C3Po: Cross-View Cross-Modality Correspondence by Pointmap Prediction

Kuan Wei Huang[*] Cornell University

Brandon Li Cornell University

Bharath Hariharan Cornell University

Noah Snavely Cornell University

Abstract

Geometric models like DUSt3R have shown great advances in understanding the geometry of a scene from pairs of photos. However, they fail when the inputs are from vastly different viewpoints (e.g., aerial vs. ground) or modalities (e.g., photos vs. abstract drawings) compared to what was observed during training. This paper addresses a challenging version of this problem: predicting correspondences between ground-level photos and floor plans. Current datasets for joint photo–floor plan reasoning are limited, either lacking in varying modalities (VIGOR) or lacking in correspondences (WAFFLE). To address these limitations, we introduce a new dataset, C3, created by first reconstructing a number of scenes in 3D from Internet photo collections via structure-from-motion, then manually registering the reconstructions to floor plans gathered from the Internet, from which we can derive correspondences between images and floor plans. C3 contains 90K paired floor plans and photos across 597 scenes with 153M pixel-level correspondences and 85K camera poses. We find that state-of-the-art correspondence models struggle on this task. By training on our new data, we can improve on the best performing method by 34% in RMSE. We also use the predicted correspondences to estimate camera poses and evaluate performance using recall metrics. Lastly, we identify open challenges in cross-modal geometric reasoning that our dataset aims to help address. Our project website is available at: https://c3po-correspondence.github.io/.

1 Introduction

A frequent experience when visiting a tourist site is finding our way around with a map. On the face of it, this is a very challenging task. The bird's-eye view offered by the map or floor plan is completely different from the view we see from the ground. Exacerbating this cross-view challenge is the fact that the modality is also different: the floor plan is abstract and has none of the visual features that we see on the ground. Yet, we regularly solve this challenge, perhaps by relying on correspondences between the two views: for instance, we might map the dome we see above us to the clear circle on the map (Figure 1, right). Given that we are able to draw these correspondences, we ask: can we get computer vision models to do the same?

The ability to draw cross-view, cross-modality correspondences between photos and abstract floor plans also has practical utility. For instance, it can allow robots to localize themselves given just a map and a few sparse views. These correspondences can also be an added source of information when solving challenging structure-from-motion (SfM) problems, since they can be used to register cameras to the floor plan and thus to each other. This may be especially useful in cases where sparse views with limited overlap lead to multiple, disconnected reconstructions that cannot be registered to each other, but could be jointly registered to a plan-view image.

[*] Corresponding email: kwhuang@cs.cornell.edu

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Track on Datasets and Benchmarks.

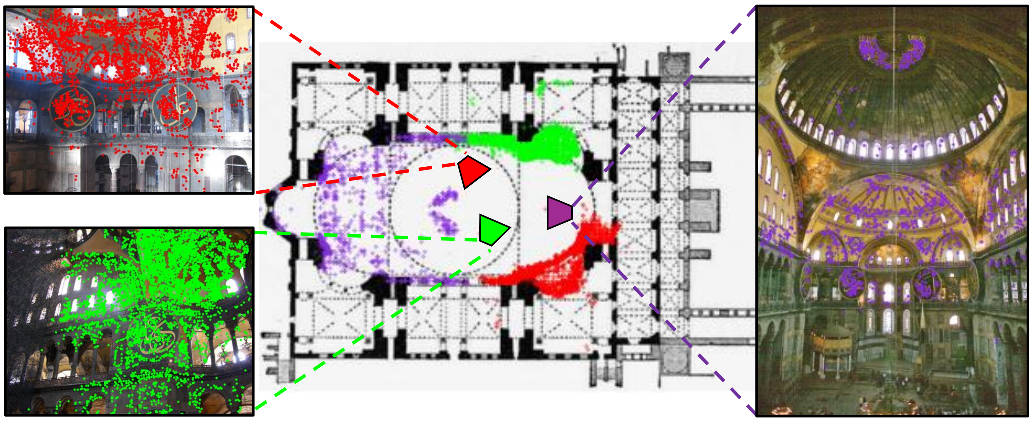

Figure 1: We present C3, a dataset of paired floor plans and photos, annotated with point correspondences and camera poses. It consists of 90K plan-photo pairs, 153M correspondences, and 85K camera poses across 597 scenes, covering a broad range of scene structures, lighting conditions, and camera poses. We show photos captured from various views within an example scene, their camera poses, and color-coded correspondences to the floor plan.

Recent advances in computer vision suggest that current state-of-the-art techniques can indeed identify correspondences even in challenging scenarios. In particular, DUSt3R [1] demonstrated the ability to draw correspondences across "nearly opposite" viewpoints. Self-supervised feature representations like DINO [2] or DIFT [3] have been shown to allow for cross-modality correspondences. Yet these models have never been tested in scenarios that combine the challenge of viewpoint and modality, a challenge exemplified in the image-to-floor plan correspondence problem. The reason is the lack of benchmarking data: there exists no prior dataset that provides ground truth correspondences across images and floor plans.

In this work, we address this gap and create a first-of-its-kind dataset, C3 (for *Cross-View Cross-Modality Correspondence*), of correspondences between floor plans and photos. To address the complexity of manually annotating correspondences, we propose a novel pipeline that uses SfM point clouds manually aligned to floor plans to infer individual point correspondences.

We use our dataset to evaluate state-of-the-art correspondence techniques ranging from DUSt3R to self-supervised features like DINO as well as methods trained explicitly for correspondence [3–7]. We find that existing models struggle to draw these correspondences, with most methods yielding errors that are more than 10% of the image size. This suggests that in spite of large advances in reconstruction, current state-of-the-art is not sufficient for solving the problem of cross-view, cross-modality correspondences.

We improve the state-of-the-art via a method we call cross-view, cross-modality correspondence by pointmap prediction (C3Po) where we fine-tune DUSt3R on our dataset, yielding a 34% reduction in error. Yet, we find that errors are much higher than classical correspondence problems. We analyze the remaining errors and find multiple challenges that are particular to this cross-view, cross-modal problem: often, ground-level photos do not provide enough context of the overall scene, and this problem is exacerbated when symmetries in the structure make the problem ambiguous. Handling this ambiguity is an open problem deserving of future research.

In sum, our contributions are:

1. We present a dataset consisting of floor plan-photo pairs collected from the Internet, along with pixel correspondences for each pair and the camera pose for each photo.
2. We demonstrate that state-of-the-art correspondence techniques fail to draw accurate correspondences between images and floor plans.
3. We adapt DUSt3R's pointmap prediction to estimate correspondences, outperforming the best baseline by 34%.

4. We show camera pose estimation as a downstream application of our predicted correspondences and evaluate its performance with recall metrics.
5. We identify systematic sources of error due to the natural ambiguity in data for future work to explore.

2 Related Work

2.1 Cross-view and Cross-modal Datasets

Since C3 is, to our knowledge, the first cross-view, cross-domain correspondence dataset, we identify the following three categories of prior datasets that are most relevant to our work: datasets with photos of scenes, floor plans, and cross-viewpoint imagery. Datasets with photos of scenes have been critical in the development of scene understanding methods [8, 9] and training the visual perception for embodied agents [10]. However, they lack scene layout images. Floor plans are a compact and generalizable 2D layout representation of 3D buildings, and prior floor plan datasets have enabled tasks like layout estimation [11, 12], scene generation [13, 14], and others [15–17]. These prior datasets, however, are limited to indoor residential buildings. The recent WAFFLE dataset [18] consists of 20K floor plans covering a wide range of building types and locations, but lacks photos of the corresponding scenes. Finally, cross-view datasets typically contain satellite and ground-level images [19] and have historically been utilized for tasks like geo-localization [20]. While the satellite images provide aerial views, they do not necessarily contain structural information in the abstracted way that floor plans do. They are also missing the correspondences between the images from the two viewpoints. Our dataset not only contains floor plan–photo pairs drawn from a broad array of architectural structures and geographical regions, but also provides 2D correspondences between the pairs and a camera pose for each photo.

2.2 Pixel Correspondence

Pixel correspondence, the process of finding matching points in different images that represent the same point in the real world, is a fundamental task in computer vision with a wide range of applications, including 3D reconstruction [21–23], motion tracking [24–26], and geo-localization [27–29]. Historical matching methods rely on manually engineered features, like SIFT [30], SURF [31], and ORB [32]. Newer strategies have shifted towards learning-based methods [33], for their ability to extract more adaptable features automatically from data. However, these features are local, limiting their robustness in capturing broader scene context. To address this, dense methods [4, 5] have been introduced to establish correspondences at a global scale for better performance, especially in extreme conditions, like large viewpoint changes, textureless areas, and poor lighting, but have yet to be tested on inputs from different visual modalities. Recently, DUSt3R [1] demonstrated the ability to learn scene geometry from pairs of photos. At its core, DUSt3R predicts a pointmap which creates a mapping from 2D image pixels to 3D scene points, and we turn this pointmap prediction into a correspondence prediction. While we find that DUSt3R alone is not effective at cross-view, cross-modality matching, we discuss how we adapt it to this task in Section 4.

3 C3: Cross-View, Cross-Modality Correspondence Dataset

Our goal is to create a dataset that consists of paired floor plans and photos and annotated correspondences between them. Two key challenges faced in building such a dataset are (1) finding a good source of floor plan images, and (2) determining correspondences between floor plans and corresponding photos. In the following sections, we describe how we address these challenges and produce our dataset.

3.1 Sourcing Floor Plans

We source our floor plans from Wikimedia Commons, following past work [18]. Wikimedia Commons is an online media repository that hosts freely licensed media content, including images, sound, and video clips. Data is organized into a hierarchy of categories and subcategories (for instance, *Cathédrale Notre-Dame de Paris* → *Interior of Cathédrale Notre-Dame de Paris* → *Plans of Cathédrale Notre-Dame de Paris*). Images and other files form the leaves of the hierarchy and can belong to multiple categories. Each file includes metadata, including captions, source, and the categories it belongs to.

To collect floor plans from Wikimedia Commons, we start by recursively traversing through the *Floor plans* category and considering every image file in the subtree. We filter these images in the following way. We observe that floor plans are typically tagged with category names that mention the structure or landmark associated with the floor plan, e.g., *Angkor Wat*, *Plans of Guy's Hospital*, and *Blenheim Palace in art*. We iterate through the category tags and infer the name of the structure or landmark by removing prefixes like "Floor plans of", "Floor plan of", "Plans of", "Plans of the" or "Maps of", and suffixes like "in art". We then check if the structure is a scene of interest by checking if it is an instance of a predefined set of scene categories as in [34] (e.g., "religious building" or "castle"; the full list is included in the supplemental material). If it is indeed a scene of interest, we manually inspect the floor plan image to ensure it is a canonical floor plan (not a section plan or drawing too abstract where scene structure is hard to extract), before adding the floor plan image to our dataset. This process results in 10,842 floor plans from a total of 6,194 scenes.

3.2 Collecting Corresponding Photos

We use MegaScenes [34] as the primary source of photos because it contains a large number of in-the-wild photos that are already grouped by scenes. We take the intersection of scenes that are represented in MegaScenes with the scenes for which we have collected floor plans above. For some scenes, we also collect additional photos from YFCC100M [35] (for reasons explained in the next section). With this process, we end up with 1,474 scenes associated with a total of 766K photos and 2,942 floor plans.

3.3 Determining Correspondences

Once we have floor plans and sets of photographs from the same scenes, the next step is to annotate correspondences between floor plans and photos. However, manually annotating correspondences at this scale is an infeasible task. Instead, we use a two-step process: 1) automatic SfM reconstruction of the photo collection corresponding to each scene using COLMAP [22], and 2) manual alignment of the resulting point clouds with the floor plan for that scene. Given a set of images, COLMAP estimates a 3D camera pose for each image, as well as a sparse point cloud corresponding to keypoints in the photo collection. Aligning these point clouds with the floor plan thus directly yields correspondences between individual photos and the floor plan and is a substantially easier task than manually annotating correspondences. We describe each step below.

3.3.1 COLMAP Reconstruction from Photo Collections

We use default parameters for feature extraction and sparse reconstruction. For scenes with a small number of photos, we use exhaustive matching with default parameters. Running exhaustive matching on scenes with a large number of photos takes a prohibitive amount of time, so we instead use vocabulary tree matching with 40 nearest neighbors.

For some scenes, COLMAP on MegaScenes photos results in disjoint components and very sparse reconstructions. As such, we augment our photo collection with YFCC100M [35] for all scenes. Since YFCC100M photos are geotagged, for each scene, we download all photos that are within a 50-meter radius of that scene's GPS location (retrieved from Wikimedia Commons). We then use this augmented set of images to perform COLMAP reconstruction.

Finally, as a post-processing step, we run COLMAP's model merger on all pairs of reconstructed 3D components for each scene. This step attempts to merge any components with overlapping cameras or 3D points and results in more unified reconstructions (although many scenes will still have separate components, e.g., for the interior and an exterior of a building).

3.3.2 Manual Alignment of Point Clouds to Floor Plans

We devised a custom user interface that displays SfM point clouds and floor plans for a given scene and allows annotators to interactively apply transformations to each scene component's point cloud to align it to the floor plan. The interface displays a floor plan and a bird's-eye view of a point cloud. This viewpoint simplifies the matching process by limiting the floor plan and point cloud transformations to only 2D translations, rotation, and scale. Once manually aligned, the transformation parameters mapping the point cloud to the floor plan, $\mathrm{T}_{\mathrm{pc}\rightarrow\mathrm{fp}}$, are saved in a database as a transformation that maps points in the point cloud to the floor plan. In this work, we repeat this alignment for the two largest reconstructed point clouds for all the scenes.

Once we have these alignments, it is fairly straightforward to obtain sparse correspondences between the floor plan and the corresponding photos. We simply take each reconstructed point $\mathbf{X}$ that is visible in a photo i and project it into both the photo (using COLMAP-estimated camera parameters T_i) and the floor plan (using the manually estimated transformation $\mathrm{T}_{\mathrm{pc}\rightarrow\mathrm{fp}}$), where $\mathbf{x}_{\mathrm{i}}$ and $\mathbf{x}_{\mathrm{fp}}$ are corresponding points:

$$\mathbf{x}_{\mathrm{fp}} = \mathrm{T}_{\mathrm{pc}\rightarrow\mathrm{fp}}\mathbf{X} \tag{1}$$

$$\mathbf{x}_i = \mathrm{T}_i\mathbf{X} \tag{2}$$

We can apply a similar transformation to obtain the camera pose of each photo.

This yields our full dataset of floor plans, corresponding photos, correspondences between the two, and camera poses of the photos, which we report in Section 3.4. Note that there is a drop in number of scenes, floor plans, and photos from Section 3.2 because not every scene has a reconstruction and many reconstructions are sparse and thus not alignable. We also manually inspect floor plans and discard composite and ambiguous ones, where an image contains floor plans of many scenes and floor plans for multiple floors of a scene, respectively.

3.4 Dataset Statistics

The C3 dataset comprises 90K floor plan-photo image pairs derived from 597 scenes. These scenes span 648 unique floor plans and include 85K photos. The dataset also includes a camera pose for each photo and 153M total pixel-level correspondences. The number of correspondences per plan-photo pair ranges from 1 to 13,262, with an average of 1,711 correspondences per pair.

We split the dataset into train and test sets by scene, ensuring no scene-level overlap between them. We train on 479 scenes, which consists of 519 unique floor plans, 66K photos (and camera poses), and 120M correspondences. We test on 118 scenes, which contains 129 unique floor plans, 19K photos (camera poses), and 33M correspondences.

4 Evaluation on C3

We first detail the correspondence baselines and show that existing baselines from the literature struggle on the cross-view cross-modality correspondence task in Section 4.1. In Section 4.2, we share our approach and discuss our results and findings. We show camera pose estimation as a downstream application of our predicted correspondences in Section 4.3.

4.1 Baseline Performance

Method. We evaluate our dataset with a combination of sparse, semi-dense, and dense matching algorithms: SuperGlue [7], LoFTR [4], DINOv2 [2], DIFT [3], RoMa [5], and MASt3R [6]. We also evaluate on DUSt3R [1]. Since our goal is to find dense correspondences between images, we make adjustments to these methods, detailed below. We start with the matching methods and leave DUSt3R and MASt3R for last. Since SuperGlue produces sparse correspondences, we perform nearest neighbor interpolation to create a dense correspondence map. DINOv2 outputs patch-level features, so we upsample the features to full image resolution using bilinear interpolation and then compute pixel correspondence with cosine similarity. For LoFTR, DIFT, and RoMa, we can sample correspondences directly with pixel coordinates. We use LoFTR's `outdoor-ds` pre-trained model and RoMa with default `scale_factor` and `upsample_factor=(512, 512)`. With DUSt3R, we input floor plan as the reference image (that is, the image that defines the 3D coordinate frame) along with a photo. This way, DUSt3R's pointmap representation maps each photo pixel to a 3D point at location (x, y, z) in the floor plan's coordinate frame. To obtain an actual pixel correspondence on the floor plan, we perform an orthographic projection $(x, y, z) \rightarrow (x, z)$, i.e., we drop the y-coordinate because the y-axis represents the up direction with respect to the coordinate frame of the first image (the floor plan in this case). With MASt3R, we provide the input images in the same order as DUSt3R.

	RMSE (↓)
SuperGlue	0.4050
LoFTR	0.2901
DINOv2	0.5338
DIFT	0.3036
RoMa	0.3308
DUSt3R	0.2925
MASt3R	0.4616
Ours	**0.1919**

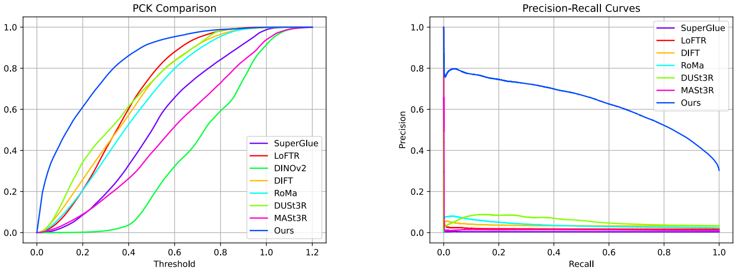

Figure 2: Quantitative results for C3 test set (floor plan and photo pairs from scenes not used during training). Left: table of RMSE values (lower is better). Our method, trained on C3 training data, achieves a significant reduction in error. Middle: Percentage of Correct Keypoints (PCK) as a function of error threshold. Right: Precision-Recall curves generated by thresholding on predicted confidence or score for each method.

To ensure dense matches, we obtain the pixel correspondences from the prediction head without any follow-up filtering steps.

Results. We report quantitative results in Figure 2. The left table lists RMSE scores for each model. To standardize the RMSE calculation, we normalize all model outputs to a range of $[0, 1]$ (that is, the image dimensions are remapped to a unit square). The middle graph shows percent of correct keypoints (PCK), which measures the proportion of predicted correspondence points that fall within a certain threshold distance of the ground truth points. In our case, our distance metric is the Euclidean distance. The right graph displays Precision-Recall (PR) curves for the methods that output a confidence score associated with the correspondences, and we consider a prediction to be correct if its Euclidean distance from the ground truth is less than 0.05 units in normalized floor plan coordinates. We show qualitative comparisons in Figure 3. Unsurprisingly, all correspondence-based methods exhibit poor performance as they have not been trained on floor plan data. Although MASt3R—a network built on the DUSt3R model for matching tasks—might be expected to outperform DUSt3R, it actually shows higher error. One explanation could be that MASt3R is performing correspondence estimation, while DUSt3R is predicting scene structure and projecting it onto the 2D floor plan which is meaningfully closer to the solution.

4.2 Cross-View Cross-Modality Correspondence by Pointmap Prediction

While the baseline results were rather poor, we observe promising geometric structures in DUSt3R outputs; the model only needed to learn the 2D translations, rotation, and scale to align to the floor plan. We therefore leverage the strong geometric prior from the pre-trained DUSt3R model and fine-tune on our dataset, with some modifications. First, we split DUSt3R's Siamese encoders, which were designed to process two input images with visual overlap. Since our inputs—floor plans and photos—are from different domains, we reason that each encoder should separately learn the distributions of each individual domain. We also find we can treat this correspondence task as a pointmap prediction problem. As explained in Section 4.1, we can set up DUSt3R's pointmap representation to map 2D points in the photo to 3D points in the floor plan coordinate frame, then project back to 2D via an orthographic projection that discards the y- (or up-) coordinate. We also experiment with discarding the z-coordinate instead, but this empirically leads to slower model convergence. Finally, we observe model overfitting on floor plans during training. To improve model generalization, we perform photometric augmentations (color jitter) and geometric augmentations (cropping and rotation) on the floor plans.

Training Setup. We train our approach for 10 epochs which takes about 3 days with 8×A6000 40GB. We use the same hyperparameters as DUSt3R. We share more details regarding our setup in the supplementary material.

Results. Quantitatively, our model displays a 34% decrease in RMSE compared to the best performing baseline. We also observe a stronger PCK and PR performance for our model. Figure 3 shows that our model predicts the correspondences accurately, sometimes less noisily than the COLMAP

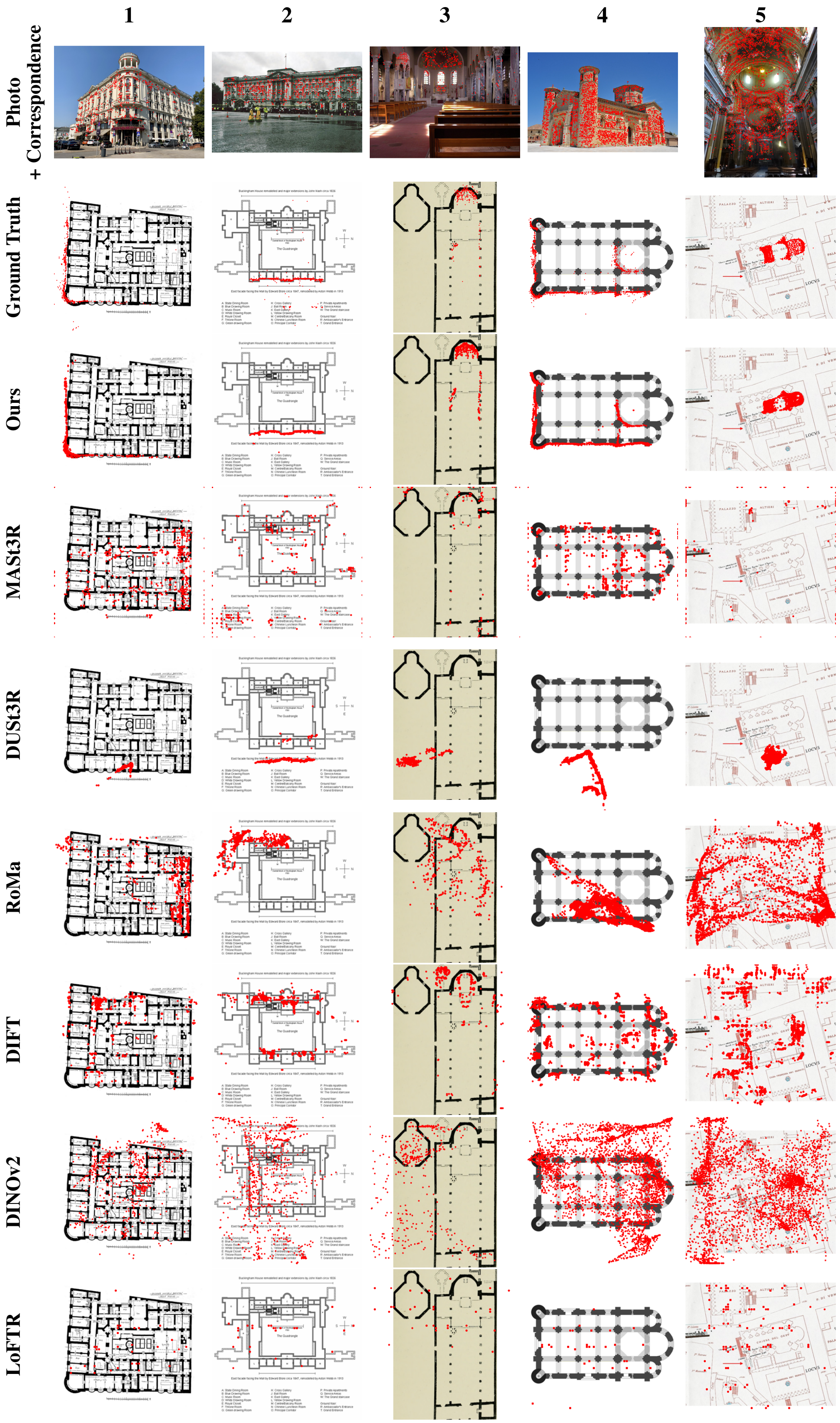

Figure 3: Qualitative results for C3 test set. Each red dot in the images represents a correspondence point.

	C3 R@							
Method	Angular				Positional			Angular + Positional
	5°	10°	20°	30°	5%	10%	20%	30°, 20%
C3Po (ours)	21.86	32.53	44.12	51.35	15.94	27.73	41.21	29.48

Table 1: Camera pose estimation results on the C3 dataset, measured by recall under varying angular, positional, and both angular and positional thresholds. The positional thresholds represent a percentage of the floor plan's diagonal length.

reconstructions. We perform the Wilcoxon signed-rank test, a non-parametric paired test, between our error and each baseline and find all P-values to be less than 0.05.

4.3 Camera Pose Estimation

Method. We now explore the task of estimating the 2D camera pose of the ground-view photo in the floor plan coordinate frame—a challenging "you are here" photo location identification problem. We first estimate epipolar geometry via correspondences found with our method. To simulate the floor plan's orthographic projection, we use a large focal length (10^7 pixels). Given matched keypoints between the photo and plan, we compute an essential matrix using OpenCV's `findEssentialMat` function, then decompose the essential matrix into camera rotation and translation. To transform the camera rotation matrix from the XY plane to the floor plan coordinate frame, which lies on the XZ plane, we apply a $-90°$ rotation about the x-axis. We find that in our case an ambiguity can arise for photos that observe a near-planar structure (as vertical planes like walls yield points that lie on a *line* in the floor plan). Such cases can yield upside-down cameras on the wrong side of the observed surface. We detect such cases automatically based on the estimated rotation matrix and correct them by reflecting the camera across the best-fit plane formed by the correspondences. Finally, we estimate the camera center by computing the epipole in the floor plan image. The epipole in the floor plan corresponds to the camera location of the ground-view photo.

Results. We evaluate camera poses estimated from correspondences predicted by C3Po on the C3 dataset using a range of angular, positional, and combined angular-positional thresholds, as reported in Table 1. Positional thresholds are expressed as the Euclidean distance between estimated and ground-truth camera centers, normalized by the diagonal length of the floor plan.

5 Open Challenges

While our method demonstrates encouraging results, we show two categories of examples where our model could improve on.

5.1 Challenge 1: Photos with Minimal Context

This case refers to floor plan-photo pairs where the photo lacks contextual information of its global surroundings. These are examples that would be challenging even for humans to reason about in terms of the general location of the correspondences on the floor plans or camera pose. For example, Figure 4 (top row) shows a close-up shot of a door and a window. The second row figure is a photo of only an artwork. In both instances, our model makes plausible predictions; for example, the door photo is mapped to one of the doors on the floor plan and is along the exterior wall of the scene. However, the answer is wrong due to lack of context. Future work could attempt to resolve this issue by, for instance, predicting distributions of correspondences, rather than regressing to a specific unimodal answer.

5.2 Challenge 2: Structural Symmetry

This challenge involves scenes with structural symmetry. Although there are subtle cues on the floor plan that can often help disambiguate scenes that feature symmetries (domes, similar walls or

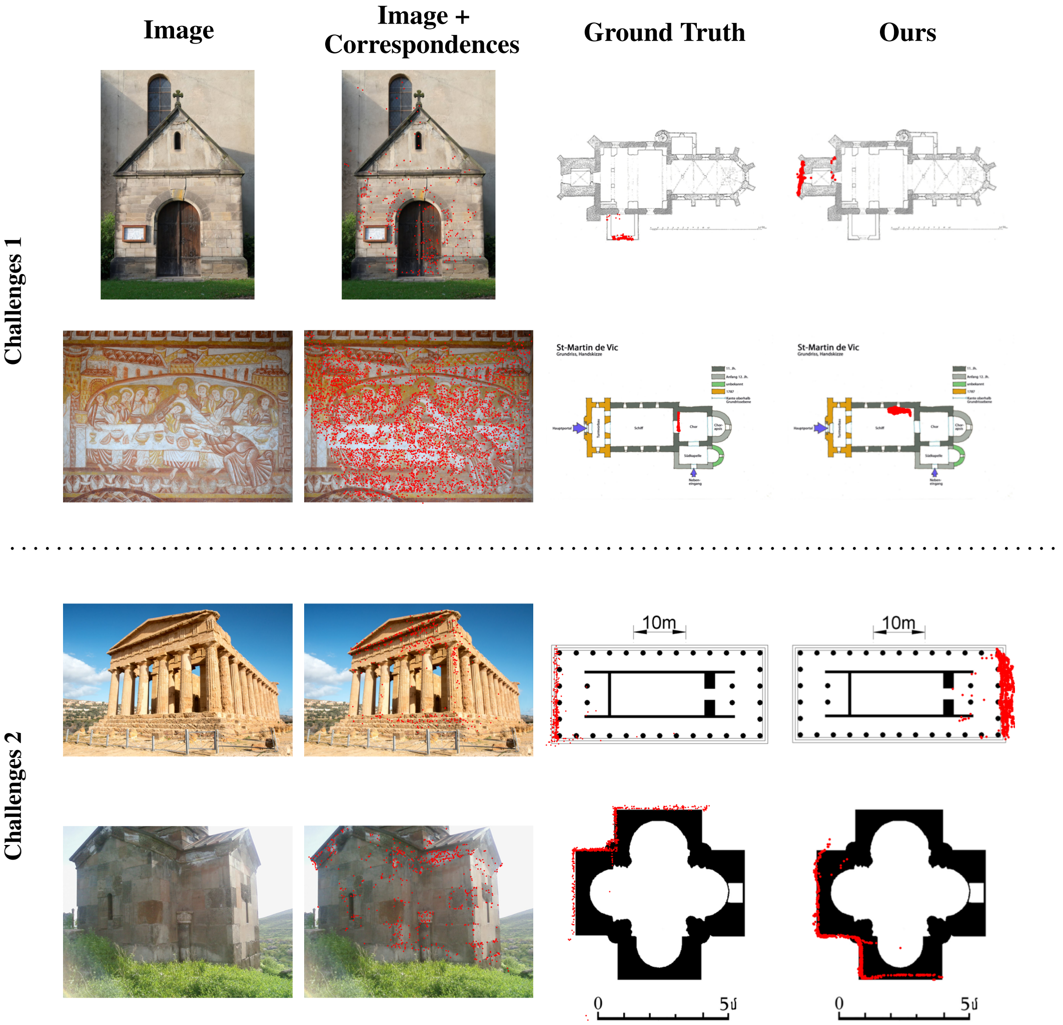

Figure 4: We share two categories of data that our model struggles on. Challenge 1, top two rows, are cases where the photo provides minimal context clues of where it could be on the floor plan. Challenge 2, bottom two rows, are scenes that exhibit structural symmetry, where multiple correspondence alignments would seem plausible.

hallways, etc.), they are difficult to identify from the photos. Figure 4 (third row) shows a photo of a temple viewed from the left side of the floor plan and the last row shows a photo of a church viewed from the top left corner. Again, our model predicts plausible correspondence configurations that are consistent with the scene geometry in both cases, and again, perhaps a more distributional approach to prediction, e.g., with diffusion models, would be more appropriate in such cases.

6 Conclusion

In this paper, we present C3, the first cross-view, cross-modality correspondence dataset. We first source floor plans and photos of scenes from the Internet and then determine their correspondences by running COLMAP followed by carefully aligning the reconstructed point clouds to the floor plans. We show that existing correspondence methods fail to accurately establish matches between floor plans and images. We propose to frame matching as a DUSt3R pointmap prediction task, and this approach outperforms the best performing baseline by 34%. We further showcase the utility of these correspondences for camera pose estimation as a downstream task. We also highlight structured failure modes for future research directions.

In addition to the open challenges we observe, our dataset could be used to enable a number of other cross-modal tasks. For instance: (1) given an image and a floor plan, localize the image on the plan (i.e., camera-to-plan relative pose), (2) given a floor plan and a camera, generate an image (i.e., floor plan-conditioned image generation), and (3) given an image, generate a complete floor plan of the structure pictured (image-to-floor-plan generation). In general, we hope that having access to quality data can help spur progress on problems involving jointly reasoning about the kinds of global, abstracted structure available in a floor plan and the local structure pictured in a photo.

Societal Impacts. Through our dataset and approach, we hope to enable researchers to develop more accurate and robust methods in areas including 3D vision, image generation, and robotics. We do not anticipate negative societal concerns.

References

[1] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *CVPR*, 2024.

[2] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," 2023.

[3] L. Tang, M. Jia, Q. Wang, C. P. Phoo, and B. Hariharan, "Emergent correspondence from image diffusion," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[4] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "LoFTR: Detector-free local feature matching with transformers," *CVPR*, 2021.

[5] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg, "Roma: Robust dense feature matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19790–19800, 2024.

[6] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3d with mast3r," 2024.

[7] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *CVPR*, 2020.

[8] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, "Joint 2d-3d-semantic data for indoor scene understanding," *arXiv preprint arXiv:1702.01105*, 2017.

[9] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *International Conference on 3D Vision (3DV)*, 2017.

[10] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson Env: real-world perception for embodied agents," in *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*, IEEE, 2018.

[11] J. Zheng, J. Zhang, J. Li, R. Tang, S. Gao, and Z. Zhou, "Structured3d: A large photo-realistic dataset for structured 3d modeling," in *Proceedings of The European Conference on Computer Vision (ECCV)*, 2020.

[12] S. Cruz, W. Hutchcroft, Y. Li, N. Khosravan, I. Boyadzhiev, and S. B. Kang, "Zillow indoor dataset: Annotated floor plans with 360º panoramas and 3d room layouts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2133–2143, June 2021.

[13] W. Wu, X.-M. Fu, R. Tang, Y. Wang, Y.-H. Qi, and L. Liu, "Data-driven interior plan generation for residential buildings," *ACM Transactions on Graphics (SIGGRAPH Asia)*, vol. 38, no. 6, 2019.

[14] C. van Engelenburg, F. Mostafavi, E. Kuhn, Y. Jeon, M. Franzen, M. Standfest, J. van Gemert, and S. Khademi, "Msd: A benchmark dataset for floor plan generation of building complexes," 2024.

[15] Y. Wu, Y. Wu, G. Gkioxari, and Y. Tian, "Building generalizable agents with a realistic and rich 3d environment," *arXiv preprint arXiv:1801.02209*, 2018.

[16] M. Vidanapathirana, Q. Wu, Y. Furukawa, A. X. Chang, and M. Savva, "Plan2scene: Converting floorplans to 3d scenes," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10733–10742, June 2021.

[17] M. Standfest, M. Franzen, Y. Schroder, L. G. Medina, Y. V. Hernandez, J. H. Buck, Y.-L. Tan, M. Niedzwiecka, and R. Colmegna, "Swiss dwellings: A large dataset of apartment models including aggregated geolocation-based simulation results covering viewshed, natural light, traffic noise, centrality and geometric analysis," 2022.

[18] K. Ganon, M. Alper, R. Mikulinsky, and H. Averbuch-Elor, "Waffle: Multimodal floorplan understanding in the wild," 2024.

[19] S. Zhu, T. Yang, and C. Chen, "Vigor: Cross-view image geo-localization beyond one-to-one retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3640–3649, 2021.

[20] P.-E. Sarlin, D. DeTone, T.-Y. Yang, A. Avetisyan, J. Straub, T. Malisiewicz, S. R. Bulo, R. Newcombe, P. Kontschieder, and V. Balntas, "OrienterNet: Visual Localization in 2D Public Maps with Neural Matching," in *CVPR*, 2023.

[21] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building rome in a day.," *Communications of the ACM*, vol. 54, no. 10, 2011.

[22] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[23] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[24] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *European Conference on Computer Vision (ECCV)*, 2020.

[25] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[26] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[27] T. Weyand, I. Kostrikov, and J. Philbin, "Planet-photo geolocation with convolutional neural networks," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pp. 37–55, Springer, 2016.

[28] B. Zhou, L. Liu, A. Oliva, and A. Torralba, "Recognizing city identity via attribute analysis of geo-tagged images," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III 13*, pp. 519–534, Springer, 2014.

[29] D. Wilson, X. Zhang, W. Sultani, and S. Wshah, "Visual and object geo-localization: A comprehensive survey," 2023.

[30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints.," *International journal of computer vision*, vol. 60, 2004.

[31] H. Bay, T. Tuytelaars, and L. Van Gool, "Bay, herbert, tinne tuytelaars, and luc van gool. "surf: Speeded up robust features.," May 2006.

[32] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf.,” in *ICCV*, pp. 2564–2571, 2011.

[33] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self-supervised interest point detection and description,” in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 224–236, 2018.

[34] J. Tung, G. Chou, R. Cai, G. Yang, K. Zhang, G. Wetzstein, B. Hariharan, and N. Snavely, “Megascenes: Scene-level view synthesis at scale,” in *ECCV*, 2024.

[35] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, “YFCC100M: The new data in multimedia research,” *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This paper's Dataset, Evaluation, Open Challenges and Conclusion sections support the claims made in the Abstract and Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss this in the Open Challenges section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: We do not include new theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The Dataset, Evaluation, and Supplemental Materials sections provide all the information needed to reproduce the main experimental results. We also share a GitHub that contains pretrained weights and a notebook for the reviewer to visualize results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We share links to our dataset hosted on Hugging Face and code hosted on GitHub.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We share this in the Dataset, Evaluation and Supplemental Material section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We discuss this in the Evaluation section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We list the compute resources in the Evaluation and Supplementary Materials sections.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answ er: [Yes]

   Justification: Our research was conducted in every aspect in alignment with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss broader impacts in the Conclusion section.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no risks for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We host our dataset on Hugging Face with the license CC-BY 4.0.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Details about the dataset, code and model could be found in the Dataset, Evaluation, and Supplemental Materials sections, as well as on the GitHub and Hugging Face repositories shared in the submission.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.