EVALUATING GROUPS OF FEATURES VIA CONSISTENCY, CONTIGUITY, AND STABILITY

Chaehyeon Kim, Weiqiu You, Shreya Havaldar & Eric Wong
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104, USA
{chaenyk,weiqiuy,shreyah,exwong}@seas.upenn.edu

ABSTRACT

Feature attributions explain model predictions by assigning importance scores to input features. In high-dimensional data such as images, these scores are often assigned to groups of features. There are various strategies for creating these groups, ranging from simple patches to deep-learning-based algorithms. Which group should be used for explanation? We formally define three key criteria for interpretable groups of features: consistency, contiguity, and stability. We find that patch-based groups outperform groups created via modern segmentation tools. [1]

1 THE IMPORTANCE OF GROUPS FOR EXPLANATIONS

When explaining high-dimensional data, feature attributions often assign importance scores to groups of features rather than individual features. In images, groups are typically collections of neighboring pixels from the same object. For example, the number of groups in an ImageNet image is much smaller than 50k, the number of pixels. By reducing mental overhead, scores for grouped features can be more accessible for humans to understand than scores for individual features.

Existing strategies for creating groups of features range from simple patches to classic or deep-learning-based segmentation algorithms. In this work, we study four techniques for dividing images into groups: classic computer vision approaches (patches (Dosovitskiy et al., 2021) and superpixels (Levner & Zhang, 2007)) as well as modern deep-learning approaches (Segment Anything Model (SAM) (Kirillov et al., 2023) and Archipelago (Tsang et al., 2020)). While groups from different approaches can be quite diverse in shape, size, and content, there is no standard benchmark for comparing groups and measuring their qualities.

To better understand groups for explanations, we propose three criteria that a group should satisfy to improve the quality of feature attributions. These criteria encourage more consistent, contiguous, and stable attributions. In the following analysis, we find evidence that despite advancements in modern segmentation tools, patch-based groups are more suitable for feature attributions.

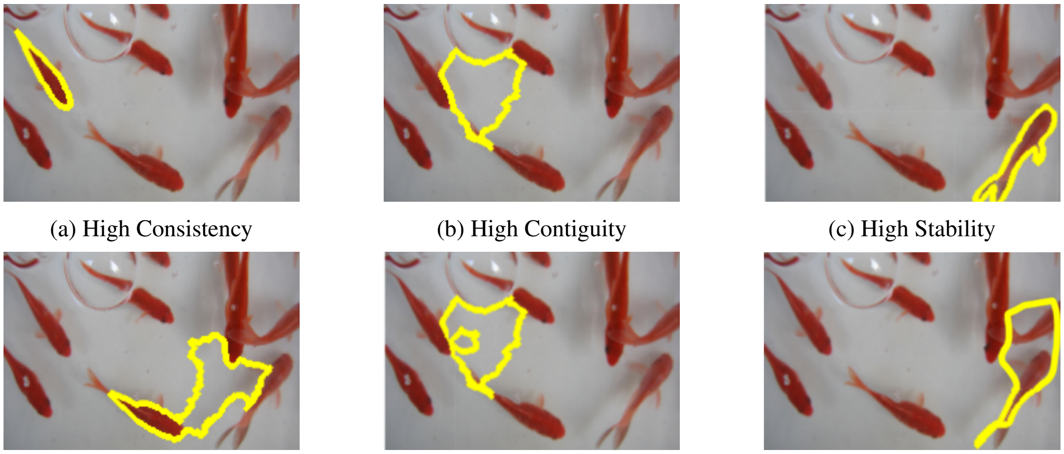
2 DESIDERATA OF GROUPS FOR FEATURE ATTRIBUTIONS

We first discuss the desiderata of groups for feature attributions. We posit that an ideal group should possess the following characteristics: the group should be consistent and contiguous to be human interpretable, and it also needs to be stable to have a reliable feature attribution. In this section, we elaborate on these properties and show the representative examples in Figure 1.

Formally, we define $x \in \mathbb{R}^{n \times p}$ to be an input such as an image, where n is the number of patches and p is patch width × patch height × 3. We also represent a boolean vector group as $\alpha \in \{0, 1\}^n$, where $\alpha_i = 1$ if the ith feature is included in the group, and 0 otherwise.

Consistency. Consistency is the semantic coherency of features within the same group. Highly consistent groups should contain only one element in a group. To calculate the consistency, we use

[1] Code is available at https://github.com/BrachioLab/exlib



(a) High Consistency

(b) High Contiguity

(c) High Stability

(d) Low Consistency

(e) Low Contiguity

(f) Low Stability

Figure 1: Illustrative examples of groups with high and low consistency, contiguity, and stability.

the embedding cosine similarity to measure the closeness or similarity between pairs of features. Formally, let $h : \mathbb{R}^{n\times p} \to \mathbb{R}^k$ be an embedding function of the input and $Z = \|\alpha\|(\|\alpha\| - 1)$, where k is the hidden size. Then, for each pair of i-th and j-th features,

$$\mathrm{Consistency}(x, \alpha) = \frac{1}{Z} \sum_{i,j:i\neq j \wedge a_i a_j = 1} \frac{h(x)_i \cdot h(x)_j}{\|h(x)_i\|\|h(x)_j\|} \tag{1}$$

Contiguity. Group contiguity captures smoothness of a group without unnecessary "holes". Then,

$$\mathrm{Contiguity}(x, \alpha) = \frac{\lambda\|K\alpha\|_{2,1}}{\sqrt{\|\alpha\|_1}} \quad \text{where} \quad K = \begin{bmatrix}(1-\rho)\nabla \\ \rho\mathrm{Id}_p\end{bmatrix} \tag{2}$$

Here, $\nabla \in \mathbb{R}^{2p\times p}$ is the spatial gradient operator, Id_p is the $p \times p$ identity operator, and the parameter ρ balances the sparsity of the input and the spatial gradient.

Stability. Stability captures robustness to change in model output when adding extra features to a group. Formally, let $h : \mathbb{R}^{n\times p} \to \mathbb{R}^k$ be a predictive model. Here, k is the number of classes. Then,

$$\mathrm{Stability}(x, \alpha) = \max_i \|h(x \odot \alpha_{+i}) - h(x \odot \alpha)\| \tag{3}$$

where α_{+i} is the group created by adding the i-th feature to α. Further details of these three properties are discussed in Appendix C.

3 Modern Grouping Methods are No Better than Patches

Finally, we analyze how different grouping methods perform across the three desiderata. Specifically, we evaluate groups using patches, superpixels, SAM, and Archipelago on the ImageNet (Deng et al., 2009) and MS-COCO (Lin et al., 2014) dataset.

We find that modern approaches such as SAM and Archipelago are significantly less reliable. These groups have stability loss as high as 17.74, about ×3 worse than other metrics. Results for all metrics and methods are in Table 1 and 2 of Appendix E. In fact, SAM segments have the lowest performance across all methods, suggesting that they are the least suitable for feature attributions. This may be because most of these segments are degenerated groups.

Surprisingly, simple patches are the most reliable option for grouping since they have the highest performance across all metrics. Further discussions are in Appendix E.

Conclusion. Currently, our results suggest that patches are the more suitable groups over modern approaches. We hope that our desiderata and corresponding analysis can guide future work in developing more reliable groups for explanations and interpretable for humans.

URM Statement

The authors acknowledge that at least one key author of this work meets the URM criteria of ICLR 2024 Tiny Papers Track.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 2331783.

References

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Suesstrunk. Slic superpixels compared to state-of-the-art superpixel methods. In *TPAMI*. 2012.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

Julius Adebayo, Michael Muelly, Hal Abelson, and Been Kim. Post hoc explanations may be ineffective for detecting unknown spurious correlation, 2022.

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey Hinton. Neural additive models: Interpretable machine learning with neural nets. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=wHkKTW2wrmm.

Afra Amini, Tiago Pimentel, Clara Meister, and Ryan Cotterell. Naturalistic causal probing for morpho-syntax. *Transactions of the Association for Computational Linguistics*, 11:384–403, 2022. URL https://api.semanticscholar.org/CorpusID:248811730.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10, 2015. URL https://api.semanticscholar.org/CorpusID:9327892.

Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. volume 39, pp. 2481–2495. 2017. doi: 10.1109/TPAMI.2016.2644615.

David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Mueller. How to explain individual classification decisions, 2009.

Shahaf Bassan and Guy Katz. Towards formal xai: Formally approximate minimal explanations of neural networks. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pp. 187–207. Springer, 2023.

Jasmijn Bastings and Katja Filippova. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2020. doi: 10.18653/v1/2020.blackboxnlp-1.14. URL http://dx.doi.org/10.18653/V1/2020.BLACKBOXNLP-1.14.

Jasmijn Bastings, Sebastian Ebert, Polina Zablotskaia, Anders Sandholm, and Katja Filippova. "will you find these shortcuts?" a protocol for evaluating the faithfulness of input salience methods for text classification. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 976–991, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.64. URL https://aclanthology.org/2022.emnlp-main.64.

Blair Bilodeau, Natasha Jaques, Pang Wei Koh, and Been Kim. Impossibility theorems for feature attribution. *ArXiv*, abs/2212.11870, 2022. URL https://api.semanticscholar.org/CorpusID:254974246.

Nitay Calderon, Eyal Ben-David, Amir Feder, and Roi Reichart. DoCoGen: Domain counterfactual generation for low resource domain adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7727–7746, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.533. URL https://aclanthology.org/2022.acl-long.533.

Mahaman Sani Chaibou, Pierre-Henri Conze, Karim Kalti, Basel Solaiman, and Mohamed Ali Mahjoub. Adaptive strategy for superpixel-based region-growing image segmentation. *Journal of Electronic Imaging*, 26(6):061605–061605, 2017.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, 2017.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. How do decisions emerge across layers in neural models? interpretation with differentiable masking. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020. doi: 10.18653/v1/2020.emnlp-main.262. URL http://dx.doi.org/10.18653/v1/2020.emnlp-main.262.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2009.

Misha Denil, Alban Demiraj, and Nando de Freitas. Extraction of salient sentences from labelled documents. *ArXiv*, abs/1412.6815, 2014. URL https://api.semanticscholar.org/CorpusID:9121062.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4443–4458, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.408. URL https://aclanthology.org/2020.acl-main.408.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Michael Eickenberg, Elvis Dohmatob, Bertrand Thirion, and Gaël Varoquaux. Total variation meets sparsity: statistical learning with segmenting penalties. In *Medical Image Computing and Computer Aided Intervention (MICCAI)*. 2015.

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. In *International Journal of Computer Vision*. 2004.

Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. In *Biological Cybernetics*. 1980.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

Shreya Havaldar, Adam Stein, Eric Wong, and Lyle Ungar. Topex: Topic-based explanations for model comparison, 2023.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. volume 9, pp. 1735–1780. 1997. doi: 10.1162/neco.1997.9.8.1735.

Sara Hooker, D. Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Neural Information Processing Systems*, 2018. URL https://api.semanticscholar.org/CorpusID:202782699.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C. Wallace. Learning to faithfully rationalize by construction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.409. URL http://dx.doi.org/10.18653/v1/2020.acl-main.409.

Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi. Representation of linguistic form and function in recurrent neural networks. *Computational Linguistics*, 43(4):761–780, December 2017. doi: 10.1162/COLI_a_00300. URL https://aclanthology.org/J17-4003.

Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. Learning the difference that makes a difference with counterfactually-augmented data, 2020.

Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un)reliability of saliency methods. *Lecture Notes in Computer Science*, pp. 267–280, 2019. ISSN 1611-3349. doi: 10.1007/978-3-030-28954-6_14. URL http://dx.doi.org/10.1007/978-3-030-28954-6_14.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *arXiv preprint arXiv:2304.02643*. 2023.

Thibault Laugel, Xavier Renard, Marie-Jeanne Lesot, Christophe Marsala, and Marcin Detyniecki. Defining locality for surrogates in post-hoc interpretablity, 2018.

Ilya Levner and Hong Zhang. Classification-driven watershed segmentation. In *IEEE Transactions on Image Processing*. 2007.

Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure, 2017.

Teng Li, Tao Mei, In-So Kweon, and Xian-Sheng Hua. Contextual bag-of-words for visual categorization. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(4):381–392, 2011. doi: 10.1109/TCSVT.2010.2041828.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollar. Microsoft coco: Common objects in context. volume abs/1405.0312. 2014. URL https://arxiv.org/abs/1405.0312.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pp. 150–158, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450314626. doi: 10.1145/2339530.2339556. URL https://doi.org/10.1145/2339530.2339556.

Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pp. 623–631, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450321747. doi: 10.1145/2487575.2487579. URL https://doi.org/10.1145/2487575.2487579.

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.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2016.11.008. URL https://www.sciencedirect.com/science/article/pii/S0031320316303582.

Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. *ACM Comput. Surv.*, 55 (13s), jul 2023. ISSN 0360-0300. doi: 10.1145/3583558. URL https://doi.org/10.1145/3583558.

Peer Neubert and Peter Protzel. Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms. In *ICPR*. 2014.

Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

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

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction, 2021.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018. doi: 10.1609/aaai.v32i1.11491. URL https://ojs.aaai.org/index.php/AAAI/article/view/11491.

Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. A consistent and efficient evaluation strategy for attribution methods. In *International Conference on Machine Learning*, 2022. URL https://api.semanticscholar.org/CorpusID:249642068.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. In *nature*, volume 5, pp. 533–536. 1986.

Madeline Chantry Schiappa, Sachidanand VS, Yunhao Ge, Ondrej Miksik, Yogesh S Rawat, and Vibhav Vineet. Robustness analysis on foundational segmentation models. *arXiv preprint arXiv:2306.09278*, 2023.

Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128:336–359, 2016. URL https://api.semanticscholar.org/CorpusID:15019293.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 3145–3153. JMLR.org, 2017.

Sandipan Sikdar, Parantapa Bhattacharya, and Kieran Heese. Integrated directional gradients: Feature interaction attribution for neural NLP models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 865–878, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.acl-long.71.

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

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

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2014. URL `https://api.semanticscholar.org/CorpusID:12998557`.

David Stutz, Alexander Hermans, and Bastian Leibe. Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166:1–27, 2018.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, 2017. URL `https://api.semanticscholar.org/CorpusID:16747630`.

Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. In *Advances in Neural Information Processing Systems*, 2020.

Bhavan Vasu and Chengjiang Long. Iterative and adaptive sampling with spatial attention for black-box model explanations. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020. doi: 10.1109/wacv45572.2020.9093576. URL `http://dx.doi.org/10.1109/WACV45572.2020.9093576`.

Andrea Vedaldi and Stefano Soatto. Quick shift and kernel methods for mode seeking. In *European Conference on Computer Vision*. 2008.

Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6707–6723, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.523. URL `https://aclanthology.org/2021.acl-long.523`.

Anton Xue, Rajeev Alur, and Eric Wong. Stability guarantees for feature attributions with multiplicative smoothing, 2023.

Weiqiu You, Helen Qu, Marco Gatti, Bhuvnesh Jain, and Eric Wong. Sum-of-parts models: Faithful attributions for groups of features, 2023.

Yilun Zhou and Julie Shah. The solvability of interpretability evaluation metrics. In Andreas Vlachos and Isabelle Augenstein (eds.), *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 2399–2415, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.182. URL `https://aclanthology.org/2023.findings-eacl.182`.

Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie A. Shah. Do feature attribution methods correctly attribute features? *ArXiv*, abs/2104.14403, 2021. URL `https://api.semanticscholar.org/CorpusID:233443847`.

Shiping Zhu, Xi Xia, Qingrong Zhang, and Kamel Belloulata. An image segmentation algorithm in image processing based on threshold segmentation. In *IEEE Conference on Signal-Image Technologies and Internet-Based System*. 2007.

Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1651–1661, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1161. URL `https://aclanthology.org/P19-1161`.

A Appendix

We include additional information detailing the evaluation of existing groups. Appendix B provides a review of studies related to our work, Appendix C discusses about the limitations of existing feature grouping methods and contains a complete description of each metric. Appendix D provides experiment details, and Appendix E shows additional results.

B Related Work

Segmentation. Image segmentation is a crucial task in computer vision and image processing. The primary objective is to divide an image into meaningful groups, which has significant applications in various fields, including medical image analysis. There are several methods available for image segmentation, ranging from simple thresholding (Zhu et al., 2007) to more complex superpixel algorithms such as Felzenswalb's method (Felzenszwalb & Huttenlocher, 2004), which is based on graphs, the Quickshift method (Vedaldi & Soatto, 2008) that approximates kernelized mean-shift, the SLIC method (Achanta et al., 2012) that is based on k-means clustering, and the Watershed method (Neubert & Protzel, 2014) that is based on the grayscale gradient image.

Modern deep learning methods such as CNN (Fukushima, 1980), Recurrent Neural Networks (RNNs) (Rumelhart et al., 1986), and Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) have yielded a new generation of image segmentation models that exhibit remarkable performance improvements. Fully Convolutional Networks (FCNs) (Long et al., 2015) replace the fully connected layers in convolutional networks with convolutional layers, allowing the network to output spatial heatmaps for segmentation. DeconvNet (Noh et al., 2015) conducts semantic segmentation based on transposed convolution using encoder (Goodfellow et al., 2016) - decoder (Badrinarayanan et al., 2017) based models.

Existing grouping methods. There are four commonly used methods of obtaining groups of features: patch, superpixels, segmentation, and Archipelago. Patch is a simple baseline that divides an image into non-overlapping, fixed-size blocks. Superpixels (Levner & Zhang, 2007; Vedaldi & Soatto, 2008; Achanta et al., 2012) are regions of visually similar pixels located adjacent to each other in an image. Segmentation models built with deep learning (Long et al., 2015; He et al., 2018; Kirillov et al., 2023) create groups based on global context and connections within the image. Lastly, Archipelago (Tsang et al., 2020) builds upon superpixel algorithms and assigns interaction scores to pairs of superpixels based on how they affect the model's predictions when perturbed, in order to create larger groups.

Though commonly used to group features, each of these methods presents drawbacks in the context of interpretability. Two key sets of drawbacks relate to the *semantics* and the *geometry* of created groups. We propose the following sets of semantic and geometric properties as a way to evaluate group interpretability.

Feature attribution. Feature attribution methods include post-hoc and built-in attributions. For post-hoc methods, gradient-based attributions include using gradients directly (Selvaraju et al., 2016; Baehrens et al., 2009; Simonyan et al., 2014; Bastings & Filippova, 2020), gradient × inputs (Sundararajan et al., 2017; Denil et al., 2014; Smilkov et al., 2017) and through propagation (Ribeiro et al., 2018; Springenberg et al., 2014; Bach et al., 2015; Shrikumar et al., 2017; Montavon et al., 2017). Other methods create a surrogate model to approximate the original model (Ribeiro et al., 2016; Lundberg & Lee, 2017; Laugel et al., 2018), or perturb inputs and observe the change in predictions (Petsiuk et al., 2018; Vasu & Long, 2020; Kaushik et al., 2020; Li et al., 2017; Kádár et al., 2017; Ribeiro et al., 2018; De Cao et al., 2020) and use manual (Kaushik et al., 2020) or automatic (Calderon et al., 2022; Zmigrod et al., 2019; Amini et al., 2022; Wu et al., 2021) counterfactual perturbation. While the above methods focus on individual features, Tsang et al. (2020); Sikdar et al. (2021) investigates feature interactions. Another line of work has built-in feature attributions including training different modules for each feature (Lou et al., 2012; 2013; Agarwal et al., 2021), or pre-selecting one or multiple groups of features to use for prediction (Jain et al., 2020; You et al., 2023).

Feature groups in images. Dividing an image into groups of pixels, or superpixels, is a long-established area of research from computer vision. A range of algorithms use geometric, spatial, and color properties (Levner & Zhang, 2007; Vedaldi & Soatto, 2008; Achanta et al., 2012) to find clusters of similar pixels (see Stutz et al. (2018) for a survey). A crude but simple approach is to divide into a uniform grid of patches (Li et al., 2011), which has recently surged in popularity for their efficiency in transformer-based architectures (Dosovitskiy et al., 2021). On the other hand, image segmentation aims to create larger groups of pixels that correspond to entire objects (Felzenszwalb & Huttenlocher, 2004; Zhu et al., 2007; Ronneberger et al., 2015; Long et al., 2015; Chen et al., 2017; He et al., 2018; Ranftl et al., 2021; Kirillov et al., 2023). The resulting groups can be more semantically meaningful than superpixels, but the performance and quality of the segments depends highly on the type of object being segmented Schiappa et al. (2023) and can require a significant amount of labeled training data. Our work aims to get more semantic groups than superpixels but without the labeled data required of segmentation models, and can be viewed as a region-growing algorithm (Chaibou et al., 2017) that uses the embedding space of a foundation model to guide the growth direction.

Groups for explanations. Feature attribution methods try to explain how different input features affect the model prediction, and they often attribute to groups of features. For images, some works assign scores to patches (Ribeiro et al., 2016; Lundberg & Lee, 2017; You et al., 2023) or segments (Tsang et al., 2020; You et al., 2023). Other works attribute to each pixel using gradients (Selvaraju et al., 2016; Baehrens et al., 2009; Simonyan et al., 2014; Bastings & Filippova, 2020), or by randomly sampling patch masks and interpolating between them (Petsiuk et al., 2018) or

Some feature attribution methods assign scores to a higher level of groups of features. Archipelago (Tsang et al., 2020) post-hoc selects groups by merging pairs of interacting features. FRESH (Jain et al., 2020) generates a group of features using top 20% tokens from an attention mechanism in a pre-trained Transformer. SOP (You et al., 2023)generates groups of features with an external attention trained end-to-end with a pre-trained model. IDG (Sikdar et al., 2021) selects groups based on a hierarchical structure. TopEx (Havaldar et al., 2023) uses topics as groups for explanations.

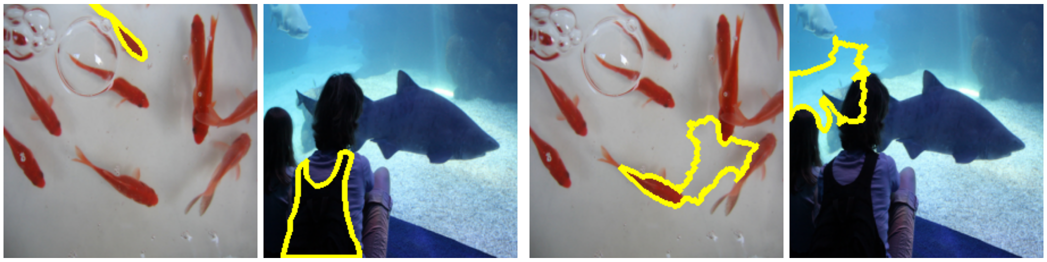
Attribution evaluation. Various metrics for evaluating feature attribution methods are studied (Nauta et al., 2023; Zhou et al., 2021; Adebayo et al., 2018; Hooker et al., 2018; DeYoung et al., 2020; Bastings et al., 2022; Rong et al., 2022; Zhou & Shah, 2023; Adebayo et al., 2022). Some perturbation metrics include insertion and deletion for images (Petsiuk et al., 2018) and comprehensiveness and sufficiency for text (DeYoung et al., 2020). The performance is also evaluated for models with built-in explanations to ensure no considerable degradation from the original model (Jain et al., 2020; You et al., 2023).

There are some debates about what constitutes good explanations. Multiple works have shown the failures of feature attributions (Bilodeau et al., 2022; Sundararajan et al., 2017; Adebayo et al., 2018; Kindermans et al., 2019). Different properties are proposed as important axioms that feature attributions should satisfy, including stability (Xue et al., 2023), minimality (Bassan & Katz, 2023), and faithfulness (You et al., 2023). However, there is still a lacking for evaluating features groups used for explanations.

C METRICS FOR GROUPS

Many explanation techniques today rely on feature groups as intermediate features for high-dimensional image data. Underlying this technique is the implicit assumption that such intermediate features are interpretable. In this section, we investigate the quality of existing feature groups for images with respect to properties important for interpretability.

This section describes in detail how we measure group consistency, contiguity, and stability. As mentioned in the main paper, $x \in \mathbb{R}^{n \times p}$ is input such as an image, and $\alpha \in \{0,1\}^n$ indicates group membership of each feature, where $\alpha_i = 1$ if the ith feature is included in the group, and 0 otherwise.



(a) High (b) High (c) Low (d) Low

Figure 2: Illustrative examples of high and low consistency. Groups with high consistency, equivalent to having high semantic similarity within a group, only depict one cohesive element: an orange fish for (a) and clothing for (b). However, groups with low consistency contain several unrelated elements: the fish and fragmented bowl in (c) and the tank with the girl's hair in (d).

C.1 Semantic Properties

For groups to be interpretable, they must make sense semantically. Grouping methods such as patch (Dosovitskiy et al., 2021), superpixels (Levner & Zhang, 2007) and Archipelago (Tsang et al., 2020) pose a challenge for human interpretation due to the unintuitive shapes of the generated groups. These groups often break apart elements or include too many dissimilar elements in one group. We formalize these semantic properties as *consistency* and *stability*.

Consistency. The consistency of a group is measured by how many unrelated elements a group contains. Impure groups lack semantic integrity, making it difficult for humans to interpret them. An ideal group contains only a single element, making the group clear and understandable to humans. Existing methods produce groups with low consistency. In Figure 4, we observe many groups that contain both people and stingrays together, despite the fact that these are two separate elements and should thus be in different groups. Also, Figure 2 illustrates examples of high and low similarity groups. Examples of high similarity only contain one object (e.g. fish, clothing), while examples of low similarity contain several unrelated objects (e.g. fish and bowl, head and background). Given that high-similarity groups are more cohesive compared to the low-similarity ones, we posit that an ideal group should have high similarity to be interpretable.

To measure consistency, we use the cosine similarity of embeddings of features within the group. More specifically, we divide the group into patches, compute the embedding similarity between each pair of patches in the group, and take the average for all the pairs in the group. We calculate the group consistency as follows:

$$\forall i, j \text{ such that } i \neq j \quad \text{Consistency}(x, \alpha) = \frac{1}{\|\alpha\|(\|\alpha\| - 1)} \sum_i \sum_j \frac{h(x)_i \cdot h(x)_j}{\|h(x)_i\| \|h(x)_j\|} \tag{4}$$

where $h : \mathbb{R}^{n \times p} \to \mathbb{R}^k$ be an embedding function of the input. To be comparable with other metrics, the consistency is then projected to consistency loss, which ranges from 0 to 1, with 0 as the best value and 1 as the worst:

$$\text{ConsistencyLoss}(x, \alpha) = 1 - \frac{\text{Consistency}(x, \alpha) + 1}{2} \tag{5}$$

When there is only one patch that contains anything in the group, we discard the group when computing the average consistency for a grouping method. This is because a group with only one patch would have the perfect consistency loss of 0, but using the singleton groups will bias the consistency metric to favor smaller groups. Lower consistency loss is more desired than higher consistency loss.

An alternative way to measure consistency when it has object labels (such as the MS-COCO dataset), we can directly measure the supervised version of consistency of each group by calculating its entropy based on the percentage of objects vs background within the group as follows:

$$\text{P}(\alpha, \alpha^*) = \frac{(\alpha^*)^\intercal \alpha}{\alpha^\intercal \mathbf{1}} \tag{6}$$

where α^* is the ground truth object and α is the predicted group. Then,

$$\begin{aligned}\mathrm{SupervisedConsistency}(\alpha, \alpha^*) = -(&P(\alpha, \alpha^*) \log P(\alpha, \alpha^*) \\ &+ (1 - P(\alpha, \alpha^*)) \log(1 - P(\alpha, \alpha^*)))\end{aligned} \tag{7}$$

If the predicted group only consist of object, the supervised consistency value is 0. Otherwise, if the predicted group is consist of 50% object and 50% background, which is not desirable the supervised consistency value is 1. Therefore the supervised consistency also ranges from 0 to 1, with 0 as the best and 1 as the worst.

Stability. Stability measures the robustness to change in model output when adding extra features to a group. Stable groups then then to contain all needed information for an element and keep elements intact. An ideal grouping method should not break apart an element into multiple different groups. Groups created by patch (Dosovitskiy et al., 2021), superpixels (Levner & Zhang, 2007), and Archipelago (Tsang et al., 2020) often segment a single element into multiple parts, resulting in unintuitive and fragmented groups. For instance, in Figure 4, we see that the stingray in the bottom right is segmented into multiple groups that divide the stingray in unintuitive ways, which can result in lack of stability. To ensure the stability of a feature group, the addition of more features should not result in a change of predicted class once the group has been selected. To expand the group's feature set, we consider all possible patches with features yet to be chosen by the group. We calculate the stability loss for adding each additional patch by computing the difference in predicted probabilities. For our experiments, we used 16×16 patches to be consistent with ViT. Then,

$$\mathrm{StabilityLoss}(x, \alpha) = \max_i \|h(x \odot \alpha_{+i}) - h(x \odot \alpha)\| \tag{8}$$

where $h : \mathbb{R}^{n \times p} \to \mathbb{R}^k$ be an predictive model that predicts for k classes, and α_{+i} is the group created by adding the i-th patch to α. The stability loss ranges from 0 to 1 with 0 being the best and 1 the worst, which matches the direction of other metrics.

A supervised way to quantify stability is to count the number of segments one element is divided into; in other words, how many fragments are in one element. Therefore, each group should have only one concept therefore maximizing the purity. Through our analysis, we observed two semantic consistency patterns: backgrounds and objects. Objects become more stable over time, but adding a new concept to them can result in a instability, and the change of stability when it changes from object to background. On the other hand, background starts as low instability, but including a new concept can destabilize them.

C.2 Geometric Properties

For groups to be easily understood by humans, their shapes need to be visually interpretable. However, groups generated by segmentation methods such as SAM (Kirillov et al., 2023) vary too much in sizes, making it difficult to discern patterns within or draw meaning from the smaller groups. Additionally, SAM does not always partition the entire image, leaving many features not included in any group. We formalize these desired geometric properties as *contiguity*.

Contiguity. The contiguity of a group depends on the shape. When forming a group, it is important to ensure that it is smoothly shaped and has no unnecessary holes. Groups with very low contiguity (i.e., extremely small, weirdly shaped) are not interpretable to humans. For instance, SAM (Kirillov et al., 2023) segments may seem reasonable at first glance (see Figure 4), but these segments have large variations in sizes; in other words, there is a small number of high-quality groups but a large number of low-quality groups. The vast majority of segments generated by SAM are too small to interpret. Furthermore, there are some groups with unnecessary holes. The size distribution of groups generated with SAM for all 1000 ImageNet classes is plotted in Figure 3, and we see that many groups generated by SAM have extremely small sizes.

To measure contiguity, we repurpose the segmentation penalty *sparse variation* (Eickenberg et al., 2015). The sparse variation penalty enforces contiguous zones in the group, and lower sparse variation values can guarantee greater contiguity within a group. We used the normalized form of *sparse variation* to make it independent from the group size. We formalize the group contiguity loss as

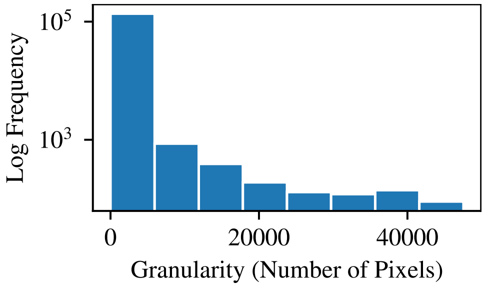


Figure 3: Distribution of granularity scores for ImageNet groups segmented by SAM. Low-granularity groups are hard to interpret, and 75% of groups are smaller than 3 pixels.

follows:

$$\text{ContiguityLoss}(x, \alpha) = \frac{\lambda \|K\alpha\|_{2,1}}{\sqrt{\|\alpha\|_1}} \quad \text{where} \quad K = \begin{bmatrix} (1-\rho)\nabla \\ \rho \text{Id}_p \end{bmatrix} \tag{9}$$

where ∇ is the spatial gradient operator. For a 2D mask of size $p = H \times W$, where H is the height and W is the width of the image. We have $\nabla \in \mathbb{R}^{2p \times p}$. The matrix Id_p is the identity operator of size $p \times p$. Finally, ρ is the parameter controlling the trade-off between coordinate sparsity and spatial regularity. We take $\rho = 0.5$ in our calculations. Lower contiguity loss is more desired than higher contiguity loss, as it indicates that there are fewer holes in the group.

D Experiment Details

D.1 Datasets

We use the ImageNet dataset (Deng et al., 2009), which consists of 1000 classes of human-annotated photographs, and the MS-COCO dataset (Lin et al., 2014), a large-scale dataset designed for object detection, segmentation, and image captioning tasks.

D.2 Model

We use a pre-trained Vision Transformer (ViT) model (Dosovitskiy et al., 2021) pre-trained on ImageNet-21k (14 million images, 21,843 classes) at resolution 224x224 and fine-tuned on ImageNet 2012 (1 million images, 1,000 classes) at resolution 224x224 [2].

D.3 Grouping Methods

Here, we describe the four grouping methods we examine.

- Patches (Dosovitskiy et al., 2021) divide an image into fixed-size blocks without overlapping regions.
- Superpixels interpret pixel values as local topography and fill areas from initial points until adjacent areas meet at delineated edges. We use watershed segmentation (Levner & Zhang, 2007) to obtain the superpixels.
- Segment Anything Model (SAM) (Kirillov et al., 2023) creates groups based on global context and relationships within the image using transformer-based architecture.
- Archipelago (Tsang et al., 2020) perturbs different segments and assigns interaction scores to pairs of segments by their influences on the model's prediction.

[2] https://huggingface.co/google/vit-base-patch16-224

	Stability Loss	Consistency Loss	Contiguity Loss
Patch	**0.43 ± 0.002**	**0.20 ± 0.03**	**0.05 ± 0.00**
Superpixel	0.49 ± 0.01	0.22 ± 0.03	0.06 ± 0.00
SAM	0.53 ± 0.01	0.26 ± 0.02	0.10 ± 0.00
Archipelago	0.50 ± 0.02	0.22 ± 0.02	0.07 ± 0.00

Table 1: Experimental results on ImageNet. We measure the mean values of Stability, Consistency , and Contiguity loss. Patches perform the highest on all group interpretability metrics.

	Sup Stability	Sup Consistency	Stability	Consistency	Contiguity
Patch	5.87 ± 2.39	**0.12 ± 0.17**	**0.50 ± 0.01**	**0.50 ± 0.04**	**0.00 ± 0.00**
Superpixel	6.80 ± 4.43	0.16 ± 0.21	0.53 ± 0.02	**0.50 ± 0.3**	0.05 ± 0.00
SAM	17.74 ± 6.19	0.13 ± 0.18	0.64 ± 0.03	**0.50 ± 0.03**	0.07 ± 0.00
Archipelago	**5.79 ± 4.23**	**0.12 ± 0.17**	0.63 ± 0.01	**0.50 ± 0.02**	0.07 ± 0.01

Table 2: Experimental results on MS-COCO. We measure the mean values of Stability, Consistency , and Constiguity loss. Given that MS-COCO contains ground truth annotations for objects in an image, we also calculate the supervised version of Stability and Consistency loss.

E Additional Results

We calculate quantitative metrics on the ImageNet dataset in Table 1, and MS-COCO dataset in Table 2. To match the number of groups, we take the average number of groups of SAM and archipelago for each example, usually 64, and apply it to the patch and superpixel methods. We then measure quantitative metrics for each group and take the average value for each image. For the consistency loss and stability loss, we use patches of size 16×16.

E.1 Best and Worse Cases

To compare the results of each method, we consider the best and worst-case scenarios for each metric. The best cases for all three metrics are all 0, and the worst is all 1. Regarding consistency and stability loss, the worst-case scenario may be calculated using another image. The lower bound for consistency might be calculated by using exactly the same embedding, and for stability, it should be calculated by having all the pixels in the image. The lower bound for the best possible sparse variation in square shape can be achieved by taking the square root of the number of pixels. On the other hand, the upper bound can be obtained by taking a square like a checkerboard. For instance, if a group of k pixels exists, the best possible sparse variation would be taking the square with one side as the square root of k. However, the worst possibility would be taking the square with one side as the square root of $2k$; in such a case, if you choose a pixel to the group, avoid selecting the pixel adjacent to it, just like in a checkerboard.

E.2 Results and Discussions

Consistency. We plot the distribution of dissimilarities of groups generated with all methods in Figure 5. In the consistency loss plot for merging 1000 ImageNet classes in Figure 5, all methods displayed a distribution similar to a skewed normal distribution with slightly different mean and variance. SAM and Archipelago showed longer tails compared to patches and superpixels. For the MS-COCO dataset, all methods displayed a similar distribution. Table 1 also indicates that the average consistency loss is similar across different methods. Both Figure 5 and Table 1 demonstrate that different grouping methods exhibit similar embedding similarities on average, and patches showed the best performance for both datasets.

Contiguity. Histograms in Figure 6 show that patches and superpixels consistently have low contiguity loss, while SAM consistently has high contiguity loss. Archipelago has a dichotomy of majority low contiguity loss with a smaller peak for high contiguity loss. We can also see from

Table 1 and 2 that SAM segments have higher average contiguity loss than the other three methods and that patches have lower maximum contiguity loss than the other three methods.

The fact that patches have the best average and maximum contiguity loss is likely because they do not have many edges. Conversely, SAM segments are the worst contiguous for both the average and the worst case, which is due to how it selects semantic segments that are diverse in shape.

Stability. Histograms in Figure 7 show diverse distributions of stability for all four methods. Patches skew to 0, while SAM segments skew toward 1. Table 1 also shows that patches have much smaller stability loss than the other three types of groups. Since the value of 0 represents the best-case scenario, patches are the most stable, while SAM segments are the least.

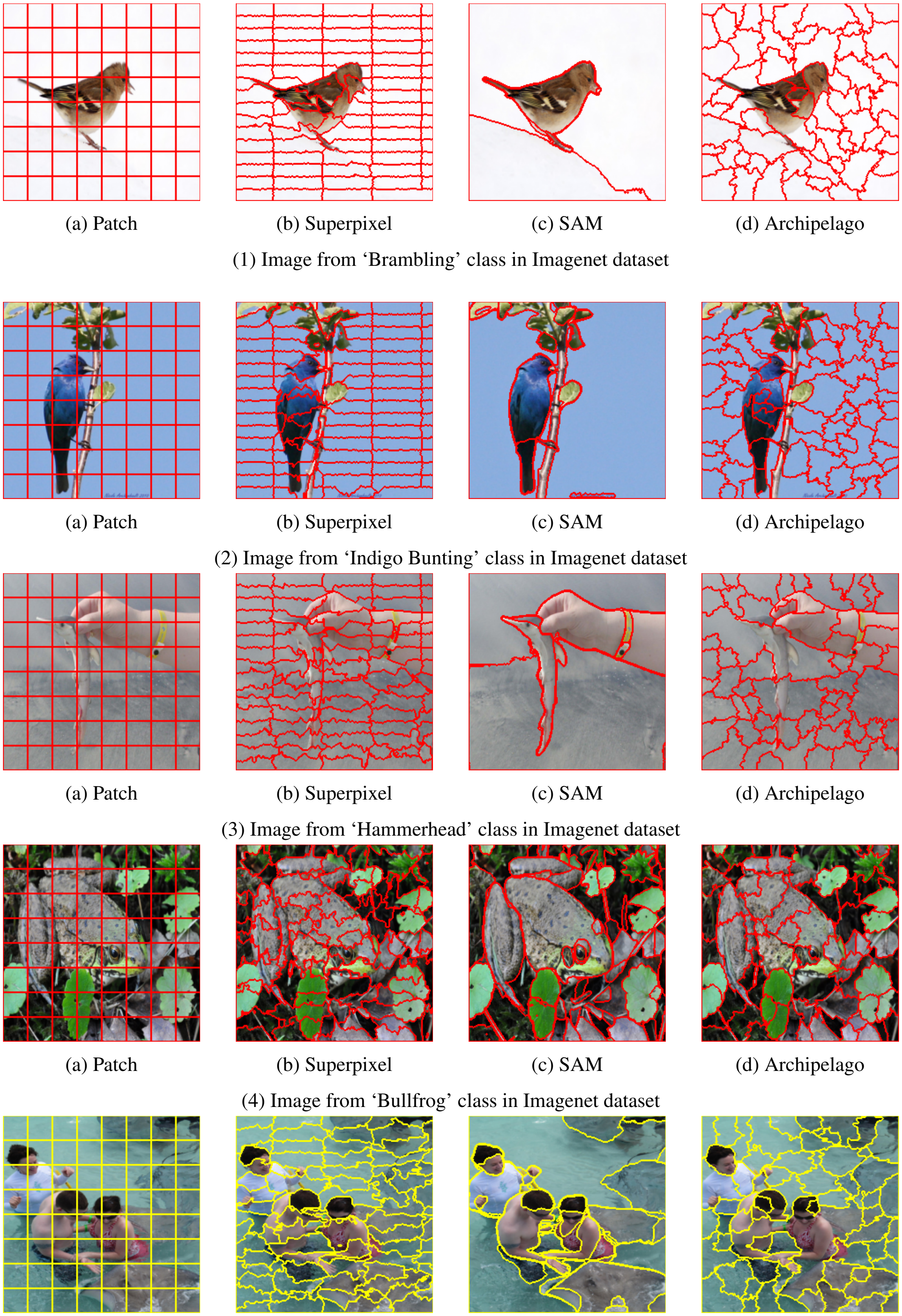
This could be attributed to the fact that the groups in SAM had varying sizes. As shown in Figure 4, some groups found by SAM are small and thin, having little contextual information. Therefore, adding one patch to a small group might result in a higher stability loss. Conversely, the patch groups have uniform sizes, allowing each group to have some contextual information.

In the case where we have object labels (such as the MS-COCO dataset), we can directly count the number of fragments each labeled object is divided into. As shown in Table 2, Archipelago and Patch groups have the low average fragmentation, indicating that these methods separate key elements into different groups.

In the case without object labels, we can measure fragmentation by computing the instability of each group via the equation presented in Figure 8. A low instability value means that adding one patch does not significantly alter the model prediction, indicating that the group has one clear meaning (and, therefore, less fragmentation). Fig 7 illustrates the average instability for the existing methods. Patches have much lower instability, and thus, less fragmentation than groups from other methods on both the ImageNet and MS-COCO datasets. Based on these results, we can conclude that patches are quantitatively less fragmented than existing groups, making it easier for humans to interpret.

Summary The statistics in Table 1 and 2 and Figure 5, 6, and 7 show that the patches have the minimum loss on average and, in the worst case, for all the properties. This indicates that the patches are the best type of group based on these desiderata.

Also, SAM and Archipelago have worse average stability and contiguity loss. Although SAM and Archipelago's Transformer-based architectures enable them to comprehend global context effectively, they are not well-suited for grouping when measured with the three desired quantitative metrics.



(a) Patch (b) Superpixel (c) SAM (d) Archipelago

(1) Image from 'Brambling' class in Imagenet dataset

(a) Patch (b) Superpixel (c) SAM (d) Archipelago

(2) Image from 'Indigo Bunting' class in Imagenet dataset

(a) Patch (b) Superpixel (c) SAM (d) Archipelago

(3) Image from 'Hammerhead' class in Imagenet dataset

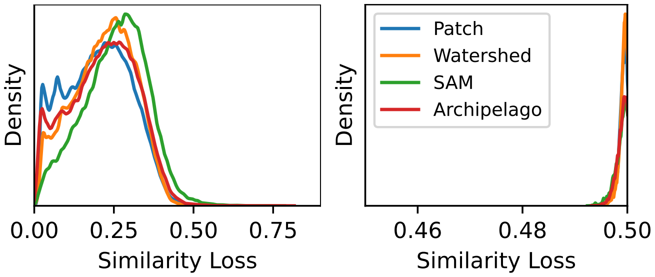
(a) Patch (b) Superpixel (c) SAM (d) Archipelago

(4) Image from 'Bullfrog' class in Imagenet dataset

(a) Patch (b) Superpixel (c) SAM (d) Archipelago

(5) Image from MS-COCO datset

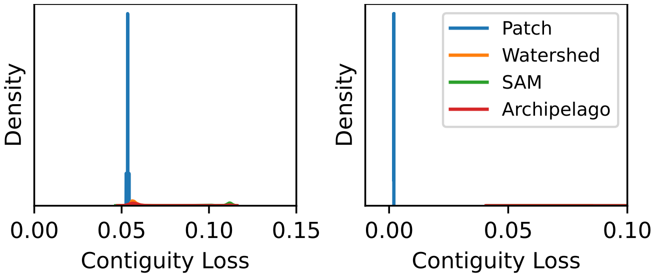
Figure 4: Examples of groups created by commonly used feature grouping methods: Grid, Watershed superpixels, SAM, and Archipelago.



(a) ImageNet

(b) MS-COCO

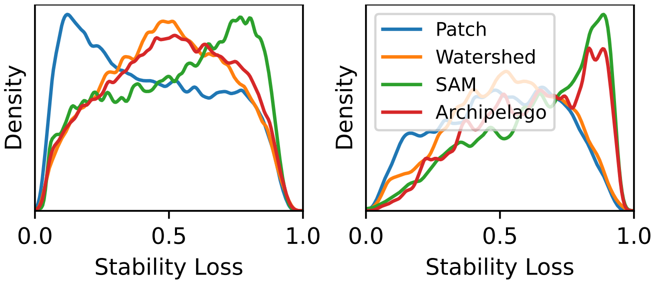
Figure 5: Similarity loss for the ImageNet and MS-COCO datasets. Calculated using semantic similarity between features within a group, similarity loss measures group purity. Patches have the lowest similarity loss, and therefore highest purity, of all groups (for both datasets, the blue line skews to the left compared to the others).



(a) ImageNet

(b) MS-COCO

Figure 6: Contiguity loss for the ImageNet and MS-COCO datasets. Calculated using the sparse variation, contiguity loss measures how smooth the group is. Patches have the lowest consistency loss, which indicate the shape of patch groups are smooth, and therefore easily interpretable to humans.



(a) ImageNet

(b) MS-COCO

Figure 7: Stability loss for the ImageNet and MS-COCO datasets. We calculate how many groups each element is fragmented into. High stability loss indicates high fragmentation, which means that the same element is divided into multiple groups. The desired group for human interpretation should contain the whole element.