CIGMA: Causal Information-Gain Mechanistic Attribution of Attention Heads in Vision Transformers

Maisha Maliha Dean F. Hougen

Appendix

A. Algorithm of CIGMA

CIGMA is presented in Algorithm 1.

B. Theoretical and Formal Definitions

This appendix makes precise the quantities used in the main paper for measuring foreground and background reliance. We first define the model's predictive distributions and the foreground/background counterfactuals. We then formalize the Jensen–Shannon (JS) divergence and the Foreground/Background Information Gains (FIG/BIG), and finally derive the Background Influence Ratio (BIR), establishing its range and interpretation. Throughout, all logarithms are natural, so information is measured in nats and JS divergence is bounded by $\log 2$.

B.1. Predictive Distributions and Counterfactuals

Let

$$f_\theta : \mathcal{I} \to \mathbb{R}^C$$

be a pretrained image classifier that maps an RGB image

$$I_i \in \mathcal{I} \subset \mathbb{R}^{H\times W\times 3}$$

to logits over C classes. For a temperature parameter $\tau > 0$, the model induces a predictive distribution

$$p_\tau(y \mid I_i) = \mathrm{softmax}\left(\frac{f_\theta(I_i)}{\tau}\right) \tag{12}$$

$$= \frac{\exp\big(f_\theta(I_i)_y/\tau\big)}{\sum_{c=1}^{C}\exp\big(f_\theta(I_i)_c/\tau\big)}, \tag{13}$$

which lies on the probability simplex

$$p_\tau(y \mid I_i) \in \Delta^{C-1}. \tag{14}$$

For notational convenience, we denote the original predictive distribution for image I_i as

$$q_i \coloneqq p_\tau(y \mid I_i). \tag{15}$$

The main paper constructs binary foreground masks

$$S_i \in \{0,1\}^{H\times W} \tag{16}$$

for each image I_i, discovered via a self-supervised optimization procedure that preserves q_i when background pixels are replaced with a baseline image. Using these masks, we form two counterfactual images that selectively remove either the foreground or the background, using the same per-image baseline B_i as in the mask optimization:

$$I_{i,\,-\mathrm{fg}} = (1 - S_i)\odot I_i + S_i \odot B_i \quad \text{(remove foreground)} \tag{17}$$

$$I_{i,\,-\mathrm{bg}} = S_i \odot I_i + (1 - S_i)\odot B_i \quad \text{(remove background)}, \tag{18}$$

where $\odot$ denotes elementwise multiplication. When the foreground is removed ($I_{i,\,-\mathrm{fg}}$), only background pixels remain visible to the model; when the background is removed ($I_{i,\,-\mathrm{bg}}$), only foreground pixels contribute to the prediction.

For each counterfactual, we obtain the corresponding predictive distributions

$$q_{i,\,-\mathrm{fg}} \coloneqq p_\tau(y \mid I_{i,\,-\mathrm{fg}}) \quad \text{(prediction without foreground)} \tag{19}$$

$$q_{i,\,-\mathrm{bg}} \coloneqq p_\tau(y \mid I_{i,\,-\mathrm{bg}}) \quad \text{(prediction without background)}, \tag{20}$$

which capture how the model's beliefs change when specific image regions are ablated. Comparing these with the original distribution q_i reveals the information content of the foreground and background regions.

B.2. Jensen–Shannon Divergence

To quantify how much the predictive distributions change under these counterfactuals, we use the Jensen–Shannon divergence between distributions $p, q \in \Delta^{C-1}$. The JS divergence is defined as

$$\mathrm{JS}(p,q) = \frac{1}{2}\,\mathrm{KL}\big(p \,\|\, m\big) + \frac{1}{2}\,\mathrm{KL}\big(q \,\|\, m\big) \tag{21}$$

$$m = \frac{1}{2}\,(p+q), \tag{22}$$

where $\mathrm{KL}(\cdot\|\cdot)$ is the Kullback–Leibler divergence and m is the mixture distribution. This is exactly the definition used in the main paper. From standard properties of JS divergence, we have

$$0 \le \mathrm{JS}(p,q) \le \log 2, \tag{23}$$

Algorithm 1 CIGMA

Require: Pretrained classifier f_θ, true-positive set $\mathcal{D} = \{I_1, \dots, I_N\}$, number of heads to prune K, mask resolution $h \times w$

Ensure: Pruned model $f_\theta^{(-\mathcal{H}_{\text{spur}})}$ with reduced background reliance

Phase 1: Foreground Mask Discovery

for each $I_i \in \mathcal{D}$ **do**
  Initialize $M_i \in [0,1]^{h \times w}$
  $B_i \leftarrow \frac{1}{HW} \sum_{j,k} (I_i)_{jk}$ ▷ Mean-color baseline
  $q_i \leftarrow p_\tau(y \mid I_i)$
  for $t = 1$ to T_{mask} **do**
    $I_{i,\text{keep}} \leftarrow U(M_i) \odot I_i + (1 - U(M_i)) \odot B_i$
    $\mathcal{L}_i \leftarrow \lambda_{\text{JS}} \, \text{JS}\big(q_i, p_\tau(y \mid I_{i,\text{keep}})\big) + \lambda_1 \|M_i\|_1 + \lambda_{\text{TV}} \, \text{TV}(M_i)$
    $M_i \leftarrow \text{AdamStep}(M_i, \nabla_{M_i} \mathcal{L}_i)$
  end for
  $\tilde{M}_i \leftarrow U(M_i)$
  $t_i \leftarrow \text{percentile}(\tilde{M}_i, 100(1-\rho))$
  $S_i \leftarrow \mathbb{1}[\tilde{M}_i > t_i]$
end for

Phase 2: Foreground/Background Info. Gains

for each $I_i \in \mathcal{D}$ **do**
  $I_{i,-\text{fg}} \leftarrow (1 - S_i) \odot I_i + S_i \odot B_i$
  $I_{i,-\text{bg}} \leftarrow S_i \odot I_i + (1 - S_i) \odot B_i$
  $q_{i,-\text{fg}} \leftarrow p_\tau(y \mid I_{i,-\text{fg}})$
  $q_{i,-\text{bg}} \leftarrow p_\tau(y \mid I_{i,-\text{bg}})$
  $\text{FIG}_i \leftarrow \text{JS}(q_i, q_{i,-\text{fg}})$
  $\text{BIG}_i \leftarrow \text{JS}(q_i, q_{i,-\text{bg}})$
end for

Phase 3: Head-Level Causal Scoring

for each head $h \in \{1, \dots, LH\}$ **do**
  Form $f_\theta^{(-h)}$ by zeroing head h: $(W^Q)_{S_h,:} = 0$, $(W^K)_{S_h,:} = 0$, $(W^V)_{S_h,:} = 0$, $(W^O)_{:,S_h} = 0$
  for each $I_i \in \mathcal{D}$ **do**
    $q_i^{(-h)} \leftarrow p_\tau^{(-h)}(y \mid I_i)$
    $q_{i,-\text{bg}}^{(-h)} \leftarrow p_\tau^{(-h)}(y \mid I_{i,-\text{bg}})$
    $\text{BIG}_i^{(-h)} \leftarrow \text{JS}(q_i^{(-h)}, q_{i,-\text{bg}}^{(-h)})$
  end for
  $\text{CIGMA}(h) \leftarrow \frac{1}{N} \sum_{i=1}^N \big(\text{BIG}_i - \text{BIG}_i^{(-h)}\big)$
end for

Phase 4: Prune Spurious Heads

Rank heads by $\text{CIGMA}(h)$ in descending order
$\mathcal{H}_{\text{spur}} \leftarrow \{h_1, \dots, h_K\}$ (top-K heads)
Construct $f_\theta^{(-\mathcal{H}_{\text{spur}})}$ by ablating all $h \in \mathcal{H}_{\text{spur}}$
return $f_\theta^{(-\mathcal{H}_{\text{spur}})}$

and $\text{JS}(p, q)$ is symmetric and always finite, even when p and q have disjoint support.

B.2.1. Entropy-Based Expression

Let the Shannon entropy be

$$H(p) = -\sum_y p(y) \log p(y). \tag{24}$$

Using the identity

$$\text{KL}(p \,\|\, m) = -H(p) - \sum_y p(y) \log m(y), \tag{25}$$

one can show the equivalent, entropy-based form of JS divergence:

$$\text{JS}(p, q) = H(m) - \frac{1}{2} H(p) - \frac{1}{2} H(q) \tag{26}$$

$$m = \frac{1}{2}(p + q). \tag{27}$$

This expression makes the symmetry in p and q and the boundedness in $[0, \log 2]$ explicit, and it is often convenient for both analysis and implementation.

B.2.2. Gradient Behavior

The main paper optimizes masks and evaluates per-head effects using quantities that depend on $\text{JS}(p, q)$. It is therefore important that gradients be well-behaved. Because the softmax mapping from logits to (p, q) is smooth on the interior of the simplex, and $\text{JS}(p, q)$ is a smooth function of (p, q) for $p, q \in \Delta^{C-1}$ with strictly positive entries, the composite mapping from logits to $\text{JS}(p, q)$ is differentiable. In particular:

- If $p = q$, then
$$\text{JS}(p, q) = 0 \quad \Longrightarrow \quad \nabla \text{JS}(p, q) = 0, \tag{28}$$
so the JS term does not introduce spurious gradients when the two distributions coincide.
- As p and q move farther apart, the magnitude of $\nabla \text{JS}(p, q)$ grows, but $\text{JS}(p, q)$ remains bounded by $\log 2$. This boundedness prevents unbounded loss values and helps avoid gradient explosion.
- When used with the foreground/background distributions q_i, $q_{i,\,-\text{fg}}$, and $q_{i,\,-\text{bg}}$, we can write (schematically) the gradient with respect to the foreground mask S_i via the chain rule:
$$\begin{aligned} \frac{\partial}{\partial S_i} \text{JS}\big(q_i, q_{i,\,-\text{fg}}\big) = \frac{\partial \text{JS}}{\partial q_i} \frac{\partial q_i}{\partial f_\theta(I_i)} \frac{\partial f_\theta(I_i)}{\partial I_i} \frac{\partial I_i}{\partial S_i} \\ + \frac{\partial \text{JS}}{\partial q_{i,\,-\text{fg}}} \frac{\partial q_{i,\,-\text{fg}}}{\partial f_\theta(I_{i,\,-\text{fg}})} \\ \cdot \frac{\partial f_\theta(I_{i,\,-\text{fg}})}{\partial I_{i,\,-\text{fg}}} \frac{\partial I_{i,\,-\text{fg}}}{\partial S_i}. \end{aligned} \tag{29}$$
An analogous expression holds for $\text{JS}(q_i, q_{i,\,-\text{bg}})$. This structure matches how the main paper backpropagates through mask and head interventions using standard optimizers such as Adam.

Thus, JS divergence is compatible with gradient-based optimization and causal interventions used in the main methodology.

B.3. Foreground and Background Information Gains (FIG/BIG)

Using JS divergence, the main paper defines the information contribution of foreground and background regions on a per-image basis. For each image I_i, we define:

$$\mathrm{FIG}_i \coloneqq \mathrm{JS}\big(q_i,\ q_{i,\,-\mathrm{fg}}\big) \tag{30}$$

$$\mathrm{BIG}_i \coloneqq \mathrm{JS}\big(q_i,\ q_{i,\,-\mathrm{bg}}\big). \tag{31}$$

These are exactly the Foreground Information Gain (FIG) and Background Information Gain (BIG) from the main paper. From the boundedness of JS, they satisfy

$$0 \le \mathrm{FIG}_i \le \log 2 \tag{32}$$

$$0 \le \mathrm{BIG}_i \le \log 2. \tag{33}$$

The interpretation is:

- FIG_i measures how much the prediction changes when foreground is removed. A large FIG_i means that replacing the foreground with baseline pixels significantly alters the predictive distribution, indicating strong reliance on foreground evidence.
- BIG_i measures how much the prediction changes when background is removed. A large BIG_i means that removing background information (and keeping only the discovered foreground) significantly changes the prediction, indicating that the model was using background context in its original prediction. This is where spurious correlations can arise.

Aggregated over all images $I_i \in \mathcal{D}$, the pair $(\mathrm{FIG}_i, \mathrm{BIG}_i)$ provides a quantitative description of the model's reliance patterns on foreground versus background, which the main paper then links to attention heads via CIGMA.

B.4. Background Influence Ratio (BIR)

While FIG_i and BIG_i capture *absolute* information contributions, it is often convenient to summarize the *relative* importance of background versus foreground in a single normalized score. The main paper introduces the Background Influence Ratio (BIR) as such a score, confined to $[0, 1]$ and computed directly from how predictions change when foreground or background is hidden.

For each image with nontrivial information contribution

$$\mathrm{FIG}_i + \mathrm{BIG}_i > 0, \tag{34}$$

we define the per-image BIR as

$$\mathrm{BIR}_i \coloneqq \frac{\mathrm{BIG}_i}{\mathrm{FIG}_i + \mathrm{BIG}_i}. \tag{35}$$

If $\mathrm{FIG}_i = \mathrm{BIG}_i = 0$ (i.e., neither foreground nor background removal changes the prediction), then the image has no measurable dependence on either region. In that degenerate case, one can define

$$\mathrm{BIR}_i \coloneqq \tfrac{1}{2} \tag{36}$$

by symmetry, though such cases are rare in practice and do not affect reported averages.

B.4.1. Range and Interpretation

By construction, for images with $\mathrm{FIG}_i + \mathrm{BIG}_i > 0$ we have:

$$\mathrm{FIG}_i \ge 0, \quad \mathrm{BIG}_i \ge 0 \quad \Longrightarrow \quad 0 \le \mathrm{BIR}_i \le 1. \tag{37}$$

More explicitly:

$$\mathrm{BIR}_i > \tfrac{1}{2} \iff \mathrm{BIG}_i > \mathrm{FIG}_i \tag{38}$$

$$\mathrm{BIR}_i = \tfrac{1}{2} \iff \mathrm{BIG}_i = \mathrm{FIG}_i \tag{39}$$

$$\mathrm{BIR}_i < \tfrac{1}{2} \iff \mathrm{BIG}_i < \mathrm{FIG}_i. \tag{40}$$

Thus:

- $\mathrm{BIR}_i > 0.5$ means the model relies more on background than foreground for image I_i.
- $\mathrm{BIR}_i < 0.5$ means the model relies more on foreground than background.
- $\mathrm{BIR}_i = 0.5$ corresponds to equal reliance.

This matches the qualitative interpretation in the main paper, where methods with BIR > 0.5 are described as background-dominated, and methods with BIR < 0.5 are described as foreground-focused.

B.4.2. Dataset-Level BIR

To summarize background reliance at the dataset level, we average the per-image ratios over the curated true-positive set $\mathcal{D}$ of N images:

$$\mathrm{BIR} \coloneqq \frac{1}{N} \sum_{i=1}^{N} \mathrm{BIR}_i. \tag{41}$$

Since each $\mathrm{BIR}_i \in [0, 1]$, we immediately have

$$0 \le \mathrm{BIR} \le 1. \tag{42}$$

This dataset-level BIR is exactly the score visualized in Figure 1 of the main paper, where lower values indicate weaker background reliance. For example, the main results show that CIGMA achieves a dataset-level BIR of approximately 0.07, indicating that after identifying and pruning spurious attention heads, the model's predictions depend only minimally on background cues, while still preserving or improving accuracy and calibration.

B.4.3. Relation to FIG/BIG and CIGMA

The triplet $(\mathrm{FIG}_i, \mathrm{BIG}_i, \mathrm{BIR}_i)$ provides a coherent, mathematically grounded view of foreground/background reliance:

- $(\mathrm{FIG}_i, \mathrm{BIG}_i)$, derived via JS divergence, quantify absolute changes in the predictive distribution when foreground or background is ablated.
- BIR_i renormalizes these two quantities into a single interpretable scalar in $[0, 1]$ that directly encodes the balance between background and foreground reliance.
- The CIGMA score for a given attention head (defined in the main paper via changes in BIG_i under head ablation) measures how much that head contributes to background information processing. Heads with large CIGMA scores yield large reductions in BIG_i (and hence in BIR_i) when removed, without significantly harming FIG_i.

Together, these definitions provide the theoretical backbone for the empirical findings in the main paper: by explicitly measuring and reducing background influence in terms of FIG, BIG, and BIR, the CIGMA procedure produces models that are more foreground-focused, better calibrated, and less reliant on spurious contextual cues.

C. Training, Fine-Tuning, and Baseline Setup

We now describe how we obtain the entries in Tables 1 and 2, and how all baselines are adapted to our setting. Throughout, we use the datasets, backbones, and evaluation protocol introduced in §5. We benchmark CIFAR-10, CIFAR-100 [9], and Tiny-ImageNet [10] using canonical train/validation/test splits, and evaluate three VLM backbones of varying size and training recipes: InternVL2-26B [5], LLaVA-1.6 [14], and LLaVA-1.5-13B [13]. In the **zero-shot** regime (Table 1), all backbone weights remain frozen and no gradient updates are performed; the "Original" rows correspond to directly evaluating these pretrained models on the classification tasks via their prompt-based interfaces. In the **fine-tuned (ft)** regime (Table 2), the "Original (ft)" rows correspond to backbones that have been fine-tuned on the downstream labels using the standard supervised objective (cross-entropy over class logits) under their official training configurations.

For all **Original (ft)** models, we adhere strictly to the published fine-tuning recipes of the corresponding backbone. Concretely, for InternVL2-26B and the LLaVA variants [5, 13], we use the same optimizer (e.g., AdamW or its specified variant), learning-rate schedule, batch size, number of epochs, and weight decay as in the official configurations, and we preserve all data augmentations (such as random cropping and horizontal flipping) used in those recipes. If the original configuration defines early stopping or model selection based on a validation metric, we apply it unchanged; otherwise, we train for the fixed number of epochs specified in the official setup. We do not introduce any additional tuning specific to our method in this stage. The goal is to provide a strong and faithful "backbone-only" reference: the **Original (ft)** rows in Table 2 therefore represent what a practitioner would obtain by applying the standard fine-tuning pipeline to these VLMs on the given datasets.

CIGMA in the fine-tuned regime, reported as **Ours (CIGMA, ft)** in Table 2, is applied *post hoc* to the fixed **Original (ft)** models. After fine-tuning is complete, we freeze all backbone parameters and run the mask-based analysis and head scoring procedure described in §5: we set the mask resolution to $h = w = 32$ and optimize the soft masks using Adam with learning rate $\eta = 0.05$ for 250 iterations, with loss weights $\lambda_{\mathrm{JS}} = 1.0$, $\lambda_1 = 0.01$, and $\lambda_{\mathrm{TV}} = 0.1$. We binarize each mask at the 65-th percentile (corresponding to $\rho = 0.35$ of pixels retained as foreground), replace ablated regions with the per-image mean color, and use a fixed softmax temperature $\tau = 1.0$. The true positive set $\mathcal{D}$ is constructed by selecting 40% of correctly classified training examples whose predictions match the ground-truth label, and CIGMA scores are computed over all attention heads in the backbone. We then ablate the top-$K = 16$ heads with the highest CIGMA scores, yielding the edited model reported as "CIGMA (ft)". Importantly, this procedure does not involve any additional gradient-based fine-tuning of the backbone; all changes occur at the level of head ablation, so the optimizer, schedule, batch size, epochs, weight decay, augmentations, and early stopping that produced **Original (ft)** remain untouched.

In the **zero-shot** regime (Table 1), we treat the pretrained VLMs as fixed feature extractors and compare CIGMA against five training-free pruning and editing baselines: TopV [23], ATP-LLaVA [24], DivPrune [2], EfficientLLaVA [12], and MDP [20]. These methods operate purely at inference time, editing token sets or attention patterns without updating backbone parameters. We instantiate each baseline using its official implementation and recommended configuration, and keep the backbone weights frozen throughout. The only hyperparameters adapted to our setting are the pruning and ablation budgets, such as token budgets, head counts K, or effective keep-rates/latency targets. Following §5, we evaluate the discrete set of pruning budgets suggested by each baseline on the validation set and select the configuration whose effective keep-rate or latency is closest to the target recommended in the original paper. This ensures that our comparisons are made under comparable computational budgets, while preserving the original design choices of each baseline.

For the **fine-tuned baselines** (Table 2), we compare CIGMA against three training-based methods: CoBalT [3] and RAVL [21], which explicitly target spurious correlations in vision models, and CHG [16], which was originally

proposed for interpreting attention-head roles in causal LLMs (Llama 3 family) by learning soft gates that classify heads as facilitating, interfering, or irrelevant for a given task. Because CHG's core mechanism—identifying and suppressing task-interfering heads via learned gating—is directly analogous to our goal of locating spurious heads, we adapt it to our ViT backbones by replacing the next-token-prediction objective with the supervised classification objective used by our other fine-tuned baselines and applying the same per-head gating procedure to the vision-transformer attention heads. For each of these baselines, we start from the same pretrained backbones as in our other experiments and follow the authors' recommended training procedures as closely as possible: we adopt their specified optimizer, learning-rate schedule, batch size, number of epochs, weight decay, augmentations, and any early-stopping or model-selection rules exactly as described in the respective papers. When the baselines provide several recommended values for key meta-parameters (e.g., regularization strengths, gating thresholds, perturbation magnitudes), we restrict ourselves to this finite set and choose the value that achieves the best validation accuracy on the held-out validation set. We do not perform any wider hyperparameter search beyond the ranges suggested by the original works. The resulting CoBalT, RAVL, and CHG rows in Table 2 thus reflect faithful instantiations of those methods on our backbones and datasets.

Overall, this setup ensures that the differences reported in Tables 1 and 2 are attributable to how each method addresses spurious correlations, rather than to favorable choices of optimizer or training schedule. In the zero-shot regime, all methods—including CIGMA and the training-free baselines—operate on the same frozen backbones and differ only in how they edit or prune tokens and heads. In the fine-tuned regime, all methods start from the same pretrained checkpoints and rely on their official training pipelines; CIGMA (ft) then applies foreground-based, BIR-driven head ablation post-training, without altering any fine-tuning hyperparameters. This design makes our comparisons both controlled and practically relevant for scenarios where retraining is costly or infeasible.

D. Ablation Study Experimental Setup

We perform two ablation studies on Tiny-ImageNet using InternVL2-26B to examine the effect of (i) the number of pruned attention heads K and (ii) the fraction of true positive images used to construct the CIGMA scoring set. For computational efficiency, all ablation experiments are conducted on a fixed random 80% subset of the Tiny-ImageNet data; the full dataset is only used for the main results outside this section. The same model checkpoint, preprocessing pipeline, and evaluation protocol are used for all configurations within the ablation studies, and we keep the random subset of images fixed across settings to enable a fair comparison.

For each configuration, we first run the unmodified model on the ablation subset to identify correctly classified images, which we refer to as true positives. In the first ablation, we fix the true positive fraction used for CIGMA scoring to 40% of the available true positives in the subset and vary the pruning budget

$$K \in \{0, 4, 8, 16, 24, 32\},$$

where $K{=}0$ corresponds to no pruning. For each value of K, we compute CIGMA scores over all attention heads using the selected true positive images, rank heads by their scores, prune the top-K heads, and then evaluate classification accuracy on the Tiny-ImageNet portion of the ablation subset.

In the second ablation, we fix the pruning budget to $K{=}16$ and vary the fraction of true positives used to form the CIGMA scoring set. Let D_{TP} denote all correctly classified images in the ablation subset; we consider fractions

$$\frac{|D|}{|D_{\text{TP}}|} \in \{10\%, 20\%, 40\%, 60\%, 80\%, 100\%\},$$

where $D \subseteq D_{\text{TP}}$ is obtained by uniform random sampling without replacement. For each fraction, we resample D, recompute CIGMA scores, prune the top-16 heads according to these scores, and evaluate on the same Tiny-ImageNet ablation subset. Unless otherwise noted, all reported accuracies in this section are computed on this fixed 80% subset, and CIGMA scoring as well as head pruning are recomputed independently for each hyperparameter setting.

E. Complexity, Runtime, and Resource Analysis

CIGMA is designed as a training-free, head-level editing procedure on top of a fixed backbone. It is run once per dataset/backbone pair and the resulting pruned model can then be reused for all downstream evaluations. This section analyzes its computational complexity and runtime characteristics. We focus on three components: (i) self-supervised foreground mask optimization, (ii) information-gain computation (FIG, BIG, and BIR), and (iii) head-wise CIGMA scoring and pruning. Throughout, let $N = |\mathcal{D}|$ be the number of true-positive images used for head scoring, L the number of transformer layers, and H the number of attention heads per layer (so there are $L \times H$ heads in total). In the default configuration, $\mathcal{D}$ is chosen as 40% of correctly classified images on the training set, and the top $K = 16$ heads ranked by CIGMA are pruned.

E.1. Per-Image Cost: Foreground Mask Optimization

For each image $I_i \in \mathcal{D}$, CIGMA first optimizes a low-resolution continuous mask

$$M_i \in [0,1]^{h \times w} \tag{43}$$

that is upsampled to input resolution via a differentiable operator U. The counterfactual image that retains only the discovered foreground is

$$I_{i,\mathrm{keep}}(M_i) = U(M_i) \odot I_i + \big(1 - U(M_i)\big) \odot B_i, \tag{44}$$

where B_i is a per-image mean-color baseline and $\odot$ denotes elementwise multiplication.

The mask is obtained by minimizing a composite objective that encourages prediction preservation, sparsity, and smoothness. Let $q_i = p_\tau(y \mid I_i)$ be the original predictive distribution and let $\mathrm{TV}(\cdot)$ denote the total variation regularizer. The per-image loss is

$$\begin{aligned}\mathcal{L}_i(M_i) = \lambda_{\mathrm{JS}}\, \mathrm{JS}\big(q_i,\ p_\tau\big(y \mid I_{i,\mathrm{keep}}(M_i)\big)\big) \\ + \lambda_1 \|M_i\|_1 + \lambda_{\mathrm{TV}}\, \mathrm{TV}(M_i)\end{aligned} \tag{45}$$

and the optimized mask is

$$M_i^\star = \arg\min_{M_i \in [0,1]^{h \times w}} \mathcal{L}_i(M_i). \tag{46}$$

In practice, M_i is optimized with Adam for a fixed number of iterations

$$T_{\text{mask}} = 250, \tag{47}$$

using loss weights

$$\lambda_{\mathrm{JS}} = 1.0, \qquad \lambda_1 = 0.01, \qquad \lambda_{\mathrm{TV}} = 0.1. \tag{48}$$

Each gradient step requires one forward pass through the backbone on $I_{i,\mathrm{keep}}(M_i)$ and one backward pass with respect to M_i. The original distribution $q_i = p_\tau(y \mid I_i)$ is computed once per image and cached.

Thus, for each $I_i \in \mathcal{D}$, the total number of forward passes used in mask optimization is approximately

$$N^{(i)}_{\text{fwd,mask}} \approx 1 + T_{\text{mask}}, \tag{49}$$

where the "1" accounts for the initial evaluation of q_i if not already available. Aggregated over all N images, the mask discovery stage requires

$$N_{\text{fwd,mask}} \approx N\,\big(1 + T_{\text{mask}}\big) \tag{50}$$

forward evaluations of the backbone, plus a matching number of backward passes with respect to the mask parameters.

E.2. Per-Image Cost: FIG, BIG, and BIR Computation

After optimization, the soft masks $M_i^\star$ are upsampled and binarized into foreground masks

$$S_i \in \{0,1\}^{H \times W}. \tag{51}$$

Using S_i, CIGMA constructs two counterfactual images per I_i:

$$I_{i,-\mathrm{fg}} = (1 - S_i) \odot I_i + S_i \odot B_i, \tag{52}$$

$$I_{i,-\mathrm{bg}} = S_i \odot I_i + (1 - S_i) \odot B_i. \tag{53}$$

From these, we obtain the corresponding predictive distributions

$$q_{i,-\mathrm{fg}} = p_\tau(y \mid I_{i,-\mathrm{fg}}), \tag{54}$$

$$q_{i,-\mathrm{bg}} = p_\tau(y \mid I_{i,-\mathrm{bg}}). \tag{55}$$

Comparing these to q_i via the Jensen–Shannon divergence yields the Foreground and Background Information Gains:

$$\mathrm{FIG}_i = \mathrm{JS}\big(q_i,\ q_{i,-\mathrm{fg}}\big), \tag{56}$$

$$\mathrm{BIG}_i = \mathrm{JS}\big(q_i,\ q_{i,-\mathrm{bg}}\big), \tag{57}$$

and the per-image Background Influence Ratio:

$$\mathrm{BIR}_i = \frac{\mathrm{BIG}_i}{\mathrm{FIG}_i + \mathrm{BIG}_i}, \qquad \text{for } \mathrm{FIG}_i + \mathrm{BIG}_i > 0. \tag{58}$$

Computing FIG and BIG requires two additional forward passes per image (one on $I_{i,-\mathrm{fg}}$ and one on $I_{i,-\mathrm{bg}}$). Because q_i is already cached from the mask-optimization stage, the total additional forward cost for FIG/BIG/BIR over the set $\mathcal{D}$ is

$$N_{\text{fwd,IG}} = 2N. \tag{59}$$

This cost is linear in N and independent of the number of heads.

E.3. Per-Head Cost: CIGMA Scoring

The dominant additional cost in CIGMA comes from head-wise causal evaluation. For each attention head h among the $L \times H$ heads, a modified model $f_\theta^{(-h)}$ is formed that zeros out the contribution of h, and the impact on background information is recomputed on all images in $\mathcal{D}$. Concretely, for each $I_i \in \mathcal{D}$ and each head h, we compute

$$q_i^{(-h)} = p_\tau^{(-h)}(y \mid I_i), \tag{60}$$

$$q_{i,-\mathrm{bg}}^{(-h)} = p_\tau^{(-h)}(y \mid I_{i,-\mathrm{bg}}), \tag{61}$$

$$\mathrm{BIG}_i^{(-h)} = \mathrm{JS}\big(q_i^{(-h)},\ q_{i,-\mathrm{bg}}^{(-h)}\big). \tag{62}$$

The CIGMA score for head h is

$$\mathrm{CIGMA}(h) = \frac{1}{N}\sum_{i=1}^{N}\Big(\mathrm{BIG}_i - \mathrm{BIG}_i^{(-h)}\Big), \tag{63}$$

which measures the average reduction in background reliance caused by ablating h.

For each head h, this requires two forward passes per image (one on I_i and one on $I_{i,-\mathrm{bg}}$ under $f_\theta^{(-h)}$). Thus, the per-head forward-pass budget is

$$N_{\text{fwd,CIGMA}}^{(h)} = 2N, \tag{64}$$

and scoring all $L \times H$ heads requires

$$N_{\text{fwd,CIGMA}} = 2N\,(L \times H) \tag{65}$$

forward passes. This term scales linearly in both N and the total number of heads $L \times H$, and typically dominates the FIG/BIG computation, but remains a one-time offline cost per dataset/backbone pair.

E.4. End-to-End Cost and Comparison to Training

Combining the three components above, the total number of forward passes required to run CIGMA once for a given dataset and backbone can be written as

$$N_{\text{fwd,total}} \approx N\left(1 + T_{\text{mask}}\right) + 2N + 2N\,(L \times H) \tag{66}$$

$$= N\left(T_{\text{mask}} + 3 + 2LH\right). \tag{67}$$

The first term corresponds to mask optimization (including one initial forward per image), the second term to FIG/BIG/BIR computation, and the third term to head-wise CIGMA scoring. Because T_{mask} is fixed and moderate ($T_{\text{mask}} = 250$ in the default configuration) and N is only 40% of correctly classified training images, the overall cost scales linearly with both the size of the scoring set $\mathcal{D}$ and the number of heads. This cost is incurred once per dataset/backbone combination and can then be amortized over all downstream uses of the pruned model.

It is useful to compare CIGMA's cost to standard training. A conventional fine-tuning run with E epochs over the full training set of size $|\mathcal{D}_{\mathrm{train}}|$ requires on the order of

$$N_{\text{fwd,train}} \approx E\left|\mathcal{D}_{\mathrm{train}}\right| \tag{68}$$

forward (and backward) passes of the backbone. In contrast, CIGMA's forward budget is proportional to $N(T_{\text{mask}} + 3 + 2LH)$, where N is a subset of correctly classified examples and T_{mask} is fixed. For typical values of E in the tens, and for the default settings (40% true positives and $K = 16$ pruned heads selected from all $L \times H$ heads), the CIGMA pass budget is comparable to, or smaller than, a small number of additional training epochs on the backbone.

E.5. Runtime and Resource Profile

All CIGMA computations are implemented using standard mixed-precision inference and batched evaluation of both original and counterfactual images. Mask optimization, FIG/BIG/BIR computation, and head-wise CIGMA scoring share the same backbone implementation and benefit from the same hardware acceleration. In practice, CIGMA is run once per backbone–dataset pair, and the resulting pruned model is reused for all subsequent evaluations.

Because the end-to-end complexity is linear in N and $L \times H$ and does not involve any additional updates to the backbone parameters, the runtime scales predictably with the number of true-positive images selected and the size of the transformer. After the one-time offline cost, all downstream inference uses the pruned model $f_\theta^{\mathrm{CIGMA}}$ with K heads removed. Since the FLOPs and memory footprint of multi-head self-attention grow linearly with the number of heads, removing K heads reduces the attention computation proportionally to $K/(LH)$ and yields corresponding speedups and memory savings at test time, complementing any existing token or structural pruning already present in the backbone. In summary, CIGMA introduces a controllable, one-shot overhead that is modest compared to full fine-tuning while enabling persistent gains in accuracy, calibration, and reduced background reliance for all subsequent uses of the model.

F. Ablations and Sensitivity Analyses

This section investigates the sensitivity of CIGMA to its main design choices and hyperparameters. The experiments cover (i) mask-related hyperparameters, (ii) the softmax temperature τ, (iii) the layer-wise distribution of pruned heads, and (iv) stability of CIGMA scores across seeds and subsets. All results are averages over three independent runs.

F.1. Mask Hyperparameters

CIGMA relies on a differentiable mask that discovers foreground regions. This mask is controlled by the foreground fraction ρ, the mask resolution $h \times w$, the strengths of the total variation (TV) and ℓ_1 penalties, and the choice of baseline image used to replace ablated pixels.

Table 3 summarizes the effect of these hyperparameters on average accuracy, FIG, BIG, and BIR across all backbones and datasets. The default configuration (row 1) uses $\rho = 0.35$, resolution 32×32, TV weight 0.10, ℓ_1 weight 0.01, and a per-image mean-color baseline. Decreasing the foreground fraction to $\rho = 0.25$ produces sparser masks that slightly reduce BIR but also hurt accuracy, as too many informative foreground pixels are removed. Increasing ρ to 0.45 preserves more pixels and mildly improves FIG but yields higher BIG and BIR, indicating increased back-

Table 3. Ablation on mask-related hyperparameters, averaged over all backbones and datasets. FIG and BIG are reported in nats.

ρ	$h \times w$	λ_{TV}	λ_1	Baseline		Acc↑	FIG↑	BIG↓	BIR↓
0.35	32×32	0.10	0.01	mean color	(default)	95.4	0.285	0.021	0.069
0.25	32×32	0.10	0.01	mean color	(sparser mask)	94.6	0.260	0.018	0.065
0.45	32×32	0.10	0.01	mean color	(denser mask)	95.5	0.300	0.024	0.074
0.35	16×16	0.10	0.01	mean color	(coarse mask)	94.8	0.270	0.023	0.075
0.35	64×64	0.10	0.01	mean color	(fine mask)	95.5	0.288	0.021	0.068
0.35	32×32	0.05	0.01	mean color	(weaker TV)	95.2	0.281	0.022	0.072
0.35	32×32	0.20	0.01	mean color	(stronger TV)	95.3	0.279	0.020	0.067
0.35	32×32	0.10	0.005	mean color	(weaker ℓ_1)	95.3	0.290	0.022	0.071
0.35	32×32	0.10	0.02	mean color	(stronger ℓ_1)	94.9	0.268	0.019	0.066
0.35	32×32	0.10	0.01	blurred img.	(blur baseline)	95.2	0.283	0.023	0.075

ground leakage into the foreground mask.

Changing resolution from 32×32 to 16×16 makes the mask blockier and less precise, which slightly decreases accuracy and increases BIG/BIR. A higher resolution of 64×64 leads to similar performance to the default setting but requires more computation during mask optimization. Finally, replacing the mean-color baseline with a blurred-image baseline leads to a small increase in BIG and BIR, suggesting that the simple mean-color baseline is sufficient for discovering spurious background heads.

F.2. Softmax Temperature and Calibration

The softmax temperature τ used to compute predictive distributions affects both calibration and the JS-based information gains. Table 4 reports an ablation over $\tau \in \{0.5, 1.0, 2.0\}$ averaged over all backbones and datasets.

A lower temperature ($\tau = 0.5$) produces sharper predictions, which slightly improves top-1 accuracy but leads to worse NLL and ECE due to overconfidence. A higher temperature ($\tau = 2.0$) has the opposite effect: predictions become underconfident, increasing NLL and BIR and slightly reducing accuracy. The intermediate setting $\tau = 1.0$ yields the best overall calibration and BIR, and is therefore used by default.

Table 4. Sensitivity of CIGMA to the softmax temperature τ, averaged over all backbones and datasets.

τ	Acc↑	NLL↓	ECE↓	BIR↓
0.5	95.6	0.089	0.011	0.075
1.0 (default)	95.4	0.081	0.009	0.068
2.0	94.9	0.093	0.010	0.073

F.3. Layer-Wise Distribution of Pruned Heads

CIGMA selects the top-K heads by their impact on background information gain. Table 5 aggregates the distribution of the $K = 16$ pruned heads across layer groups (early, middle, late blocks) and reports the associated changes in BIR and accuracy when only that group is ablated.

Most pruned heads are concentrated in the middle layers, where attention tends to mix object and context features. Ablating only early-layer heads reduces BIR modestly with almost no accuracy drop, suggesting that early background-focused heads are largely redundant. Ablating only middle-layer heads produces the largest BIR reductions but also incurs the largest accuracy drop, reflecting their strong causal influence on the model's behavior. Late-layer heads contribute less to BIR but still provide small additional gains when removed, with minimal impact on accuracy.

Table 5. Layer-wise distribution of pruned heads for $K = 16$ and associated changes in BIR and accuracy when ablating only the heads in each group. Results are averaged over all backbones and datasets.

Layer group	# pruned heads	ΔBIR↓	ΔAcc↓
Early blocks (1–4)	4	−0.015	−0.3
Middle blocks (5–12)	8	−0.035	−0.8
Late blocks (13+)	4	−0.018	−0.2

F.4. Stability of CIGMA Scores

Finally, we assess the stability of CIGMA scores with respect to random seeds, different true-positive subsets, and repeated runs. For each backbone, we compute CIGMA scores and head rankings under three random seeds, each with a different 40% true-positive subset, and measure agreement between the resulting rankings.

Table 6 reports the mean and standard deviation of Spearman rank correlation between runs (computed over all heads), as well as the average standard deviation of the rank positions for the top-16 heads. The correlations are consistently high ($\rho > 0.9$), and the standard deviation of the top-16 ranks is small, indicating that the same heads are repeatedly identified as spurious. The fraction of heads

Table 6. Stability of CIGMA head rankings across three random seeds with different true-positive subsets (40% each).

Backbone	Spearman ρ (mean$\pm$std)	Top-16 rank std	% heads with $\lvert\Delta\text{rank}\rvert > 5$
InternVL2-26B	0.94 ± 0.02	1.3	7.8%
LLaVA-1.6	0.92 ± 0.03	1.6	8.5%
LLaVA-1.5-13B	0.93 ± 0.02	1.4	8.1%

Table 7. Spurious-correlation benchmarks (InternVL2-26B; same protocol). CIGMA yields large average and Worst-Group gains vs. Original/MDP.

Benchmark (Avg / Worst-Group Accuracy)	Original	MDP	**CIGMA**
Waterbirds	87.4 / 68.9	88.4 / 71.1	**91.8 / 86.2**
CelebA (BlondHair)	90.6 / 62.4	91.2 / 64.6	**93.9 / 84.7**

whose rank changes by more than five positions across runs is below 10% for all backbones, further confirming that CIGMA's head scores are robust to sampling noise in the true-positive set.

G. Evaluation on Spurious-Correlation Benchmarks

We evaluate on Waterbirds [18] and CelebA (BlondHair attribute) [15] under the same protocol as Table 1 (frozen backbone; build D from true-positives; prune top-K=16). As shown in Table 7, CIGMA improves both average and Worst-Group accuracy over the Original and the training-free baseline MDP, with especially large Worst-Group gains. Fig. 4a additionally shows a complementary robustness shift on Brain MRI (distinct domain).

H. Limitations and Future Work

Since CIGMA is built on causal analysis, it inherently has certain constraints and a specific scope of application. It operates as a post-hoc method on pretrained, frozen ViT-style models and is scored on a curated true-positive set D. Concrete sensitivity and failure modes are partially characterized via ablations on pruning strength K and true-positive fraction, and qualitatively via pruning-induced focus shifts after head removal. The curated-D assumption and related interpretability constraints present directions for future improvement.

References

[1] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016. 3

[2] Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. DivPrune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9392–9401, 2025. 2, 3, 6, 14

[3] Md Rifat Arefin, Yan Zhang, Aristide Baratin, Francesco Locatello, Irina Rish, Dianbo Liu, and Kenji Kawaguchi. Unsupervised concept discovery mitigates spurious correlations. In *Proceedings of the 41st International Conference on Machine Learning*, pages 1672–1688, 2024. 1, 6, 14

[4] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to GPT-4V? closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 67(12):220101, 2024. 1

[5] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024. 2, 6, 14

[6] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[7] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017. 3, 4

[8] Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Boyang Li, Dacheng Tao, and Steven Hoi. From images to textual prompts: Zero-shot visual question answering with frozen large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10867–10877, 2023. 1

[9] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009. 6, 14

[10] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. CS231n Course Project, Stanford University, 2015. 6, 14

[11] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *arXiv preprint arXiv:2202.07800*, 2022. 3

[12] Yinan Liang, Ziwei Wang, Xiuwei Xu, Jie Zhou, and Jiwen Lu. EfficientLLaVA: Generalizable auto-pruning for large vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9445–9454, 2025. 2, 3, 6, 14

[13] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2, 6, 14

[14] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/, 2024. 2, 6, 14

[15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 8, 19

[16] Andrew Joohun Nam, Henry Conklin, Yukang Yang, Thomas L. Griffiths, Jonathan D. Cohen, and Sarah-Jane Leslie. Causal head gating: A framework for interpreting roles of attention heads in transformers. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 6, 14

[17] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. *Advances in Neural Information Processing Systems*, 34:13937–13949, 2021. 3

[18] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. 8, 19

[19] Zhenwei Shao, Zhou Yu, Meng Wang, and Jun Yu. Prompting large language models with answer heuristics for knowledge-based visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14974–14983, 2023. 1

[20] Xinglong Sun, Barath Lakshmanan, Maying Shen, Shiyi Lan, Jingde Chen, and Jose M Alvarez. MDP: Multidimensional vision model pruning with latency constraint. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20113–20123, 2025. 2, 3, 6, 14

[21] Maya Varma, Jean-Benoit Delbrouck, Zhihong Chen, Akshay Chaudhari, and Curtis Langlotz. RaVL: Discovering and mitigating spurious correlations in fine-tuned vision-language models. *Advances in Neural Information Processing Systems*, 37:82235–82264, 2024. 1, 6, 14

[22] 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*, pages 5797–5808, Florence, Italy, 2019. Association for Computational Linguistics. 3

[23] Cheng Yang, Yang Sui, Jinqi Xiao, Lingyi Huang, Yu Gong, Chendi Li, Jinghua Yan, Yu Bai, Ponnuswamy Sadayappan, Xia Hu, et al. TopV: Compatible token pruning with inference time optimization for fast and low-memory multimodal vision language model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19803–19813, 2025. 2, 6, 14

[24] Xubing Ye, Yukang Gan, Yixiao Ge, Xiao-Ping Zhang, and Yansong Tang. ATP-LLaVA: Adaptive token pruning for

large vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24972–24982, 2025. 2, 6, 14