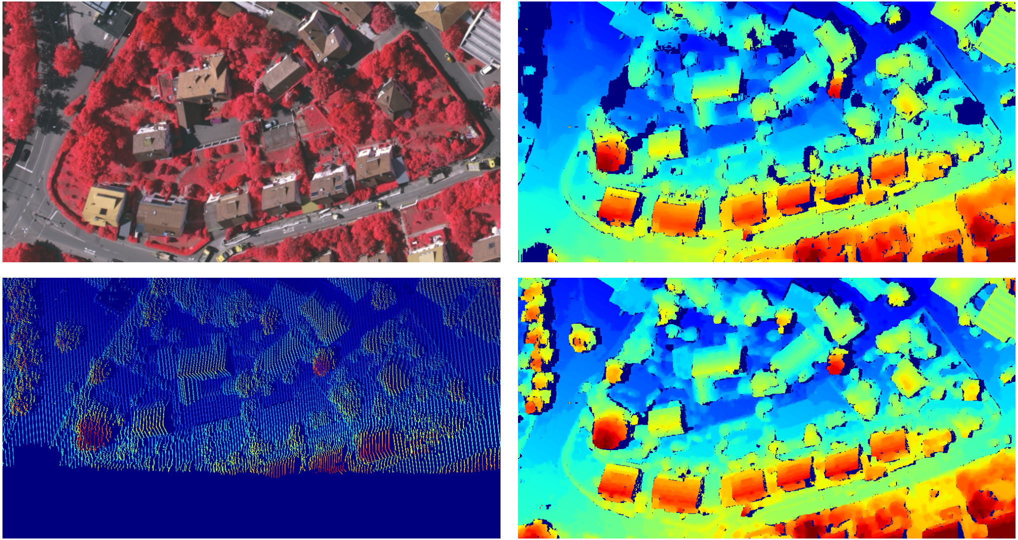
$$\begin{aligned}
\mathrm{Acc}_d(\mathcal{P}_L, \mathcal{P}_S) &= \frac{\sum_{i=1}^{|\mathcal{P}_L \cap \mathcal{P}_S|} \delta_d(\mathcal{P}_L(i), \mathcal{P}_S(i))}{|\mathcal{P}_L \cap \mathcal{P}_S|} \\
\delta_d(x, y) &= \begin{cases} 1 & \text{if } \mathrm{dist}(x, y) \leq d \\ 0 & \text{else} \end{cases}
\end{aligned}$$

Evaluation

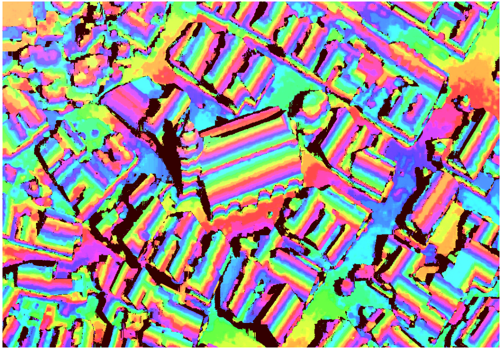
Model	**Recall [%]**	**Accuracy [%]** 0.3m	0.5m	1m
SGM	76.0	52.5	69.8	86.7
Pt-Net	87.7	62.9	76.4	87.1
Training 1	92.1	**65.2**	78.6	88.9
Training 2	**92.4**	64.5	**78.7**	**89.3**

- One disparity corresponds to a 3D distance of 0.55 to 0.72 meters
- Each training iteration increases recall and overall accuracy
- Increase in recall of 16.4 percent points
- Increase of accuracy between 2.6 and 12.7 percent points
- Note: The 3D laser reference data contains a small amount of outliers

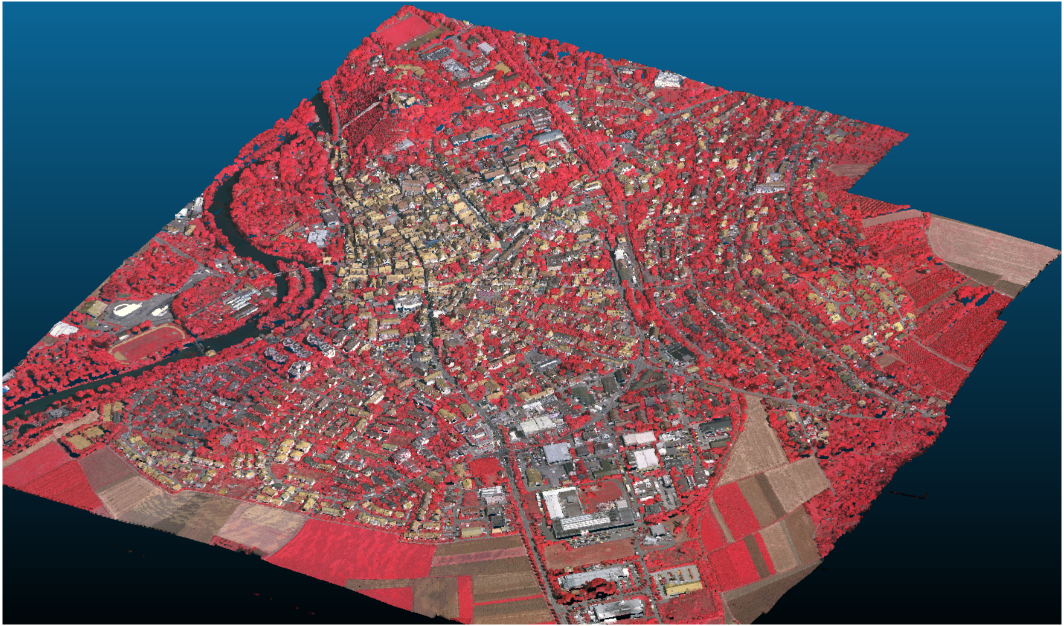
Evaluation



Evaluation



Evaluation



Conclusion

- Practical approach bridging the gap between learning based approaches and classical energy based models
- Self-supervised learning
  - enables deep learning for stereo without labeled ground truth
  - improves accuracy
  - leads to significantly denser reconstructions

Conclusion

- Practical approach bridging the gap between learning based approaches and classical energy based models
- Self-supervised learning
  - enables deep learning for stereo without labeled ground truth
  - improves accuracy
  - leads to significantly denser reconstructions

Thank you for your attention!