# Towards Unconstrained Cross-View Pose Estimation

## Supplementary Material

## Overview

In this supplementary material, we expand upon our implementation details (Sec. A), gnomonic projections (Sec. B), model recall (Sec. C), and on model biases (Sec. D).

## A. Additional Implementation Details

In this section, we expand upon some of the implementation details provided in the paper.

### A.1. Orientation Bins

In our model, an arbitrary number of orientation bins can be used in the Pose Region Predictor. In our case, we use 8 bins, discretizing the orientation space into 45° sections. In practice, the higher number of bins, the greater a reduction in pose space that can be performed in the first stage of our inference prediction pipeline; however, given the efficient pose-queryable predictor allowing for a high number of parallel sampling, it is generally unnecessary to reduce beyond 45°. We find that greater than 45° can result in slightly less performance, since the sampling and refinement stages in our inference-time predictions may be more affected by local minima. Importantly, we find that local minima tend to occur around 90° and 180° offset from the true orientation due to the often grid-like nature of roads, and this can affect performance (refer to Sec. C).

### A.2. Training Negative Pose Sampling

For sampling negative poses for the contrastive loss used to optimize the pose-queryable predictor, we described using two distributions: uniform and informed according to the pose region predictions. In practice, we find that leveraging the predicted pose region probabilities allows for training to be accelerated since harder negatives are sampled more frequently, however exclusively using this distribution or sampling too high a proportion according to it can result in the undesired behaviour of poses in the less probable, less sampled regions predicting higher scores, potentially causing issues in the refinement stage of our inference process. Hence, it is important to sample from both during training. In practice, we found that having a fixed 2-to-1 uniform-to-informed ratio worked well. While we experimented with dynamically increasing the informed pose region probability sampling over the course of training, we did not find a notable improvement; while intuitively one may expect it to be better, the predicted pose region probabilities are effectively already uniform early in training as they are essentially random since the predictor isn't performant yet.

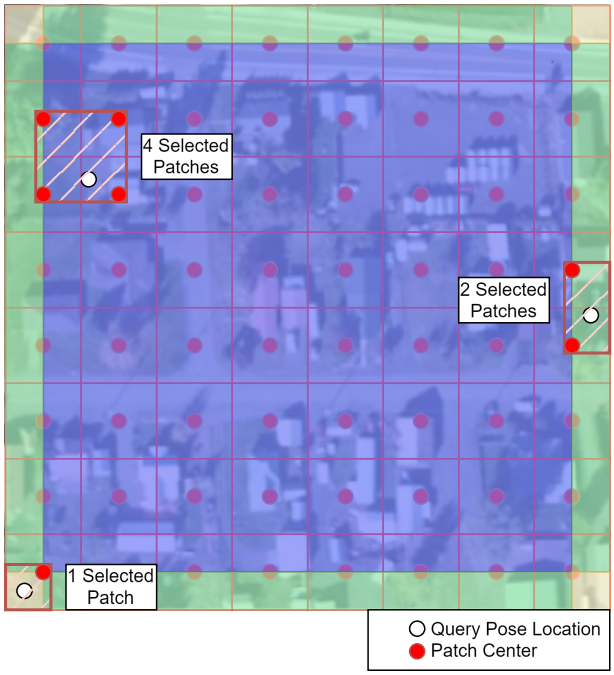



Figure 1. This Figure shows an example of how an image is patched and those patches are selected in the Pose Queryable Predictor. Aerial image is from CVUSA [**?** ], sourced from Bing Maps © Microsoft Corporation.

### A.3. Pose-Queryable Predictor

As described in the main paper: given a pose, the pose-queryable predictor leverages the selection and aggregation strategy from bilinear interpolation. Consider that in bilinear interpolation-based sampling, given a 2D grid of values (e.g. image pixels), the interpolated value for points within that grid is a weighted mean of the neighboring grid values. In particular, defining those neighboring grid points on a unit square with values given by function $f$, the formula is given by:

$$f(x,y) = (1-x)(1-y)f(0,0) + x(1-y)f(1,0) + (1-x)yf(0,1) + xyf(1,1)$$

In our case, we can leverage the differentiability of this simple function to resolve the non-differentiable patch boundaries from before by interpreting the Pose-Conditioned MLP as function $f$ over a 2D grid of patch features, located at their centers. Now, querying a score for a pose entails predicting a score for each neighboring patch only, and aggregating those predictions according to the above.

In addition to making the pose space differentiable, another benefit is that it retains the locality of poses to nearby patches which intuitively better predict them. In particular, under this regime the bilinear interpolation-based aggregation can be interpreted as taking the weighted mean of the predictions, with higher confidence on patches whose centers are located closer to the candidate location.

A visualization of the selection process is shown in Figure 1; here, all selected patches' output features for a query pose are independently used to predict the pose score with the pose-conditioned MLP, then a weighted average is taken where the weights are derived according to bilinear interpolation for the query pose's location with respect to the the selected patches' center locations; normalized to sum to 1 for cases where less than four are selected.

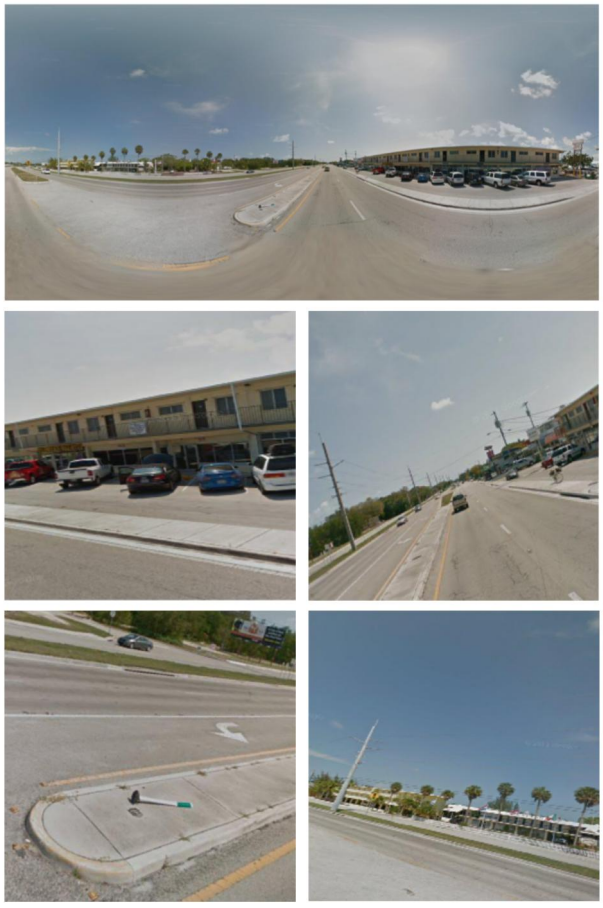


Figure 2. This Figure demonstrates examples of possible gnomonic/rectilinear projections from a panoramic image. Panorama is from CVUSA [**?** ], sourced from Google Street View © Google, Inc.

### A.4. Inference: Pose Prediction Pipeline

In the pose prediction pipeline, there is a tradeoff between the number of samples queried in parallel in the 2nd stage (continuous pose sampling) and the number of iterations performed in the 3rd stage (pose refinement). In general, increasing both improves performance, at a minor memory cost for the sampling number and a time cost for the iteration number. Some impacts on performance as a result of this selection are shown in table 1.

| | | ↓Localization (m) | | ↓Orientation (°) | |
|---|---|---|---|---|---|
| Iter | Samples | Mean | Median | Mean | Median |
| 1 | 1 | 6.4 | 2.3 | 19.5 | 13.7 |
| | 4 | 6.2 | 2.2 | 10.7 | 5.0 |
| | 12 | 6.1 | 2.1 | 8.3 | 2.7 |
| | 32 | 6.0 | 1.9 | 7.7 | 2.2 |
| 2 | 1 | 6.3 | 2.3 | 19.4 | 13.7 |
| | 4 | 6.1 | 2.1 | 10.7 | 4.9 |
| | 12 | 5.9 | 2.0 | 8.3 | 2.7 |
| | 32 | 5.9 | 1.9 | 7.3 | 2.0 |
| 10 | 1 | 6.3 | 2.3 | 19.4 | 13.7 |
| | 4 | 6.0 | 2.1 | 10.2 | 4.8 |
| | 12 | 5.9 | 2.0 | 8.3 | 2.7 |
| | 32 | 5.9 | 1.9 | 7.1 | 1.8 |

Table 1. Performance over localization and orientation over VIGORsame-area split with semi-positives, different pose sample numbers (Samples) and pose refinement iterations (Iter).

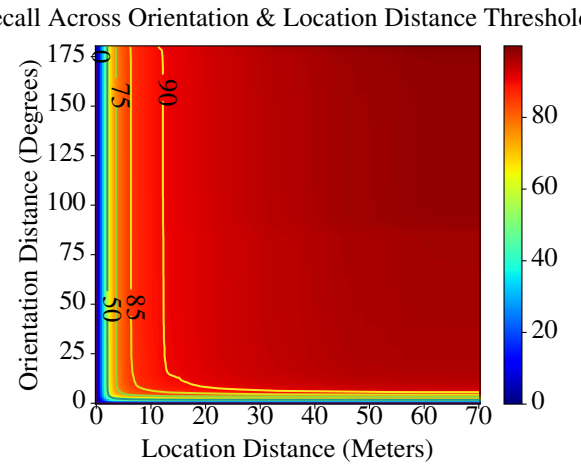
**Pose Refinement:** In this last stage, note that as there may be some level of misalignment between the Pose-Queryable Predictor's output and the Pose Region Predictor's scores, we let this final optimization go outside of the region selected in the first stage. This is useful for cases where the best pose is near the border between different regions, potentially resulting in similarly high scores and the wrong region being selected.

## B. Example Projections

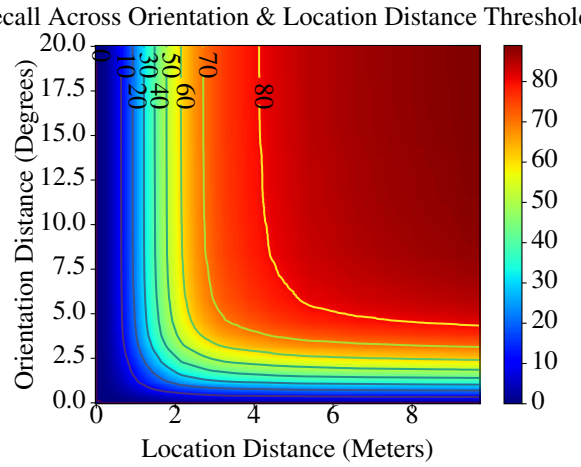
Figure 2 shows an assortment of possible rectilinear projections our projection process may produce given a panoramic image. Note that this is an example image from CVUSA, showing that our projections properly account for the reduced FoV so long as the required information is available for the view; to guarantee this, we reduce the range of yaw we sample from with CVUSA so that the projected views don't sample from the unavailable regions.

## C. Recall Comparison

In addition to measuring median and mean localization and orientation, it can also be useful to look at how the model's performance varies using recall at different location and orientation distance thresholds to better understand failure cases and general behaviour. As an example, we visualize this for the unaligned, positive-only, same-area VIGOR evaluation as a heatmap in Figure 3, where the coordi-




(a) Recall curve over larger range.




(b) Recall curve over smaller range.

Figure 3. These heatmaps display the recall curves across different distance and orientation thresholds over the VIGOR [**?** ], positive-only, same-area evaluation. 3b is the same as 3a, but zoomed in to a smaller distance range. In particular, the value at a specific (location,orientation) coordinate indicates the recall where a true positive is defined as having less than both that location distance and orientation distance.

nates are the location and orientation thresholds and the color/value at a particular location is the recall at those thresholds. Here, we can see that nearly all poses that are localized well are also oriented well, but even those that are localized more poorly are often oriented well still. This can be explained due to the natural road alignment present in the VIGOR dataset, where these poorer localizations occur down the road from the ground-truth pose, but general building and road alignment allows for correct orientation to be predicted still. Additionally, we can also note a small gap in localization where few predictions are localized really close to the ground-truth; this is likely indicative of the misalignment of **?** ]'s labels as noted by **?** ]. Lastly, a minor plateau with respect to orientation can be seen particularly in the recall gradient for the bottom few % of predictions at the 90° and 180° mark; this is due to the grid-like nature of roads such that those poor predictions align along the wrong road/road-direction.

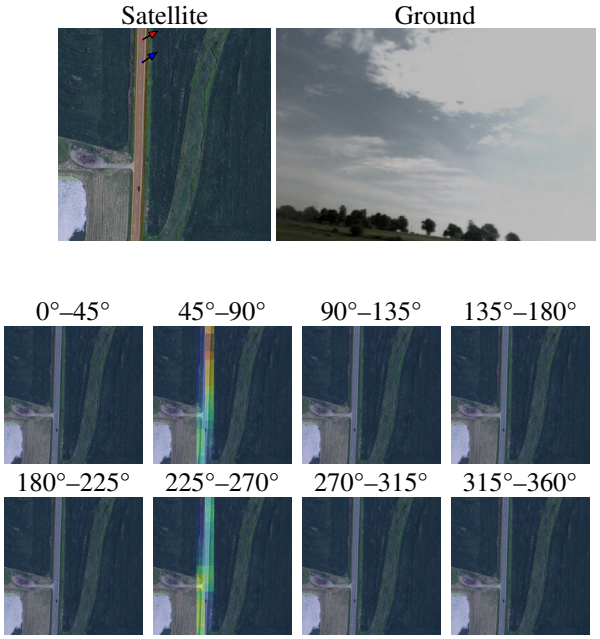



Figure 4. Visualization of model prediction over a difficult gnomonic-projection image. The pose scores across location at the ground-truth orientation are visualized as the overlayed heatmap on top of the satellite image, where red is higher, and dark blue is lower. Additionally, the ground truth location and direction is indicated by a red arrow, and the predicted, a blue arrow. Below, the Pose Region Predictor's region probability scores are shown, where red is again higher, and blue lower. Aerial image is from CVUSA [**?** ], sourced from Bing Maps © Microsoft Corporation, ground image from CVUSA [**?** ], sourced from Google Street View © Google, Inc.

## D. Biases Example

An example of the model predicting a reasonable pose in a harder case can be seen in Figure 4 which demonstrates some interesting properties learned by the model. In this example, the ground image is almost entirely of the sky, with only a small section of ground-level features visible far away; as such, there is little to no feature matching that could normally occur that would enable a good prediction. Despite this, the model does so by leveraging a few characteristics that the standard matching-based approaches would be unable to use.

For one, since our model has a general transformer-based design that enables learning arbitrary prediction strategies, it has learned the underlying bias of the datasets to be road

aligned and leverages this information to constrain predictions when other features are not available. Depending on context, being able to learn such biases from the dataset can be negative or positive. When operating on images that occur according to the distribution of training images, then this is a positive feature that enables making high quality predictions that otherwise wouldn't be possible. On the other hand, if an image is sufficiently different from the training data, such as a non-road aligned image in this case, then performance will be hindered by this.

For another, the model has also learned to partially leverage sun positioning to help constrain it's orientation predictions. This information is not typically capturable in geometric-based approaches, however is the type of meta-information that may be leveraged by human experts. In this example, the sun isn't directly visible, but the gradient in brightness in the sky has allowed the model to infer some information. Specifically, both VIGOR and CVUSA are collected in the US, which is in the northern hemisphere; this has resulted in the sun positioning being generally present in the southern half of the sky, which the model has roughly learned, despite the lack of direct temporal consistency between cross-view image pairs. In this and other similar cases, the model uses this to produce slightly higher probabilities in regions having such a sun alignment; in this example, that is oriented more towards the right.

Finally, due to our training and evaluation including matching poses that are on the edges of the corresponding satellite imagery, the model has also learned to leverage the absence of features in prediction. In this case, the ground image can see trees off in the distance, but no trees are visible in the satellite view. Given the previously described properties, this allows the model to constrain predictions towards the top section of the road, to account for the trees needing to be out-of-frame.