TURN THE BOTTLENECK KNOB: TOKEN-VALVE PATH EXPLANATIONS FOR VISION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Turn the Bottleneck Knob (TBK) is a post-hoc explanation method for frozen Vision Transformers. It combines shared token gates on selected pre-attention branches, finite-step target cross-entropy optimization, and integration of absolute gate sensitivities along a linear opening path. The resulting scores measure unsigned sensitivity per unit gate coordinate; they do not identify minimal or necessary tokens. We give a shared-parameter derivative and a loss-change upper bound, distinguish these properties from causal faithfulness, and clarify why a finite-noise information-bottleneck motivation does not establish deterministic information compression. Existing ImageNet-1K results on ViT-B/16 and DeiT-B/16 show a competitive but mixed proxy-metric profile. A separate ViT-Base follow-up supports an Insertion gain over CoIBA with a paired confidence interval excluding zero; its ROAD, Deletion, and Infidelity intervals include zero. Smoothing materially affects the Insertion comparison. The evidence supports this particular attribution pipeline in the evaluated setting, with unresolved implementation and protocol details stated explicitly.

1 INTRODUCTION

Vision Transformers (ViTs) (Dosovitskiy et al., 2021) represent images through interacting tokens. In post-deployment analysis, the model may be available as frozen weights while training data and retraining access are unavailable. This motivates an explanation procedure that operates on a single input with white-box forward/backward access. The practical question is which internal interventions change a specified target score, without updating model weights.

Gradient and path methods (Simonyan et al., 2014; Sundararajan et al., 2017) measure sensitivity in chosen coordinates; perturbation methods (Petsiuk et al., 2018) evaluate changes under input interventions. Information-bottleneck methods instead introduce restrictions on intermediate representations (Schulz et al., 2020; Hong et al., 2025). These approaches motivate internal control variables, but their different interventions and objectives do not provide interchangeable notions of importance.

We propose TBK a particular combination of a deterministic shared token gate, per-input endpoint optimization, and unsigned path integration. A gate scales the output of the pre-attention LayerNorm at selected blocks. Stage A performs a finite number of cross-entropy updates to obtain a target-conditioned gate configuration. Stage B integrates the magnitude of the total derivative with respect to each shared gate along a linear path to that configuration. The scores summarize sensitivity along this intervention path, while the residual stream continues to carry information even when these gates are closed.

The contribution is this algorithmic combination and its reported empirical behavior, rather than a new information-theoretic equivalence or a guarantee of necessary evidence. We make three aspects explicit: the branch-level intervention and its shared derivative; the relation between unsigned sensitivity, endpoint-weighted sensitivity, and signed Integrated Gradients (IG); and the scope of the existing metric evidence. The associated mathematical identities use standard chain-rule and triangle-inequality arguments. They do not establish ranking quality, and the empirical comparisons remain essential.

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.

2 Related Work

Gradient, attention, and perturbation explanations. IG (Sundararajan et al., 2017) attributes a signed output difference along a path. Guided IG (Kapishnikov et al., 2021), Expected Gradients (Erion et al., 2021), scale-space attribution (Xu et al., 2020), FastIG (Hesse et al., 2021), and iterated integrated attribution (Barkan et al., 2023) vary paths, references, or integration. Transformer-specific approaches aggregate attention and relevance (Abnar & Zuidema, 2020; Chefer et al., 2021b;a). Boundary methods include AGI (Pan et al., 2021), MFABA (Zhu et al., 2024b), and local attribution (Zhu et al., 2024c).

Recent ViT methods provide important alternatives. LeGrad (Bousselham et al., 2025) uses attention-map gradients aggregated across layers. Transformer Input Sampling (TIS) (Englebert et al., 2023) samples input-token subsets. MDA (Walker et al., 2025) optimizes patch ordering and magnitudes using attribution metrics. TokenTM (Wu et al., 2024) measures token transformations and aggregates their effects with attention. LibraGrad (Mehri et al., 2025) modifies backward gradient flow. TBK instead optimizes forward branch gates and differentiates the resulting gated network along a gate path. These distinctions identify the intervention and score, not superiority: the five methods are not included in the existing numerical comparisons.

Bottleneck attribution. IBA (Schulz et al., 2020) and M2IB (Wang et al., 2023) motivate explanations using representation bottlenecks; CoIBA (Hong et al., 2025) extends this line for ViTs and already shares a damping ratio across layers. NIB (Zhu et al., 2025) also addresses limitations including intrinsic randomness, so neither cross-layer sharing nor determinism alone is a novelty claim. TBK uses neither sampled bottleneck noise nor an explicit information penalty in its operational objective. Its distinctive choice is to combine shared branch gates with CE endpoint optimization and unsigned sensitivity integration. The classical IB perspective (Tishby et al., 2000; Alemi et al., 2017) supplies background, but TBK is not an optimizer of a nontrivial compression–prediction trade-off.

3 Methodology

3.1 The gated network

Let f_θ be a frozen classifier, x an image with N patch tokens, and y a specified target class. Token 0 denotes CLS. We use a shared vector $\alpha \in [0,1]^{N+1}$ over selected blocks $\mathcal{S}$. Writing $H^{(\ell)}$ for a block input, the pre-normalized intervention is

$$\begin{aligned} h^{(\ell)} &= \mathrm{LN}_1^{(\ell)}(H^{(\ell)}), & z^{(\ell)} &= \mathrm{diag}(\alpha) h^{(\ell)}, \\ U^{(\ell)} &= H^{(\ell)} + \mathrm{Attn}^{(\ell)}(z^{(\ell)}), & H^{(\ell+1)} &= U^{(\ell)} + \mathrm{MLP}^{(\ell)}(\mathrm{LN}_2^{(\ell)}(U^{(\ell)})). \end{aligned} \tag{1}$$

Unselected blocks use their ordinary forward computation. The source manuscript identifies selected `norm1` modules; Equation 1 makes explicit the output-scaling interpretation of that description. The original archive contains no main-result implementation. A subsequently recovered server source uses a forward hook that scales `norm1` outputs exactly this way, supporting the graph specification by static inspection. This does not establish that its current bytes generated every historical result (Section 5.2); no experiment was rerun.

The same α_i scales token i at every selected occurrence, including CLS; the displayed patch map drops CLS after scoring. Thus sharing uses $N+1$ gate variables rather than $|\mathcal{S}|(N+1)$ independent variables. It reduces freedom but constrains layer-specific interventions, whose relative value has not been isolated experimentally. For the evaluated 12-block backbones the reported selection is zero-indexed blocks 4–11.

The reference $\alpha = 0$ closes only the selected branch inputs. Residual routes, earlier blocks, and possible attention/projection biases remain; it does not erase tokens or block all information. At $\alpha = 1$, the selected scalings are identities. We denote its logits by $r_{\theta,\alpha}(x)$ and probabilities by $p_{\theta,\alpha}(x) = \mathrm{softmax}(r_{\theta,\alpha}(x))$. The intended target loss is $\mathcal{L}_{\mathrm{logit}}(\alpha) = -\log p_{\theta,\alpha}(x)_y$. However, the recovered follow-up wrapper returns probabilities to `CrossEntropyLoss`, yielding $\mathcal{L}_{\mathrm{prob}}(\alpha) = -\log[\mathrm{softmax}(p_{\theta,\alpha}(x))]_y$. These are different objectives. We use $\mathcal{L}$ for the scalar loss actually chosen; the derivative and path results apply to either. The follow-up evidence must be interpreted with

the probability-input surrogate, not relabeled as a logit-CE experiment. Its targets are loaded labels from a saved tensor, not recomputed original predictions; their construction remains undocumented.

3.2 Stage A: finite-step endpoint optimization

TBK parameterizes $\alpha = \sigma(a)$ and applies T optimizer updates to $\mathcal{L}(\sigma(a))$, with θ fixed. The endpoint is

$$\widehat{\alpha} = \sigma(a^{(T)}), \qquad a^{(k+1)} = \mathcal{U}_k\big(a^{(k)}, \nabla_a \mathcal{L}(\sigma(a^{(k)}))\big), \tag{2}$$

where $\mathcal{U}_k$ denotes the chosen numerical optimizer (including any optimizer state). This notation defines a finite output, not an attained $\arg\min$. Static inspection of the recovered source identifies Adam with learning rate 0.1, raw gates initialized to zero (opening 0.5), and fixed step budgets. Its linkage to the original main-table execution remains unverified. The source keeps model weights outside the optimizer; gradients are computed but only gate parameters are updated.

No compression or sparsity penalty is present in the default objective. For example, if a binary target logit is $\alpha_1 + \alpha_2$, CE decreases as both gates open even when either could support the decision. Likewise, $\log(1 + e^{-\alpha})$ has no minimizer on the open sigmoid interval $(0, 1)$. Hence a finite partially open endpoint does not establish necessity, minimum sufficiency, or optimizer convergence. Target CE encourages the specified class score; it does not constrain agreement with the original output distribution or even guarantee the final predicted label.

What the information-bottleneck motivation establishes. For Gaussian X with variance $v > 0$ and independent $\varepsilon_\sigma \sim \mathcal{N}(0, \sigma^2)$, the finite-noise channel $Z_\sigma = \alpha X + (1 - \alpha)\varepsilon_\sigma$ has

$$I(X; Z_\sigma) = \tfrac{1}{2} \log\left(1 + \frac{\alpha^2 v}{(1-\alpha)^2 \sigma^2}\right). \tag{3}$$

This is increasing in $\alpha \in (0, 1)$ for fixed $\sigma > 0$, but diverges as $\sigma \to 0$. Exactly deterministic nonzero scaling preserves the information in X: its MI is infinite for this continuous example, and equals $H(X)$ for finite-entropy discrete X. Thus finite-noise monotonicity does not justify a deterministic compression ordering. TBK's deterministic scaling is an intervention design choice. The population CE variational bound also does not imply single-example MI preservation; details are in Appendix A.

3.3 Stage B: unsigned gate sensitivity

For the linear path $\alpha(t) = t\widehat{\alpha}$, define

$$g_i(t) = \partial_i \mathcal{L}(t\widehat{\alpha}), \qquad s_i = \int_0^1 |g_i(t)|\, dt, \qquad i = 0, \ldots, N. \tag{4}$$

The score measures sensitivity per unit gate coordinate, integrated over the chosen path parameter. Absolute value removes cancellation across path locations of the *total* derivative, discards its direction, and can retain oscillatory magnitude. It does not generally remove saturation or numerical noise.

Lemma 1 (Shared-gate derivative)**.** *For the graph in Equation 1, the total shared-parameter derivative is*

$$\partial_i \mathcal{L}(\alpha) = \sum_{\ell \in \mathcal{S}} \left\langle \overline{z}_i^{(\ell)}(\alpha), h_i^{(\ell)}(\alpha) \right\rangle, \qquad \overline{z}_i^{(\ell)} = \frac{\partial \mathcal{L}}{\partial z_i^{(\ell)}}. \tag{5}$$

Each adjoint and feature is evaluated in the same forward/reverse graph. Equation 4 takes absolute value after *summing these occurrences.*

This follows by summing the local multiplication-node derivatives with respect to the shared parameter; the reverse adjoints account for downstream effects. Features vary with earlier gates. Layer contributions may cancel before taking absolute value. A single last-layer inner product would omit other occurrences.

Theorem 1 (Continuous path bounds)**.** *Suppose $\mathcal{L}(t\widehat{\alpha})$ is absolutely continuous, satisfies the chain rule almost everywhere, and its partial derivatives along the path are integrable. Let $\Delta\mathcal{L} = \mathcal{L}(\widehat{\alpha}) - \mathcal{L}(0)$ and $\mathrm{IG}_i = \widehat{\alpha}_i \int_0^1 g_i(t)\, dt$. Then*

$$\sum_i \mathrm{IG}_i = \Delta\mathcal{L}, \qquad |\Delta\mathcal{L}| \le \sum_i |\mathrm{IG}_i| \le \sum_i \widehat{\alpha}_i s_i \le \sum_i s_i. \tag{6}$$

Algorithm 1 TBK for one input: recovered numerical ordering

Require: Frozen classifier, input x, target y, selected sites $\mathcal{S}$, budgets $T, m \geq 2$
1: Initialize raw gate $a = 0$; attach the same $\sigma(a)$ at every selected site.
2: **for** $k = 1, \ldots, T$ **do**
3: Recompute the gated forward pass; update a with Adam on the declared scalar $\mathcal{L}(\sigma(a))$.
4: **end for**
5: Detach endpoint $\widehat{\alpha} = \sigma(a)$ from the optimization graph.
6: **for** $j = 1, \ldots, m$ **do**
7: Create leaf $\beta_j = (j/m)\widehat{\alpha}$; recompute the forward pass with this shared gate.
8: Store $v_{j,i} = |\partial\mathcal{L}(\beta_j)/\partial\beta_{j,i}|$ for all tokens including CLS.
9: **end for**
10: Set $Q_i = m^{-1}(v_{1,i}/2 + \sum_{j=2}^{m-1} v_{j,i} + v_{m,i}/2)$.
11: Drop CLS; reshape patches; standardize, clamp, add the declared smoothing map, resize and normalize.
12: **return** Postprocessed patch map; Q remains a separate numerical readout.

The fundamental theorem of calculus and triangle inequality prove the result (Section 4). This is an upper bound, with equality possible, and can be arbitrarily loose. It holds for any qualifying endpoint, including random ones; it does not certify causal faithfulness or image-space Insertion performance.

Signed IG has completeness; neither s_i nor $\widehat{\alpha}_i s_i$ has signed completeness. Omitting the endpoint factor intentionally emphasizes sensitivity rather than displacement-weighted magnitude and can reverse rankings. For $\mathcal{L} = C + 10\alpha_1 + \alpha_2$ and $\widehat{\alpha} = (0.001, 0.9)$, $s_1 = 10 > s_2 = 1$ while the weighted scores are $0.01 < 0.9$. A controlled comparison of these alternatives remains necessary to justify this ranking choice empirically. The bound includes CLS; summing only displayed patch scores need not bound the total loss change.

Numerical integration and maps. The existing experiments report $m = 20$ integration steps. For a full-interval right-endpoint approximation $\widetilde{s}_i = m^{-1} \sum_{j=1}^{m} |g_i(j/m)|$, a K_i-Lipschitz integrand gives $|\widetilde{s}_i - s_i| \leq K_i/(2m)$ (Section 4). The recovered code instead applies a trapezoidal rule to nodes $1/m, \ldots, 1$, omitting the interval $[0, 1/m]$. This is not the full-interval right-endpoint average; Section 4 gives its separate conditional error bound. The code then standardizes the patch map, clamps negative entries, adds a 3×3 mean-pooled map, upsamples, and min–max normalizes. These operations can change rankings and scales. The continuous theorem is about raw s_i, not automatically the implemented quadrature or postprocessed maps.

3.4 Executable ordering and computational cost

Algorithm 1 separates the optimized endpoint from the score readout. The path stage treats its endpoint as fixed: it differentiates the fresh gate at each path point, not through the optimizer trajectory. This distinction matters because differentiating through Adam would define a different attribution object. Likewise, summing absolute values of individual layer contributions before integration would define a different score from the shared derivative used here. The recovered follow-up uses $\mathcal{L}_{\text{prob}}$, while the intended logit-loss variant uses $\mathcal{L}_{\text{logit}}$. Algorithm 1 states their common ordering without assigning the same measured performance to both objectives. A batch introduces the mean-loss scaling described below. The m stored derivative vectors require $O(m(N+1))$ storage in a direct implementation; an online quadrature accumulator can reduce readout storage to $O(N+1)$ without altering the stated arithmetic weights. Each stage still needs the network activations required for reverse differentiation.

Ignoring fixed preprocessing and evaluation overhead, the method uses $T+m$ gated forward/backward evaluations per input. Gate sharing reduces parameter count, not network passes; throughput remains hardware- and batching-dependent. The reported timing and its missing details are retained in Appendix E.

4 INTERPRETING AND APPROXIMATING THE PATH SCORE

Shared derivative. Temporarily give each multiplication site its own gate $\alpha_i^{(\ell)}$. Its local derivative is $\langle \overline{z}_i^{(\ell)}, h_i^{(\ell)} \rangle$. Tying all these gates to α_i sums the derivatives. Reverse adjoints already propagate downstream dependence, so differentiating features again in each local term would double count paths. For $z_1 = \alpha x$, $z_2 = \alpha z_1$, and $\mathcal{L} = z_2$, the two local contributions are both αx, giving $2\alpha x$ rather than the last-site-only value αx.

Completeness and upper bounds. Under Theorem 1's assumptions,

$$\Delta\mathcal{L} = \int_0^1 \frac{d}{dt}\mathcal{L}(t\widehat{\alpha})dt = \sum_i \widehat{\alpha}_i \int_0^1 g_i(t)dt = \sum_i \mathrm{IG}_i.$$

Apply the triangle inequality first across coordinates and then within each integral. Since $0 \le \widehat{\alpha}_i \le 1$, this gives Equation 6. The sharper weighted quantity is $\sum_i \widehat{\alpha}_i s_i$. Equality in the first inequality requires all nonzero IG_i to have the same sign; equality in the second requires $g_i(t)$ to have a constant weak sign almost everywhere for each coordinate with positive endpoint. Equality between weighted and unweighted sums requires $(1 - \widehat{\alpha}_i)s_i = 0$ for every i. For strictly interior sigmoid endpoints, the last equality therefore requires all $s_i = 0$. A linear loss with same-sign slopes attains the weighted bound, establishing its tightness over this class.

Cancellation and lack of faithfulness guarantees. For $\mathcal{L} = C + \alpha_1 - \alpha_2$ and endpoint (a, a), $0 < a < 1$, the endpoint change is zero, but $s_1 + s_2 = 2$ and the weighted sum is $2a$. Perturbing the second slope to $-1 + \delta$ gives endpoint change $a\delta$ and an arbitrarily large bound-to-change ratio as $\delta \downarrow 0$. Even a single coordinate can accumulate large unsigned magnitude: $\mathcal{L}(\alpha) = C + \varepsilon \sin(k\alpha)$ with $C > \varepsilon > 0$ has endpoint change at most 2ε while its absolute derivative integral grows with k. Taking $p_y = e^{-\mathcal{L}}$ realizes a valid binary-class CE example. These bounds cannot be inverted into necessity, confidence, or ranking-reliability claims.

Conditional quadrature error. If $f_i(t) = |g_i(t)|$ is K_i-Lipschitz and the right-endpoint rule uses $t_j = j/m$, then

$$\left| \frac{1}{m}\sum_{j=1}^{m} f_i(t_j) - \int_0^1 f_i(t)dt \right| \le \sum_{j=1}^{m} \int_{(j-1)/m}^{j/m} K_i(t_j - t)dt = \frac{K_i}{2m}.$$

Consequently $|\Delta\mathcal{L}| \le \sum_i \widehat{\alpha}_i(\widetilde{s}_i + K_i/(2m))$ under these conditions. This controls only integration error for fixed endpoints; it does not bound optimizer error or effects of smoothing. For the rule actually found in the recovered source, with $m \ge 2$,

$$Q_i = \frac{1}{m}\left[\tfrac{1}{2} f_i(1/m) + \sum_{j=2}^{m-1} f_i(j/m) + \tfrac{1}{2} f_i(1) \right].$$

It approximates $\int_{1/m}^1 f_i(t)dt$. If $f_i \le M_i$ on $[0, 1/m]$ and is K_i-Lipschitz on the remaining interval, the missing interval contributes at most M_i/m, while summing the trapezoidal interval errors gives at most $K_i(m-1)/(2m^2)$. Therefore

$$|Q_i - s_i| \le \frac{M_i}{m} + \frac{K_i(m-1)}{2m^2}.$$

For a constant integrand c, this rule returns $(1 - 1/m)c$ instead of c, demonstrating the truncation directly. No bounds on M_i, K_i are supplied, so $m = 20$ remains uncertified. We document the existing computation rather than silently replacing it and retaining its numbers.

4.1 LOSS CHOICE AND THE UNITS OF GATE SENSITIVITY

The distinction between the two CE objectives also changes the derivative being integrated. Let r be the logits, $p = \mathrm{softmax}(r)$ and $q = \mathrm{softmax}(p)$. Their logit gradients are

$$\nabla_r \mathcal{L}_{\mathrm{logit}} = p - e_y, \qquad \nabla_r \mathcal{L}_{\mathrm{prob}} = (\mathrm{diag}(p) - pp^\top)(q - e_y). \tag{7}$$

The first equality is the usual CE derivative; the second applies the softmax Jacobian to the outer CE derivative. Even with the same gate endpoint and network, these gradients are generally different vectors. For a binary target with probability $p_y = p$ and logit difference $d = r_y - r_{\neg y}$, the outer target probability is $q_y = \sigma(2p - 1)$, and

$$\frac{d\mathcal{L}_{\text{prob}}}{dd} = -2p(1-p)(1-q_y), \qquad \frac{d\mathcal{L}_{\text{logit}}}{dd} = -(1-p).$$

Thus the former multiplies the latter by $2p(1 - q_y)$. This factor varies along a gate path. As $p \to 0$, probability-input CE has a vanishing derivative with respect to d although logit CE's derivative approaches -1. This is an analytical saturation example, not a measured failure rate. The binary losses favor the same target direction, but neither their optimization trajectories nor their integrated rankings are thereby identical. Multiclass softmax coupling further rules out treating the two gradients as a universal scalar rescaling. Recovering the wrapper is therefore material to the explanation definition.

Unsigned coordinate sensitivity also has units. Consider a positive diagonal change of variables $\alpha_i = c_i\beta_i$, $c_i > 0$, with endpoint $\widehat{\beta}_i = \widehat{\alpha}_i/c_i$ and the same physical intervention path. For $\widetilde{\mathcal{L}}(\beta) = \mathcal{L}(c \odot \beta)$, the chain rule gives

$$s_i^{(\beta)} = c_i s_i^{(\alpha)}, \qquad \widehat{\beta}_i s_i^{(\beta)} = \widehat{\alpha}_i s_i^{(\alpha)}. \tag{8}$$

Signed path contributions have the same invariance as the endpoint-weighted quantity. Raw unsigned rankings can change under coordinate-specific scaling. TBK fixes the opening fraction in $[0, 1]$ as its coordinate convention; it does not claim invariance to arbitrary reparameterization. This comparison holds the endpoint and physical path fixed and transforms the admissible domain accordingly. Rerunning finite Adam in new coordinates would additionally change the endpoint, so the identity is not an optimizer-invariance result. Together with the cancellation example, it separates three questions: which intervention is applied, which scalar is differentiated, and which units define the score. Those choices must be held fixed before a ranking comparison can isolate an attribution mechanism.

The omitted CLS coordinate. The path theorem includes CLS, whereas the displayed map omits it. Its patch-only counterpart is

$$|\Delta\mathcal{L} - \text{IG}_0| \le \sum_{i=1}^{N} \widehat{\alpha}_i s_i.$$

This follows by removing the CLS term from the signed completeness identity. A patch map alone therefore need not bound the full loss change: the missing CLS contribution may be nonzero even when all patch scores are small. Subsequent map normalization does not recover that omitted contribution.

5 Experiments

5.1 Reported setup and evidence provenance

We retain the existing ImageNet-1K (Deng et al., 2009) results on ViT-B/16 and DeiT-B/16 (Dosovitskiy et al., 2021; Touvron et al., 2021). No new experiment is introduced here. Main-table step budgets are $T = 50$ for ViT and $T = 30$ for DeiT, both with $m = 20$. The separate ViT follow-up uses $T = 20, m = 20$, 1000 images, and a 200-image Infidelity subset. Its paired intervals therefore do not quantify uncertainty for the exact main-table configuration.

We report ROADCombined (Rong et al., 2022) (larger MoRF/LeRF separation is better), Insertion AUC (higher), Deletion AUC (lower) (Petsiuk et al., 2018), and Infidelity (lower) (Yeh et al., 2019). Infidelity depends on attribution magnitudes, normalization, and perturbations; it is not a measure of rank stability. The source does not specify all metric parameters, sample identities, filtering, and baseline configurations. Section 5.2 separates known settings from missing facts. Consequently, numerical comparisons are reported historical results, not an independently reproduced benchmark or fully verified parity claim. The original archive has no main-result implementation; subsequently recovered server sources clarify some details but do not fully establish historical result provenance.

The main quantitative baselines are exactly those in Table 2: Generic, IBA, IIA, ViT-CX (Xie et al., 2022), AGI, LA, BIG, M2IB, MFABA, NIB, SG (Smilkov et al., 2017), SM, and CoIBA. IG, Guided-IG, Fast-IG, EG, DeepLIFT (Shrikumar et al., 2017), and ISA (Zhu et al., 2024a) also appear in the archived qualitative grids; they are not additional main-table numerical comparisons.

5.2 Operational specification and protocol identity

The operational sequence is: choose x, y and selected sites; share sigmoid gates across those sites; update their raw parameters for T CE steps with frozen model weights; evaluate total gate derivatives along $t\widehat{\alpha}$; integrate their absolute values; remove CLS for the patch map; apply the declared postprocessing. All forward passes must recompute features at that path point. The recovered source creates a fresh leaf α_t at each path point and differentiates the shared variable, consistent with the total derivative in Lemma 1. It includes all 197 tokens during differentiation, drops CLS afterward, and uses a 14×14 patch grid. This is static source verification of these operations, not execution verification.

The inspected 12-block model hooks agree with Equation 1. Additional backbone names in the source are configuration candidates, not evidence of evaluation; the reported main comparisons remain limited to ViT-Base and DeiT-Base.

Table 1: Recoverable protocol settings. "Unspecified" records a missing fact, not a default value.

Item	Main comparison	ViT follow-up
Optimization steps	ViT 50; DeiT 30	20
Integration steps	20	20
Image count	Unspecified	1000; Infidelity 200
Layer set	Blocks 4–11	Blocks 4–11
Default smoothing	Reported mean 3×3	Add mean 3×3
Reported repeats	No main-table variance	Five 200-image runs

Recovered source-specific settings. The inspected follow-up source initializes a separate 197-dimensional raw gate for each image, uses Adam (0.1) and a batch-mean CE loss, and shares each image's gates across selected layers. Batch averaging rescales gate gradients; fixed optimizer epsilon means exact batch-size invariance should not be assumed. After the truncated trapezoidal accumulation, patch scores are standardized per image using their mean and standard deviation plus 10^{-8}, clamped at zero, and transformed as $S \leftarrow S + \mathrm{AvgPool}_{3,1,1}(S)$ (zero padding), rather than replaced by a mean-filtered map. The candidate main helper then bilinearly upsamples to 224×224 with `align_corners=False` and min–max normalizes with 10^{-8} stabilization. The follow-up imports a separate normalization utility, whose exact equivalence also needs verification. Its "raw" ablation skips additive smoothing but still includes standardization, clamping, and map normalization; it is not the raw theoretical s_i.

The follow-up loader takes a prefix of saved image and label tensors; the tensor's original sampling/filtering and label construction remain undocumented. The wrapper applies ImageNet normalization with means $(0.485, 0.456, 0.406)$ and standard deviations $(0.229, 0.224, 0.225)$, followed by a pretrained model and Softmax. CE on this wrapper therefore uses $\mathcal{L}_{\mathrm{prob}}$. Insertion/Deletion use zero substrate and an evaluation step equal to image width. ROAD uses a separate logits wrapper and software ROADCombined evaluator. The Infidelity helper uses 100 perturbation samples by default, a seeded NumPy generator, and a fitted scalar rescaling of each explanation's predicted changes before squared-error calculation. Thus near-equal Infidelity does not directly validate unnormalized gate sensitivities. These are code-level observations; full evaluator/version/config bindings are still needed.

The recovered default run configuration specifies batch size 8, seed 3407, 1000 images, 200 Infidelity images, and 100 perturbation samples. Its saved summary agrees with the default follow-up means at the reported precision, establishing an artifact-to-table correspondence; it does not supply an immutable source-commit binding. The follow-up defines "gate std." as each image's standard deviation across patch gates, then averages over images; it is not repeated-run uncertainty of a metric. Near-one means a patch gate is at least 0.95. Exact main-table sampling, target choice, baseline

Table 2: Main results on ImageNet-1K. ROAD/Insertion ↑, Deletion/Infidelity ↓; best per column in **bold**. FPS comparisons are in Appendix E.

	ViT-Base				DeiT-Base			
Method	ROAD	Ins.	Del.	Infid.	ROAD	Ins.	Del.	Infid.
Generic	0.1242	0.4217	0.2347	0.01130	-0.0712	0.3220	0.4443	0.01452
IBA	0.2743	0.5550	0.1427	**0.01092**	0.2680	0.5837	0.2077	0.01439
IIA	0.1477	0.4709	0.2230	0.01098	-0.0745	0.3343	0.4336	0.01440
ViT-CX	0.2439	0.5135	0.1630	0.01126	0.0542	0.4248	0.3626	0.01441
AGI	0.2575	0.5405	0.0428	0.01190	0.1739	0.5236	0.1022	0.01440
LA	0.2514	0.5614	0.0322	0.01104	0.1459	0.6132	0.0982	**0.01439**
BIG	0.1650	0.4767	0.0606	0.01202	0.1011	0.4621	0.1236	0.01499
M2IB	0.1650	0.4366	0.1952	0.01164	0.1821	0.5005	0.2496	0.01440
MFABA	0.1358	0.4661	0.0696	0.01134	-0.0275	0.3563	0.2171	0.01512
NIB	0.1494	0.4031	0.2024	0.01093	0.1498	0.4785	0.2653	0.01439
SG	0.1470	0.3330	**0.0134**	0.25022	0.1556	0.4860	**0.0701**	0.06343
SM	0.1454	0.2517	0.0400	0.47687	0.1491	0.3613	0.1043	0.18208
CoIBA	0.2829	0.5713	0.1164	0.01168	0.2797	0.5779	0.1807	0.01442
TBK (ours)	**0.2852**	**0.5954**	0.1188	**0.01092**	**0.2875**	**0.6136**	0.1748	**0.01439**

hyperparameters, checkpoint hashes, and all code-to-result bindings remain unresolved. No original experiment code was added to the immutable ZIP, and no measurement or implementation was changed in this revision.

Batch-loss scaling. The batched follow-up averages its loss over B images with separate gates. For independent per-image forwards, its gradient for image b is $B^{-1}\partial_i\mathcal{L}_b$, so its raw quadrature estimates $s_{b,i}/B$ with the truncation error described above. The single-image formulas use $\mathcal{L}_b$; applying the theorem directly to the batch loss instead concerns the mean loss change. Map normalization does not turn this scaled, truncated, postprocessed readout into a completeness certificate.

5.3 Main results and interpretation

TBK has the highest displayed ROAD and Insertion means on both backbones. Infidelity ties several methods, and Deletion is mixed: CoIBA has lower Deletion on ViT, while TBK has lower Deletion on DeiT; SG has the lowest Deletion in both columns. These means alone do not establish significant differences. No main-table standard deviations were supplied, and follow-up reruns cannot supply them retrospectively.

The paired follow-up comparison with CoIBA (Table 3) supports an Insertion difference of 0.02289, with 95% interval $[0.01672, 0.02899]$. ROAD, Deletion, and Infidelity intervals include zero. Smoothing contributes materially to this comparison: raw TBK Insertion is 0.57623, whereas mean-3×3 smoothing yields 0.59355; CoIBA's follow-up mean is 0.57066. The corresponding mean differences are 0.00557 and 0.02289, but no paired interval for the raw difference is supplied. Neither raw-map significance nor a uniformly superior four-metric profile follows.

The existing endpoint and layer ablations are informative but limited. Learned gates outperform the all-open and random-gate alternatives on several displayed means. This is evidence for the tested pipeline, not a proof of necessary token selection. Initialization matters: the low-initialization diagnostic changes ROAD from 0.28495 to 0.20865 and Insertion from 0.59355 to 0.51040. Layers 4–11 perform best on ROAD, Insertion, and Deletion among the tested layer sets, without establishing an optimal selection rule. No shared-versus-independent-gate comparison is available.

The step ablations show metric plateaus over parts of the reported ranges (Appendix C). They do not establish convergence of each gate vector or numerical certification of each integral. Five repeated 200-image runs have identical displayed Insertion and Deletion but small ROAD variation (Table 7); this concerns aggregate repeatability. Rank robustness to inputs, initialization, optimization, or discretization is untested. A CoIBA β change alters its objective, whereas a TBK step-budget change alters its numerical procedure; their qualitative comparison is not a matched robustness test.

Existing crate, scoreboard, king-crab, and bookshop examples are retained at larger scale in Appendix D. These selected illustrations show differences in map concentration and spatial coverage. Object localization is not ground-truth evidence of what a classifier uses: background context may contribute to its decision. The examples are neither random sampling nor newly evaluated failure cases.

5.4 Paired uncertainty and the effect of postprocessing

Paired comparison. Only the Insertion interval excludes zero in a direction favoring TBK. The interval is for the follow-up protocol, not the main ViT $T = 50$ setting. Bootstrap details beyond the supplied table are unavailable; the displayed interval is not a substitute for those records.

Table 3: Paired bootstrap comparison between TBK and CoIBA. ROAD/Insertion are higher-is-better, while Deletion/Infidelity are lower-is-better; therefore only the Insertion interval provides a clear advantage claim here.

Metric	n	TBK mean	CoIBA mean	Mean diff.	95% CI	TBK win rate
ROAD	1000	0.28495	0.28287	0.00208	$[-0.00174, 0.00608]$	0.527
Insertion	1000	0.59355	0.57066	**0.02289**	$[\mathbf{0.01672}, \mathbf{0.02899}]$	0.627
Deletion	1000	0.11934	0.11785	0.00149	$[-0.00176, 0.00474]$	0.427
Infidelity	200	0.00973	0.00974	-0.00001	$[-0.00006, 0.00004]$	0.575

Postprocessing. Additive smoothing changes evaluated rankings. The row labeled Raw skips smoothing but retains other map transformations. Raw TBK has higher ROAD and lower Deletion than the default, while smoothing increases Insertion from 0.57623 to 0.59355. The paired default advantage over CoIBA does not establish a significant raw-map advantage. Constant/random/zero-map Infidelity controls and matched normalization/postprocessing comparisons are unavailable.

Table 4: Smoothing ablation for ViT-Base. Gate statistics are unchanged because smoothing is applied after saliency estimation.

Smoothing	ROAD	Ins.	Del.	Infid.	Gate mean	Gate std.	Near-one ratio
Mean 3×3	0.28495	0.59355	0.11934	0.00973	0.65616	0.18761	0.00000
Raw	**0.28681**	0.57623	**0.10492**	**0.00972**	0.65616	0.18761	0.00000
Mean 5×5	0.28227	**0.60056**	0.11870	0.01002	0.65616	0.18761	0.00000
Gaussian 3×3	0.28560	0.59046	0.11685	**0.00972**	0.65616	0.18761	0.00000

6 Limitations and Conclusion

TBK offers a concrete internal sensitivity interface for frozen ViT/DeiT models: finite CE endpoint optimization followed by unsigned gate-path integration. Its mathematical bounds describe that path, while the strongest inferential empirical evidence is the smoothed follow-up Insertion comparison. Deterministic scaling does not establish MI compression, CE optimization does not establish necessity, and the unsigned scores do not conserve a signed output difference.

The evaluation is limited to ImageNet-1K and two non-hierarchical base models. IN-A/IN-R, Swin, localization or ground-truth attribution benchmarks such as FunnyBirds, and human-centered validation remain untested. Hierarchical models require an explicit correspondence between tokens across changing resolutions; the same shared vector cannot simply be assumed applicable. Component comparisons against signed/weighted gate IG and preservation diagnostics (label agreement, target probabilities, loss traces) are also missing. Recovering the exact result-generating implementation and metric protocol is necessary before reproducibility can be claimed. TBK requires multiple white-box forward/backward passes and is slower than one-shot gradients, with the historical throughput and its timing limitations given in Appendix E.

AI USE STATEMENT

Generative AI tools assisted with drafting and revising parts of the manuscript. The authors reviewed the suggested text and remain responsible for the experimental design, implementation, data, numerical results, methodological claims, and final text.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *ACL*, 2020.

Alexander A Alemi et al. Deep variational information bottleneck. In *ICLR*, 2017.

Oren Barkan, Yuval Asher, Amit Eshel, Noam Koenigstein, et al. Visual explanations via iterated integrated attributions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2073–2084, 2023.

Walid Bousselham, Angie Boggust, Sofian Chaybouti, Hendrik Strobelt, and Hilde Kuehne. LeGrad: An explainability method for vision transformers via feature formation sensitivity. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20336–20345, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Bousselham_LeGrad_An_Explainability_Method_for_Vision_Transformers_via_Feature_Formation_ICCV_2025_paper.html.

Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 397–406, 2021a.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *CVPR*, 2021b.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Alexandre Englebert, Sédrick Stassin, Géraldin Nanfack, Sidi Ahmed Mahmoudi, Xavier Siebert, Olivier Cornu, and Christophe De Vleeschouwer. Explaining through transformer input sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 806–815, 2023. URL https://openaccess.thecvf.com/content/ICCV2023W/NIVT/html/Englebert_Explaining_Through_Transformer_Input_Sampling_ICCVW_2023_paper.html.

Gabriel Erion et al. Improving performance of deep learning models with axiomatic attribution priors and Expected Gradients. *Nature Machine Intelligence*, 2021.

Robin Hesse, Simone Schaub-Meyer, and Stefan Roth. Fast axiomatic attribution for neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 19513–19524, 2021.

Jung-Ho Hong, Ho-Joong Kim, Kyu-Sung Jeon, and Seong-Whan Lee. Comprehensive information bottleneck for unveiling universal attribution to interpret vision transformers. In *CVPR*, 2025.

Andrei Kapishnikov, Subhashini Venugopalan, Besim Avci, Ben Wedin, Michael Terry, and Tolga Bolukbasi. Guided integrated gradients: An adaptive path method for removing noise. In *arXiv preprint arXiv:2106.09788*, 2021.

Faridoun Mehri, Mahdieh Soleymani Baghshah, and Mohammad Taher Pilehvar. LibraGrad: Balancing gradient flow for universally better vision transformer attributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 67–78, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Mehri_LibraGrad_Balancing_Gradient_Flow_for_Universally_Better_Vision_Transformer_Attributions_CVPR_2025_paper.html.

Deng Pan, Xin Li, and Dongxiao Zhu. Explaining deep neural network models with adversarial gradient integration. In *Thirtieth International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *BMVC*, 2018.

Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. A consistent and efficient evaluation strategy for attribution methods. *arXiv preprint arXiv:2202.00449*, 2022.

Karl Schulz et al. Restricting the flow: Information bottlenecks for attribution. *ICLR*, 2020.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *ICML*, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop*, 2014.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. In *Allerton Conference on Communication, Control, and Computing*, 2000.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

Chase Walker, Sumit Jha, and Rickard Ewetz. Metric-driven attributions for vision transformers. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4e21153e79aff242492146d78d09fcdb-Abstract-Conference.html.

Ying Wang, Tim GJ Rudner, and Andrew G Wilson. Visual explanations of image-text representations via multi-modal information bottleneck attribution. *Advances in Neural Information Processing Systems*, 36:16009–16027, 2023.

Junyi Wu, Bin Duan, Weitai Kang, Hao Tang, and Yan Yan. Token transformation matters: Towards faithful post-hoc explanation for vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10926–10935, 2024. URL https://arxiv.org/abs/2403.14552.

Weiyan Xie, Xiao-Hui Li, Caleb Chen Cao, and Nevin L Zhang. ViT-CX: Causal explanation of vision transformers. *arXiv preprint arXiv:2211.03064*, 2022.

Shawn Xu, Subhashini Venugopalan, and Mukund Sundararajan. Attribution in scale and space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9680–9689, 2020.

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in) fidelity and sensitivity of explanations. *Advances in neural information processing systems*, 32, 2019.

Zhiyu Zhu, Huaming Chen, Xinyi Wang, Jiayu Zhang, Zhibo Jin, Jason Xue, and Jun Shen. Iterative search attribution for deep neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 62332–62348. PMLR, 2024a. URL https://proceedings.mlr.press/v235/zhu24a.html.

Zhiyu Zhu, Huaming Chen, Jiayu Zhang, Xinyi Wang, Zhibo Jin, Minhui Xue, Dongxiao Zhu, and Kim-Kwang Raymond Choo. Mfaba: A more faithful and accelerated boundary-based attribution method for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024b.

Zhiyu Zhu, Zhibo Jin, Jiayu Zhang, and Huaming Chen. Enhancing model interpretability with local attribution over global exploration. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 5347–5355, 2024c.

Zhiyu Zhu et al. Narrowing information bottleneck theory for multimodal image-text representations interpretability. In *ICLR*, 2025.

A INFORMATION-THEORETIC BACKGROUND AND ITS LIMITS

For a population distribution $p(Y, Z)$ and a variational classifier q_θ, assuming the expectations exist,

$$\mathrm{CE}(q_\theta) = H(Y \mid Z) + \mathbb{E}_Z \,\mathrm{KL}(p(\cdot \mid Z) \| q_\theta(\cdot \mid Z)), \tag{9}$$

$$I(Z;Y) \geq H(Y) - \mathrm{CE}(q_\theta). \tag{10}$$

This population lower bound motivates a predictive surrogate (Alemi et al., 2017); its variational gap can change with the representation. Reducing one sample's CE does not establish increased population MI. For a point-mass explanation problem with fixed y, $H(Y) = I(Z;Y) = 0$, although the model's CE may vary. The operational objective therefore targets a class score, not a verified information-preservation constraint.

Lemma 2 (Finite positive-noise Gaussian channel)**.** *Let $X \sim \mathcal{N}(\mu_X, v)$ with $v > 0$ and independent $\varepsilon_\sigma \sim \mathcal{N}(0, \sigma^2)$ with $\sigma > 0$. For $Z_\sigma = \alpha X + (1-\alpha)\varepsilon_\sigma$, $0 < \alpha < 1$,*

$$I(X; Z_\sigma) = \tfrac{1}{2} \log \left(1 + \frac{\alpha^2 v}{(1-\alpha)^2 \sigma^2} \right).$$

It is strictly increasing in α at fixed positive noise and diverges to $+\infty$ for every fixed $\alpha > 0$ as $\sigma \to 0$.

proof. Gaussian entropy is $\frac{1}{2}\log(2\pi e \ \mathrm{Var})$. The marginal variance of Z_σ is $\alpha^2 v + (1-\alpha)^2\sigma^2$, and its conditional variance given X is $(1-\alpha)^2\sigma^2$. Subtracting the entropies yields the formula. The ratio $\alpha/(1-\alpha)$ increases on $(0,1)$, proving the monotonicity; the divergence follows directly. Equal input/noise variances recover $\frac{1}{2}\log(1 + \alpha^2/(1-\alpha)^2)$. □

Although $Z_\sigma \to \alpha X$ in mean square, the MI limit is not a finite deterministic bottleneck. Nonzero scalar multiplication is invertible: continuous nondegenerate Gaussian X has $I(X; \alpha X) = +\infty$, while discrete finite-entropy X has $I(X; \alpha X) = H(X)$. A fair Bernoulli variable retains one bit for both $\alpha = 0.01$ and 0.99. Only $\alpha = 0$ makes this scalar channel constant. No Gaussianity or sub-Gaussianity assumption is needed for TBK's deterministic graph, and no unproved extension of the noisy formula is used to justify its compression.

B EXISTING FOLLOW-UP PROTOCOL CHECKS

The source reports these checks on 1000 ViT-Base images, with Infidelity on a configured 200-image subset, $T = 20$, $m = 20$, and default blocks 4–11. Source inspection clarifies their loader and some evaluator settings, but original image identities, tensor provenance, and complete historical execution bindings remain unresolved. Values below are retained as reported.

Endpoint diagnostics. The learned default has mean opening 0.65616 and reported near-one ratio 0.00000. Partial opening after finite optimization does not prove minimality or necessity. The all-open and random alternatives have weaker ROAD and Insertion means. Initialization materially affects results: the Init 0.1 row is substantially worse on these metrics, while Init 0.9 has slightly higher Insertion and more near-one gates. The L1 rows are separate historical diagnostic variants; they do not change the default CE-only algorithm.

Layer selection. The default set has the best displayed ROAD, Insertion, and Deletion among the tested subsets. This measures the effect of changing the intervention, not intrinsic semantic content of those layers or universal optimality. Infidelity differences are small and favor other subsets in some rows.

Repeatability. These five 200-image runs repeat aggregate metrics; they do not measure input-perturbation robustness or rank agreement. The displayed ROAD variation also means aggregate outputs are not identical across all reported metrics. Deterministic gate equations alone do not fix every numerical or evaluator source of variation.

Table 5: Stage A gate diagnostics. ROAD/Insertion are higher-is-better; Deletion/Infidelity are lower-is-better.

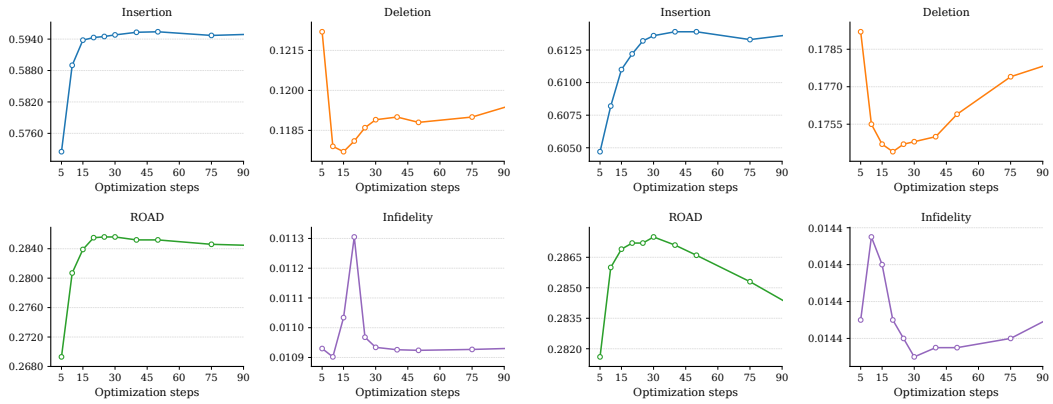
Setting	ROAD	Ins.	Del.	Infid.	Gate mean	Gate std.	Near-one ratio
Learned default	**0.28495**	0.59355	0.11934	0.00973	0.65616	0.18761	0.00000
All-open	0.27162	0.58247	0.11995	0.01012	1.00000	0.00000	1.00000
Random gate	0.23597	0.54896	0.14700	0.00985	0.50129	0.28743	0.04987
Init 0.1	0.20865	0.51040	0.15781	**0.00972**	0.29054	0.17584	0.00000
Init 0.9	0.28322	**0.59552**	0.12119	0.00974	0.82507	0.14212	0.41119
$\lambda = 10^{-4}$	0.28488	0.59396	0.11940	0.00973	0.65507	0.18774	0.00000
$\lambda = 5\times10^{-4}$	0.28354	0.59278	0.11957	0.00973	0.65087	0.18770	0.00000
$\lambda = 10^{-3}$	0.28233	0.59105	0.12024	0.00973	0.64567	0.18705	0.00000

Table 6: Existing target-layer ablation (ViT-Base); values and tested layer sets are unchanged.

Layer set	ROAD	Ins.	Del.	Infid.	Gate mean	Gate std.	Near-one ratio
Default 4–11	**0.28495**	**0.59355**	**0.11934**	0.00973	0.65616	0.18761	0.00000
Early 0–3	0.20452	0.53496	0.17763	0.00972	0.57018	0.20440	0.00000
Mid 4–7	0.25743	0.57469	0.13939	0.00972	0.56931	0.22609	0.00000
Late 8–11	0.25388	0.57026	0.14413	0.00971	0.71473	0.12359	0.00000
Last2 10–11	0.23861	0.55235	0.14721	**0.00971**	0.73453	0.11010	0.00000
Sparse 4/6/8/10	0.24300	0.55404	0.15077	0.00977	0.57469	0.25518	0.00000
All12	0.22394	0.54077	0.16208	0.00974	0.56156	0.23981	0.00000

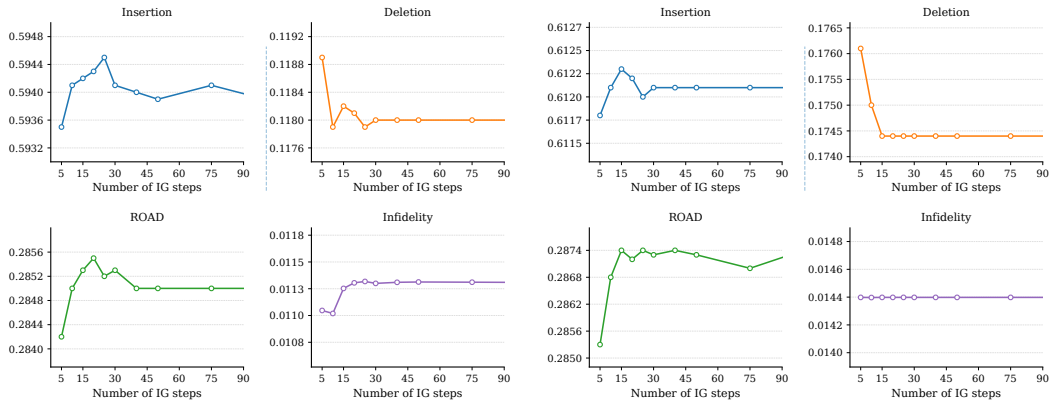
Table 7: Existing repeated 200-image runs for the learned default. These are aggregate repeatability checks, not rank-stability tests.

Run	ROAD	Ins.	Del.	Gate mean	Near-one ratio
Seed 3407	0.28623	0.59026	0.11236	0.65368	0.00000
Seed 3408	0.28669	0.59026	0.11236	0.65368	0.00000
Seed 3409	0.28749	0.59026	0.11236	0.65368	0.00000
Seed 3410	0.28650	0.59026	0.11236	0.65368	0.00000
Seed 3411	0.28673	0.59026	0.11236	0.65368	0.00000



(a) ViT-Base (b) DeiT-Base

Figure 1: **Ablation on optimization steps (`opt_steps`) in Stage A.** Existing proxy-metric curves show plateaus over parts of the tested range. These aggregate observations do not establish convergence of the gate vector, necessary pathways, or a stable optimizer solution.



(a) ViT-Base (b) DeiT-Base

Figure 2: **Ablation on integration steps (`ig_steps`) in Stage B.** Existing proxy-metric curves vary modestly over parts of the tested range. Metric plateaus do not certify quadrature error or stability of per-image gate rankings.

C EXISTING STEP ABLATIONS

The archived curves vary optimization and integration budgets. Several metric curves plateau around 15–20 steps in the evaluated range. This is a metric-level observation, not proof that each endpoint converges, each target prediction is preserved, or quadrature noise cancels. Absolute-value accumulation can preserve gradient oscillations. The reported main defaults remain 50 optimization steps for ViT, 30 for DeiT, and 20 integration steps for both. Layer set, initialization, optimizer, and smoothing are additional material choices beyond these two budgets.

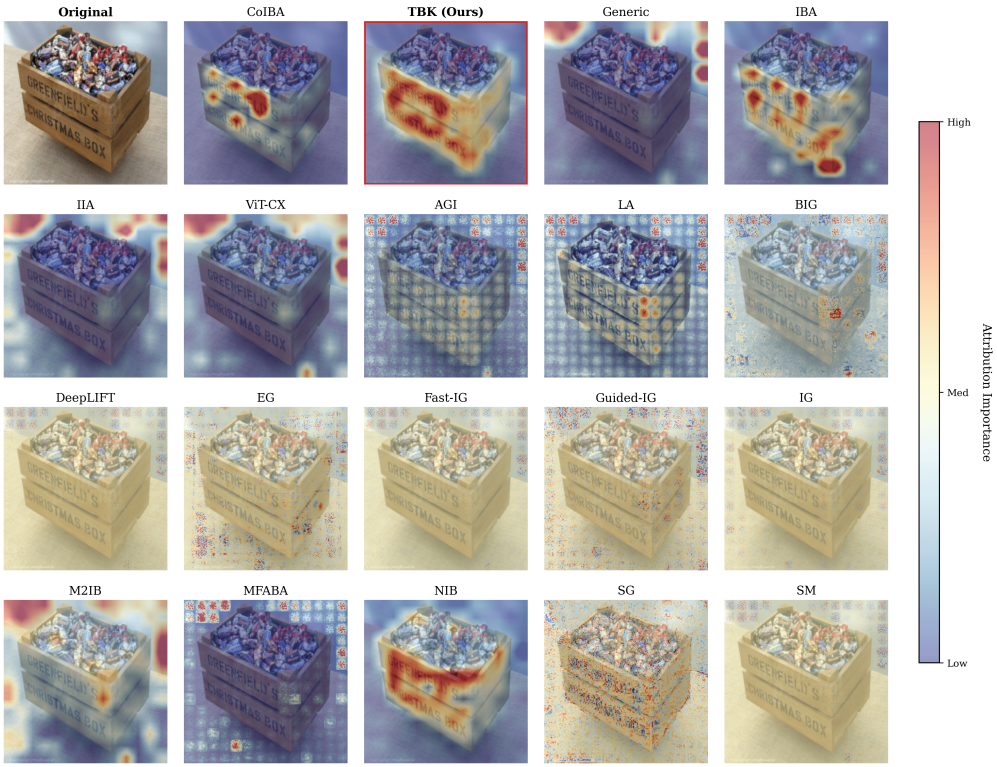


Figure 3: Existing selected crate example. Spatial concentration differs across methods; no ground-truth attribution is provided.

D SELECTED QUALITATIVE EXAMPLES

The following are existing selected illustrations, not random samples or a new failure-case evaluation. The crate, scoreboard, and king-crab grids show how different methods distribute saliency over object and context. Such visual agreement with object outlines does not establish the model's true evidence. The source's visualization-normalization claim could not be independently checked against plotting code. Full grids are retained for inspection.

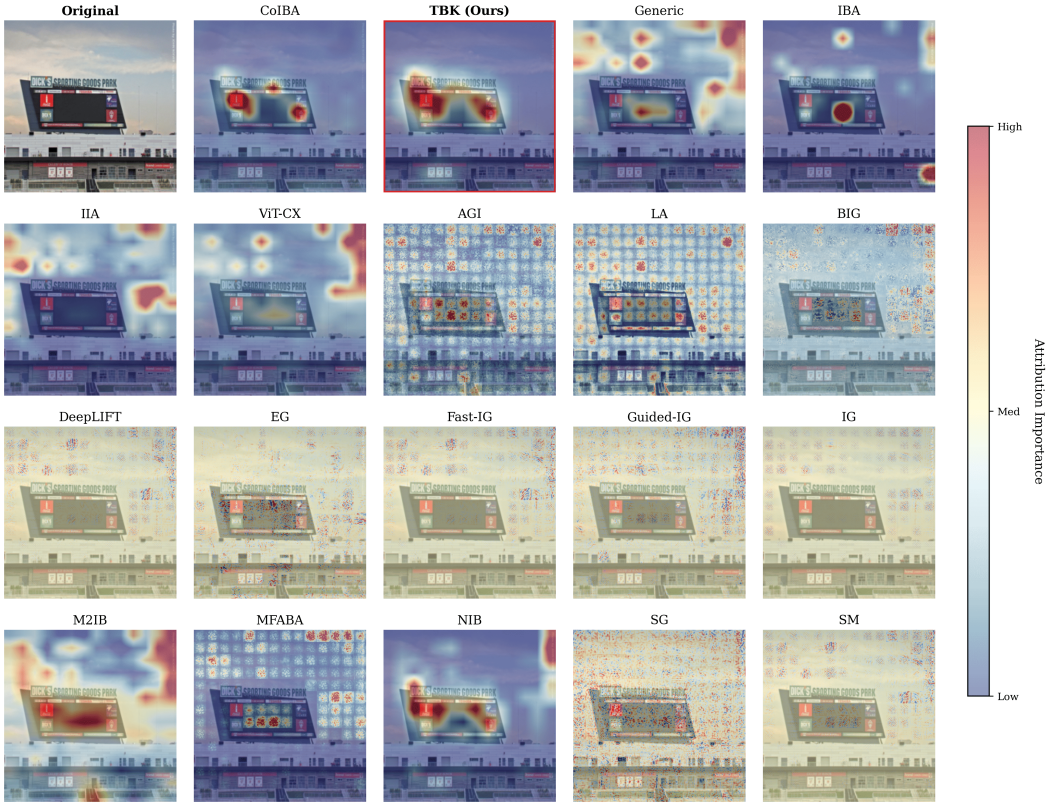


Figure 4: Existing selected scoreboard example; attribution on background is not by itself an error.

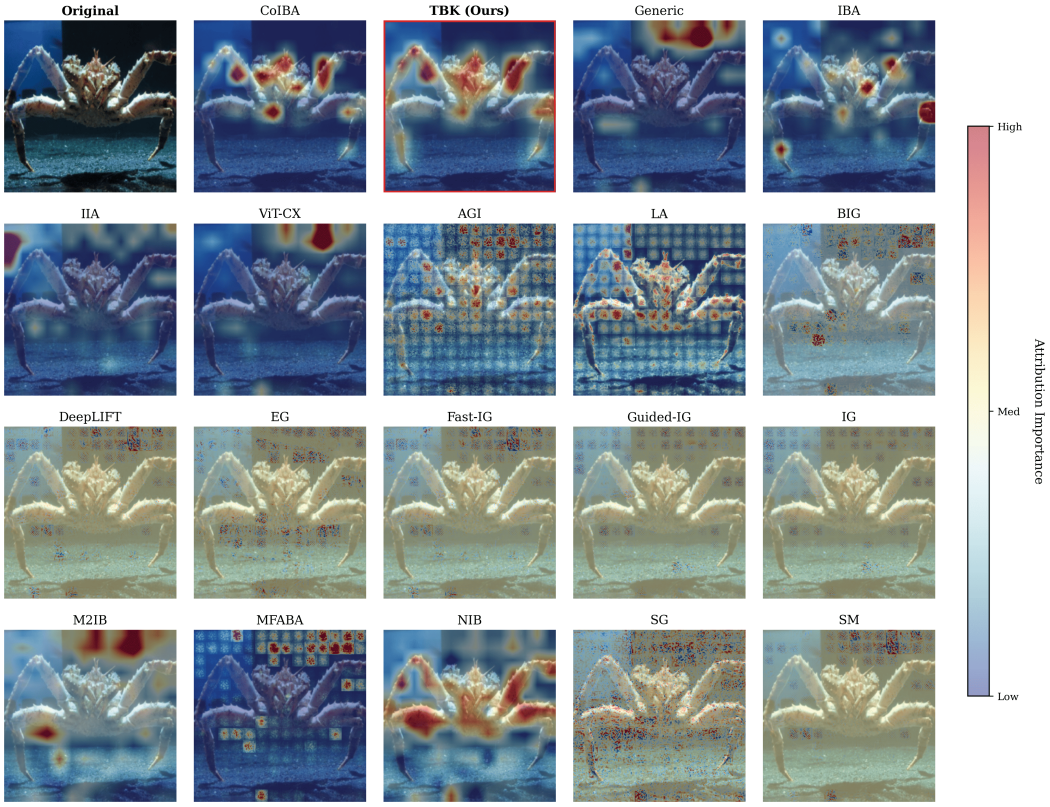


Figure 5: Existing selected king-crab example; visual coherence does not prove joint causal support.

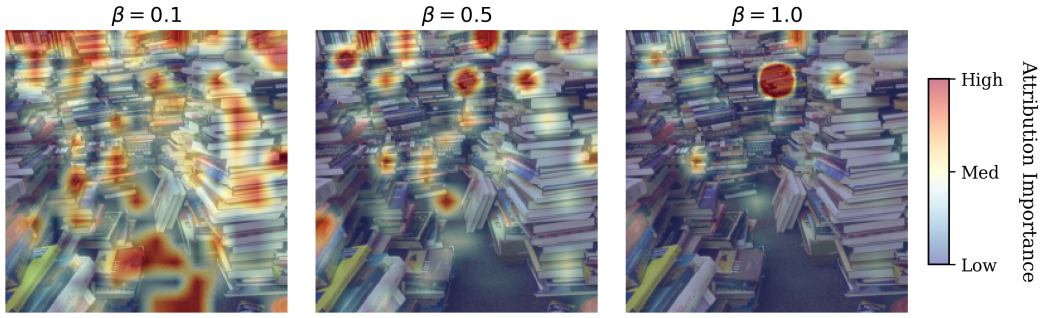


Figure 6: Existing CoIBA bookshop maps for different objective weights; the example illustrates objective dependence.

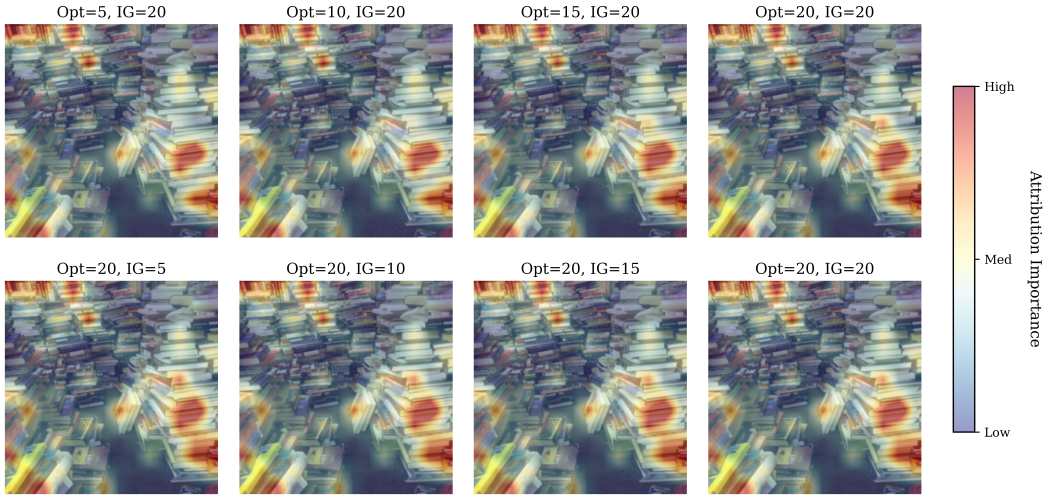


Figure 7: Existing TBK bookshop maps for different step budgets. Similar visual focus in one selected example does not establish general rank robustness.

The bookshop comparison changes CoIBA's objective weight β and, separately, TBK's step budgets. These are different interventions in the explanation procedure. A changed β is not itself randomness, and this example does not establish a matched ranking-robustness advantage.

E REPORTED EFFICIENCY AND INFRASTRUCTURE

The source identifies `blocks.4.norm1` through `blocks.11.norm1` for 12-block models, including the evaluated `vit_base_patch16_224.orig_in21k_ft_in1k` and `deit_base_patch16_224`. Its additional lists name `vit_tiny_patch16_224`, `vit_small_patch16_224`, `deit_tiny_patch16_224`, and `deit_small_patch16_224`, and propose blocks 8–23 for `vit_large_patch16_224`. Those lists are not evidence of evaluations on these additional models. Equation 1 aligns with the recovered forward-output hook. No checkpoint/source/config hash binding was supplied for every main-table result, so the match cannot establish historical identity by itself.

The reported FPS values favor TBK over CoIBA on these two backbones but favor several simpler methods over TBK. Stage A and Stage B require multiple forward/backward passes, with cost approximately proportional to $T + m$ for a fixed architecture. The archive does not establish batch size, warm-up, synchronization, timing scope, or baseline sampling budgets, so the values are descriptive and not a controlled throughput reproduction.

Table 8: Efficiency comparison (FPS ↑). Higher is better.

Method	ViT-Base FPS	DeiT-Base FPS
Generic	4.48	4.66
IBA	1.34	1.61
IIA	1.94	2.84
Vitcx	2.39	5.26
AGI	0.11	0.16
LA	0.08	0.10
BIG	0.23	0.22
M2IB	3.76	2.89
MFABA	48.85	36.29
NIB	23.15	18.24
SG	7.38	7.29
SM	66.27	74.37
CoIBA	0.78	0.34
TBK (ours)	0.92	1.14

Table 9: Hardware and software configuration of the experimental server.

Component	**Specification**
GPUs	2 × NVIDIA L40S (46 GB memory per GPU)
CPU	Intel Xeon Gold 6442Y (24 physical cores, 2.6 GHz)
System Memory	503 GB RAM
Operating System	Linux (CentOS/RHEL-based distribution)
Kernel Version	4.18.0-553.40.1.el8_10.x86_64
NVIDIA Driver	570.144
CUDA Version	12.8

F OUTSTANDING EVIDENCE REQUIREMENTS

Further tests should isolate unsigned, signed, absolute-signed and endpoint-weighted scores with identical hooks, endpoints and postprocessing; compare independent layer gates; and measure original-versus-gated predictions. Recent ViT baselines, shifted data and hierarchical backbones would address scope. Rank perturbations and Infidelity scale controls would assess stability and metric interpretation. These experiments, ground-truth attribution benchmarks and human audits remain outstanding; the present bounds concern only the specified graph and path.