Revisiting Geometric Obfuscation with Dual Convergent Lines for Privacy-Preserving Image Queries in Visual Localization

Supplementary Material

In this supplementary material, we first provide the detailed mathematical derivation for Proposition 1 in Sec. A. We then detail the architectures of the inversion models used in our experiments in Sec. B. Following this, we present extended qualitative evaluations demonstrating the robustness of DCL against inversion attacks in Sec. C. Subsequently, we clarify the system architecture of our privacy-preserving localization framework in Sec. D. In Sec. E, we discuss the degree of privacy spectrum across various paradigms. In Sec. F and Sec. G, we provide implementation details and additional experimental analyses, including geometric degeneracy and ablation studies. Finally, we detail the baseline configurations and practical limitations of iterative server-side attacks in Sec. H.

A. Mathematical Proofs of Proposition 1

We provide the detailed mathematical proof for Proposition 1, which establishes that the recovered keypoint parameter t_i^* is a weighted average of individual intersection parameters $\{t_{i,j}^*\}$.

The attack's cost function (in the main paper Eq. (3)), seeks to find the point $\mathbf{x}_i^* = \hat{\mathbf{x}}_i$ on the target line $\mathbf{l}_i$ that minimizes the sum of squared point-to-line distances:

$$f(\hat{\mathbf{x}}_i) = \sum_{\mathbf{l}_j \in \mathcal{N}(\mathbf{l}_i)} d(\mathbf{l}_j, \hat{\mathbf{x}}_i)^2. \tag{1}$$

We parameterize the point $\hat{\mathbf{x}}_i$ along the line $\mathbf{l}_i$ using the scalar parameter t_i, where the true keypoint $\mathbf{x}_i$ is assumed to be at $t_i = 0$. We assume that $\mathbf{v}_i$ is a normalized direction vector (i.e., $\|\mathbf{v}_i\| = 1$):

$$\hat{\mathbf{x}}_i(t_i) = \mathbf{x}_i + t_i \mathbf{v}_i. \tag{2}$$

By substituting Eq. (2) into the cost function, $f(\hat{\mathbf{x}}_i)$ becomes a function of t_i, which we denote as $f(\hat{\mathbf{x}}_i(t_i))$.

Proposition 1. *For the inter-anchor scenario, the minimized cost function $f(\hat{\mathbf{x}}_i(t_i))$ is convex quadratic. The recovered point parameter t_i^* is the weighted average of the individual parameters $\{t_{i,j}^*\}$ of the intersection points between the target line $\mathbf{l}_i$ and each neighbor line $\mathbf{l}_j$:*

$$t_i^* = \frac{\sum_j (w_{i,j} t_{i,j}^*)}{\sum_j w_{i,j}}. \tag{3}$$

Proof. The point-to-line distance $d(\mathbf{l}_j, \hat{\mathbf{x}}_i)$ is calculated using the Euclidean distance in 2D. For convenience in the derivation below, we adopt an abuse of notation where the squared distance $d(\mathbf{l}_j, \hat{\mathbf{x}}_i)^2$ is expressed using the squared norm of a 3D cross product. This is valid by treating all 2D vectors ($\mathbf{x} \in \mathbb{R}^2$, including $\mathbf{v}_i$, $\mathbf{v}_j$, and $\mathbf{a}_2$) as 3D vectors with a zero z-component (i.e., $\mathbf{x} \to (\mathbf{x}, 0)$). The direction vector $\mathbf{v}_j$ is also assumed to be normalized ($\|\mathbf{v}_j\| = 1$).

We consider the contribution of a single neighbor line $\mathbf{l}_j$, which passes through an anchor $\mathbf{a}_2$ with direction $\mathbf{v}_j$. The squared point-to-line distance, $f_j(\hat{\mathbf{x}}_i(t_i)) = d(\mathbf{l}_j, \hat{\mathbf{x}}_i(t_i))^2$, is calculated using the cross product:

$$f_j(\hat{\mathbf{x}}_i(t_i)) = \|(\hat{\mathbf{x}}_i(t_i) - \mathbf{a}_2) \times \mathbf{v}_j\|^2$$

Substituting $\hat{\mathbf{x}}_i(t_i) = \mathbf{x}_i + t_i \mathbf{v}_i$:

$$f_j(\hat{\mathbf{x}}_i(t_i)) = \|((\mathbf{x}_i - \mathbf{a}_2) + t_i \mathbf{v}_i) \times \mathbf{v}_j\|^2$$

Using the distributive property of the cross product $\mathbf{v}_i \times \mathbf{v}_j$ and expanding the squared norm, we obtain the quadratic form $f_j(\hat{\mathbf{x}}_i(t_i)) = A_{i,j} t_i^2 + B_{i,j} t_i + C_{i,j}$, where:

$$A_{i,j} = \|\mathbf{v}_i \times \mathbf{v}_j\|^2 \tag{4}$$

$$B_{i,j} = 2\,((\mathbf{x}_i - \mathbf{a}_2) \times \mathbf{v}_j) \cdot (\mathbf{v}_i \times \mathbf{v}_j) \tag{5}$$

$$C_{i,j} = \|(\mathbf{x}_i - \mathbf{a}_2) \times \mathbf{v}_j\|^2. \tag{6}$$

The total cost $f(\hat{\mathbf{x}}_i(t_i)) = \sum_j f_j(\hat{\mathbf{x}}_i(t_i)) = (\sum_j A_{i,j}) t_i^2 + (\sum_j B_{i,j}) t_i + (\sum_j C_{i,j})$ is also a quadratic and convex function. Since $A_{i,j} = \|\mathbf{v}_i \times \mathbf{v}_j\|^2 \geq 0$ for all j, the coefficient of the quadratic term, $\sum_j A_{i,j}$, is non-negative. The parameter t_i^* that minimizes $f(\hat{\mathbf{x}}_i(t_i))$ satisfies $\frac{df}{dt_i}(t_i^*) = 0$:

$$t_i^* = -\frac{\sum_j B_{i,j}}{2 \sum_j A_{i,j}}. \tag{7}$$

Since the individual optimum $t_{i,j}^*$ minimizes $f_j(\hat{\mathbf{x}}_i(t_i))$, we have $t_{i,j}^* = -B_{i,j}/(2A_{i,j})$, which implies $B_{i,j} = -2A_{i,j} t_{i,j}^*$. Substituting this relationship yields the weighted average form:

$$t_i^* = \frac{\sum_j (A_{i,j} t_{i,j}^*)}{\sum_j A_{i,j}}. \tag{8}$$

By setting the weights $w_{i,j} = A_{i,j}$ (Eq. (4)), we complete the proof of the proposition. ■

Corollary 1.1 (Corollary 1.1 from Main Paper). *The weight $w_{i,j}$, which is defined as $A_{i,j}$ in Proposition 1 (Eq. (4)), for each offset contribution $t_{i,j}^*$ is determined solely by the angle $\theta_{i,j}$ between the target line $\mathbf{l}_i$ and its*

neighboring line $\mathbf{l}_j$*. This relationship is established by applying the geometric definition of the cross product to the term* $A_{i,j}$*:*

$$w_{i,j} = A_{i,j} = \|\mathbf{v}_i \times \mathbf{v}_j\|^2 = \sin^2(\theta_{i,j}). \quad (9)$$

B. Inversion Model Configuration and Setup

We present inversion experiments for two representative descriptors: SIFT [11] and SuperPoint [7]. We selectively applied an inversion model for each descriptor. For SIFT, we utilized InvSfM [18], a widely adopted baseline in Privacy-Preserving Visual Localization with publicly available pre-trained weights. Designed to reconstruct images from 3D SfM point clouds, InvSfM employs a three-stage cascaded architecture: VisibNet for visibility estimation, CoarseNet for initial reconstruction, and RefineNet for final enhancement. Although originally trained on projected 3D point clouds, the model is capable of performing inversion using 2D query feature maps extracted from single images.

For SuperPoint, since no compatible pre-trained weights exist, we trained a new model using the U-Net-based inversion architecture adopted in [6, 12]. Unlike InvSfM, which utilizes a complex cascaded structure often aided by auxiliary depth or RGB inputs, we employ a simplified 2D U-Net trained on SuperPoint keypoints and descriptors [7]. This strictly adheres to the cloud-based localization scenario where raw images are not transmitted to the server.

C. Extended Evaluation

Robustness against SuperPoint inversion. We provide uncurated qualitative results for a stress-test using SuperPoint features with an expanded neighborhood (K=100), where DCL's robustness continues to hold (see Fig. 2). Additionally, Fig. 3 visualizes the inversion network's output. This image shows the result of feeding the recovered keypoint locations, which are captured from a ×2 zoomed-out, at the K=20 neighborhood setting, into the network. This zoomed-out view is necessary as most points of DCL are recovered far outside the original image boundary. This visualization provides clear empirical evidence of our method's robustness against the inversion attack.

Robustness against SIFT inversion. We also provide qualitative results to visually validate DCL's robustness against the geometry-recovery attack [5], conducted in the oracle setting. Fig. 4 shows the inversion results for SIFT features (K=20), corresponding to the quantitative metrics (Table 1). These results confirm that while Random Lines [20] and Coordinate Permutation [15] yield recognizable inversions, DCL demonstrates consistent robustness against the attack by rendering the output unrecognizable.

Dataset	Metrics	Feature points	Random lines [20]	Coordinate permut. [15]	DCL (ours)
7-scenes	PSNR(↓)	15.38	14.30	13.83	**11.02**
	SSIM(↓)	0.604	0.531	0.509	**0.481**
	LPIPS(↑)	0.540	0.595	0.614	**0.720**
Cambridge	PSNR(↓)	18.59	17.00	16.29	**9.765**
	SSIM(↓)	0.674	0.556	0.505	**0.308**
	LPIPS(↑)	0.445	0.512	0.549	**0.801**
Aachen	PSNR(↓)	18.43	17.16	16.44	**9.73**
	SSIM(↓)	0.602	0.488	0.425	**0.248**
	LPIPS(↑)	0.406	0.470	0.520	**0.806**

Table 1. Mean quality metrics for images reconstructed from SIFT [11] features using the InvSfM [18] network.

D. Privacy Preserving Localization Framework

To mitigate potential privacy risks, DCL adopts a privacy-preserving(P.P.) architecture where the raw query image is never transmitted to the server. Consequently, the extraction of all visual features is performed on the client-side. The client computes the image retrieval features, keypoint descriptors, and DCL features, and transmits only these compact representations to the server.

Upon receiving the features, the server executes the standard localization pipeline following HLoc [19]. Specifically, the server performs image retrieval against the database using the received global features, followed by feature matching and 6-DoF pose estimation.

Importantly, we employ this hierarchical pipeline, which includes an image retrieval step, primarily to ensure a fair comparison with the non-privacy-preserving baseline [19]. Furthermore, our proposed geometry obfuscation method is agnostic to the localization pipeline. It can be readily applied to architectures that omit the image retrieval stage, such as direct 2D-3D matching against a global map.

E. Discussions on Privacy Preserving Paradigms

Privacy in image-query obfuscation is not binary, but depends on the adversary model and the amount of recoverable information.

Learning-based: Recently, learning-based localization methods (*e.g.* Scene Coordinate Regression (SCR) [1, 2, 4, 21]), which directly map image pixels to 3D world coordinates, have achieved impressive localization performance. While this implicit map representation could offer privacy in a client-based (map distribution) setting, it poses a high risk for cloud-based visual localization. This approach typically requires uploading the raw query image (or query features) for inference[†], making it unsuitable as a Privacy-Preserving Image Query (PPIQ) solution.

Distance between anchors	Aachen Day			Aachen Night		
	0.25m,2°	0.5m,5°	5m,10°	0.25m,2°	0.5m,5°	5m,10°
H	**41.0**	**57.9**	**72.7**	13.3	**21.4**	**38.8**
$2H$	34.1	53.2	71.5	**14.3**	20.4	37.8
$3H$	26.8	49.5	70.3	12.2	24.5	37.8

Table 2. Ablation study of anchor point distance on localization accuracy on the Aachen Day & Night dataset. We report the recall at thresholds 0.25m/2°, 0.5m/5°, and 5m/10° (%). H denotes the height of the image query.

Descriptor-free: A common limitation of descriptor-free approaches is their inability to conceal keypoint positions, leaving the 2D layout vulnerable to image content inference. As shown by the original InvSfM work (cf. Fig.6 in [18]), a fair-well image can be reconstructed from keypoint locations and color information even without descriptors, which mainly add textural details.

Segmentation-based obfuscation: A recent diffusion-based attack [17] shows that segmentation-based methods are not immune to leakage. However, they still provide a higher degree of privacy than geometric obfuscation methods [15, 20] by more effectively obscuring fine details.

Prior geometry-based obfuscation: Prior geometry-based schemes [15, 20] may still retain a degree of privacy depending on the recovery conditions, although the evaluation in our main paper focuses on the worst-case upper-bound setting of [5].

F. Implementation Details

Dataset and scene selection. Consistent with prior works [16, 22], our evaluation excludes the Street and Great Court scenes from the Cambridge dataset [8] because of their excessive geometric outliers.

Pose estimation parameters. For the RANSAC-based pose estimation, we configure the inlier reprojection error thresholds on a per-dataset basis. The 2D keypoint thresholds (ϵ_{pt}) are set to $4px$ for 7Scenes, $18px$ for the Cambridge dataset, and $8px$ for Aachen. Consequently, the inlier thresholds for 2D lines (ϵ_{line}) are defined as their corresponding point-based values divided by $\sqrt{2}$ (i.e., $\epsilon_{pt}/\sqrt{2}$), resulting in $4/\sqrt{2}px$, $18/\sqrt{2}px$, and $8/\sqrt{2}px$, respectively.

Baseline framework and fair comparison. To ensure a rigorous and fair evaluation, we re-implemented the baseline methods, including HLoc [19] and Random Lines [20], within our unified testing framework using PoseLib [10]. Applying the same Lo-RANSAC loop across all methods, including DCL and the baselines [19, 20], guarantees an evaluation under identical conditions, enabling a direct and equitable comparison of accuracy and recall performance.

Consistent with prior works, all experiments reported on 7Scenes, Cambridge Landmarks, and Aachen utilize Structure-from-Motion (SfM) based pseudo ground truth (pGT) poses [3] to ensure fair comparison across methods.

Exclusion of baselines. The pose estimation results for the coordinate permutation [15] are omitted from our comparison. The official implementation for this method is not publicly available, and our own efforts to replicate the results reported in their paper were unsuccessful.

G. Additional Experiments and Analysis

Partition boundary analysis. Note that while we use a vertical partition for simplicity, the scheme is fundamentally applicable to any dividing line (e.g., horizontal). We conducted an empirical analysis by setting the boundary as a horizontal midline that crosses the center of the image. The experiment revealed a significant increase in degenerate cases, which grew from 4 to 39 in the 7Scenes dataset. This increased degeneracy stems from the inherent nature of indoor environments, which frequently possess textureless ceilings and floors. A horizontal boundary risks isolating these sparse areas into a single region, leading to geometric instability. Conversely, our vertical midline intersects the feature-rich central band, ensuring a more balanced keypoint distribution.

Effect of anchor length on Aachen dataset. Due to space constraints in the main manuscript, the ablation study on the impact of anchor length for the Aachen Day-Night dataset is presented here. Consistent with the observations on the 7Scenes and Cambridge Landmarks datasets (in the main paper), the Aachen dataset achieves the best localization performance when the anchor length is set to the full image height (H) as shown in Table. 2.

H. Details on Server-Side Attacks

Baseline server-side attack. As introduced in the main paper, we consider a challenging server-side adversary [20] that leverages 2D-3D correspondences. To simulate the server-side attack, we first performed the geometry-recovery attack [5], setting K=20 nearest neighbors for each outlier. Note, we assumed perfect identification of these neighbors. Then, if the neighborhood contains projected inlier points, we used a point-to-point distance instead of point-to-line distance in Eq. (3) in the main paper [9]. If the

[†]A notable example is Niantic's Lightship VPS, a cloud-based service that sends raw camera frames to the cloud [14]. It uses ACE [4], a production-scale SCR relocalizer, in its operations [13].

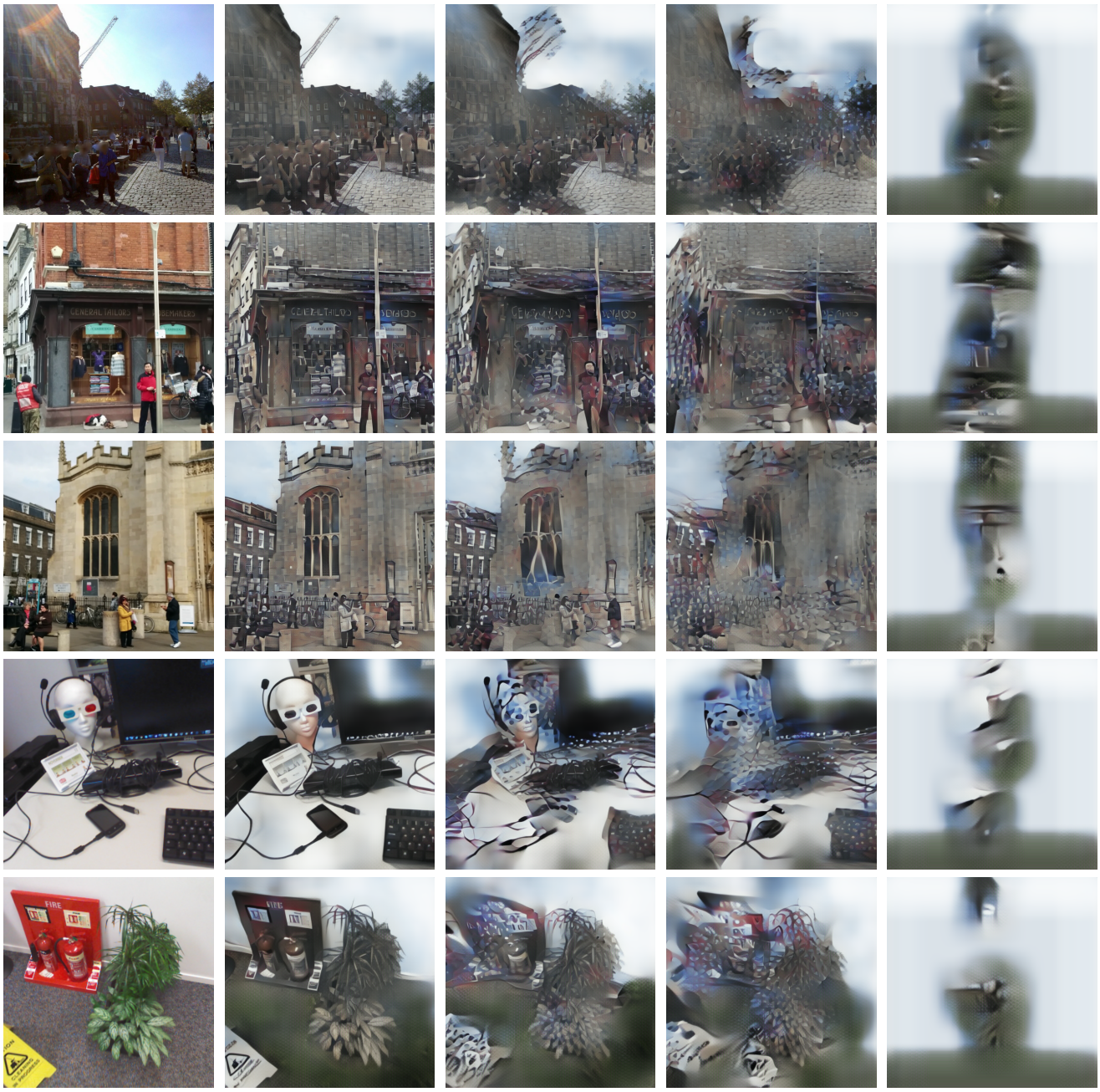
Figure 1. Visualization of the iterative server-side attack on three different scenes. In (a), black points indicate inlier keypoints from the server map, while colored points indicate ground truth (user's) query keypoints (left) or their estimated locations after running the iterative attack on DCL (right). For each scene, same colored points in (left) and (right) indicate the same keypoint. In (b), the original query image (left) and the image revealed from the estimated points (right) are shown for each scene.

inilers are not in the neighborhood, the server can still proceed with recovery using only the obfuscated line neighbors as in Sect. 4.2 of the main text [9].

Iterative server-side attack and its practical limitations. To evaluate robustness against stronger adversaries, we implemented an iterative server-side attack. In each step, we estimate the 2D points from the 2D lines which are anticipated to comprise the biggest number of neighboring (server-side) inliers, and add them to the inlier set for the next iteration.

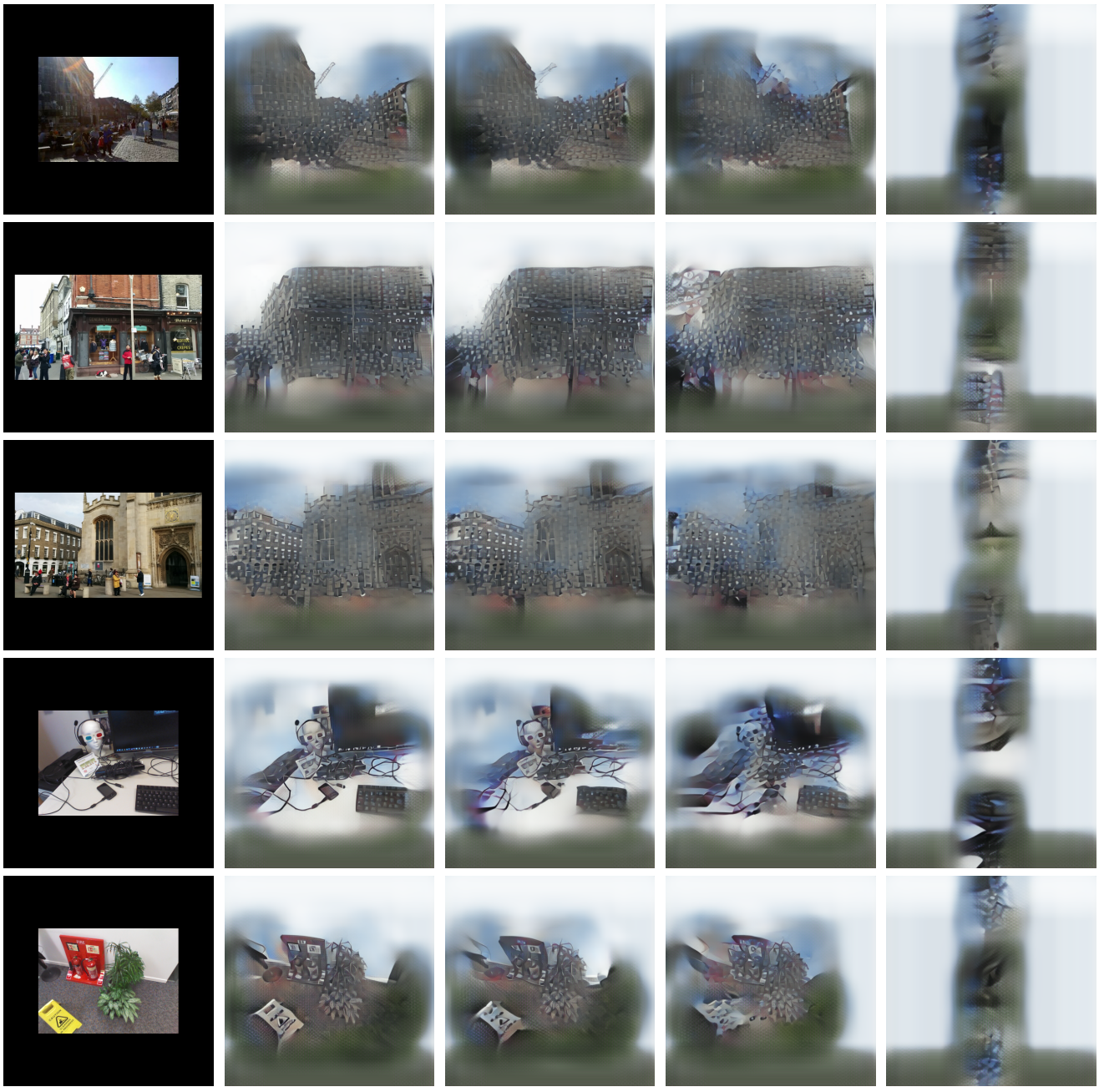
Since nearby obfuscated 2D lines exhibit similar directions, their estimated keypoints are usually found at similar locations provided they share a large overlap of server-side inliers. We observe this effect as well as accumulation of errors lead to a *spiral-like* drift as shown in Fig. 1, preventing accurate recovery. Quantitatively, the mean point estimation errors for the iterative attack in Scenes 1, 2, and 3 are 163, 55, and 122 pixels, respectively. This demonstrates the structural advantage of DCL rather than an outlier artifact of the recovery algorithm [5].

Furthermore, it is important to note that this attack assumes 100% accurate neighborhood estimation between server-side inliers and *outliers* defined as keypoints from dynamic, privacy-sensitive objects. Since [5] relies on local pattern matching, forming accurate inlier-outlier neighborhoods can be inherently challenging due to several factors, including i) inliers being potentially geometrically far from outliers, ii) the transient nature of privacy-sensitive objects (*e.g.* motion or occlusions) changing local patterns and the matching result, and iii) variations in SfM hyperparameters that significantly affect inlier keypoint distribution, potentially leading to poor generalization of attack models similar to [5].



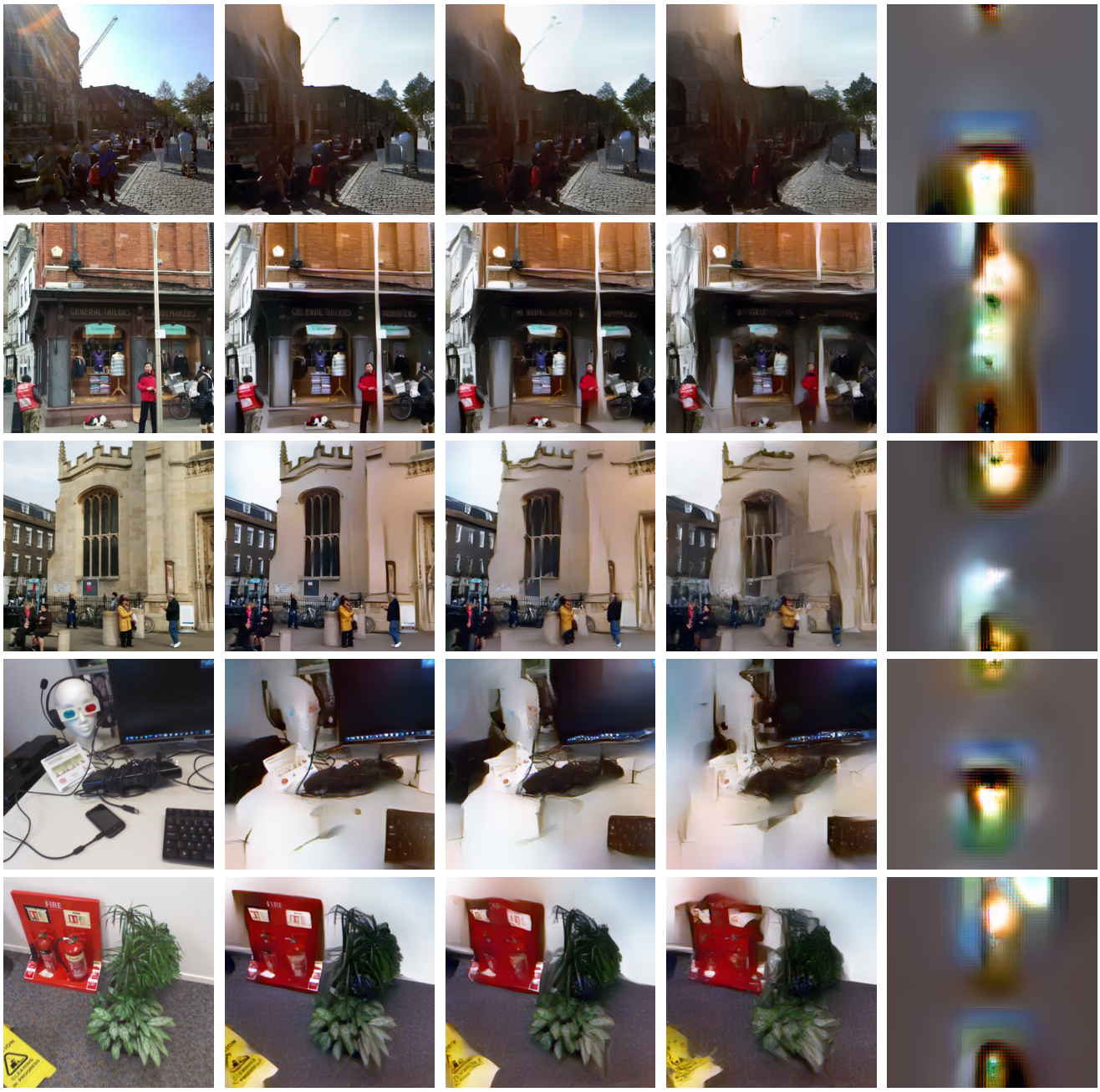
(a) Original Image (b) Feature Points [11] (c) Random Lines [20] (d) Coord. Perm. [15] (e) **DCL (ours)**

Figure 2. Qualitative inversion results using **SuperPoint** features against the geometry-recovery attack [5] in an **expanded neighborhood** ($K = 100$) setting. The reconstructed images are organized by dataset: Row 1 (Aachen), Rows 2–3 (Cambridge), and Rows 4–5 (7Scenes). Feature positions are recovered via various methods: (a) Feature Points, (b) Random Lines [20], (c) Coordinate Permutation [15], and (d) Dual Convergent Lines (ours).



(a) Original Image (b) Feature Points [11] (c) Random Lines [20] (d) Coord. Perm. [15] (e) **DCL (ours)**

Figure 3. Visualization of the inversion network's output using **SuperPoint** features after the geometry-recovery attack [5]. The results are shown at the K=20 neighborhood setting in a $\times 2$ **zoomed-out view** because most recovered points fall outside the original image boundary. The reconstructed images are organized by dataset: Row 1 (Aachen), Rows 2–3 (Cambridge), and Rows 4–5 (7Scenes). This visualization provides clear empirical evidence of the effect of recovered keypoint displacement.



(a) Original Image (b) Feature Points [11] (c) Random Lines [20] (d) Coord. Perm. [15] (e) **DCL (ours)**

Figure 4. Inversion results generated by the **InvSfM** [18] model using **SIFT** features, conducted in the **oracle setting** with $K = 20$. The reconstructed images are organized by dataset: Row 1 (Aachen), Rows 2–3 (Cambridge), and Rows 4–5 (7Scenes). Feature positions are recovered via various methods: (a) Feature Points, (b) Random Lines [20], (c) Coordinate Permutation [15], and (d) Dual Convergent Lines (ours).

References

[1] Eric Brachmann and Carsten Rother. Visual camera re-localization from RGB and RGB-D images using DSAC. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. 2

[2] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6684–6692, 2017. 2

[3] Eric Brachmann, Martin Humenberger, Carsten Rother, and Torsten Sattler. On the limits of pseudo ground truth in visual camera re-localisation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6218–6228, 2021. 3

[4] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5044–5053, 2023. 2, 3

[5] Kunal Chelani, Assia Benbihi, Fredrik Kahl, Torsten Sattler, and Zuzana Kukelova. Obfuscation based privacy preserving representations are recoverable using neighborhood information. In *International Conference on 3D Vision (3DV)*, 2025. 2, 3, 4, 5, 6

[6] Deeksha Dangwal, Vincent T. Lee, Hyo Jin Kim, Tianwei Shen, Meghan Cowan, Rajvi Shah, Caroline Trippel, Brandon Reagen, Timothy Sherwood, Vasileios Balntas, Armin Alaghi, and Eddy Ilg. Analysis and mitigations of reverse engineering attacks on local feature descriptors, 2021. 2

[7] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW)*, pages 224–236, 2018. 2

[8] Alex Kendall, Matthew Grimes, and Roberto Cipolla. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2938–2946, 2015. 3

[9] Jeonggon Kim, Heejoon Moon, and Je Hyeong Hong. Revisiting Geometric Obfuscation for Privacy-Preserving Image Queries in Visual Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 3, 4

[10] Viktor Larsson. PoseLib - Minimal Solvers for Camera Pose Estimation. https://github.com/vlarsson/PoseLib, 2020. Accessed: 2022-10-30. 3

[11] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2, 5, 6, 7

[12] Tony Ng, Hyo Jin Kim, Vincent T Lee, Daniel DeTone, Tsun-Yi Yang, Tianwei Shen, Eddy Ilg, Vassileios Balntas, Krystian Mikolajczyk, and Chris Sweeney. NinjaDesc: Content-concealing visual descriptors via adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12797–12807, 2022. 2

[13] Niantic Labs. Niantic research presents: State of the art in relocalization with machine learning, 2023. 3

[14] Niantic Spatial, Inc. Niantic spatial vps. https://www.nianticspatial.com/docs/nsdk/features/lightship_vps/, 2025. 3

[15] Linfei Pan, Johannes L. Schönberger, Viktor Larsson, and Marc Pollefeys. Privacy preserving localization via coordinate permutations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18174–18183, 2023. 2, 3, 5, 6, 7

[16] Maxime Pietrantoni, Gabriela Csurka, and Torsten Sattler. Gaussian splatting feature fields for (privacy-preserving) visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1082–1092, 2025. 3

[17] Maxime Pietrantoni, Torsten Sattler, and Gabriela Csurka. Vulnerability of privacy-preserving visual localization against diffusion-based attacks. https://openreview.net/forum?id=NmWf0gLufZ, 2025. 3

[18] Francesco Pittaluga, Sanjeev J Koppal, Sing Bing Kang, and Sudipta N Sinha. Revealing scenes by inverting structure from motion reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 145–154, 2019. 2, 3, 7

[19] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12716–12725, 2019. 2, 3

[20] Pablo Speciale, Johannes L Schonberger, Sudipta N

Sinha, and Marc Pollefeys. Privacy preserving image queries for camera localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1486–1496, 2019. 2, 3, 5, 6, 7

[21] Fangjinhua Wang, Xudong Jiang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. Glace: Global local accelerated coordinate encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[22] Qunjie Zhou, Sérgio Agostinho, Aljosa Osep, and Laura Leal-Taixé. Is geometry enough for matching in visual localization? In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 407–425. Springer, 2022. 3