TOWARDS LVLM-AIDED ALIGNMENT OF TASK-SPECIFIC VISION MODELS

Alexander Koebler[1,*], **Christian Greisinger**[2], **Jan Paulus**[3], **Ingo Thon**[4], **Florian Buettner**[1,5,6]
[1]Goethe University Frankfurt [2]University of Technology Nuremberg
[3]Nuremberg Institute of Technology [4]Siemens AG
[5]German Cancer Research Center (DKFZ) [6]German Cancer Consortium (DKTK)

ABSTRACT

In high-stakes domains, small task-specific models are crucial due to their low computational requirements and the availability of numerous methods to explain their results. However, these explanations often reveal that the models do not align well with human domain knowledge, relying instead on spurious correlations. This might result into brittle behaviour once deployed in the real-world. To address this issue, we introduce a novel and efficient method for aligning small task-specific vision models with human domain knowledge by leveraging the generalization capabilities of a Large Vision Language Model (LVLM). Our LVLM-Aided Visual Alignment (LVLM-VA) method provides a bidirectional interface that translates model behavior into natural language and human class-level instructions into image-level critiques, enabling effective interaction between domain experts and the model. We show that our method improves model performance whilst drastically reducing the need for extensive fine-grained feedback.

1 INTRODUCTION

In an era of increasingly large general-purpose models, reliable small task-specific vision models are still of vital importance. This is especially true in many high-stakes domains where interpretability and trustworthiness demands are rigorous. For these non-functional requirements, current Large Vision Language Models (LVLMs) fall short (Guan et al., 2024; Yang et al., 2021). However, ensuring the continuous reliability of small task-specific models also remains a challenge (Decker et al., 2023). In this work, we introduce a synergetic approach leveraging the benefits of both paradigms.

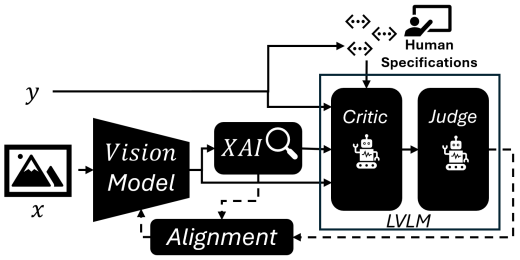

Figure 1: LVLM-Aided Visual Alignment (LVLM-VA) of a small task-specific vision model with human knowledge, using Explanable AI (XAI) in conjunction with a Large Vision Language Model (LVLM) Critic & Judge pair. The knowledge is induced into the system via human defined class-level specifications contained in prompts.

Spurious correlations in the training data set can cause a model to learn shortcuts, resulting in brittle behaviour when used in the real world (Lapuschkin et al., 2019; Rueckel et al., 2020). One way to tackle this issue and increase the reliability of models is to explicitly incorporate human domain knowledge into the model training pipeline (Von Rueden et al., 2021). Recent works have addressed this challenge by fine-tuning the model with human critique based on explanations of the current model behaviour (Teso & Kersting, 2019; Ross et al., 2017). Thereby, these methods improve the alignment of the model with human reasoning. However, they often require extensive fine-grained feedback for each image (Schramowski et al., 2020). Furthermore, explanations of the current model behaviour and feedback on potential errors must be provided directly in the image space (Ross et al., 2017). This results in inefficient and non-human-centered interaction with the model. Gu et al. (2024) introduce

*Work done while at Siemens AG. Correspondence: <alexander.koebler@gmx.de>

an approach to use an LVLM to provide explanations of the model's decision in natural language. They do not consider the natural language interface to inject human feedback back into the model. However, the bidirectional process of adapting an ML model is not only important to consider for incorporating human knowledge and values but also for increasing user trust (Shen et al., 2024). In this work, we propose a novel LVLM-aided approach to align small task-specific vision models with human domain knowledge. The LVLM acts as a translator in both directions: First, it transforms explanations of current model behaviour from image space into natural language, highlighting spurious correlations. Second, it translates human domain knowledge about the vision task, represented in natural language, into instance-wise critiques in image space. Thus, the LVLM provides a more intuitive interface for domain experts to actively steer the model and critically evaluate its reasoning. We show that our approach can drastically reduce the amount of fine-grained feedback required for debugging vision models while effectively increasing their performance.

2 Detecting Spurious Correlations

In the initial step, a combination of Explainable AI (XAI) and an LVLM is used to steer the behaviour of the vision model f and reduce spurious correlations. This combination generates an instance-wise corrective signal based on class-level human specifications. First, explanation maps $M(f, x)$ in image space are generated on a set of alignment samples x that may deviate from the training distribution. The explanations represent a proxy of the current model behaviour to the LVLM. This step is agnostic to the specific XAI approach. Thus, a variety of methods can be considered that highlight the model's attention in the image (Lundberg & Lee, 2017; Binder et al., 2016; Ribeiro et al., 2016; Zhou et al., 2016). Since Yang et al. (2023) show that pre-segmentation of images improves the performance of LVLMs on vision tasks, we introduce a subsequent segmentation step. Note that, as we later want to identify which part of the explanation should be adapted, we segment the explanation map as shown in Figure 2 instead of the original input image. Subsequently, the segmented explanation map C, together with the original image x and the ground truth label y, are provided to the LVLM-based Critic g. To facilitate g in detecting if the vision model f relies on spurious features, it is instructed to utilize a chain-of-thought process. The introduced prompt guides the model to examine the original image and identify which areas belong to the ground truth class y, determine for each segmented cluster which parts of the original image are included, combine both insights, and describe if a cluster covers a relevant region, and lastly provide a verdict whether a cluster is relevant based on the previous insights. To further steer the LVLM in this process and emphasize what important concepts define a particular class, class-specific prompts include human-defined descriptions about how to accurately recognize the class. Those prompts are selected on the basis of the label of the alignment sample. As these descriptions allow scaling class-level human feedback to instance-wise critique, they drastically decrease human effort for aligning the model. To subsequently utilize the LVLM assessment and reduce task complexity, an LLM Judge h, which can be the same model as g, is used to derive a final binary verdict R for x from the free-form output of g. It classifies whether a cluster represents a spurious feature yielding a single binary value R_j per cluster j in C. The class-independent prompt to the judge h is further used to steer the final verdict by including example pairs of Critic assessments and corresponding binary human verdicts. Besides aligning the final verdict with human knowledge, literature shows that such few-shot examples can also drastically increase the performance of an LLM on specified tasks (Brown et al., 2020). We refer to the Appendix for more details about the prompts as well as example outputs.

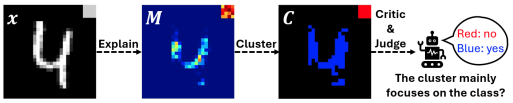

Figure 2: Verdict generation by Critic & Judge pair based on clustered explanations for an MLP trained on digit classification with artificial decoys. The example image shows a confounder in the top right corner.

3 LVLM-Aided Visual Alignment (LVLM-VA)

Different previous works have focused on correcting model explanations (Ross et al., 2017; Slany et al., 2022) by aligning them with fine-grained human feedback (Schramowski et al., 2020) in the form of instance-wise corrections in image space. In our novel LVLM-VA approach, we utilize the

Right for the Right Reasons (RRR) loss function introduced by Ross et al. (2017) for the alignment:

$$L(\theta, X, y, A) = \sum_{n=1}^{N}\sum_{k=1}^{K} -y_{nk}\log(\hat{y}_{nk}) + \lambda_1 \sum_{n=1}^{N}\sum_{d=1}^{D}\left(A_{nd}\frac{\partial}{\partial x_{nd}}\sum_{k=1}^{K}\log(\hat{y}_{nk})\right)^2 + \lambda_2\sum_i \theta_i^2$$

Here, N is the number of used alignment samples, K refers to the number of classes, and D is the dimensionality of the input x. The first term "right answers" corresponds to the cross-entropy loss, optimizing the model to make correct classification predictions. The second term "right reasons" ensures that the model's decisions are based on relevant features by reducing the gradient in areas deemed irrelevant by experts via a binary mask A, steering the model to focus on important features and avoid spurious correlations. Additionally, an optional term "regularization" on the model parameters θ can be added to prevent overfitting. We automatically transfer the binary verdicts generated via the Critic & Judge pair into the correctory maps A:

$$A = \sum_{j=1}^{J} R_j \cdot \mathbf{1}\left[C = j\right] \ ,$$

where the cluster verdict R_j is applied to the corresponding cluster j in the segmented explanation map C such that A only features clusters considered to be spurious. By this, we render the previously required tedious per-instance interaction to generate the expert maps obsolete.

4 Experiments

Experimental Setup We evaluate our work on a digit classification dataset with artificial decoys, which is used in literature to study model debugging (Ross et al., 2017). Each image in the dataset contains a grey patch in a random corner. While the shade of grey for the samples in the training set depends on the digit k ($255 - 25 \cdot k$), it is chosen randomly in the test set. With this, these patches represent simple shortcut candidates for the model in the training set, but harmful confounders in the test set. This is also represented by an accuracy on the training set of 99% and an accuracy on the test set of 54% for the initially trained Multi-Layer-Perceptron (MLP). For debugging, we draw $B = 256$ samples x_a from the alignment set, which in turn is a distinct subset drawn from the test distribution and thus not subject to the spurious correlations contained in the training data. As a sampling strategy, we have drawn the B samples where the original model f has the highest output entropy. We then use DeepLiftSHAP (Lundberg & Lee, 2017) to generate the explanation maps $M(x_a, f)$. They are segmented into clusters using a Gaussian mixture model (Dempster et al., 1977). The segmented explanations C, in combination with the input image x_a and the ground truth label y_a, are fed to the Critic, which is represented by a GPT-4 Vision model. The final verdict is generated by a GPT-4 model (Achiam et al., 2023). For the alignment step using the RRR loss, the alignment samples x_a are mixed with the training samples x_s in each batch of size I with a ratio of $\frac{I_{x_a}}{I_{x_s}} = \frac{1}{8}$. Given that an epoch for N_{Train} training samples consists of $\frac{N_{Train}}{I_{x_s}}$ train iterations which is usually greater than $\frac{B}{I_{x_a}}$, the alignment samples are oversampled. With this procedure, we aim to avoid catastrophic forgetting of previously learned correct features while steering the model towards neglecting spurious correlations.
Throughout our evaluation, we assess the benefit of the different components of LVLM-VA. Thus, we report results when excluding the "right answer" term ($\lambda_1 = 0$) and solely providing (x_a, y_a)-pairs during fine-tuning. Additionally, we evaluate the effect of removing the introduced segmentation step. For this purpose, we directly use image-level verdicts to generate the correctory maps A as

$$A = R \cdot \mathbb{I}\left(M(x, f), c\right) \qquad \text{with } \mathbb{I}(M, c) = \begin{cases} 1 & \text{if } M \geq c \\ 0 & \text{if } M < c \end{cases} \ ,$$

where the optimal binarization threshold c is determined to be 0.68. As baselines for our method, we use the intuitive lower and upper bounds for human involvement given by not tuning the model at all and using instance-wise human-generated expert masks A, respectively. We refer to the Appendix for more details to the experimental setup.

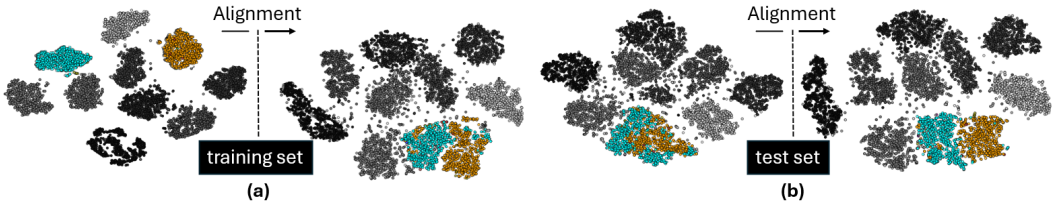

Figure 3: t-SNE plot of the embeddings of the MLP before and after the alignment step using LVLM-VA.

Results As a prerequisite, we initially evaluate the alignment of the verdicts generated by the Critic&Judge pair with human generated ground truth assessments. When directly asking if the shown explanation map $M(x_a, f)$ includes any spurious features, we observe a significant increase in the accuracy of the verdicts from 59% to 78% when providing examples to the Judge via few-shot learning. Our proposed segmentation of the explanations achieves another considerable improvement. It simplifies the task for the Critic into only answering if the specific cluster includes spurious features. With this approach, a verdict accuracy of 89% can be reached. Figure 3 highlights that using the generated verdicts in the alignment reduces the model's reliance on simple shortcuts. This leads to a less distinct clustering of the embeddings on the training set but better class discrimination on the test set. In order to also quantitatively evaluate the increased alignment of the model and the success of the debugging step, we introduce an alignment metric adapted from (Kohlbrenner et al., 2020; Koebler et al., 2024) between the ground truth masks $A^{(GT)}$, i.e., the artificial decoys in the corner, and the explanation masks after alignment $M(x, f)$ for N samples as

$$\mu_{Align} = 1 - \frac{\sum_{n=1}^{N} \sum_{d=1}^{D} A_{n,d}^{(GT)} M_d(x_n, f)}{\sum_{n=1}^{N} \sum_{d=1}^{D} A_{n,d}^{(GT)}} .$$

As shown in Table 1, our novel LVLM-VA approach can drastically increase the model's performance on the test set, clearly outperforming the baseline solely relying on semantic labels. Further, our approach is on-par with utilizing fine-grained instance-wise human feedback whilst drastically reducing the required human effort. The approach without segmentation presents some challenges, which are reflected in the lower accuracy and alignment. In some cases, the Critic struggles to correctly interpret the different degrees of importance from the complex color-shaded explanation map. Furthermore, in many cases, the model f might focus on spurious and non-spurious features, in which case reducing to a single verdict is insufficient.

Setup	Accuracy	Alignment
Initial Model	0.5449	0.5605
Only Labels ($\lambda_1 = 0$)	0.7544	0.6076
LVLM-VA (no clustering)	0.9744	0.9805
LVLM-VA	**0.9806**	**0.9967**
Human Verdicts	0.9833	0.9999

Table 1: Comparison of LVLM-VA with the initial model, the model fine-tuned without "right reason" term, and the model aligned with human defined masks.

We conducted an ablation study on budget size B, to further investigate the need for fine-grained human involvement required to provide ground truth labels. Figure 4 shows that even significantly smaller budget sizes greatly improve model performance. In addition, segmenting the explanations provides increased benefits in the low-budget regime.

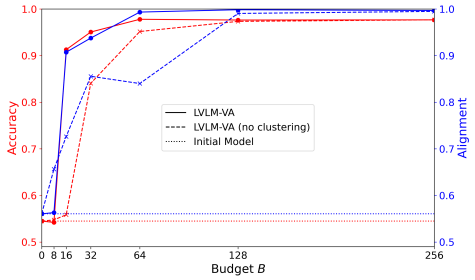

Figure 4: Change of the test accuracy and alignment for LVLM-VA using different budgets B with and without clustering of the explanation maps.

5 Conclusion

We proposed LVLM-Aided Visual Alignment (LVLM-VA) as a novel approach to correct spurious correlations and thus increase the performance of small task-specific vision models. LVLM-VA translates model behavior into natural language and incorporates human class-level descriptions via

instance-wise critique into the model. By this, we provide an efficient human-centered interface to align the model with domain knowledge without the need for extensive fine-grained feedback. An interesting avenue for future research is to scale the method to high-dimensional datasets, potentially using concept-based explanation methods to further increase the fidelity of the alignment between the human and the ML model.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *Artificial Neural Networks and Machine Learning–ICANN 2016: 25th International Conference on Artificial Neural Networks, Barcelona, Spain, September 6-9, 2016, Proceedings, Part II 25*, pp. 63–71. Springer, 2016.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Thomas Decker, Ralf Gross, Alexander Koebler, Michael Lebacher, Ronald Schnitzer, and Stefan H Weber. The thousand faces of explainable ai along the machine learning life cycle: industrial reality and current state of research. In *International Conference on Human-Computer Interaction*, pp. 184–208. Springer, 2023.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1): 1–22, 1977.

Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Ming Tang, and Jinqiao Wang. Anomalygpt: Detecting industrial anomalies using large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1932–1940, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14375–14385, June 2024.

Alexander Koebler, Christian Greisinger, Jan Paulus, Ingo Thon, and Florian Buettner. Through the eyes of the expert: Aligning human and machine attention for industrial ai. In *Artificial Intelligence in HCI: 5th International Conference, AI-HCI 2024, Held as Part of the 26th HCI International Conference, HCII 2024, Washington, DC, USA, June 29–July 4, 2024, Proceedings, Part II*, pp. 407–423, Berlin, Heidelberg, 2024. Springer-Verlag. ISBN 978-3-031-60613-7. doi: 10.1007/978-3-031-60611-3_28. URL https://doi.org/10.1007/978-3-031-60611-3_28.

Maximilian Kohlbrenner, Alexander Bauer, Shinichi Nakajima, Alexander Binder, Wojciech Samek, and Sebastian Lapuschkin. Towards best practice in explaining neural network decisions with lrp. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2020.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch, 2020.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications*, 10, 03 2019. doi: 10.1038/s41467-019-08987-4.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: training differentiable models by constraining their explanations. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pp. 2662–2670. AAAI Press, 2017. ISBN 9780999241103.

Johannes Rueckel, Lena Trappmann, Balthasar Schachtner, Philipp Wesp, Boj Hoppe, Nicola Fink, Jens Ricke, julien Dinkel, Michael Ingrisch, and Bastian Sabel. Impact of confounding thoracic tubes and pleural dehiscence extent on artificial intelligence pneumothorax detection in chest radiographs. *Investigative Radiology*, Publish Ahead of Print, 07 2020. doi: 10.1097/RLI.0000000000000707.

Patrick Schramowski, Wolfgang Stammer, Stefano Teso, Anna Brugger, Franziska Herbert, Xiaoting Shao, Hans-Georg Luigs, Anne-Katrin Mahlein, and Kristian Kersting. Making deep neural networks right for the right scientific reasons by interacting with their explanations. *Nature Machine Intelligence*, 2(8):476–486, 2020.

Hua Shen, Tiffany Knearem, Reshmi Ghosh, Kenan Alkiek, Kundan Krishna, Yachuan Liu, Ziqiao Ma, Savvas Petridis, Yi-Hao Peng, Li Qiwei, et al. Towards bidirectional human-ai alignment: A systematic review for clarifications, framework, and future directions. *arXiv preprint arXiv:2406.09264*, 2024.

Emanuel Slany, Yannik Ott, Stephan Scheele, Jan Paulus, and Ute Schmid. Caipi in practice: towards explainable interactive medical image classification. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pp. 389–400. Springer, 2022.

Stefano Teso and Kristian Kersting. Explanatory interactive machine learning. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, pp. 239–245, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450363242. doi: 10.1145/3306618.3314293. URL https://doi.org/10.1145/3306618.3314293.

Laura Von Rueden, Sebastian Mayer, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Raoul Heese, Birgit Kirsch, Julius Pfrommer, Annika Pick, Rajkumar Ramamurthy, et al. Informed machine learning–a taxonomy and survey of integrating prior knowledge into learning systems. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):614–633, 2021.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

Xu Yang, Hanwang Zhang, Guojun Qi, and Jianfei Cai. Causal attention for vision-language tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9847–9857, 2021.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

A Appendix

The following sections provide further details about the training procedure and the used prompts.

A.1 Training Procedure

In this section we would like to provide further details on the training procedure of the used two layer Multi-Layer Perceptron (MLP) with a width of 256.

Initial Model In the initial phase, the model is trained for a maximum of 100 epochs with early stopping using an Adam optimizer, a learning rate of 0.001, and a batch size of 64.

Alignment The explanations for the current behaviour of the model f are generated using DeepLiftSHAP (Lundberg & Lee, 2017) implemented by (Kokhlikyan et al., 2020). To remove noise from the explanations and produce a more coherent clustering, a threshold for the model's attention is applied to the explanation maps. They are then clustered using a Gaussian Mixture Model implemented in (Pedregosa et al., 2011). In the fine-tuning step using the RRR loss, the learning rate is reduced to $10e^{-4}$ and the training time is extended to 800 epochs. Every batch is augmented by 8 samples from the budget B yielding a new batch size of $64 + 8 = 72$. The samples in the budget B are sampled from a distinct alignment set of size 2000 split from the original validation set to prevent leakage.

A.2 Prompting the Critic & Judge

In this section we document the prompts, including few-shot examples, used for the Critic and Judge in Figure 5 and Figure 6 respectively. We further provide the class-level descriptions in Table 2 used to steer the model's alignment and finally we present a detailed example output of the models in Figure 7.

Class	Description
0	A closed, continuous loop with no starting or ending point, representing a circle or oval shape.
1	A single, straight vertical line, typically with a small base or serif at the bottom.
2	A curved line starting from the top, forming an open loop to the right, and then descending in a diagonal line toward the left.
3	Two small, open, curved loops stacked vertically, each curving to the right, connected in the middle.
4	A vertical line with an angled horizontal line starting from its midpoint, and a diagonal line connecting the top of the vertical line to the bottom of the horizontal line.
5	A horizontal line at the top connected to a vertical line descending downward, which then curves sharply to the left and forms an open loop.
6	A vertical line starting from the top, curving downward to the left, and forming a closed loop at the bottom.
7	A horizontal line at the top connected to a diagonal line that descends toward the left, with no curves or loops.
8	Two distinct loops one on the top and one on the bottom connected in the middle.
9	A small loop at the top with a vertical line descending downward from the loop's right side.

Table 2: A summary of all classes and the corresponding descriptions used in the Critic prompt.

Critic Prompt w/ Clustering

```
"The first image is the original input image of class {str(label)}, which
can be remembered as {class_description}. The second image is a modified
version of the visualization map from class {str(label)} in the original
image. This visualization map consist of {str(num_clusters)} clusters
with the colors {cluster_colors}, where each cluster describes an area of
focus from the original image. First, examine the original image to
identify which parts belong to class {str(label)}. Then, look at the
second image to see the {str(num_clusters)} clusters for
class {str(label)}. For each cluster {cluster_colors}, describe the area
where the model focuses to predict class {str(label)}. Determine whether
each cluster is within the boundaries of the class {str(label)}. A
cluster supports the correct prediction only if it focuses on an area
clearly within the class {str(label)}. If a cluster is outside the class
structure, clearly state that this cluster does not support the correct
prediction. Do not provide introductory sentences."
```

Critic Prompt w/o Clustering

```
"The first image is the original input image of class {str(label)}, which
can be remembered as {class_description}. The second image is a
visualization map of class {str(label)} in the original image. The
visualization colormap ranges from blue (indicating minimal focus) to red
(indicating strong focus), with intermediate colors (yellow/orange)
indicating moderate attention. First, examine the original image to
identify which parts belong to class {str(label)}. Then, you look at the
second image to see the overall visualization map for class {str(label)}.
Describe the most concentrated regions where the model focuses to predict
the class {str(label)}. Determine whether each of the most concentrated
regions is within the boundaries of the class {str(label)}. A
concentrated region supports the correct prediction only if it focuses on
an area clearly within the class {str(label)}. If a concentrated region
is outside the class structure, clearly state that this region does not
support the correct prediction. Do not provide introductory sentences."
```

Figure 5: Prompts for the Crtic model to generate free-form judgments about the model's reliance on spurious features. The prompts are provided with the ground truth label, a class-level description, and the number of clusters if applicable.

Judge Prompt w/ Clustering

```
You have the task to translate the responses of a large vision language model (LVLM) into readable JSON
format for further processing. The task of the LVLM was, for each of the two clusters in the second
image, to identify whether the focus aligns with any part of the class in the first image or not. First,
you read through the LVLM response. Then you identify for each of the two clusters (red and blue)
whether the focus was on the class or not. Then you construct a valid JSON with "color" and "verdict" as
keys. The "color" key should strictly be either "red" or "blue". The "verdict" key should strictly be
either "yes" or "no". For example, "color: red" and "verdict: yes" means that the red cluster did focus
on the class. "color: blue" and "verdict: no" means that the blue cluster did not focus on the class.
IMPORTANT: Please make sure to only return in valid JSON format, with the "output" key as a list of
JSON. The list should strictly contain TWO elements for each of the clusters in red and blue.
For clarification, here are a few examples:

Example 1 Input: "The original image depicts a pixelated digit 8, with the upper and lower loops of the
digit clearly visible. In the second image, the red cluster is located on the upper loop of the digit 8,
and the blue cluster is located on the lower loop of the digit 8. Both clusters are within the
boundaries of the digit 8, indicating that they both support the correct prediction of the digit."
Example 1 JSON:
{{
  "output": [
      {{
          "color": "red"
          "verdict": "yes"
      }},
      {{
          "color": "blue"
          "verdict": "yes"
      }}
  ]
}}

Example 2 Input: "The original image shows the digit 1, which is composed of a vertical column of pixels
slightly slanted to the right. In the second image, the red cluster is located centrally over the main
vertical body of the digit 1, which supports the correct prediction of the digit. The blue cluster,
however, is located at the bottom left corner of the image, away from the digit 1, and therefore does
not support the correct prediction of the digit."
Example 2 JSON:
{{
  "output": [
      {{
          "color": "red"
          "verdict": "yes"
      }},
      {{
          "color": "blue"
          "verdict": "no"
      }}
  ]
}}
===== END OF EXAMPLES =====
```

Judge Prompt w/o Clustering

```
You have the task to translate the responses of a large vision language model (LVLM) into
readable JSON format for further processing.
The task of the LVLM was, for each of the two most concentrated regions in the second image, to
identify whether the focus aligns with any part of the class in the first image or not.
First, you read through the LVLM response. Then you identify for each of the most concentrated
regions whether the focus was on the class or not.
Then you construct a valid JSON with only "verdict" and "reason" as key. The "verdict" key
should strictly be either "yes" or "no". The "reason" key should bring the response of the LVLM
into valid JSON format for parsing without changing its structure.
For example, "verdict: yes" means that all of the most concentrated regions focus on the class.
"verdict: no" means that at least one or more most concentrated region(s) did not focus on the
class.
IMPORTANT: Please make sure to only return in valid JSON format, with the "output" key as a list
of JSON. The list should strictly contain ONE element for all of the most concentrated regions.
For clarification, here are a few examples:

Example 1 Input: "Based on the visualization map, the machine learning model focused on the
bottom left corner of the image, indicated by the high redness in that area, rather than the
digit '4' located in the center."
Example 1 JSON:
{{
    "verdict": "no",
    "reason": "Based on the visualization map, the machine learning model focused on the bottom
left corner of the image, indicated by the high redness in that area, rather than the digit '4'
located in the center."
}}

Example 2 Input: "The machine learning model appears to have focused primarily on the digit '9'
in the center of the image to come up with its prediction. The highest concentration of redness,
which indicates where the model focused the most, is on the central part of the image where the
digit is located. There is some attention on the bottom left corner where the decoy is located,
but it is significantly less than on the digit itself."
Example 2 JSON:
{{
    "verdict": "yes",
    "reason": "The machine learning model appears to have focused primarily on the digit '9' in
the center of the image to come up with its prediction. The highest concentration of redness,
which indicates where the model focused the most, is on the central part of the image where the
digit is located. There is some attention on the bottom left corner where the decoy is located,
but it is significantly less than on the digit itself."
}}
===== END OF EXAMPLES =====
```

Figure 6: Prompts for the Judge model to generate binary verdicts if the model is predominantly focused on areas relevant with respect to the actual class. The model is provided with few-shot examples, including human-defined input-output pairs.

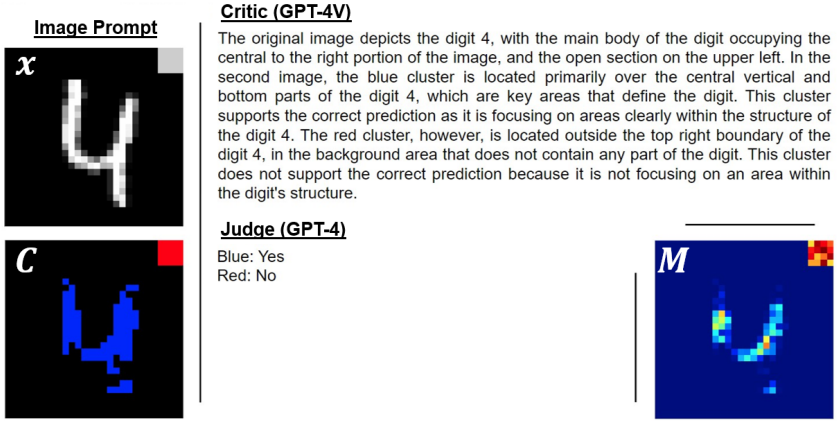

Figure 7: Example output of the Critic & Judge pair for an input where the model f focuses on the confounder in the top right corner. The models correctly identify this failure and assign the correct binary verdict to the corresponding cluster.