A Deep Learning Approach to Camera Pose Estimation

Federico Izzo Francesco Bozzo

University of Trento

January 12, 2022

Introduction

With this work we are going to present:

- the exploration of multiple dataset generation techniques;
- the COLMAP reconstruction of Povo 1 second floor;
- the development of relative and absolute pose estimation models;
- the fine-tuning of absolute pose estimation models;
- the post-processing of the model outputs;
- the model deployment using a FastAPI web-server.

Camera pose estimation

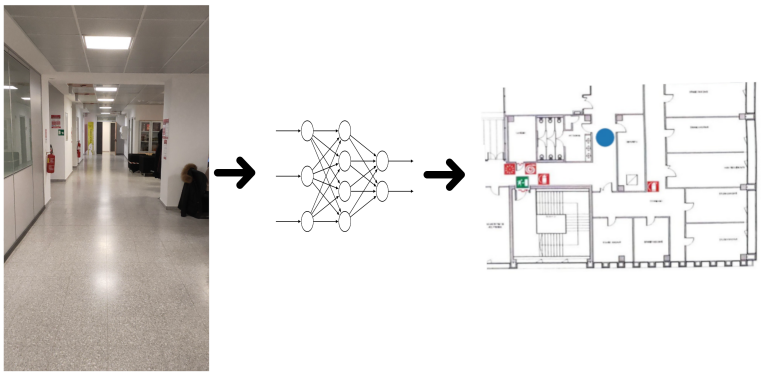

Figure: Camera pose estimation example.

The **camera pose estimation** E is a task that, given an **image** I_c acquired with a camera, processes it and returns a pair of **position** x_c and **orientation** q_c.

$$E(I_c) = (x_c, q_c)$$

Dataset generation tested approaches

Four different techniques have been tried during the project:

- **IMU sensors**: not enough accurate;
- **digital video**: not available in the university environment;
- **motion capture system**: not associable with a recorder video;
- **strcuture from motion**: usable in the context of this project

COLMAP

COLMAP is **structure from motion** tool that allows ot reconstruct 3D representations of an environment using images of it. (non so che immagine mettere qui)

Povo 1 second floor dataset

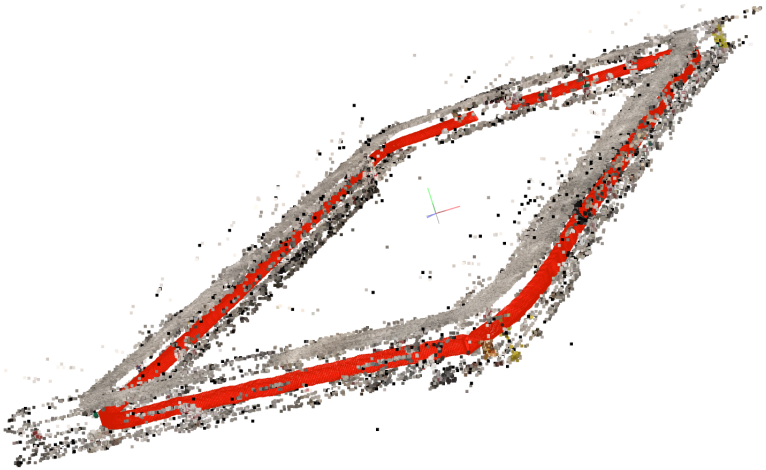

Figure: Trajectory and extracted features of Povo 2

After many attempts of environment reconstructions with COLMAP the final labeled dataset of **Povo 2 second floor** was created. **Red** lines is the **trajectory** and other points are **features**.

Coordinate Reference System

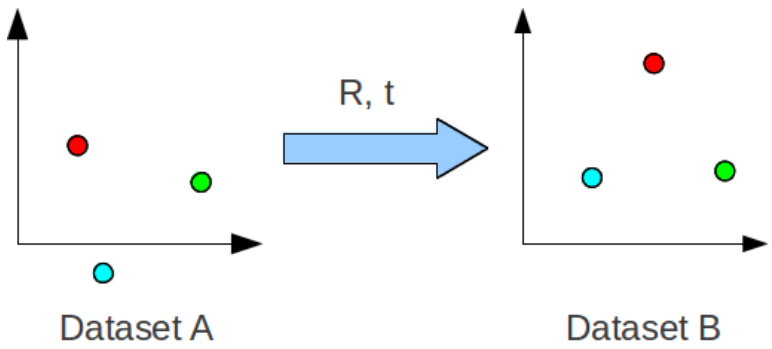

Figure: Rigid transform.

The **Coordinate Reference System** chosen by the reconstruction of COLMAP may not concide. It is required to apply some **transformations** to poses extracted from COLMAP model:

1. **scale**;
2. **translate**;
3. **rotate**;

The MeNet Model for Relative Pose Estimation

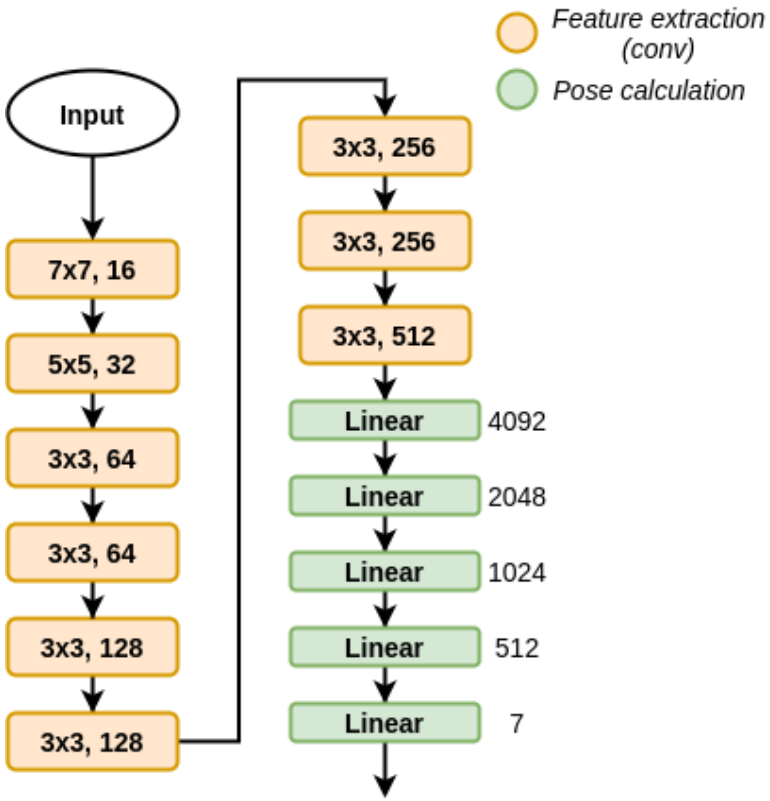

Figure: MeNet model architecture.

The **MeNet** model is targeted for **relative pose estimation**. The input of the network consists in a stack of two images: the goal is to estimate the relative pose of the second image with respect to the first one.

The PoseNet Model for Absolute Pose Estimation

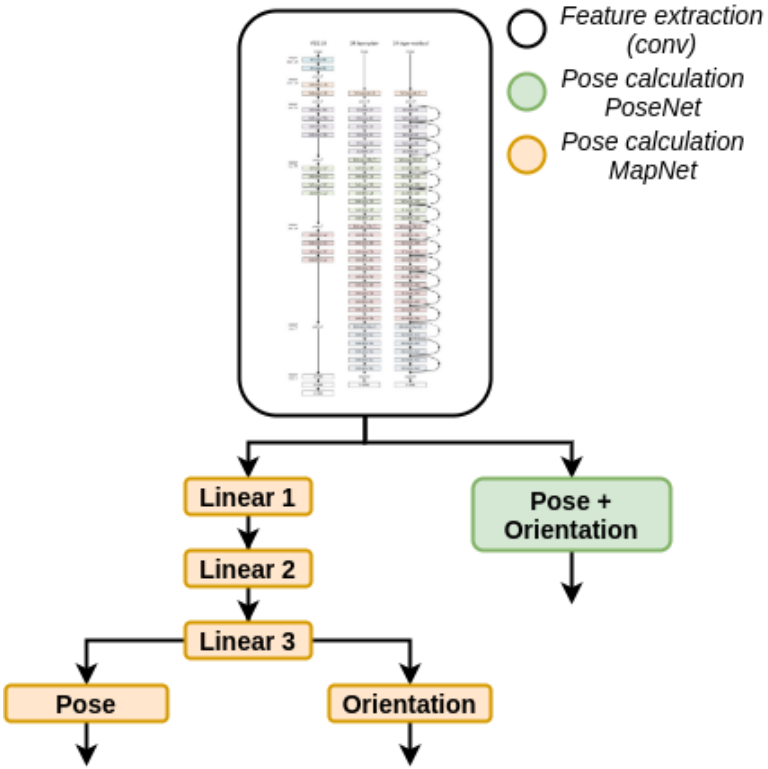

Figure: PoseNet model architecture.

The **PoseNet** model for absolute pose estimation is made up by two components:

- feature extraction through a sequence of convolutional layers (*backend*);
- pose regression on the extracted features using linear layers.

The PoseNet losses

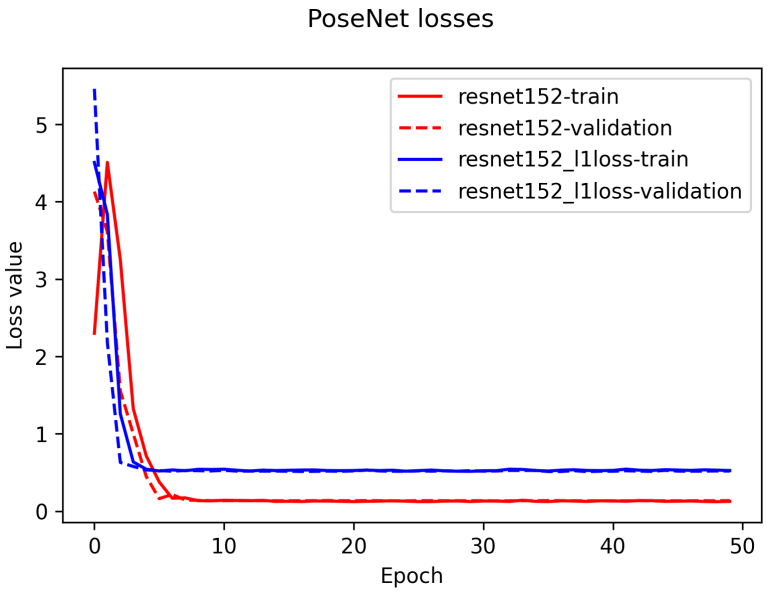

Figure: PoseNet losses

Non ho idea di come far stare la tabella

$$Loss(w) = \frac{1}{N}\sum_{i=1}^{N} h(P^i, \hat{P}^i) + \alpha h(Q^i, \hat{Q}^i) \quad (1)$$

The PoseNet losses

Figure: PoseNet predictions.

The predictions of the PoseNet
(non so cosa scrivere)

The MapNet Model for Absolute Pose Estimation

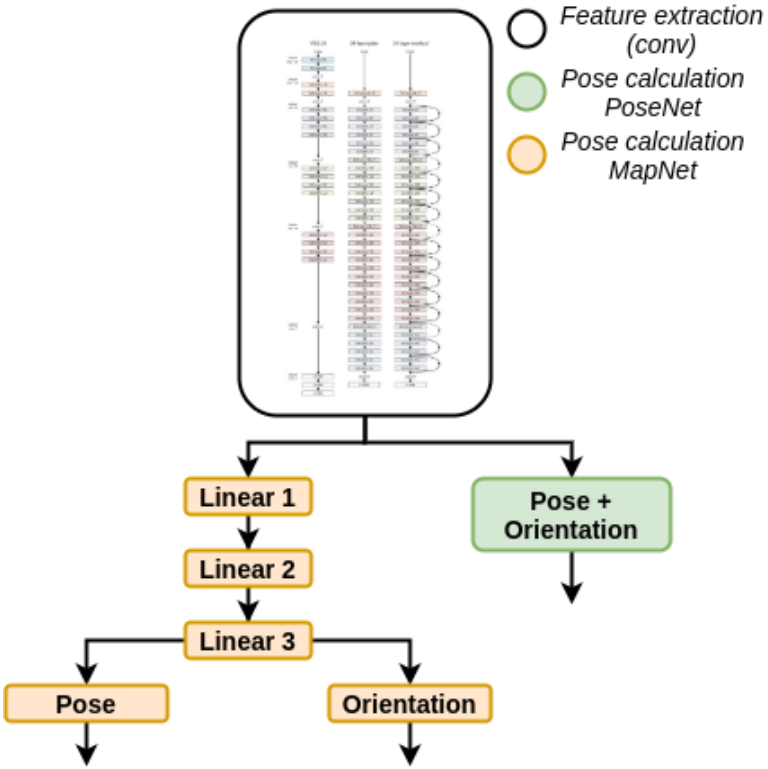

Figure: MapNet model architecture.

The **MapNet** model for absolute pose estimation represents an evolution of the PoseNet model with improvements:

- increase the number of final linear layers;
- penalize both absolute and relative errors in the loss.

The MapNet losses

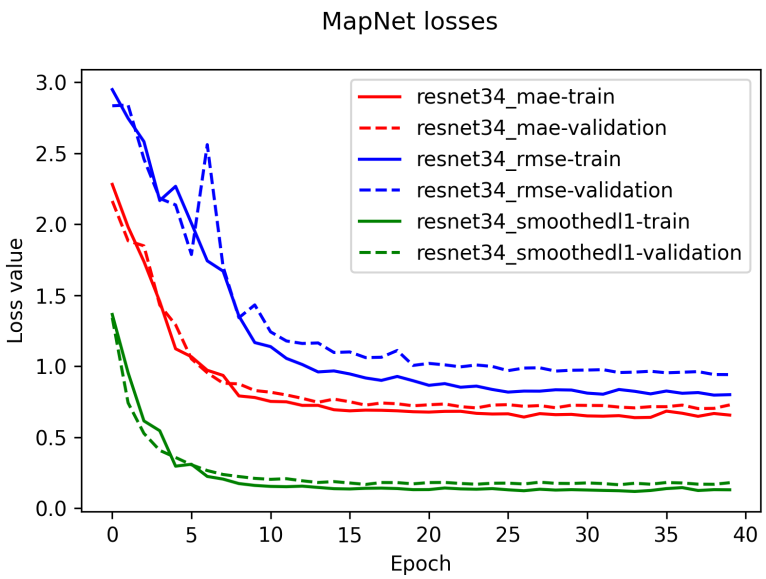

Figure: MapNet losses

Non ho idea di come far stare la tabella

$$L_{\mathcal{D}}(\Theta) = \sum_{i=1}^{|\mathcal{D}|} h([\hat{P}^i \hat{Q}^i], [P^i Q^i])$$

$$+ \alpha \sum_{i,j=1, i \neq j}^{|\mathcal{D}|} h([\hat{P}^{ij} \hat{Q}^{ij}], [P^{ij} Q^{ij}]) \quad (2)$$

The MapNet losses

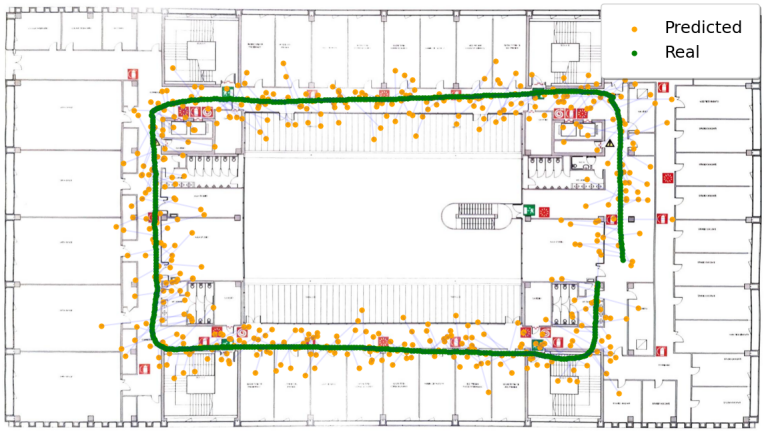

Figure: MapNet predictions.

The predictions of the MapNet
(non so cosa scrivere)

Post-processing algorithm

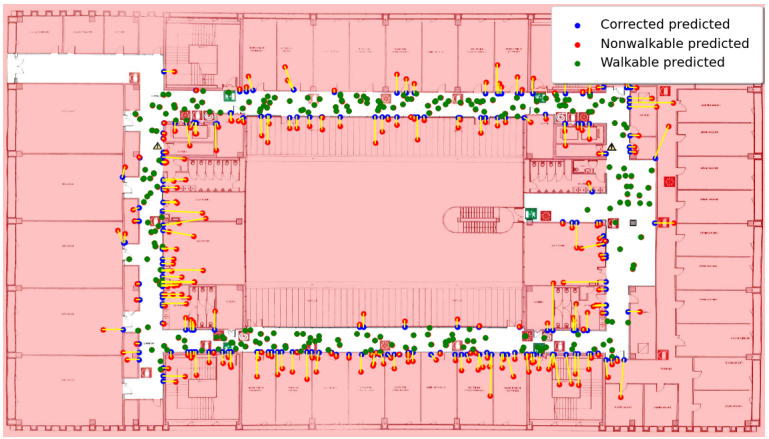

Figure: Prediction map after post-processing procedure.

Post-process technique base on Euclidean distance in order to fix non walkable predictions.

The MapNet Model for Absolute Pose Estimation

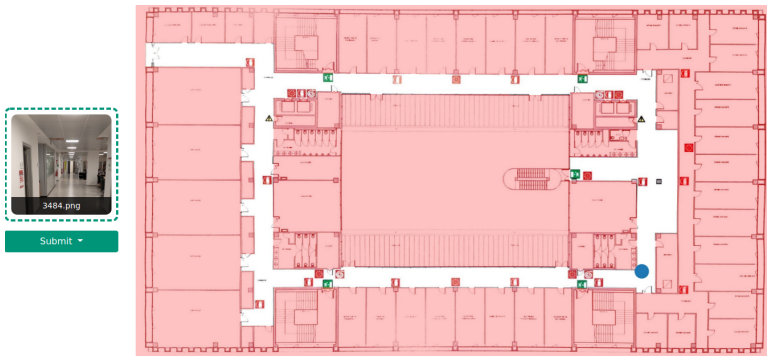

Figure: FastAPI dashboard that serves the model.

TODO

Conclusions

To summarize, the final results presented in this work are:

- the exploration of multiple dataset generation techniques;
- the COLMAP reconstruction of Povo 1 second floor;
- the development of relative and absolute pose estimation models;
- the fine-tuning of absolute pose estimation models;
- the post-processing of the model outputs;
- the model deployment using a FastAPI web-server.