GEOCIRCUIT: TRAINING-FREE TASK-AWARE HEAD-CIRCUIT EXTRACTION FOR VISUAL GEOMETRY TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Feed-forward 3D reconstruction models such as the Visual Geometry Grounded Transformer (VGGT) use a single network to predict camera pose, depth, and point maps. However, their deployment has been constrained by the billion-scale parameter and quadratic complexity. Existing VGGT acceleration methods mainly reduce tokens or sparsify attention entries, while the redundancy and task-dependent roles of complete attention heads remain underexplored. We instead ask whether a frozen visual geometry transformer contains smaller, structurally executable head circuits specialized for different geometric task sets. To investigate this question, we present **GeoCircuit**, which instruments both frame and global attention with complete-head gates and treats the pretrained model as a frozen parent network of candidate geometric subnetworks. Using dense-model predictions as label-free task teachers, **GeoCircuit** estimates task-aware gate Gauss-Newton matrices and applies shrinkage to preserve useful cross-head interactions while suppressing estimation noise. Given a target head retention ratio, **GeoCircuit** directly selects a fixed-budget circuit through a normalized multi-task objective, producing subnets of different sizes without fine-tuning. With 50% of heads retained, **GeoCircuit** outperforms state-of-the-art method at the corresponding retention setting by 5.93 percentage points on camera pose estimation and improves the point map estimation results by 43.5%.

1 INTRODUCTION

Feed-forward visual geometry models recover 3D scene structure from multiple images in a single forward pass. The Visual Geometry Grounded Transformer (VGGT) alternates frame and global attention to jointly predict camera poses, depth, and point maps (Wang et al., 2025a). However, its billion-scale parameter count and repeated attention operations impose substantial computational and memory costs. Moreover, downstream applications do not always require all of these outputs. For example, PointWorld (Huang et al., 2026) uses VGGT for camera pose initialization in robotic data annotation, while Prior Depth Anything (Wang et al., 2025c) uses VGGT's depth and confidence predictions for depth refinement. These costs and selective output requirements motivate identifying smaller subnetworks that preserve the predictions each application needs.

Existing VGGT acceleration methods reduce numerical precision, merge tokens, or sparsify attention (Feng et al., 2026; Shen et al., 2026; Wang et al., 2026). Several studies further show that VGGT's sensitivity varies across tasks: FGQ (Zhang et al., 2026) reports different quantization sensitivity profiles across tasks, blocks, and channels, while HeSS (Kim et al., 2026) identifies attention heads with different sensitivities to camera pose and point cloud errors. However, they generally keep every complete attention head active. Therefore, whether complete heads can be removed from both frame and global attention to form smaller task-aware subnetworks remains unexplored.

Circuit extraction provides a useful perspective for studying this question. In mechanistic interpretability, a circuit is commonly viewed as a subset of model components that supports a specific behavior (Elhage et al., 2021; Conmy et al., 2023). Prior studies have shown that Transformer heads contain substantial redundancy while some heads learn specialized functions (Michel et al., 2019; Voita et al., 2019). These findings motivate investigating whether VGGT contains shared and task-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

dependent computation. We therefore define a geometric head circuit as a subset of complete attention heads that preserves a requested set of outputs from the frozen model. Extracting circuits for pose, depth, and point-map prediction allows us to identify the heads shared across tasks and those required by individual tasks, providing both compact execution paths and a clearer understanding of how VGGT supports multiple geometric outputs.

However, extracting circuits from VGGT presents two linked challenges. First, selecting heads independently can overlook their interactions: heads with overlapping functions may have little effect when removed individually, yet removing them together can impair performance (Kwon et al., 2022). Second, estimating these interactions involves a trade-off between calibration cost and reliability. Using a few unlabeled input images to assess head removal keeps extraction inexpensive, but provides limited data for estimating a full interaction matrix, whose size grows quadratically with the number of heads. For covariance-based estimates, sampling noise becomes a concern when the number of heads is large relative to the calibration sample size (Ledoit & Wolf, 2004). The challenge is therefore to account for head interactions while keeping their estimation reliable and inexpensive. Moreover, training or fine-tuning a separate model for each task and budget would add further computational cost. These considerations motivate a training-free method that estimates the joint effect of head removal from a small calibration set and extracts circuits with a prescribed number of retained heads.

To this end, we introduce **GeoCircuit**, a training-free framework that places complete-head gates in both frame and global attention. Dense-model predictions provide label-free targets for pose, depth, and point-map fidelity losses. For each task, a gate-space generalized Gauss–Newton matrix models individual head sensitivities and interactions between heads removed together. We shrink the estimated matrix toward its diagonal to reduce interaction noise, then select a circuit under a prescribed head budget. A normalized multi-task objective balances fidelity across the requested outputs. The retained heads can be packed into narrower query, key, value, and output projections, yielding a structurally smaller attention network with unchanged pretrained weights. Experiments on VGGT demonstrate the effectiveness of this approach. At 50% head retention, **GeoCircuit** improves camera pose results by 5.93 percentage points and improves point map estimation results by 43.5% relative to HeSS at the corresponding retention setting. We further measure end-to-end inference latency on 100-frame sequences, demonstrating that our method delivers $1.63\times$ to $1.80\times$ speedup over VGGT, and increasing to $1.95\times$ to $2.34\times$ when combined with SparseVGGT (Wang et al., 2026).

Our main contributions are summarized as follows:

- We formulate VGGT efficiency as task-aware geometric circuit extraction. This formulation treats a frozen VGGT as a parent network containing candidate head subnetworks and considers complete heads across both frame and global attention.
- We introduce a label-free and training-free circuit estimator based on gate-space generalized Gauss-Newton matrices. Our shrinkage formulation uses cross-head interactions while reducing the effect of noise from limited calibration data.
- We develop a fixed-budget selector that extracts circuits for individual geometric tasks or a requested combination of tasks. The selected circuits can be structurally packed for efficient execution without fine-tuning the original model.
- We evaluate the extracted circuits through end-to-end task accuracy, interaction-aware versus independent selection, cross-task circuit transfer, and measured hardware efficiency.

2 Related Work

Efficient Feed-forward 3D Reconstruction. Recent work accelerates feed-forward 3D reconstruction through quantization, token reduction, and sparse attention. QuantVGGT (Feng et al., 2026) mitigates activation outliers and stabilizes calibration, while FGQ (Zhang et al., 2026) incorporates task-dependent sensitivity into quantization. FastVGGT (Shen et al., 2026) merges redundant tokens for global attention, and Spark3R (Tang et al., 2026) combines query-token merging with key–value token pruning. SparseVGGT (Wang et al., 2026) introduces block-sparse global attention, while AVGGT (Sun et al., 2026) converts early global layers to frame attention and subsamples keys and

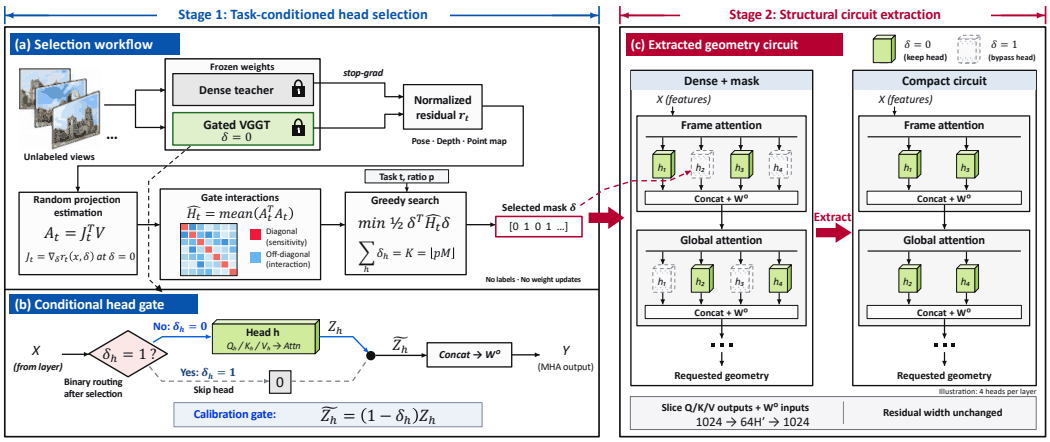


Figure 1: Overview of **GeoCircuit**. (a) Frozen dense-model predictions on unlabeled views define task residuals. Random projections estimate head sensitivities and interactions for greedy selection under a head budget. (b) Binary gates retain ($\delta_h = 0$) or bypass ($\delta_h = 1$) each head. (c) Extracting compact circuits by selected masks.

values. HeSS (Kim et al., 2026) further uses curvature-based sensitivity to geometric errors to redistribute attention-block budgets among global-attention heads. Our focus is task-conditioned circuit extraction: identifying subsets of complete heads across frame and global attention that jointly preserve requested geometric outputs. Whereas HeSS allocates sparsity within heads, **GeoCircuit** models cross-head interactions using label-free dense-model targets to select a fixed circuit for each task set and head budget, yielding compact executable subnetworks without updating pretrained weights.

Circuit Extraction. Circuit extraction seeks subnetworks that preserve a pretrained transformer's behavior on a specified task (Elhage et al., 2021). ACDC (Conmy et al., 2023) iteratively removes connections through activation patching, while EAP (Syed et al., 2024) and EAP-IG (Hanna et al., 2024) use gradient attribution to accelerate circuit discovery. Edge Pruning (Bhaskar et al., 2024) optimizes connection masks, while DiscoGP (Yu et al., 2025) jointly learns weight and connection masks. **GeoCircuit** shares their objective of preserving task-specific behavior and extends this perspective to multi-view geometry. We extract circuits of complete attention heads across frame and global attention, using dense-model predictions as label-free targets. Our shrinkage-regularized quadratic model captures joint head effects and guides selection under a fixed retention budget, without iterative mask training. The extracted circuits support task-level sufficiency and necessity evaluation and can be structurally packed for efficient inference.

3 Proposed GeoCircuit Framework

3.1 Preliminaries and Circuit Formulation

Visual Geometry Grounded Transformer. The Visual Geometry Grounded Transformer (VGGT) is a feed-forward model that predicts 3D scene geometry from multiple images (Wang et al., 2025a). Given images $x = (I_v)_{v=1}^V$ of the same scene, its backbone alternates frame attention, which processes tokens within each image, and global attention, which exchanges information across views. The resulting representations support camera pose, depth, and point-map prediction. For circuit extraction, we consider all attention heads in these alternating backbone layers, covering both within-image and cross-view computation. Let L be the number of candidate attention layers and H_ℓ the number of heads in layer ℓ. The total number of candidate heads is $M = \sum_{\ell=1}^{L} H_\ell$.

Circuit Extraction. Following the circuit view of neural networks, we define a circuit as a subnetwork of the frozen model comprising selected components and the original computations connecting them to the requested outputs. We extract head-level circuits by applying binary gates to

complete attention heads while keeping all other model components unchanged. The removal indicator $\delta_{\ell,h} \in \{0,1\}$ equals 1 when head h in layer ℓ is removed and 0 otherwise. For input tokens X_ℓ, the gated attention output is

$$A_\ell(X_\ell;\delta_\ell) = b_\ell^O + \sum_{h=1}^{H_\ell}(1-\delta_{\ell,h})Z_{\ell,h}(X_\ell)W_{\ell,h}^O, \tag{1}$$

where $Z_{\ell,h}(X_\ell)$ is the output of head h, $W_{\ell,h}^O$ is its block of the output projection, and b_ℓ^O is the shared output bias. Removing a head eliminates its entire projected contribution while preserving the shared bias and residual connection. Collecting the removal indicators across layers gives $\delta \in \{0,1\}^M$, with $\delta = \mathbf{0}$ recovering the dense model.

Task Set and Circuit Size. Circuit extraction is guided by the geometric outputs to preserve and the number of heads to retain. We denote the nonempty set of requested tasks by $\mathcal{T} \subseteq \{\text{pose}, \text{depth}, \text{point}\}$. For a target head retention ratio $\rho \in [0,1]$, we require

$$\sum_{i=1}^{M}(1-\delta_i) = M - K, \qquad K = \lfloor (1-\rho)M \rfloor. \tag{2}$$

Here K is the number of excluded heads, and the circuit $\mathcal{C}(\delta) = \{i \in \{1,\dots,M\} : \delta_i = 0\}$ contains $M - K = \lceil \rho M \rceil$ heads. Thus, $\rho = 1$ recovers the dense model, and smaller values specify smaller circuits. Among circuits of this size, we seek one that best preserves the dense model's predictions for $\mathcal{T}$ on unlabeled calibration scenes. The budget applies across all candidate layers, allowing the retained head count to vary across layers and attention types. Once selected for a task set and budget, the circuit is fixed across inference scenes. Since evaluating every candidate circuit is impractical, we next construct a tractable predictor of the fidelity loss caused by joint head removal.

3.2 Reliable Modeling of Head Interactions

Extracting a task-conditioned circuit requires assessing how well a set of retained heads preserves the requested geometric outputs. As discussed in the introduction, scoring heads independently overlooks their interactions, while estimating these interactions from limited calibration data can be unreliable. We address these challenges by constructing a task-specific predictor of circuit fidelity. Using dense-model predictions as label-free targets, we derive a local quadratic model that captures both individual head sensitivities and pairwise interactions. We estimate this model through random projections of output residuals, then shrink its interaction terms toward zero to limit the influence of estimation noise. The resulting predictor guides circuit selection at the specified circuit size.

Defining label-free task fidelity. Circuit selection requires measuring how well a candidate preserves the requested geometric outputs without ground-truth labels. We use the dense model's predictions as fixed targets and compare each circuit against them on the same input scenes. Let $F_t(x;\delta)$ denote the prediction for task t on scene x, and let $y_t^0(x) = F_t(x;\mathbf{0})$ be the dense prediction with all heads retained. A task-specific transform ψ_t converts each prediction into a vector of geometric quantities for comparison: relative rotations, relative translations, and focal lengths for pose; log-depth values for depth; and corresponding 3D points in a shared coordinate system for point maps. The transform normalizes translations and 3D points by a dense-model scene scale and scales each residual block so its squared norm averages over its coordinates. Dense-model confidence further weights the depth and point comparisons. All scales, confidence weights, and validity masks are fixed from the dense prediction, so each circuit is assessed using the same reference. We define the task residual vector $r_t(x,\delta)$ as the coordinate-wise difference between the circuit's and dense model's transformed predictions:

$$r_t(x,\delta) = \psi_t(F_t(x;\delta)) - \psi_t(y_t^0(x)). \tag{3}$$

The fidelity loss is half the mean squared residual norm over the N unlabeled calibration scenes in $\mathcal{D}_{\text{cal}}$:

$$\mathcal{L}_t(\delta) = \frac{1}{2N}\sum_{x\in\mathcal{D}_{\text{cal}}} \|r_t(x,\delta)\|_2^2. \tag{4}$$

When all heads are retained, the circuit matches the dense model, making the residual and loss zero. This provides the starting point for the local approximation below. Calibration and selection use frozen weights and evaluation mode.

Modeling individual and joint head effects. Circuit selection need to compare many head subsets, but evaluating each one with a full network pass is impractical. We therefore approximate the fidelity loss with a quadratic predictor: squared removal indicators describe individual head effects, and products of two indicators describe pairwise interactions. To derive this predictor, we temporarily treat each binary indicator δ_i as a real-valued variable in Eq. 1. A head's contribution is multiplied by $1 - \delta_i$, so an intermediate value such as $\delta_i = 0.5$ halves that contribution. Allowing continuous gate values lets us differentiate the task residual with respect to the gates at $\delta = \mathbf{0}$, where all heads are retained. We use these derivatives to construct the local fidelity predictor. During circuit selection, each gate is restricted to 0 or 1, so each head is either fully retained or removed. Let $J_t(x)$ denote the residual Jacobian, whose ith column measures the response to a small change in δ_i. Averaging the inner products between these columns over calibration scenes defines the matrix H_t:

$$J_t(x) = \left.\frac{\partial r_t(x,\delta)}{\partial \delta}\right|_{\delta=\mathbf{0}}, \qquad H_t = \frac{1}{N}\sum_{x\in\mathcal{D}_{\text{cal}}} J_t(x)^\top J_t(x). \tag{5}$$

The following proposition establishes why $H_t \in \mathbb{R}^{M\times M}$ determines the local fidelity predictor.

Proposition 3.1 (Local circuit fidelity) *Suppose each residual in Eq. 3 is twice continuously differentiable near $\delta = \mathbf{0}$. Since the dense-model residual is zero, $\nabla\mathcal{L}_t(\mathbf{0}) = \mathbf{0}$ and $\nabla^2\mathcal{L}_t(\mathbf{0}) = H_t$. Thus, the second-order Taylor approximation of the fidelity loss at the dense model is*

$$q_t(\delta) = \frac{1}{2}\delta^\top H_t \delta. \tag{6}$$

Appendix A.1 provides a self-contained proof and an approximation error bound. At the dense model, both the fidelity loss and its first derivative are zero because the predictions match the reference exactly. The second-order Taylor approximation therefore contains only the quadratic term, which estimates the loss caused by perturbation in head contributions. The matrix H_t is the gate-space generalized Gauss-Newton matrix (Martens, 2020), and the proposition shows that it is also the exact loss Hessian at this point.

To apply this predictor to actual circuits, we return to binary indicators. Since $\delta_i^2 = \delta_i$, its squared terms become individual removal costs, while the products retain the pairwise interactions:

$$q_t(\delta) = \underbrace{\frac{1}{2}\sum_{i=1}^{M} H_{t,ii}\delta_i}_{\text{individual sensitivities}} + \underbrace{\sum_{i<j} H_{t,ij}\delta_i\delta_j}_{\text{pairwise interactions}}. \tag{7}$$

The diagonal measures each head's local sensitivity. An off-diagonal entry $H_{t,ij}$ is the additional predicted fidelity loss from removing heads i and j together, relative to the sum of their separate predicted losses. A positive entry indicates reinforcing residual changes while a negative entry indicates partial cancellation. Since the loss is measured at the final task output, these terms can connect heads in different layers and in frame and global attention.

Estimating interactions with random projections. Estimating H_t requires information about how each head affects the task residual vector $r_t(x,\delta)$, which measures differences between the circuit's and dense model's transformed predictions. Because this vector contains many geometric coordinates, explicitly constructing its Jacobian with respect to all head gates is expensive. We instead project the residual vector onto random directions and differentiate each resulting scalar to obtain head sensitivities along that direction. For every scene and task, we draw S independent vectors $v_{t,x,s}$ whose entries are independently $+1$ or -1 with equal probability. Each vector has the same dimension as the residual vector and satisfies $\mathbb{E}[v_{t,x,s}v_{t,x,s}^\top] = I$. The gradient of a projected residual with respect to the removal indicators at the dense model is

$$a_{t,x,s} = \nabla_\delta\left[v_{t,x,s}^\top r_t(x,\delta)\right]\big|_{\delta=\mathbf{0}} = J_t(x)^\top v_{t,x,s}. \tag{8}$$

Averaging the outer products of these gradients over scenes and probes gives the interaction estimate:

$$\widehat{H}_t = \frac{1}{NS}\sum_{x\in\mathcal{D}_{\text{cal}}}\sum_{s=1}^{S} a_{t,x,s}a_{t,x,s}^\top. \tag{9}$$

The following proposition establishes the properties of this estimate.

Proposition 3.2 (Unbiased interaction estimation) *For a fixed calibration set and the random probes defined above, the estimator in Eq. 9 is unbiased for H_t and positive semidefinite:*

$$\mathbb{E}_v[\widehat{H}_t] = H_t, \qquad \widehat{H}_t \succeq 0. \tag{10}$$

Here, the expectation is over the random probes. Thus, the estimator recovers both individual head sensitivities and pairwise interactions in expectation, and each realization defines a nonnegative quadratic predictor. Appendix A.2 provides the proof.

Each differentiation returns an M-dimensional vector containing all head sensitivities to one projected residual. Thus, S such operations per scene and task estimate the interaction matrix without storing the full residual Jacobian. We differentiate the projected residual because its gate derivatives can be nonzero at the dense model, whereas the gradient of the squared fidelity loss vanishes there. All model weights remain frozen. With finitely many probes, individual interaction estimates can still be noisy, motivating the shrinkage step below.

Stabilizing interactions with shrinkage. Noise in the estimated interactions can mislead circuit selection. To limit its influence, we shrink the interaction terms toward zero while preserving individual head sensitivities. Let $\alpha \in [0, 1]$ denote the weight on the estimated interactions and Diag keep only a matrix's diagonal entries. The stabilized estimate is

$$\widetilde{H}_t = (1 - \alpha)\,\mathrm{Diag}(\widehat{H}_t) + \alpha\widehat{H}_t, \qquad \alpha \in [0, 1], \tag{11}$$

This leaves individual sensitivities unchanged and multiplies every pairwise interaction by α, so smaller values impose stronger shrinkage. The matrix remains positive semidefinite. At $\alpha = 0$, the predictor uses only individual sensitivities; at $\alpha = 1$, it uses the full estimated interactions without shrinkage. We denote the resulting predictor by $\widetilde{q}_t$, where the tilde indicates that H_t in Eq. 6 is replaced by the shrinkage-stabilized estimate $\widetilde{H}_t$.

3.3 Task-Conditioned Circuit Extraction

The fidelity predictors provide the costs for selecting a circuit at the requested head retention ratio. Selection combines the task losses into a normalized objective and uses greedy head removal followed by swaps to refine the circuit. Validation against the dense model determines the interaction weight α, after which we pack the selected circuit for execution.

Balancing fidelity across requested tasks. A joint circuit should preserve all requested outputs, but pose, depth, and point-map losses measure different quantities. Combining these losses requires normalization so that their scales do not dominate selection. We specify a positive reference loss β_t for each task and express its predicted fidelity loss relative to this scale. The reference losses are fixed before circuit selection and shared across candidate circuits. We define $\Phi_{\mathcal{T}}$ as the largest normalized predicted loss across the requested tasks and minimize it subject to Eq. 2:

$$\Phi_{\mathcal{T}}(\delta) = \max_{t \in \mathcal{T}} \frac{\widetilde{q}_t(\delta)}{\beta_t}, \qquad \min_{\delta} \Phi_{\mathcal{T}}(\delta). \tag{12}$$

For one task, this reduces to minimizing its predicted fidelity loss. For multiple tasks, it minimizes the largest normalized loss, preventing a small loss on one task from compensating for a large relative loss on another.

Selecting heads with greedy search. Finding the best circuit by exhaustive search is impractical because of the number of possible head subsets. We instead start from the dense model and remove one head at a time until $M - K = \lceil \rho M \rceil$ heads remain. Each choice accounts for the heads already removed through the joint fidelity objective. Let e_i be the vector with a 1 at position i and zeros elsewhere, so $\delta + e_i$ additionally removes a currently retained head i. The corresponding change in task t's predicted loss, denoted by $m_{t,i}(\delta)$, is

$$m_{t,i}(\delta) = \widetilde{q}_t(\delta + e_i) - \widetilde{q}_t(\delta) = (\widetilde{H}_t\delta)_i + \frac{1}{2}\widetilde{H}_{t,ii}. \tag{13}$$

At each step, we remove the head that minimizes the updated joint objective $\Phi_{\mathcal{T}}(\delta + e_i)$. For a single task, this is the head with the smallest $m_{t,i}(\delta)$. For multiple tasks, we compare the updated

normalized losses across all requested tasks, since the task with the largest loss can change after a removal. The vector $\widetilde{H}_t\delta$ is updated by adding column i of $\widetilde{H}_t$, avoiding repeated matrix–vector products. Appendix B details the subsequent swap refinement and the complete search procedure.

Choosing shrinkage by circuit validation. The interaction weight α determines how much estimated interactions influence the selected circuit. To choose it, we compare the resulting circuits against the dense model on separate unlabeled scenes, using their actual outputs to assess fidelity. For each $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$, we construct the task matrices and run the same circuit search at the requested head retention ratio. We then evaluate each candidate circuit against the dense model on an unlabeled validation set. Let $\mathcal{L}_t^{\text{val}}$ denote Eq. 4 evaluated on that set. We select the candidate with the smallest $\max_{t\in\mathcal{T}} \mathcal{L}_t^{\text{val}}(\delta)/\beta_t$. The task scales remain fixed across all candidates, and no geometric labels or weight updates are needed.

4 Evaluation

4.1 Experimental setup

Evaluation settings. We primarily evaluate **GeoCircuit** on the open-source VGGT-1.2B model (Wang et al., 2025a) using a single NVIDIA A100 GPU with 40 GB of memory. We set the interaction weight to $\alpha = 0.5$, with circuit size specified by the head retention ratio ρ. Calibration uses 30 unlabeled RGB images from five 7-Scenes sequences (Shotton et al., 2013), with supervision provided by the frozen dense model and no pretrained weight updates. We compare against HeSS (Kim et al., 2026), which assigns different token-level sparsity ratios to attention heads according to their sensitivity. Both methods are applied to all frame and global attention heads. To evaluate compatibility with block-sparse attention, we also apply **GeoCircuit** to SparseVGGT (Wang et al., 2026), fixing its attention-block pruning ratio at 25% or 50%. For these hybrids, ρ specifies the percentage of heads retained, independently of the fixed block sparsity. Detailed calibration and implementation settings are provided in Appendix D. To further test generalization, we also extend **GeoCircuit** to π^3 model (Wang et al., 2025b). The result can be found in Appendix F.

Dataset and Metrics. We evaluate camera pose estimation on Co3Dv2 (Reizenstein et al., 2021), which contains videos of everyday objects with camera-pose annotations. Point-map reconstruction is evaluated on DTU (Jensen et al., 2014), a multi-view stereo (MVS) benchmark with controlled image capture and reference 3D scans. Following VGGT (Wang et al., 2025a), each image pair's pose error is the maximum of its relative rotation and translation angular errors. We report the area under the cumulative error curve (AUC), computed up to each evaluation threshold and normalized by the threshold. For MVS, we report Accuracy (Acc), Completeness (Comp), and Chamfer Distance (CD). Accuracy is the mean nearest-neighbor Euclidean distance from predicted points to ground-truth points, while Completeness measures the reverse direction. Chamfer Distance is their arithmetic mean. Higher AUC and lower Acc, Comp, and CD indicate better performance.

4.2 Performance vs retention ratio

Figure 2 compares **GeoCircuit**, HeSS, and SparseVGGT+**GeoCircuit** as retention decreases from 100% to 50%, with SparseVGGT block pruning fixed at 25% or 50%. At 50% retention, **GeoCircuit** exceeds HeSS by 5.93 percentage points in AUC@30 on Co3Dv2. On DTU, it reduces Chamfer distance by 43.5% relative to HeSS at the same operating point. Combining circuit extraction with SparseVGGT further improves pose performance: SparseVGGT+**GeoCircuit** with 25% block pruning gains 1.02 AUC@30 points over standalone **GeoCircuit** at this head budget. At both block pruning levels, SparseVGGT+**GeoCircuit** maintains lower reconstruction errors than HeSS, although its errors are higher than those of standalone **GeoCircuit**. These results demonstrate that task-conditioned circuit extraction remains effective alongside block-sparse attention.

At 100% head retention, SparseVGGT+**GeoCircuit** retains all heads and therefore corresponds to SparseVGGT alone. Reducing head retention to 90% improves AUC@30 by 2.56 percentage points under 50% block pruning. This is because the removed heads contribute little to the target task while still propagating substantial errors introduced by block pruning to the final output (see Appendix E). By excluding these heads, **GeoCircuit** suppresses error propagation while preserving task-relevant computation, improving pose accuracy over SparseVGGT alone.

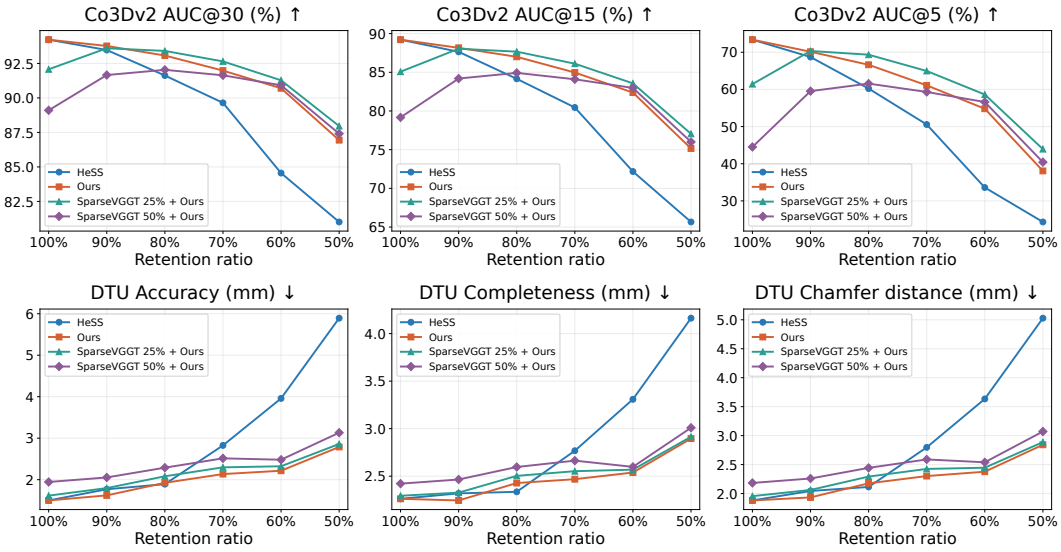


Figure 2: Comparison of **GeoCircuit**, HeSS, SparseVGGT+**GeoCircuit** across retention ratios on Co3Dv2 and DTU. Higher AUC and lower Acc, Comp, and CD indicate better performance.

4.3 Task-conditioned evaluation

To assess whether task-specific selection benefits the corresponding task, we compare pose-conditioned and point-map-conditioned masks at matched head retention ratios on both benchmarks, as shown in Figure 3. On Co3Dv2, the pose-conditioned mask achieves 93.52 AUC@30 at 80% retention, compared with 66.43 for the point-map-conditioned mask. The advantage widens at 50% retention, where the two masks achieve 87.15 and 6.32, respectively. On DTU, the point-map-conditioned mask yields a lower CD of 1.21 mm at 80% retention and 2.08 mm at 50%, compared with 1.79 mm and 4.77 mm for the pose-conditioned mask. These results support task-conditioned circuit extraction: matching the selection objective to the evaluated task yields better performance, with larger gains at 50% retention.

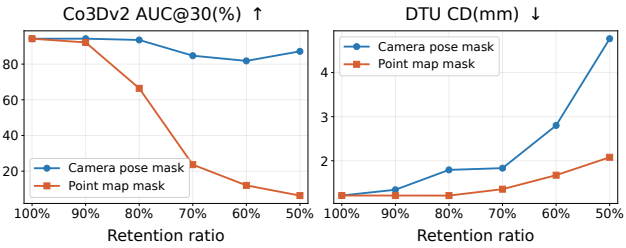


Figure 3: Pose-conditioned and point-map-conditioned circuits evaluated on Co3Dv2 and DTU at matched head retention ratios.

4.4 Sensitivity study

We vary the interaction weight $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$ across head retention ratios from 90% to 50%. Recall that $\alpha = 0$ uses only individual head sensitivities, while $\alpha = 1$ includes all estimated interactions without shrinkage. We also visualize the shrinkage map in Appendix C. As shown in Table 1, no single value performs best at every retention ratio. Nevertheless, $\alpha = 0.5$ achieves both the highest mean AUC@30 on Co3Dv2 (91.29%) and the lowest mean Acc on DTU (2.137 mm), with each mean computed over the five sparse settings. It also performs best on both metrics at 50% retention, the smallest circuit budget evaluated. Under Eq. 11, this setting halves the estimated off-diagonal interactions while preserving individual head sensitivities. Together, these results support the existing choice of $\alpha = 0.5$ as a shared default across tasks and circuit sizes.

Table 1: Sensitivity to interaction weight α. Mean denotes the unweighted average over the five sparse retention ratios (90%–50%), excluding the dense model (100%). Bold indicates the best value within each metric for each sparse row and the mean row.

	Co3Dv2 AUC@30 (%) ↑					DTU Acc (mm) ↓				
Retention	$\alpha = 0$	0.25	0.5	0.75	1	$\alpha = 0$	0.25	0.5	0.75	1
100%	94.22	94.22	94.22	94.22	94.22	1.499	1.499	1.499	1.499	1.499
90%	93.94	93.88	93.77	**94.02**	93.65	1.626	1.639	1.621	**1.576**	1.645
80%	89.71	92.82	93.07	**93.10**	93.04	1.822	1.986	1.924	1.965	**1.814**
70%	87.26	92.13	91.98	**92.59**	92.19	2.305	2.159	2.135	2.065	**2.034**
60%	87.68	**91.51**	90.70	88.76	88.67	2.850	2.349	**2.216**	2.855	3.066
50%	85.78	83.60	**86.94**	86.00	85.22	3.247	3.267	**2.788**	3.086	2.869
Mean	88.88	90.79	**91.29**	90.89	90.55	2.370	2.280	**2.137**	2.309	2.286

4.5 Efficiency analysis

We evaluate end-to-end inference latency on three Co3Dv2 sequences (handbag, sandwich, and parking meter), using 100 distinct frames per sequence and a single NVIDIA A100 40GB GPU per run. We time the complete forward pass, including all prediction heads, excluding model loading, preprocessing, and downstream reconstruction. Our method uses 50% head retention, and SparseVGGT uses 50% block sparsity. As shown in Figure 4, physically removing the pruned heads achieves $1.63\times$ to $1.80\times$ speedup over VGGT, compared with $1.31\times$ to $1.44\times$ for SparseVGGT. Combining our method with SparseVGGT increases the speedup to $1.95\times$ to $2.34\times$ over VGGT, providing an additional $1.49\times$ to $1.63\times$ acceleration over SparseVGGT alone. For example, on the sandwich sequence, our integration reduces SparseVGGT latency from 10.60 s to 6.52 s, compared with 15.28 s for VGGT. These results demonstrate that shrinkage-based head pruning yields practical inference gains and provides complementary acceleration when combined with sparse attention.

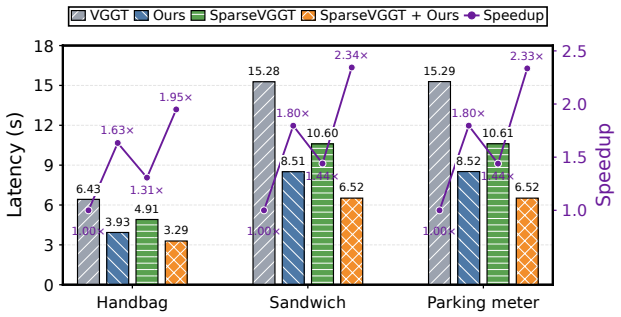


Figure 4: End-to-end latency and speedup relative to VGGT.

5 Conclusion

We presented **GeoCircuit**, a training-free framework for extracting compact, task-aware head circuits from a frozen visual geometry transformer. Treating the pretrained model as a parent network, **GeoCircuit** selects complete heads across frame and global attention using dense-model predictions as label-free targets. Task-aware gate-space Gauss-Newton matrices model joint head effects, while shrinkage limits the influence of estimation noise. A normalized multi-task objective guides selection at a prescribed head retention ratio, yielding structurally executable circuits for different tasks without fine-tuning. Experiments on camera pose estimation and point-map reconstruction show that the extracted circuits better preserve task performance than HeSS at reduced retention ratios, supporting task-aware circuit extraction as a way to specialize a unified geometry model.

AI use statement

In this work, we used generative AI tools to assist with manuscript drafting and language polishing, as well as the development and debugging of experimental code. All experimental and debugging suggestions generated by AI were carefully reviewed and approved by the authors. The authors take

full responsibility for the final manuscript, experimental results, and associated code, including all AI-assisted content.

Ethics statement

This work studies task-conditioned circuit extraction for efficient 3D reconstruction using the publicly available VGGT model (Wang et al., 2025a) and existing benchmark datasets. The reported experiments involve no human-subject studies or collection of new personal data. Our focus is algorithmic analysis and geometric prediction; downstream deployment should account for privacy, fairness, and safety considerations in the intended application.

Reproducibility statement

To support reproducibility, we describe the circuit formulation and selection procedure in Section 3 and Appendix B. Experimental settings, datasets, and evaluation metrics are provided in Section 4, with calibration and implementation details in Appendix D. Appendix A provides proofs of the theoretical results. We will publicly release the source code for reproducing our results upon acceptance.

References

Adithya Bhaskar, Alexander Wettig, Dan Friedman, and Danqi Chen. Finding transformer circuits with edge pruning. *Advances in Neural Information Processing Systems*, 37:18506–18534, 2024.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

Weilun Feng, Haotong Qin, Mingqiang Wu, Chuanguang Yang, Yuqi Li, Xiangqi Li, Zhulin An, Libo Huang, Yulun Zhang, Michele Magno, et al. Quantized visual geometry grounded transformer. In *International Conference on Learning Representations*, volume 2026, pp. 59817–59838, 2026.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. *arXiv preprint arXiv:2403.17806*, 2024.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026.

Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 406–413, 2014.

Yongsung Kim, Wooseok Song, Jaihyun Lew, Hun Hwangbo, Jaehoon Lee, and Sungroh Yoon. Hess: Head sensitivity score for sparsity redistribution in vggt. *arXiv preprint arXiv:2603.25336*, 2026.

Woosuk Kwon, Sehoon Kim, Michael W Mahoney, Joseph Hassoun, Kurt Keutzer, and Amir Gholami. A fast post-training pruning framework for transformers. *Advances in Neural Information Processing Systems*, 35:24101–24116, 2022.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? *Advances in neural information processing systems*, 32, 2019.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10881–10891. IEEE, 2021.

You Shen, Zhipeng Zhang, Yansong Qu, Xiawu Zheng, Jiayi Ji, Shengchuan Zhang, and Liujuan Cao. Fastvggt: Fast visual geometry transformer. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2930–2937, 2013.

Xianbing Sun, Zhikai Zhu, Zhengyu Lou, Bo Yang, Jinyang Tang, Liqing Zhang, He Wang, and Jianfu Zhang. Avggt: Rethinking global attention for accelerating vggt. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 251–260, 2026.

Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 407–416, 2024.

Zecheng Tang, Jiaye Fu, Qiankun Gao, Haijie Li, Yanmin Wu, Jiaqi Zhang, Siwei Ma, and Jian Zhang. Spark3r: Asymmetric token reduction makes fast feed-forward 3d reconstruction. *arXiv preprint arXiv:2605.06270*, 2026.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 5797–5808, 2019.

Chung-Shien Brian Wang, Christian Schmidt, Jens Piekenbrinck, and Bastian Leibe. Block-sparse global attention for efficient multi-view geometry transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14546–14555, 2026.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025b.

Zehan Wang, Siyu Chen, Lihe Yang, Jialei Wang, Ziang Zhang, Hengshuang Zhao, and Zhou Zhao. Depth anything with any prior. *arXiv preprint arXiv:2505.10565*, 2025c.

Lei Yu, Jingcheng Niu, Zining Zhu, Xi Chen, and Gerald Penn. Sheaf discovery with joint computation graph pruning and flexible granularity. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8833–8848, 2025.

Yipu Zhang, Jintao Cheng, Weilun Feng, Jiehao Luo, Chuanguang Yang, Zhulin An, Yongjun Xu, and Wei Zhang. Not all tasks quantize equally: Fisher-guided quantization for visual geometry transformer. *arXiv preprint arXiv:2605.15828*, 2026.

A Analysis of the Fidelity Predictor

A.1 Local approximation of task fidelity

Proof of Proposition 3.1. The proof first identifies the loss derivatives at the dense model and then shows that they yield the stated approximation. Throughout, the removal indicators are real-valued in a neighborhood of $\mathbf{0}$, and the dense targets and all residual weights remain fixed. At $\delta = \mathbf{0}$, the circuit and dense predictions coincide, so $r_t(x, \mathbf{0}) = \mathbf{0}$ and $\mathcal{L}_t(\mathbf{0}) = 0$. Write $r_{t,k}(x, \delta)$ for residual coordinate k and $J_t(x, \delta)$ for the Jacobian evaluated at δ, with $J_t(x, \mathbf{0}) = J_t(x)$. Differentiating the squared residual loss in Eq. 4 gives

$$\nabla \mathcal{L}_t(\delta) = \frac{1}{N} \sum_{x \in \mathcal{D}_{\text{cal}}} J_t(x, \delta)^\top r_t(x, \delta), \tag{A.1}$$

$$\nabla^2 \mathcal{L}_t(\delta) = \frac{1}{N} \sum_{x \in \mathcal{D}_{\text{cal}}} \left[J_t(x, \delta)^\top J_t(x, \delta) + \sum_k r_{t,k}(x, \delta) \nabla^2 r_{t,k}(x, \delta) \right]. \tag{A.2}$$

At $\delta = \mathbf{0}$, the zero residual makes the gradient and the second term in the Hessian vanish. The remaining Hessian term is exactly the matrix in Eq. 5:

$$\nabla \mathcal{L}_t(\mathbf{0}) = \mathbf{0}, \qquad \nabla^2 \mathcal{L}_t(\mathbf{0}) = \frac{1}{N} \sum_{x \in \mathcal{D}_{\text{cal}}} J_t(x)^\top J_t(x) = H_t. \tag{A.3}$$

To establish the local approximation, consider a perturbation $u \in \mathbb{R}^M$ whose segment $\{su : 0 \le s \le 1\}$ lies in the differentiable neighborhood, and define $g(s) = \mathcal{L}_t(su)$. The chain rule gives $g''(s) = u^\top \nabla^2 \mathcal{L}_t(su) u$. Since $g(0) = g'(0) = 0$, integrating twice yields $g(1) = \int_0^1 (1 - s) g''(s)\, ds$. Subtracting $q_t(u) = u^\top H_t u / 2$ therefore gives

$$\mathcal{L}_t(u) - \frac{1}{2} u^\top H_t u = \int_0^1 (1 - s)\, u^\top \left[\nabla^2 \mathcal{L}_t(su) - H_t \right] u\, ds. \tag{A.4}$$

The absolute error is at most

$$\frac{\|u\|_2^2}{2} \sup_{0 \le s \le 1} \|\nabla^2 \mathcal{L}_t(su) - H_t\|_{\text{op}}, \tag{A.5}$$

where $\|\cdot\|_{\text{op}}$ is the operator norm. Continuity of the loss Hessian makes the supremum tend to zero as $u \to \mathbf{0}$. Hence the approximation error divided by $\|u\|_2^2$ tends to zero, establishing that q_t is the second-order Taylor approximation. □

Bounding the approximation error. A stronger smoothness condition gives an explicit bound on the error away from the dense model. Suppose the loss Hessian is κ_t-Lipschitz in operator norm along the segment from $\mathbf{0}$ to u. Then $\|\nabla^2 \mathcal{L}_t(su) - H_t\|_{\text{op}} \le \kappa_t s \|u\|_2$. Applying this bound inside the integral above gives

$$|\mathcal{L}_t(u) - q_t(u)| \le \kappa_t \|u\|_2^3 \int_0^1 s(1 - s)\, ds = \frac{\kappa_t}{6} \|u\|_2^3. \tag{A.6}$$

For a binary removal vector with K active entries, $\|\delta\|_2 = \sqrt{K}$. Even if these assumptions hold along the entire segment, the bound grows as $K^{3/2}$. The local result therefore does not ensure fidelity at low head retention ratios; the extracted circuits are assessed against the dense model on unlabeled validation scenes.

A.2 Properties of the interaction estimate

Proof of Proposition 3.2. Fix the calibration set and take expectations only over the random probes. Since $a_{t,x,s} = J_t(x)^\top v_{t,x,s}$ and $\mathbb{E}_v[v_{t,x,s} v_{t,x,s}^\top] = I$,

$$\mathbb{E}_v[a_{t,x,s} a_{t,x,s}^\top] = J_t(x)^\top \mathbb{E}_v[v_{t,x,s} v_{t,x,s}^\top] J_t(x) = J_t(x)^\top J_t(x). \tag{A.7}$$

Averaging over scenes and probes gives $\mathbb{E}_v[\widehat{H}_t] = H_t$ as defined in Eq. 5. For any $z \in \mathbb{R}^M$,

$$z^\top \widehat{H}_t z = \frac{1}{NS} \sum_{x \in \mathcal{D}_{\text{cal}}} \sum_{s=1}^{S} (a_{t,x,s}^\top z)^2 \geq 0, \tag{A.8}$$

which proves positive semidefiniteness. □

The estimator targets the curvature averaged over the chosen calibration scenes. Its unbiasedness with respect to random probes does not remove error from using a finite set of scenes to represent the inference distribution.

B Circuit Search Details

For a fixed task set, head retention ratio ρ, and interaction weight α, set $K = \lfloor (1-\rho)M \rfloor$ as in Eq. 2. Maintain the current removal vector δ, each task's predicted loss $q_t = \widetilde{q}_t(\delta)$, and the vector $b_t = \widetilde{H}_t \delta$. Initialize all three to zero. The task matrices and reference scales remain fixed throughout the search.

Adding a removal. For each retained head i, compute $m_{t,i} = b_{t,i} + \widetilde{H}_{t,ii}/2$ and score the candidate by

$$\max_{t \in \mathcal{T}} \frac{q_t + m_{t,i}}{\beta_t}. \tag{B.1}$$

Choose the lowest-scoring candidate, breaking ties by head index. Set $\delta_i = 1$ and update $q_t \leftarrow q_t + m_{t,i}$ and $b_t \leftarrow b_t + \widetilde{H}_{t,:,i}$ for every task. Repeat until $\sum_i \delta_i = K$, leaving $M - K = \lceil \rho M \rceil$ retained heads. In particular, minimizing the maximum marginal cost alone would not implement the joint objective, because the current task losses q_t need not be equal.

Exchanging two heads. After exactly K removals, we refine the circuit by exchanging a removed head with a retained head. Each swap preserves the circuit size. Let i be removed and j retained. Restoring i and removing j gives $\delta' = \delta - e_i + e_j$. Expanding the quadratic yields

$$s_{t,ij} = \widetilde{q}_t(\delta') - \widetilde{q}_t(\delta) = b_{t,j} - b_{t,i} + \frac{\widetilde{H}_{t,ii} + \widetilde{H}_{t,jj}}{2} - \widetilde{H}_{t,ij}. \tag{B.2}$$

Score each pair by $\max_{t \in \mathcal{T}} (q_t + s_{t,ij})/\beta_t$ and choose the smallest score, breaking ties by the ordered pair of head indices. Accept the swap only if it strictly lowers the current joint objective. Then update $\delta \leftarrow \delta - e_i + e_j$, $q_t \leftarrow q_t + s_{t,ij}$, and $b_t \leftarrow b_t - \widetilde{H}_{t,:,i} + \widetilde{H}_{t,:,j}$. Stop when no improving pair remains. Strict decrease prevents revisiting a circuit, so the procedure terminates on the finite feasible set. The result is optimal with respect to a single swap, but need not be globally optimal.

Search cost and constrained variants. Given the task matrices, greedy selection costs $O(|\mathcal{T}|MK)$ operations. A full scan of swap candidates costs $O(|\mathcal{T}|K(M-K))$, with an additional $O(|\mathcal{T}|M)$ update per accepted swap. The number of accepted swaps depends on the instance. Storing all task matrices costs $O(|\mathcal{T}|M^2)$. These costs exclude calibration and the forward evaluations used to select α.

For prescribed per-layer circuit sizes $H_\ell - K_\ell$, where K_ℓ is the number of excluded heads in layer ℓ, restrict greedy candidates to layers that have not reached their exclusion count, with $\sum_\ell K_\ell = K$. During refinement, allow swaps only within a layer. This preserves every layer's budget while still accounting for interactions across layers through the task matrices.

C Shrinkage Hessian map visualization

The shrinkage maps reveal a strongly structured interaction pattern rather than uniform coupling across the 768 gates, as shown in Figure 5. For the full map, the largest and densest responses occur in the global–global quadrant, with a concentrated band of interactions among the later layer/head groups; frame–frame and frame–global blocks are comparatively weaker and more diffuse. This

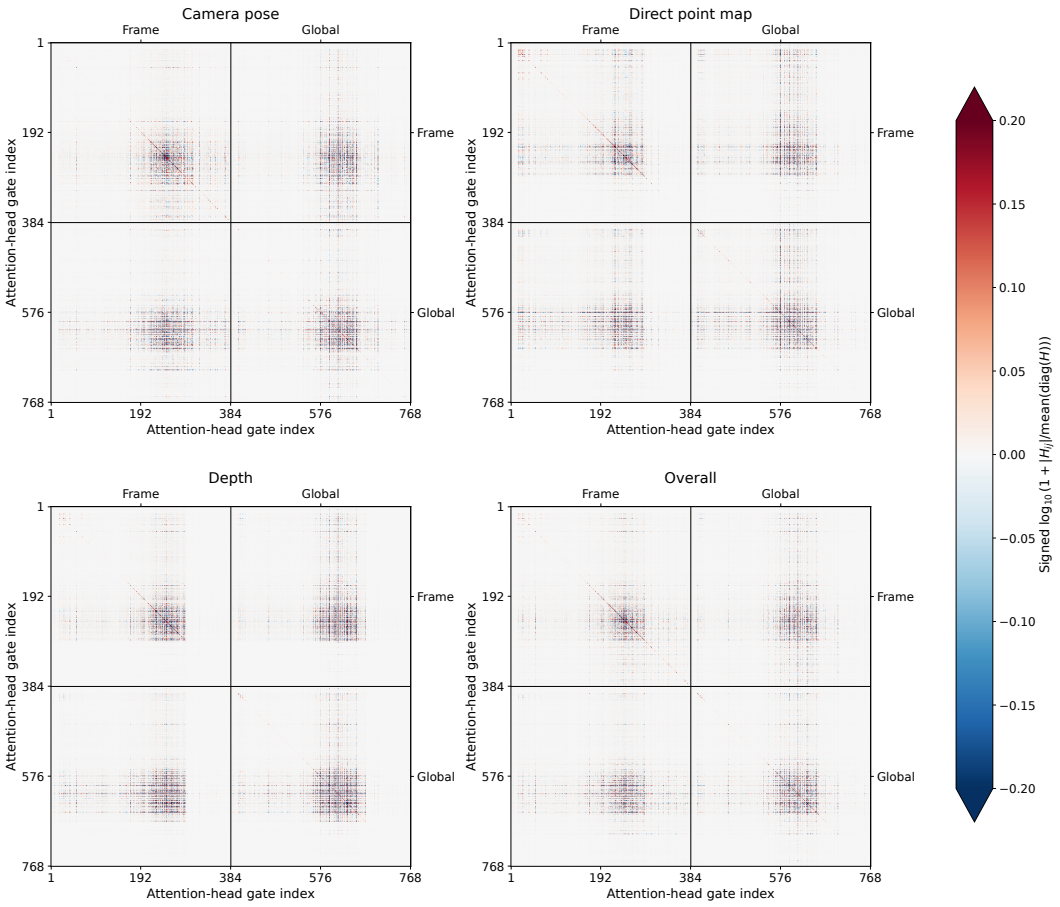


Figure 5: Shrinkage Hessian map for VGGT at $\alpha = 0.5$.

shared late-global structure is consistent with a common set of global attention heads contributing to multi-view geometry, while the task-specific differences explain why a single task-conditioned mask need not be optimal for every residual. The red/blue entries represent signed pairwise curvature after shrinkage, so they indicate interaction direction and strength rather than head importance by themselves. Because all panels use per-task mean-diagonal normalization, the maps support structural comparisons within and across tasks but do not establish that one task has larger absolute curvature or that the off-diagonal terms alone cause the downstream pruning gains.

D Calibration and Extraction Settings

Calibration and validation inputs. The calibration set comprises `chess/seq-03`, `chess/seq-05`, `fire/seq-03`, `heads/seq-01`, and `office/seq-02` from 7-Scenes (Shotton et al., 2013). We take the first six frames listed in each sequence's `kf40` manifest, giving 30 RGB images organized into five six-view inputs ($N = 5$ in Eq. 4). The validation set contains six views each from `fire/seq-04` and `pumpkin/seq-07`, totaling 12 additional images. Calibration and validation use distinct sequences, although both include the `fire` scene. Images are resized bicubically to width 224, with the aspect-ratio-derived height rounded to a multiple of 14, and RGB values are scaled to $[0, 1]$.

Frozen teacher and head gates. Supervision consists of detached predictions from the frozen dense VGGT-1.2B model on the same images; no ground-truth labels are used. All pretrained weights remain frozen, and derivatives are taken only with respect to 768 scalar gates covering complete heads in both frame and global attention. The implementation uses retention gates $g_i = 1 - \delta_i$, initialized to one, which corresponds to the dense model at $\delta = \mathbf{0}$ in the main text. Model execution and gate gradients use FP32, while curvature accumulation uses FP64.

Shrinkage and circuit allocation. We use $\alpha = 0.5$ in Eq. 11, giving equal weight to the diagonal approximation and the full estimated matrix. Circuit selection uses the global retained-head budget in Eq. 2, with no minimum retained-head count imposed on individual attention modules.

E Block-Sparse Attention Analysis

Using the same experimental configuration as the main evaluation, Figure 6 relates the heads selected for removal by **GeoCircuit** to the attention errors introduced by block sparsity. Removed heads concentrate in the later global-attention layers, where their sensitivity percentiles are generally low. Many of these same heads exhibit larger squared attention residuals than heads in earlier layers under both 25% and 50% block pruning, with residuals generally increasing at the higher pruning level. This overlap highlights a distinction between task importance and robustness to sparsification: a head with low estimated task sensitivity can still incur a substantial perturbation when attention blocks are removed.

This pattern suggests a mechanism for the benefit of combining **GeoCircuit** with SparseVGGT. By accounting for the estimated task-fidelity cost of joint head removal, **GeoCircuit** can discard low-sensitivity heads while limiting the estimated loss of task-relevant computation. When these heads also carry large sparsification errors, removing them can limit the propagation of such perturbations to subsequent layers and the final prediction. This interpretation is consistent with the end-to-end results in Figure 2: at 50% block pruning, reducing overall head retention from 100% to 90% improves Co3Dv2 AUC@30 by 2.56 percentage points. Thus, task-aware circuit extraction can improve the accuracy of a block-sparse model in addition to reducing its active head count.

F Applying GeoCircuit to π^3

As shown in Figure 7, our method better preserves π^3's pose estimation and reconstruction quality under aggressive sparsification. On Co3Dv2, it achieves higher AUC@30 and AUC@15 than HeSS at every evaluated nonzero sparsity level. At 50% requested retention, our method improves AUC@30, AUC@15, and AUC@5 by 4.91, 6.44, and 9.60 percentage points, respectively. Its advantage also extends to DTU reconstruction at this retention level: accuracy and completeness errors

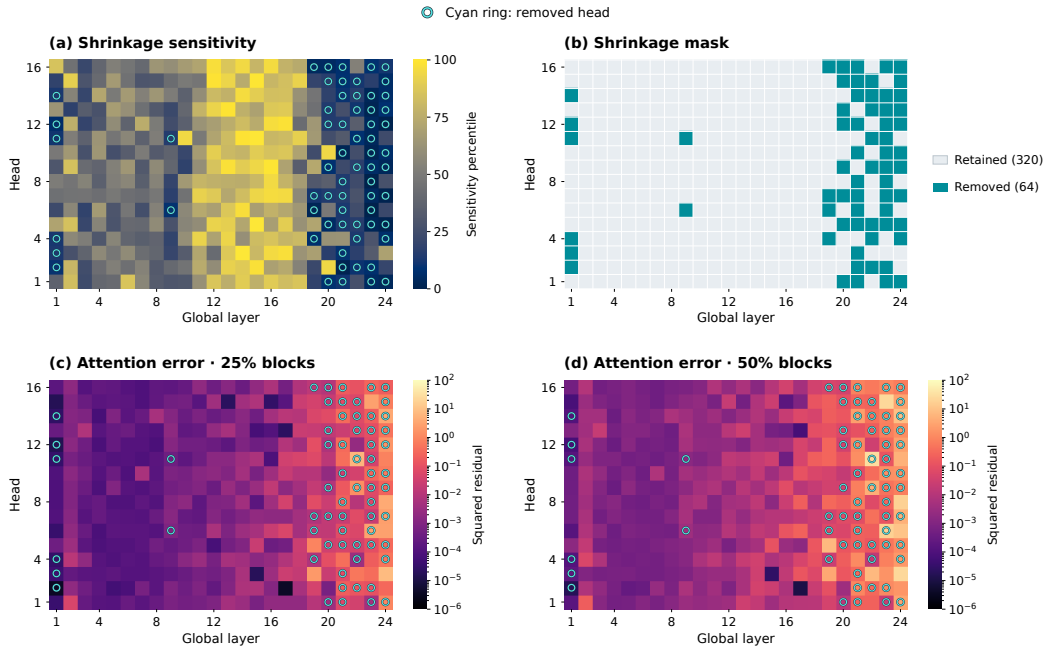


Figure 6: Alignment between head selection and block-sparse attention errors. Each cell corresponds to one of 16 heads in one of 24 global-attention layers. (a) Head sensitivity under the shrinkage estimator ($\alpha = 0.5$), expressed as percentiles; lower values indicate lower estimated sensitivity. (b) The corresponding selection mask, retaining 320 and removing 64 of the 384 global-attention heads shown. (c,d) Squared attention residuals under 25% and 50% block pruning, respectively, using a shared logarithmic color scale. Cyan rings in (a,c,d) mark the heads removed by **GeoCircuit**.

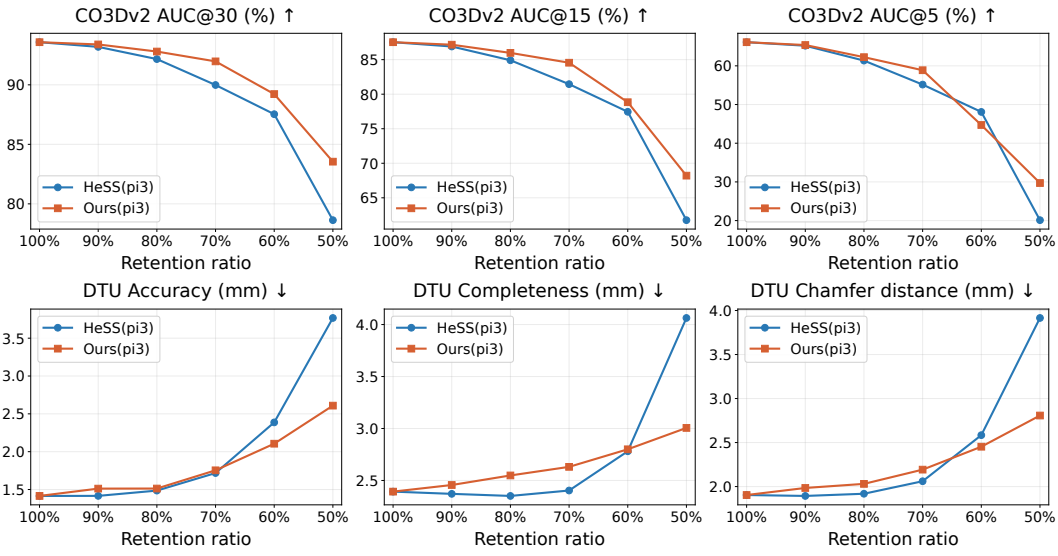


Figure 7: Comparison of **GeoCircuit**, HeSS on π^3 across retention ratios on Co3Dv2 and DTU. Higher AUC and lower Acc, Comp, and CD indicate better performance.

decrease by 30.8% and 26.0%, respectively, while Chamfer distance falls from 3.92 to 2.81 mm–a 28.3% reduction. These improvements across all six metrics at 50% retention support our method's effectiveness in preserving π^3's geometric predictions at high sparsity.

Table 2: Sufficiency and necessity of task-conditioned circuits at 50% circuit size. Co3Dv2 evaluates the pose-conditioned circuit. DTU evaluates the point-map-conditioned circuit using direct-point reconstruction.

	Co3Dv2	DTU	
Intervention	AUC@30 (%) ↑	Acc (mm) ↓	CD (mm) ↓
Dense model	94.22	1.393	1.919
Keep task circuit	87.15	2.622	2.679
Keep matched random	75.42	7.613	6.922
Remove task circuit	0.76	13.094	18.918
Remove matched random	1.16	11.742	14.421

G SUFFICIENCY AND NECESSITY

We evaluate each task-conditioned circuit on its corresponding task: the pose-conditioned circuit on Co3Dv2 camera pose estimation and the point-map-conditioned circuit on DTU direct-point reconstruction. For each task, we assess sufficiency by retaining its circuit $\mathcal{C}$ and necessity by disabling the same circuit and retaining its exact complement. Table 2 reports the 50% circuit size, where both interventions leave 384 active heads. Keeping the selected circuit achieves 87.15 AUC@30 on Co3Dv2 and 2.622 mm Acc on DTU, compared with 75.42 and 7.613 mm for random circuits matched in head count within every attention module. Removing the selected circuit raises DTU Acc to 13.094 mm, versus 11.742 mm for matched random removal, with the same ordering for CD. Moreover, pose AUC@30 falls to 0.76 under selected-circuit removal and 1.16 under random removal. These results support the value of task-aware head selection.

H LIMITATIONS

The fidelity predictor is a local quadratic approximation around the dense network (Appendix A.1). Shrinkage reduces interaction-estimation noise but does not account for higher-order effects of joint head removal. Our experiments establish effectiveness over retention ratios of 50%-100%; extending to smaller circuits may benefit from updating the predictor during extraction. Moreover, the selection budget constrains head count, while execution cost also depends on attention type, input size, and hardware. Our A100 experiments demonstrate practical acceleration, but the current objective does not directly optimize latency. Incorporating measured execution costs into circuit selection could further improve efficiency for specific deployment settings.

I VISUALIZATION RESULT

Figure 8 compares direct point-map reconstructions on four DTU scenes. Original VGGT is unpruned (100% retention). Our point-map-conditioned circuit retains 50% of complete attention heads (384 of 768), while HeSS retains 50% of eligible patch-to-patch attention blocks in frame and global attention, with special-token connections kept dense. These retention ratios use different sparsity units and do not imply matched FLOPs.

At these settings, **GeoCircuit** preserves the dense model's main object structure more closely in these examples. The difference is particularly visible in the building facade and Buddha statue, where HeSS exhibits pronounced geometric distortion.



Figure 8: Qualitative DTU direct point-map comparison. From left to right: original VGGT (100% retention), HeSS (50% eligible patch attention-block retention), and **GeoCircuit** (50% head retention).