OrienterNet: Visual Localization in 2D Public Maps with Neural Matching

Paul-Edouard Sarlin¹ Daniel DeTone² Tsun-Yi Yang² Armen Avetisyan² Julian Straub² Tomasz Malisiewicz² Samuel Rota Bulo² Richard Newcombe² Peter Kontschieder² Vasileios Balntas²

¹ ETH Zurich  ² Meta Reality Labs

Abstract

Humans can orient themselves in their 3D environments using simple 2D maps. Differently, algorithms for visual localization mostly rely on complex 3D point clouds that are expensive to build, store, and maintain over time. We bridge this gap by introducing OrienterNet, the first deep neural network that can localize an image with sub-meter accuracy using the same 2D semantic maps that humans use. OrienterNet estimates the location and orientation of a query image by matching a neural Bird’s-Eye View with open and globally available maps from OpenStreetMap, enabling anyone to localize anywhere such maps are available. OrienterNet is supervised only by camera poses but learns to perform semantic matching with a wide range of map elements in an end-to-end manner. To enable this, we introduce a large crowd-sourced dataset of images captured across 12 cities from the diverse viewpoints of cars, bikes, and pedestrians. OrienterNet generalizes to new datasets and pushes the state of the art in both robotics and AR scenarios. The code is available at github.com/facebookresearch/OrienterNet.

1. Introduction

As humans, we intuitively understand the relationship between what we see and what is shown on a map of the scene we are in. When lost in an unknown area, we can accurately pinpoint our location by carefully comparing the map with our surroundings using distinct geographic features.

Yet, algorithms for accurate visual localization are typically complex, as they rely on image matching and require detailed 3D point clouds and visual descriptors [18,31,38,39,53,57,60]. Building 3D maps with LiDAR or photogrammetry [2,22,43,61,67] is expensive at world scale and requires costly, freshly-updated data to capture temporal changes in visual appearance. 3D maps are also expensive to store, as they are orders of magnitude larger than basic 2D maps. This prevents executing localization on-device and usually requires costly cloud infrastructure. Spatial localization is thus a serious bottleneck for the large-scale deployment of robotics and augmented reality devices. This disconnect between the localization paradigms of humans and machines leads to the important research question of How can we teach machines to localize from basic 2D maps like humans do?

This paper introduces the first approach that can localize single images and image sequences with sub-meter accuracy given the same maps that humans use. These planimetric maps encode only the location and coarse 2D shape of few important objects but not their appearance nor height. Such maps are extremely compact, up to 10^2 times smaller in size than 3D maps, and can thus be stored on mobile devices and used for on-device localization within large areas. We demonstrate these capabilities with OpenStreetMap (OSM) [46], an openly accessible and community-maintained world map, enabling anyone to localize anywhere for free. This solution does not require building and maintaining costly 3D maps over time nor collecting potentially sensitive mapping data.

Concretely, our algorithm estimates the 3-DoF pose, as position and heading, of a calibrated image in a 2D map. The estimate is probabilistic and can therefore be fused with an inaccurate GPS prior or across multiple views from a multi-camera rig or image sequences. The resulting solution is significantly more accurate than consumer-grade GPS sensors and reaches accuracy levels closer to the traditional pipelines based on feature matching [57,60].
work that mimics the way humans orient themselves in their environment when looking at maps, i.e., by matching the metric 2D map with a mental map derived from visual observations [37, 45]. OrienterNet learns to compare visual and semantic data in an end-to-end manner, supervised by camera poses only. This yields accurate pose estimates by leveraging the high diversity of semantic classes exposed by OSM, from roads and buildings to objects like benches and trash cans. OrienterNet is also fast and highly interpretable. We train a single model that generalizes well to previously-unseen cities and across images taken by various cameras from diverse viewpoints – such as car-, bike- or head-mounted, pro or consumer cameras. Key to these capabilities is a new, large-scale training dataset of images crowd-sourced from cities around the world via the Mapillary platform.

Our experiments show that OrienterNet substantially outperforms previous works on localization in driving scenarios and vastly improves its accuracy in AR use cases when applied to data recorded by Aria glasses. We believe that our approach constitutes a significant step towards continuous, large scale, on-device localization for AR and robotics.

2. Related work

We can localize an image in the world using several types of map representations: 3D maps built from ground images, 2D overhead satellite images, or simpler planimetric maps from OpenStreetMap. Table 1 summarizes their differences.

### Mapping with ground-level images

<table>
<thead>
<tr>
<th>Map type</th>
<th>SFM SLAM</th>
<th>Satellite images</th>
<th>OpenStreetMap (our work)</th>
</tr>
</thead>
<tbody>
<tr>
<td>What?</td>
<td>3D points +features</td>
<td>pixel intensity</td>
<td>polygons, lines, points</td>
</tr>
<tr>
<td>Explicit geometry?</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Visual appearance?</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Freely available</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Storage for 1 km²</td>
<td>42 GB</td>
<td>75 MB</td>
<td>4.8 MB</td>
</tr>
<tr>
<td>Size reduction vs SFM</td>
<td>-</td>
<td>550×</td>
<td>8800×</td>
</tr>
</tbody>
</table>

Table 1. Types of maps for visual localization. Planimetric maps from OpenStreetMap consist of polygons and lines with metadata. They are publicly available for free and do not store sensitive appearance information, as opposed to satellite images and 3D maps built with SFM. They are also compact: a large area can be downloaded and stored on a mobile device. We show that they encode sufficient geometric information for accurate 3-DoF localization.

2. Related work

We can localize an image in the world using several types of map representations: 3D maps built from ground images, 2D overhead satellite images, or simpler planimetric maps from OpenStreetMap. Table 1 summarizes their differences.

### Mapping with ground-level images

- **What?**
  - SFM: 3D points + features
  - Satellite images: pixel intensity
  - OpenStreetMap (our work): polygons, lines, points

- **Explicit geometry?**
  - SFM: ✓
  - Satellite images: ✓
  - OpenStreetMap (our work): ✓

- **Visual appearance?**
  - SFM: ✓
  - Satellite images: ✓
  - OpenStreetMap (our work): ✓

- **Freely available**
  - SFM: ✓
  - Satellite images: ✓
  - OpenStreetMap (our work): ✓

- **Storage for 1 km²**
  - SFM: 42 GB
  - Satellite images: 75 MB
  - OpenStreetMap (our work): 4.8 MB

- **Size reduction vs SFM**
  - SFM: -
  - Satellite images: 550×
  - OpenStreetMap (our work): 8800×

Our approach, called OrienterNet, is a deep neural network that mimics the way humans orient themselves in their environment when looking at maps, i.e., by matching the metric 2D map with a mental map derived from visual observations [37, 45]. OrienterNet learns to compare visual and semantic data in an end-to-end manner, supervised by camera poses only. This yields accurate pose estimates by leveraging the high diversity of semantic classes exposed by OSM, from roads and buildings to objects like benches and trash cans. OrienterNet is also fast and highly interpretable. We train a single model that generalizes well to previously-unseen cities and across images taken by various cameras from diverse viewpoints – such as car-, bike- or head-mounted, pro or consumer cameras. Key to these capabilities is a new, large-scale training dataset of images crowd-sourced from cities around the world via the Mapillary platform.

Our experiments show that OrienterNet substantially outperforms previous works on localization in driving scenarios and vastly improves its accuracy in AR use cases when applied to data recorded by Aria glasses. We believe that our approach constitutes a significant step towards continuous, large scale, on-device localization for AR and robotics.

Effective surface representation is the most common approach to date. Place recognition via image retrieval provides a coarse localization given a set of reference images [4, 23, 32, 72]. To estimate centimeter-accurate 6-DoF poses, algorithms based on feature matching require 3D maps [31, 39, 57, 60]. These are composed of sparse point clouds, which are commonly built with Structure-from-Motion (SfM) [2, 22, 36, 43, 61, 67] from sparse points matched across multiple views [9, 58, 53]. The pose of a new query image is estimated by a geometric solver [10, 26, 33] from correspondences with the map. While some works [70, 79] leverage additional sensor inputs, such as a coarse GPS prior location, gravity direction, and camera height, recent localization systems are highly accurate and robust mostly thanks to learned features [18, 19, 51, 58, 73].

This however involves 3D maps with a large memory footprint as they store dense 3D point clouds with high-dimensional visual descriptors. There is also a high risk of leaking personal data into the map. To mitigate this, some works attempt to compress the maps [12, 13, 39] or use privacy-preserving representations for the scene appearance [20, 44, 81] or geometry [68, 69]. These however either degrade the accuracy significantly or are easily reverted [50].

**Localization with overhead imagery** reduces the problem to estimating a 3-DoF pose by assuming that the world is mostly planar and that the gravity direction is often given by ubiquitous onboard inertial sensors. A large body of work focuses on cross-view ground-to-satellite localization. While more compact than 3D maps, satellite images are expensive to capture, generally not free, and still heavy to store at high resolution. Most approaches only estimate a coarse position through patch retrieval [30, 63, 65, 82]. In addition, works that estimate an orientation are not accurate [62, 64, 77], yielding errors of over several meters.
Other works rely on sensors that directly provide 3D metric information, such as 2D intensity maps from Lidar [8, 40] or radar [7, 71]. They all perform template matching between 2D map and sensor overhead views, which is both accurate and robust, but require expensive specialized sensors, unsuitable for consumer AR applications. Our work shows how monocular visual priors can substitute such sensors to perform template matching from images only.

**Planimetric maps** discard any appearance and height information to retain only the 2D location, shape and type of map elements. OSM is a popular platform for such maps as it is free and available globally. Given a query area, its open API exposes a list of geographic features as polygons with metadata, including fine-grained semantic information with over a thousand different object types. Past works however design detectors for a single or few semantic classes, which lacks robustness. These include building outlines [5, 6, 15–17, 74, 75], road contours [21, 54] or intersections [41, 47, 78], lane markings [25, 48], street furniture [14, 76], or even text [27].

Recent works leverage more cues by computing richer representations from map tiles using end-to-end deep networks [56, 80]. They estimate only a coarse position as they retrieve map tiles with global image descriptors. In indoor scenes, floor plans are common planimetric maps used by existing works [28, 42]. They require height or visibility information that is typically not available for outdoor spaces. Our approach yields a significant step up in accuracy and robustness over all previous works by combining the constraints of projective geometry with the expressivity of end-to-end learning, leveraging all semantic classes available in OSM.

3. Localizing single images in 2D maps

**Problem formulation:** In a typical localization scenario, we aim to estimate the absolute 6-DoF pose of an image in the world. Under realistic assumptions, we reduce this problem to estimating a 3-DoF pose \( \xi = (x, y, \theta) \) consisting of a location \((x, y) \in \mathbb{R}^2\) and heading angle \(\theta \in (-\pi, \pi]\). Here we consider a topocentric coordinate system whose \(x\text{-}\text{y}-z\) axes correspond to the East-North-vertical directions.

First, we can easily assume to know the direction of the gravity, an information that humans naturally possess from their inner ear and that can be estimated by the inertial unit embedded in most devices. We also observe that our world is mostly planar and that the motion of people and objects in outdoor spaces is mostly restricted to 2D surface. The precise height of the camera can always be estimated as the distance to the ground in a local SLAM reconstruction.

**Inputs:** We consider an image \(I\) with known pinhole camera calibration. The image is rectified via a homography computed from the known gravity such that its roll and tilt are zero – its principal axis is then horizontal. We are also given a coarse location prior \(\xi_{\text{prior}}\). This can be a noisy GPS position or a previous localization estimate and can be off by over 20 meters. This is a realistic assumption for a consumer-grade sensor in a multi-path environment like a urban canyon.

The map data is queried from OSM as a square area centered around \(\xi_{\text{prior}}\) and whose size depends on how noisy the prior is. The data consists of a collection of polygons, lines, and points, each of a given semantic class and whose coordinates are given in the same local reference frame.

**Overview – Figure 2:** OrienterNet consists of three modules: 1) The image-CNN extracts semantic features from the image and lifts them to an orthographic Bird’s-Eye View (BEV) representation \(T\) by inferring the 3D structure of the scene. 2) The OSM map is encoded by the map-CNN into a neural map \(F\) that embeds semantic and geometric information. 3) We estimate a probability distribution over camera poses \(\xi\) by exhaustively matching the BEV against the map.

3.1. Neural Bird’s-Eye View inference

**Overview:** From a single image \(I\), we infer a BEV representation \(T \in \mathbb{R}^{L \times D \times N}\) distributed on a \(L \times D\) grid aligned with the camera frustum and composed of \(N\)-dimensional features. Each feature on the grid is assigned a confidence, yielding a matrix \(C \in [0, 1]^{L \times D}\). This BEV is akin to a mental map that humans infer from their environment when self-localizing in an overhead map [37, 45].

Cross-modal matching between the image and the map requires extracting semantic information from visual cues. It has been shown that monocular depth estimation can rely on semantic cues [3] and that both tasks have a beneficial synergy [29, 34]. We thus rely on monocular inference to lift semantic features to the BEV space. Following past works that tackle semantic tasks [49, 52, 55], we obtain the neural BEV in two steps: i) we transfer image features to a polar representation by mapping image columns to polar rays, and ii) we resample the polar grid into a Cartesian grid (Fig. 3).

**Polar representation:** A CNN \(\Phi_{\text{image}}\) first extracts a \(U \times V\) feature map \(X \in \mathbb{R}^{U \times V \times N}\) from the image. We consider \(D\) depth planes sampled in front of the camera with a regular interval \(\Delta_d\), i.e., with values \(\{i \cdot \Delta_d\mid i \in \{1, \ldots, D\}\}\). Since the image is gravity-aligned, each of the \(U\) columns in \(X\) corresponds to a vertical plane in the 3D space. We thus map each column to a ray in the \(U \times D\) polar representa-
We then obtain the distribution \( \alpha \) with identical models and directly regress \( \text{val} \). We first rasterize the areas, lines, and points. The resulting feature grid is then processed by a small CNN from \( \Phi_{\text{BEV}} \) that outputs the neural BEV map. It is then encoded into the neural map \( \Phi_{\text{map}} \), which extracts geometric features useful for localization. \( \Phi_{\text{map}} \) is not normalized as we let \( \Phi_{\text{map}} \) modulate its norm as importance weight in the matching. Examples in Fig. 4 reveal that \( \Phi \) often looks like a distance field where we can clearly recognize distinctive features like corners or adjoining boundaries of buildings.

\( \Phi_{\text{map}} \) also predicts a unary location prior \( \Omega \in \mathbb{R}^{W \times H} \) for each cell of the map. This score reflects how likely an image is to be taken at each location. We rarely expect images to be taken in, for example, rivers or buildings.

### 3.3. Pose estimation by template matching

**Probability volume:** We estimate a discrete probability distribution over camera poses \( \xi \). This is interpretable and fully captures the uncertainty of the estimation. As such, the distribution is multimodal in ambiguous scenarios. Figure 4 shows various examples. This makes it easy to fuse the pose estimate with additional sensors like GPS. Computing this volume is tractable because the pose space has been reduced to 3 dimensions. It is discretized into each map location and \( K \) rotations sampled at regular intervals.

This yields a \( W \times H \times K \) probability volume \( P \) such that \( P(\xi | I, \text{map, } \xi_{\text{prior}}) = P(\xi) \). It is the combination of an image-map matching term \( M \) and the location prior \( \Omega \):

\[
P = \text{softmax} (M + \Omega) .
\]

\( M \) and \( \Omega \) represent image-conditioned and image-independent un-normalized log scores. \( \Omega \) is broadcasted along the rotation dimension and softmax normalizes the probability distribution.

**Image-map matching:** Exhaustively matching the neural map \( \Phi \) and the BEV \( T \) yields a score volume \( M \). Each element is computed by correlating \( \Phi \) with \( T \) transformed by the corresponding pose as

\[
M[\xi] = \frac{1}{UZ} \sum_{p \in (U \times Z)} \Phi(\xi(p)) \top (T \odot C) [p] ,
\]

where \( \xi(p) \) transforms a 2D point \( p \) from BEV to map coordinate frame. The confidence \( C \) masks the correlation to ignore some parts of the BEV space, such as occluded areas. This formulation benefits from an efficient implementation by rotating \( T \) \( K \) times and performing a single convolution as a batched multiplication in the Fourier domain [7, 8].

**Pose inference:** We estimate a single pose by maximum likelihood: \( \xi^* = \arg\max_\xi P(\xi | I, \text{map, } \xi_{\text{prior}}) \). When the distribution is mostly unimodal, we can obtain a measure of uncertainty as the covariance of \( P \) around \( \xi^* \) [7].
Figure 4. **We train a single model that generalizes well across many datasets.** OrienterNet handles different cameras, street-level viewpoints, and unseen cities and is thus suitable for both AR and robotics. Overlaid on the input maps, the single-image predictions (black arrow $\Rightarrow$) are close to the ground truth (red arrow) and more accurate than the noisy GPS (blue dot $\circ$). The model effectively leverages building corners and boundaries, crosswalks, sidewalks, road intersections, trees, and other common urban objects. Likelihood maps can be multi-modal in scenes with repeated elements (last two rows) - we show the predicted orientations at local maxima as small arrows.
4. Sequence and multi-camera localization

Single-image localization is ambiguous in locations that exhibit few distinctive semantic elements or repeated patterns. Such challenge can be disambiguated by accumulating additional cues over multiple views when their relative poses are known. These views can be either sequences of images with poses from VI SLAM or simultaneous views from a calibrated multi-camera rig. Figure 5 shows an example of such a difficult scenario disambiguated by accumulating predictions over time. Different frames constrain the pose in different directions, e.g., before and after an intersection. Fusing longer sequences yields a higher accuracy (Fig. 6).

Let us denote $\xi_i$ the unknown absolute pose of view $i$ and $\xi_{ij}$ the known relative pose from view $j$ to $i$. For an arbitrary reference view $i$, we express the joint likelihood over all single-view predictions as

$$P(\xi_i | \{I_j\}, \text{map}) = \prod_k P(\xi_i \oplus \xi_{ij} | I_j, \text{map}), \quad (6)$$

where $\oplus$ denotes the pose composition operator. This is efficiently computed by warping each probability volume $P_j$ to the reference frame $i$. We can also localize each image of a continuous stream via iterative warping and normalization, like in the classical Markov localization [11, 66].

Table 2. OrienterNet outperforms existing architectures, which include: a) map tile retrieval by matching global embeddings [56, 77], b) featuremetric refinement [62] from an initial pose, and c) OrienterNet assuming a planar scene [62] instead of inferring monocular depth. We report the position and orientation recall ($R$).

5. Training a single strong model

Supervision: OrienterNet is trained in a supervised manner from pairs of single images and ground truth (GT) poses. The architecture is differentiable and all components are trained simultaneously by back-propagation. We simply maximize the log-likelihood of the ground truth pose $\xi$: $\text{Loss} = - \log P(\xi | I, \text{map}, \xi_{\text{prior}}) = - \log P(\xi)$. The trilinear interpolation of $P$ provides sub-pixel supervision.

Training dataset: We train a single model that generalizes to unseen locations with arbitrary kinds of images. We collect images from the Mapillary platform, which exposes the camera calibration, noisy GPS measurement, and the 6-DoF pose in a global reference frame, obtained with a fusion of SfM and GPS. The resulting Mapillary Geo-Localization (MGL) dataset includes 760k images from 12 cities in Europe and the US, captured by cameras that are handheld or mounted on cars or bikes, with GT poses and OSM data. Models trained on MGL generalize well to other datasets thanks to the diversity of cameras, locations, motions, and maps. All images are publicly available under a CC-BY-SA license via the Mapillary API. We believe that this dataset will significantly facilitate research on visual geo-localization.

Implementation: $\Phi_{\text{image}}$ and $\Phi_{\text{map}}$ are U-Nets with ResNet-101 and VGG-16 encoders. $\Phi_{\text{BEV}}$ has 4 residual blocks. We use $S=32$ scale bins, $K=512$ rotations. The BEV has size $L \times D = 32 \times 32$ m with resolution $\Delta = 50$ cm. For training, we render maps $W \times H = 128 \times 128$ m centered around points randomly sampled within 32 m of the GT pose. Localizing in such maps takes 94 ms on an NVIDIA RTX 2080 GPU.
We evaluate our single model for localization in the context of both driving and AR. Figure 4 shows qualitative examples, while Fig. 5 illustrates the effectiveness of multi-frame fusion. Our experiments show that: 1) OrienterNet is more effective than existing deep networks for localization with 2D maps; 2) Planimetric maps help localize more accurately than overhead satellite imagery; 3) OrienterNet is significantly more accurate than an embedded consumer-grade GPS sensor when considering multiple views.

### 6.1. Validating design decisions

**Setup:** We evaluate the design of OrienterNet on the validation split of our MGL dataset. This ensures an identical distribution of cameras, motions, viewing conditions, and visual features as the training set. We report recall of positions and rotation errors at the three thresholds 1/3/5m and 1/3/5°.

**Comparing model architectures:** We compare OrienterNet to alternative architectures trained on the same dataset: a) Map retrieval [77] replaces the BEV inference and matching by a correlation of the neural map and with a global image embedding. We predict a rotation by considering 4 different neural maps for the N-S-E-W directions. This formulation also regresses a probability volume, and is trained identically to OrienterNet. It mimics the retrieval of densely-sampled map patches [56] but is significantly more efficient and practical. b) Featuremetric refinement [59,62] updates an initial pose by warping a satellite view to the image assuming that the scene is planar, at a fixed height, and gravity-aligned. We replace the satellite view by an OSM map tile. This formulation requires an initial orientation (during both training and testing), which we sample within 45° of the ground truth. c) OrienterNet (planar) replaces the occupancy by warping the image features with a homography as in [62].

**Analysis – Table 2:** OrienterNet is significantly more accurate than all baselines at all position and rotation thresholds. a) Map retrieval disregards any knowledge of projective geometry and performs mere recognition without any geometric constraint. b) Featuremetric refinement converges to incorrect locations when the initial pose is inaccurate. c) Inferring the 3D geometry of the scene is more effective than assuming that it is planar. This justifies our design decisions.

**Model interpretability:** We visualize in Fig. 7 multiple internal quantities that help us understand the predictions.

### 6.2. Application: autonomous driving

**Dataset:** We consider the localization in driving scenarios with the KITTI dataset [24], following the closest existing setup [62]. To evaluate the zero-shot performance, we use their Test2 split, which does not overlap with the KITTI and MGL training sets. Images are captured by cameras mounted on a car driving in urban and residential areas and have GT poses from RTK. We augment the dataset with OSM maps.

**Setup:** We compute the position error along directions perpendicular (lateral) and parallel (longitudinal) to the viewing axis [62] since the pose is generally less constrained along the road. We report the recall at 1/3/5m and 1/3/5°. The original setup [62] assumes an accurate initial pose randomly sampled within ±20m and ±10° of the GT. OrienterNet does not require such initialization but only a coarse position-only prior. For fair comparisons, we nevertheless restrict the pose space to the same interval centered around the initial pose. We render 64 m × 64 m map tiles and resize the images such that their focal length matches the median of MGL.

**Baselines:** We report approaches based on satellite maps and trained by [62] on KITTI. VIGOR [82] and DSM [64] both perform patch retrieval with global descriptors but respectively estimate an additional position offset or the orientation. We also evaluate the featuremetric refinement [59,62] and baselines based on OSM maps, described in Sec. 6.1. As each scene is visited by a single trajectory, we cannot

### Table 3. Localization in driving scenarios with the KITTI dataset.

<table>
<thead>
<tr>
<th>Map Approach</th>
<th>Training dataset</th>
<th>Lateral R@Xm 1m 3m 5m</th>
<th>Longitudinal R@Xm 1m 3m 5m</th>
<th>Orientation R@X° 1° 3° 5°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite DSM [64] KITTI</td>
<td>10.77 31.37 48.24 3.87 11.73 19.50</td>
<td>3.53 14.09 23.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIGOR [82] KITTI</td>
<td>17.38 48.20 70.79 4.07 12.52 20.14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OrienterNet refinement [62] KITTI</td>
<td>27.82 59.79 72.89 5.75 16.36 26.48</td>
<td>18.42 49.72 71.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satellite DSM [64] KITTI</td>
<td>50.83 78.10 82.22 17.75 40.32 52.40</td>
<td>31.03 66.76 76.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OrienterNet (a)</td>
<td>53.51 88.57 94.67 26.25 59.84 70.76</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OrienterNet (b)</td>
<td>79.71 97.44 98.67 55.21 95.27 99.51</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OrienterNet (c)</td>
<td>51.26 84.77 91.81 22.39 46.79 57.81</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OrienterNet (d)</td>
<td>65.91 92.76 96.54 33.07 65.18 75.15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIGOR [82]</td>
<td>17.38 48.20 70.79 5.75 16.36 26.48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSM [64]</td>
<td>27.82 59.79 72.89 5.75 16.36 26.48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MGL</td>
<td>37.47 66.24 72.89 5.94 16.88 26.97</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MGL+KITTI</td>
<td>65.91 92.76 96.54 33.07 65.18 75.15</td>
<td>77.87 97.76 100.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KITTI</td>
<td>51.26 84.77 91.81 22.39 46.79 57.81</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MGL+KITTI</td>
<td>65.91 92.76 96.54 33.07 65.18 75.15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KITTI</td>
<td>51.26 84.77 91.81 22.39 46.79 57.81</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6.3. Application: autonomous driving
We now consider the localization of head-mounted devices for AR. We evaluate approaches based on 3D maps and image matching.

**Results:** Table 3 (a-b) shows that OrienterNet outperforms all existing approaches based on both satellite and OSM maps, in all metrics. OrienterNet exhibits remarkable zero-shot capabilities as it outperforms approaches trained on KITTI itself. The evaluation also demonstrates that planimetric maps yield better localization, as refinement and refinement approaches based on them outperform those based on satellite images. The recall at 3m/3° is saturated to over 95% by fusing the predictions from sequences of only 20 seconds.

**Generalization:** Table 3 (c-d) shows that training OrienterNet solely on KITTI results in overfitting, as the dataset is too small to learn rich semantic representations. Our larger MGL dataset alleviates this issue and enables cross-modal learning with rich semantic classes. Pre-training on MGL and fine-tuning on KITTI yields the best performance.

### 6.3. Application: augmented reality

We now consider the localization of head-mounted devices for augmented reality (AR). We show that OrienterNet is more accurate than a typical embedded GPS sensor.

**Dataset:** There is no public benchmark that provides geo-aligned GT poses for images captured with AR devices in diverse outdoor spaces. We thus record our own dataset with Aria [1] glasses. It exhibits patterns typical of AR with noisy consumer-grade sensors and pedestrian viewpoints and motions. We include two locations: i) downtown Seattle, with high-rise buildings, and ii) Detroit, with city parks and lower buildings. We record several image sequences per city, all roughly following the same loop around multiple blocks. We record calibrated RGB images and GPS measurements for each and obtain relative poses and gravity direction from an offline proprietary VI SLAM system. We obtain pseudo-GT global poses by jointly optimizing all sequences based on GPS, VI constraints, and predictions of OrienterNet.

**Single-frame localization – Table 4:** OrienterNet is consistently more accurate than the GPS, which is extremely noisy in urban canyons like Seattle because of multi-path effects. The performance is however significantly lower than with driving data (Sec. 6.2), which highlights the difficulty of AR-like conditions and the need for further research.

**Multi-frame:** We now fuse multiple GPS signals or predictions of OrienterNet over the same temporal interval of 10 consecutive keyframes, using imperfect relative poses from VI SLAM. The fusion more than doubles the accuracy of OrienterNet but marginally benefits the GPS sensor because of its high, biased noise, especially in Seattle.

**Limitations:** Localizing an image or a sequence is challenging when the environments lacks distinctive elements or when they are not registered in the map. OSM may also be spatially inaccurate. Appendix A shows some failure cases.

### 7. Conclusion

OrienterNet is the first deep neural network that can localize an image with sub-meter accuracy within the same 2D planimetric maps that humans use. OrienterNet mimics the way humans orient themselves in their environment by matching the input map with a mental map derived from visual observations. Compared to large and expensive 3D maps that machines have so far relied on, such 2D maps are extremely compact and thus finally enable on-device localization within large environments. OrienterNet is based on globally and freely available maps from OpenStreetMap and can be used by anyone to localize anywhere in the world.

We contribute a large, crowd-sourced training dataset that helps the model generalize well across both driving and AR datasets. OrienterNet significantly improves over existing approaches for 3-DoF localization, pushing the state of the art by a large margin. This opens up exciting prospects for deploying power-efficient robots and AR devices that know where they are without costly cloud infrastructures.
References


[34] Dong Lao, Alex Wong, and Stefano Soatto. Does monocular depth estimation provide better pre-training than classification for semantic segmentation? *arXiv*:2203.13987, 2022. 3

[41] Wei-Chiu Ma, Shenlong Wang, Marcus A. Brubaker, Sanja Fidler, and Raquel Urtasun. Find your way by observing the sun and other semantic cues. In ICRA, 2017. 3
[53] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. ORB: An efficient alternative to SIFT or SURF. In ICCV, 2011. 1, 2
[54] Philipp Ruchti, Bastian Steder, Michael Ruhnke, and Wolfram Burgard. Localization on OpenStreetMap data using a 3d laser scanner. In ICRA, 2015. 3
[56] Noe Samano, Mengjie Zhou, and Andrew Calway. You are here: Geolocation by embedding maps and images. In ECCV, 2020. 3, 4, 6, 7, 8
[57] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In CVPR, 2019. 1, 2
[60] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In ECCV, 2012. 1, 2
[70] Linus Svärm, Olof Enqvist, Fredrik Kahl, and Magnus Oskarsson. City-scale localization for cameras with known vertical direction. TPAMI, 2017. 2


[76] Li Weng, Valérie Gouet-Brunet, and Bahman Soheilian. Semantic signatures for large-scale visual localization. *Multimedia Tools and Applications*, 2021. 3

[77] Zimin Xia, Olaf Booij, Marco Manfredi, and Julian FP Kooij. Visual cross-view metric localization with dense uncertainty estimates. In *ECCV*, 2022. 2, 6, 7, 8


