nature portfolio

Peer Review File

STAIG: Spatial Transcriptomics Analysis via Image-Aided Graph Contrastive Learning for Domain Exploration and Alignment-Free Integration

Corresponding Author: Professor Kenta Nakai

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Attachments originally included by the reviewers as part of their assessment can be found at the end of this file.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
Key results
In this study, the authors employ a graph contrastive learning algorithm (STAIG) with the help of H&E images to generate embedded representations of spatial spots across multiple ST datasets. These embeddings are applicable to spatial clustering and batch integration. The visual feature embeddings from H&E images, not directly incorporated into the final embeddings, are mainly used to generate a probability matrix for augmenting the graph of each dataset. The authors conducted a series of experiments using various ST datasets to show the STAIG's superiority in discerning domains in tissues including human and mouse brains, human breast cancers, and zebrafish melanoma. They also attempt to demonstrate STAIG's exceling in integrate ST datasets derived from both vertical and horizontally alignable tissue slices. Finally, authors claims that STAIG can identify tumor microenvironments with elevated expressions of tumor-related marker genes.

Validity
I have some concerns about the validity of the results:
• Using STAIG's example codes, we fail to obtain the spatial clustering results as good as those claimed in the manuscript: For the DLPFC 151673 dataset (Fig.2c), the ARI for the spatial clustering results is 0.61, which is significantly lower than the reported 0.68 and the competing method GraphST (0.64).
• STAIG is essentially a method for single dataset rather than for multiple datasets, foundationally not very different from existing methods like SpaGCN. This can be told from the calculations of the block diagonal adjacency matrix A in equation (1) and the edge weight matrix W in equation (3). That is, spots from different datasets never interact with each other, with no information exchanged among them. Basically, the model should behave same as the case where the model is trained using the datasets one by one. Indeed, in our attempts to repeat the results for integrating horizontal slices, which have larger batch effects, using the provided example codes, we failed to reproduce the results in Fig.5c. Additionally, the UMAPs indicate suboptimal performance of the integration. As the two batches are well-separated while spots of same domains are dispersed.
• In Fig.3, authors claim the image features generated by BYOL can reflect the DLPFC stratifications. These results seem to be too good given the limitations of BYOL under this scenario. Therefore, the validity of Fig.3 needs further clarifications. Indeed, in our experiment using their codes to repeat Fig.3b, we failed to obtain a result as good as the one shown in the manuscript.

Significance
The contributions of this study include:
• Introduce a novel way of generating augmented ST graph for contrastive learning, based on H&E images.
• The endeavor to identify TME from tissue regions labeled as healthy.
Despite these contributions, this study has limited significances in terms of its novelty in both targeted problems and the methodology:
• The work focuses on generating spot embeddings for spatial clustering and batch alignment tasks, both of which have

been extensively studied in recent years. To name a few, STAGATE1, GraphST2, STAligner3, SpaGCN4 and conST5.
• Although the authors claim STAIG is alignment-free, they do not consider batch effects among datasets but simply putting them together, which makes STAIG no different from methods specifically designed for single dataset (e.g., SpaGCN). So STAIG should not be considered as a method for handling multiple samples.
• From the perspective methodological framework, it is very similar to conST in aspects including visual feature extraction and the idea of using contrastive graph learning.
• STAIG can hardly be recognized as a multimodal learning algorithm as the visual features are only used for graph augmentation rather than being incorporated into the final embeddings.
• STAIG is very slow compared to other similar methods like SpaGCN and STAGATE, as the clustering of a single dataset can take up to 4 hours.

Data and methodology
The data used in this study is public available, I have no particular concern about it. For the methodology, I have the following concerns:
• The authors use BYOL to extract cell morphology features from H&E image. It may be unnecessary or even worsen the extracted features, as it is well-documented that a simple U-Net can effectively handle this task6. Authors need to justify their choice here.
• BYOL needs data augmentation. Unlike general images (e.g., ImageNet), the data augmentation techniques may not be applicable here, as histology images typically lack clear semantic segments (i.e., kind of fuzzy and blurred), potentially leading to significant biases. Authors need to clarify the usefulness of the data augmentation techniques.
• The parameters of the band-pass filter are not explicitly stated in the paper, which is a potential pitfall, given the different qualities between tissue slices or even within the same slice. An arbitrary setting may severely deteriorate the image (e.g., over-blurring of the image). Authors need to explain the standards for choosing these parameters.
• The normalization of the aggregated raw gene expression matrix X on line 296 needs further explanation. This normalization potentially downweighs the importance of datasets with low sequencing depth, which yet may hold important information.
• The slice integration seems unnecessary, as the way A constructed in equation (1) and W constructed in equation (3) means spots in different datasets would never interact with each other. So, this slice integration step is meaningless. Please comment on this.
• The authors used the DS algorithm, which essentially a K-means, to exclude negative samples as those within the same cluster. This step needs further clarification as the cluster size generated by K-means can have a large variance. This means many spots can end up in a single cluster and many cluster can have only one spot.
• Please clarify the default value for the probability matrix (i.e., without labels) on line 328.
• In equation (7), what are the parameters for the Bernoulli sampling?

Analytical approach
I have the following concerns:
• The impacts of batch effects on slice integration should be analyzed more thoroughly, given the batch effects among vertical DLPFC slices are minor.
• The efficacy of image feature extraction needs more detailed analysis. For example, which part of BYOL is most critical for this effective feature extraction. Ablation studies or comparison with another computer vision representation learning framework may help on this point.
• The choice of the significance threshold log2FC for DE analysis should be discussed.
• The statistical tests for the GO analysis (BH corrected test?) should be stated.
• In addition to ARI or SC, the results should also show one more metric (e.g., NMI).
• In addition to the UMAPs, some metrics should be used to measure the batch mixing effects (e.g., BatchKL).
• In the paragraph (line 206-208), the authors claim differential expressions of muscle and tumor marker genes in the two clusters of slice A and B. However, it is really difficult to tell such differences from Fig.7f.
• The study of TME in Fig.7 should include a competing method TESLA7 specifically designed for detecting TME using ST data.
Suggested improvements
The suggested improvements are:
• The authors should add an experiment for discussing the use of different methods (e.g., kernels) other than Euclidean distance for constructing the image distance matrix D on line 311.
• The authors can include an additional experiment for testing datasets without H&E images.
• The choice of masked ratio in equation (7) should be discussed, as the quality of the final embeddings hinges on this choice.
• The ability of STAIG in integrating non-vertical and non-horizontal slices (e.g., two slices of the same tissue but from different individuals in different studies, and cannot be horizontally stitched) should be analyzed.

Clarity and context
• On line 145, the “(Fig.4f and Supplementary Fig. S7)” appears twice in a row.
• On line 197, the traditional ST methods should be explicitly named.
• In Fig.7f legend, it should be fine-grained instead of “fine-tuned”.
References
• The H&E image patch segmentation idea is also very similar to HistToGene8, so this paper should be added to reference.
• Another ST computational methods, TESLA7, for studying TME should be referred and compared.

Reports References
1. Dong, K. & Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. Nat Commun 13, 1739 (2022).
2. Long, Y. et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. Nat Commun 14, 1155 (2023).
3. Zhou, X., Dong, K. & Zhang, S. Integrating spatial transcriptomics data across different conditions, technologies and developmental stages. Nature Computational Science 3, 894-906 (2023).
4. Hu, J. et al. SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. Nat Methods 18, 1342-1351 (2021).
5. Zong, Y. et al. conST: an interpretable multi-modal contrastive learning framework for spatial transcriptomics. bioRxiv, 2022.2001. 2014.476408 (2022).
6. Falk, T. et al. U-Net: deep learning for cell counting, detection, and morphometry. Nature methods 16, 67-70 (2019).
7. Hu, J. et al. Deciphering tumor ecosystems at super resolution from spatial transcriptomics with TESLA. Cell systems 14, 404-417. e404 (2023).
8. Pang, M., Su, K. & Li, M. Leveraging information in spatial transcriptomics to predict super-resolution gene expression from histology images in tumors. bioRxiv, 2021.2011. 2028.470212 (2021).

(Remarks on code availability)
Results in Fig.2c, Fig.3b and Fig.5c are not quite reproducible

Reviewer #2

(Remarks to the Author)
The authors presented STAIG, a sophisticated graph-based deep learning model dedicated to the representation learning and spatial domain identification of spatial transcriptomics data. The model integrates gene expression, spatial coordinates, and histological images using graph-contrastive learning and advanced feature extraction. While the benchmarks delineated in the paper appear to be relatively thorough and the results yielded particularly promising, there remain certain concerns regarding detailed model design and evaluation that merit further clarification.

Major concerns

1. According to the method description, the overall spatial graph in STAIG is simply composed of disjoint subgraphs corresponding to individual slices, without any inter-slice connectivity. Consequently, all positive and negative pairs in contrastive learning come from the same slice only. Thus, it is not particularly clear why STAIG is able to achieve batch integration across slices. The examples showcased in the manuscript comprises are confined to slices from the same experiment, where batch effect is presumably minimal. Would STAIG work well with more substantial batch effects, e.g., those that may arise from different ST technologies?
2. STAIG employs a band-pass filter to mitigate noise and uneven staining in the histological images. Is it necessary to adjust the band-pass filter when applied to different datasets? Furthermore, how much does this filter step contribute to the overall performance?
3. It was mentioned in the method section that 512×512 pixel patches were extracted when partitioning the H&E images. Does that correspond to the physical dimension of the ST spots? How would STAIG adapt to ST data and H&E images with alternative spatial resolutions?
4. When no image is available, STAIG conducts graph augmentation by removing edges at a fixed probability. Would it improve performance if the removal probabilities were determined based on transcriptome similarity between the spots?
5. The hyperparameter optimization was conducted on the DLPFC dataset only, which raises questions regarding its generalizability. Does the optimal hyperparameter setting hold in other datasets, or when imaging data is not available?
6. It was mentioned in the method section that the clustering results in DLPFC were additionally smoothed for the STAIG method. It is not particularly clear whether the same procedure was applied for other benchmarked methods, which could have noticeable impact on the evaluation metrics. Based on the visualization in Fig. 2e, it seems that the clusters are spatially scattered for stLearn and Seurat, suggesting that they might not have been smoothed?
7. In the "image feature extraction" section, the authors claimed that STAIG clusterings based on image features highly conforms with manually annotated regions. However, such conformity is not visually apparent in Fig. 3. E.g., the demarcation between WM and layer 6 doesn't seem clear (Fig. 3a). It would be helpful if the authors highlight the mentioned demarcation lines in each sub-panel of Fig. 3 to ease comparison across panels.
8. The authors identified a tumor-like interface and a muscle-like interface in zebrafish melanoma. As the spots in the ST slices correspond to multiple cells, is this partition due to different cell type proportions per spot or genuine difference in single-cell gene expression?
9. Most use cases in the manuscript were based on spatial barcoding technologies, which offer relatively high numbers of detected genes. Would STAIG work well for imaging-based technologies as well, e.g., MERFISH, seqFISH?

Minor concerns

1. It was mentioned in the introduction section that "recent graph-based approaches for ST data are flawed" due to "artificially structured graphs". This is a rather bold claim, and worth more detailed explanation as to why the graphs can be "artificially structured"?

2. The training details of BYOL is not sufficiently explained. Was the BYOL model trained on a per-dataset basis, or pre-trained on a larger corpus of histological images?
3. Canonical BYOL uses the online network as the final network, but line 288 states that the "target network" was used after BYOL training. Is that a typo?
4. ARI or SC scores are not shown for Fig. S11, despite being shown for all other relevant figures.
5. Fig. 2F and S3 differ by a 90 degree rotation, which is unnecessary.
6. The superscript notations for augmented graphs are not consistent. Most occurrences contained no parentheses, but line 347–352 used parentheses.
7. The figure numbers in Table S1 are incorrect. E.g. there is no Fig. 3f (last row in Table S1) in the manuscript.
8. Brightness in Fig. 6e bottom panels are not properly normalized.

(Remarks on code availability)
The authors have released `STAIG` as a Github repo, and provided necessary instructions on environment management and usage. The codes seem well structured and the tutorial runs without error, except that some comments and print messages are written in Chinese rather than English.

Reviewer #3

(Remarks to the Author)
The authors propose a novel approach, STAIG, to extract features from multi-modal spatial transcriptomics data. The proposed approach shows state-of-the-art clustering scores in various spatial transcriptomics datasets.

The introduction quickly cites several competing approaches while providing minor information, being a bit distracting. It could be improved by providing additional details and how they differ from the proposed approach.

The methods and results are clearly described. Figures with small legends are hard to read, even digitally, due to the rasterization of text.

Comments:

Line 276: What are the parameters of the Gaussian filter and the band-pass filter? The authors suggest these filtering steps are very important. A study showing how they impact the algorithm's performance would be extremely helpful for researchers building upon this work and could be included in the ablation studies section.

Line 314: Why is the log plus one transform applied to equation (3) if it’s being transformed by e^x on equation (4)? And why is it necessary to do the element-wise product with A in equation (3) if equation (4) is conditioned to A_ij == 1?

A scale with numbers in Figures 4f, 6e, and 7f would improve the evaluation of the different gene expressions in the different regions.

Figure 6e: It’s unclear how appropriate it is to apply a statistical test to compare the mean value of groups that, by definition, have different averages, since they represent a mixture of Gaussians. Moreover, some of the test assumptions are violated. For example, the cells are not independent because they have a spatial correlation. I suggest removing the p-values.

(Remarks on code availability)

Creating a Python package and additional documentation could improve the code repository.

I did not run the code to confirm its reproducibility.

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
Thanks for the authors’ responses on the points I raised. After reading the rebuttal letter, I have the following concerns:
Major
1. Questionable reproducibility and local performance of STAIG
I appreciate the authors' provision of a pre-configured cloud server, which facilitates the verification of STAIG's performance. However, I remain concerned about its reproducibility and performance consistency across different environments. Our tests on a local server demonstrated significant discrepancies compared to the results reported in the manuscript (refer to the attached results in the reviewer's response email). This variation might stem from inherent limitations in STAIG’s methodology, particularly the probabilistic perturbation in the graph's local structure during contrastive learning, which results in highly variable augmented graphs G1 and G2 in each training instance. To mitigate this, I suggest resampling G1 and G2 multiple times and using their average structure, although this may increase training complexity and time. I recommend other reviewers also test STAIG on their local machines to verify STAIG’s reproducibility at the user end.

2. Theoretically suspicious batch correction
The approach to contrasting learning described in Equation 9 permits interactions across batches but may not effectively mitigate batch effects. Specifically, in equation 9, positive samples are all from the same batch, while samples from different batches only serve as negative samples in the denominator. Theoretically, batch effects can help minimize this loss function —its loss declines when the dissimilarity of such negative samples are increased, which the existence of batch effects can contribute to. In other words, rather than distancing cross-batch negative samples, batch effects are effectively addressed only when pulling closer cross-batch positive samples, which STAIG fails to achieve, at least theoretically.

3. Insufficient task and methodological novelty
The novelty of STAIG is still questionable from both task-oriented and methodological perspectives. The tasks of spatial clustering and slice alignment are well-trodden areas, while the ideas of applying graph convolutional networks, histology image integration, and graph contrastive learning to spatial clustering have been previously explored in other studies. Moreover, the techniques used in STAIG, such as BYOL, local graph structure perturbation, GCN, are common in the field of deep learning.

4. Overly ad hoc parameter settings
The adjustments of critical parameters without detailed justification, such as the increase in the number of selected highly variable genes from 3000 to 5000 and the alteration in edge removal probabilities (from 0.3 to 0.2), raise concerns about STAIG's adaptability to diverse datasets. Other ad hoc parameters include the p_min, p_max on line 409, the Bernoulli masking ratio on line 427, the number of clusters in DS filtering on line 457, and many other hyperparameters in network architecture, image processing and loss functions. While a few ad hoc parameters are generally acceptable, too many such parameters undermine the model's robustness and generalizability.

5. Overly casual clustering in DS filtering
Given the number of clusters in the DS filtering is key to the determination of negative samples and subsequent graph contrastive learning, its setting necessitates further investigation. The authors should provide original images of the clustering results to allow the interpretation of labeling samples in the same cluster as positives while in distinct clusters as negatives. Meanwhile, given the batch effects among H&E images of different slices, it is very likely that image patches of different tissue domain types from same slice can end up in the same cluster, while those of the same tissue type from different slices in different clusters. Such clustering results are not desirable as they are inconsistent with the images' semantic meaning.

6. Insufficient mathematical solidity
STAIG falls short of mathematically solidity, as reflected not only by excessive ad hoc parameter settings but also the lack of succinct mathematics and efficient computations. For example, the use of a logarithmic operation in Equation 3 followed by an exponential in Equation 4 seems unnecessary and only incurs extra computational costs. This could be streamlined by skipping equation 3 and directly employing D^{img}+1 in Equation 4. Additionally, the computation of the distance matrix D^{img} could be optimized by leveraging the sparsity of the adjacency matrix, rather than using the computationally intensive function like "euclidean_distances = cdist(node_emb, node_emb, metric='euclidean')" on line 106 in adata_preprocessing.py. Not only this brutal-force calculation is not inefficient, but also potentially leads to memory issues in case of large-scale data and high-dimensional embeddings.

Minor:
None

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)
I appreciate the authors' extensive efforts in conducting additional benchmarks with new data and new methods, and revising the manuscript to integrate the new results, which have improved the rigor of the manuscript and largely addressed my previous concerns. However, I still have some remaining questions and concerns that require further clarification:

- The authors explained cross-slice interaction in the STAIG model by referring to negative samples in the contrastive loss. If I understand it correctly, the negative samples include not only pre-existing edges that get removed based on image feature similarity, but also edges that do not exist in the original adjacency graph, such as those across slices. If that was the case, the second term in the denominator of Eq. 9 involving a sum over all such negative samples would be highly inefficient to compute. I suspect that some sort of negative sampling strategy was actually employed instead of computing the exhaustive sum, which was not clearly reflected in Eq. 9. Furthermore, the normalization of negative samples does not seem to align with that of positive samples in Eq. 9. Additionally, I am also unclear why f^1,2(i, i) is included in the denominator; should this not be a positive edge? Lastly, I still find it counterintuitive why batch effect correction works in STAIG, which obviously still lacks positive samples across slices. "Emphasizing on subtle local dynamics over broad global changes" is posited as an explanation, but remains a fuzzy intuition. Could ablation experiments help clarify the underlying mechanism of STAIG's

batch effect correction, and verify the authors' intuitions provided here?
- Regarding the "artificially structured graphs," I now understand its meaning, but still have a minor complaint regarding the assertion that the artificial graph structure in STAGATE does not change. My understanding is that the graph attention mechanism in STAGATE could effectively reduce the influence of unreasonable edges through low attention scores, which might mitigate this issue. A more distinct comparison between STAIG and STAGATE might focus on STAIG's integration of image features rather than solely relying on transcriptomic data.
- The colors representing "intra-view non-neighbors" and "inter-view non-neighbors" in Fig. 1e do not match the contrastive learning illustration, which may confuse readers.
- In line 463, the second highlighted term should be I(h_i^1).
- The panel order in Fig. 3c does not match those in Fig. 3a, b.

(Remarks on code availability)
Language used in the code has been rectified.

Reviewer #3

(Remarks to the Author)
The authors have addressed my comments and most comments from the other reviews.

The revised version of the manuscript clearly improves the original version, especially by the inclusion of results from imaging-based spatial transcriptomics.

However, as raised by reviewers #1 and #2, the claims of global interaction between multiple datasets could be excessive. While the responses clearly show that equation 9 is a global computation, they are not connected through adjacencies, only through their embedding space --- which is the default for deep learning training with multiple samples --- it's unclear how much of that can be qualified as a novelty once the unnecessary terms are removed and the formulas simplified.

This doesn't disqualify the paper's claims and experiments that they obtained a better integration, but the description could be more transparent.

This extends to equation 4, as acknowledged during their response. Despite being originally motivated by softmax, once you apply "log" and then "exp," it would be clearer to show that "Pij" is a normalized distance, which raises the question of why the +1 is an important constant to this normalized distance.

(Remarks on code availability)

Version 2:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
Thanks for the authors' response to my previous concerns, although some of them are not fully addressed. Therefore, in the following questions, I stay with my previous questions with numbering consistent with those in my last review letter:

Major

1. The authors claim that they have updated STAIG code and incorporated resampling process into the construction of G1 and G2 to improve STAIG's performance stability. However, I do not see the incorporation of resampling process in their method section. Moreover, the authors should show experimental results, including the ablation study on the resampling process, to prove the improved stability due to these efforts. Given the randomness inherent to STAIG, it is better to add a section in the manuscript, describing the stability issue and authors' efforts on addressing this issue. Also, given the authors' claim the update results in a slight decrease in computational speed as well as the extra costs associated with the resampling process, STAIG's scalability against different sample sizes after these updates should be demonstrated.

2.The authors have updated STAIG to restrict negative sample pairs to the same slice during integration and claimed that it significantly improves the integration. However, this update fails to address my previous concern about shortening distances between inter-batch positive pairs, which can truly mitigate batch effects in theory. Moreover, this update seems to contradict the authors' statement in the previous response, where I quote:

"Firstly, let's consider STAIG's neighbor contrastive loss functon(9) (Line 460). In this equation, the numerator represents the computation of similarities between a sample and its immediate neighbors, treated as positive pairs within the augmented graph. Conversely, the denominator accounts for the interactions with negative samples across the entire samples (The second term of the denominator). This global computation not only normalizes the impact of local interactions but also indirectly bridges different datasets by adjusting each data point's influence in relation to the overarching data characteristics. Therefore, although direct dataset interactions are not modeled by the matrices A and W, the denominator

plays a crucial role in facilitating an indirect information exchange, thereby significantly enhancing the model's capacity to generalize across diverse datasets".

This statement clearly indicates the importance of using cross sample negative pairs to make STAIG as a multi-batch model, which is however denied in this response. Although the authors have conducted ablation studies, attempting to demonstrate this point empirically, the seemingly good results may be purely random, particularly considering its theoretical deficiency. Restricting negative sample pairs to the same slice during integration essentially forbids interactions between samples across batches and makes STAIG a method for single batch only. In summary, I do not think authors fully address my concern about STAIG's effectiveness in batch integration, at least theoretically.

3.Graph construction: Admittedly, few existing methods augment graph using H&E image. I believe this is due to several reasons: Firstly, Only a limited ST platforms provide H&E image. Using gene similarity or proximity is for general applicability, at least better than the ad hoc parameter 0.2 used in the fixed edge removal in STAIG when H&E image is unavailable. Secondly, H&E image quality cannot be guaranteed, with different tissue types appearing similarly in the image. For example, a TME microenvironment may encompass many tumor subtypes but appear as a homogeneous tumor. Nonetheless, such subtle differences can be detected by gene expression data at the molecular level. Therefore, authors need to demonstrate H&E images' edge over ST data even when the H&E image quality is poor. Otherwise, authors need to provide corresponding solutions.

H&E image utilization: Spatial clustering methods in ST that effectively use the H&E image data are not limited. For example, iSTAR and conST are well-known such methods.

Batch integration: Since authors provide a claim contradictory to their previous claim about the using of cross-batch negative samples in the response to my question 2, as well as fail to address my concern about STAIG's batch integration from a theoretical perspective, I am not convinced about authors' claim "STAIG's unique framework achieves batch integration that prior methods could not".

3. Thanks for the authors' detailed explanation about their choice of these hyperparameters, though I am a little surprised when reading authors' response "Regarding the adjustment of edge removal probabilities. Initially, we did not perform sensitivity testing for the parameter p during our early experiments.". Given the key role of p in the absence of H&E image for most ST platforms, the lack of sensitivity test leaves the choice of 0.3 in the initial submission purely based on speculation, not even on less rigorous empirical sensitivity test, let alone solid theoretical foundation.

4. Since the authors have removed the DS filtering in scenarios involving images from difference slices, corresponding results should be updated to reflect this change. For example, if Fig.7 involves DS filtering in the previous version, its results need to be updated.

5. The authors fail to understand my concern. Please see the simple math below:
$W=log\left(D^{img}\right)\rightarrow e^{W_{i,j}}=e^{log\left(D_{i,j}\right)}=D_{i,j}\rightarrow P_{i,j}=\frac{D_{i,j}}{\sum D_{i,k}A_{i,k}}$

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)
I appreciate the authors' efforts in addressing my previous questions. The clarity of the manuscript has significantly improved. Following the recommendation of reviewer 1, I attempted to install and run the software as an end user. I am glad to report that the obtained results were nearly identical to those presented in the manuscript when executed on a local server equipped with A100 GPUs.

I have one remaining request concerning the newly-added ablation experiments on batch integration. Typically, batch correction carries the risk of over-correction, which may result in the removal of genuine biological signals. Therefore, in addition to the iLISI and batchKL metrics that quantify the extent of batch mixing, I suggest that the authors also report metrics related to the preservation of biological signals in these ablation tests (such as NMI and ARI), as is already included in previous parts of the manuscript.

(Remarks on code availability)

Version 3:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

Thanks for the authors' diligent efforts in addressing my previous concerns. The overall quality of the manuscript has improved, with a few remaining points that I hope the authors can address.
Major

1. Time Costs and Hardware Configurations: While the authors have addressed the stability concern through testing, I would like to see the incurred time costs. Specifically, the authors should report STAIG's running time in minutes, relative to the number of datasets and total number of spots, along with the hardware configurations of the test environment.
2. Batch Integration Clarity: I remain unclear about the batch integration process. Specifically, does any information exchange occur between batches during STAIG's batch alignment? If so, how is this exchange achieved without interactions between dataset-specific adjacency matrices in the $\mathcal{A}$ matrix and cross-sample negative samples? Additionally, I seek clarification on the difference between using an aggregated $\mathcal{A}$ matrix (similar to whole batch training) versus training each dataset separately with their own adjacency matrices (akin to mini-batch training). I would appreciate it if the authors could explain this distinction using straightforward mathematical language.
3. Systematic Hyperparameter Selection: I urge the authors to adopt a more systematic approach to hyperparameter selection. For instance, they should consider the four types of feature gene selection methods: selecting highly variable genes (HVGs) 1, minimizing redundancy in gene selection2, 3, utilizing co-expression gene networks4, 5, and selecting spatially variable genes (SVGs) 6, 7. At a minimum, the authors should cite these methods and, if feasible, conduct tests to demonstrate a rigorous and comprehensive approach to hyperparameter selection.

1. Chen, H.-I.H., Jin, Y., Huang, Y. & Chen, Y. Detection of high variability in gene expression from single-cell RNA-seq profiling. BMC genomics 17, 119-128 (2016).
2. Deng, T. et al. A cofunctional grouping-based approach for non-redundant feature gene selection in unannotated single-cell RNA-seq analysis. Briefings in Bioinformatics 24, bbad042 (2023).
3. Ding, C. & Peng, H. Minimum redundancy feature selection from microarray gene expression data. Journal of bioinformatics and computational biology 3, 185-205 (2005).
4. Langfelder, P. & Horvath, S. WGCNA: an R package for weighted correlation network analysis. BMC bioinformatics 9, 1-13 (2008).
5. Zhang, T. & Wong, G. Gene expression data analysis using Hellinger correlation in weighted gene co-expression networks (WGCNA). Computational and structural biotechnology journal 20, 3851-3863 (2022).
6. Hu, J. et al. SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. Nature methods 18, 1342-1351 (2021).
7. Svensson, V., Teichmann, S.A. & Stegle, O. SpatialDE: identification of spatially variable genes. Nature methods 15, 343-346 (2018).

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)
I appreciate the authors for addressing my previous concerns. I have no new questions but would like to revisit the first question raised by reviewer 1 regarding the lack of theoretical guarantees in STAIG's batch integration capability, which was also my primary concern in previous rounds.

The authors' rationale that model generalizability is the key to its batch correction capability makes sense. The challenge of batch effects in many existing embedding techniques (e.g., PCA or VAE) stems from their requirement to reconstruct the original expression data, thereby preserving any major data variation in the embeddings, including batch effects. Common strategy to mitigate this issue include incorporating "batch" as a covariate in the embedding process, allowing batch variation to be explained away[1], and explicitly pulling batches together with cross-batch anchors[2]. Alternatively, eliminating the need for embeddings to reconstruct the entire expression profile offers a third solution. Early attempts using supervised classification[3] or STAIG's use of self-supervised contrastive learning fall into this category. By eliminating the data reconstruction requirement, embeddings are less inclined to retain batch variations, and more likely to use up all explanation power for capturing within-batch variations, which leads to integration when combined with generalizability across batches.

Nevertheless, the absence of a theoretical guarantee means that the effectiveness of STAIG in batch correction could vary, particularly as "generalizability" of deep learning models can be unstable, and really depends on the extent of distribution shift between batches. It is conceivable that STAIG's batch correction capabilities might falter as batch effects intensify beyond a certain generalizable limit. Demonstrating this through synthetic augmentation of batch effects between slices could help identify STAIG's operational limits, and provide valuable guidelines for users regarding the reliability of STAIG in batch effect correction under varying conditions.

[1] Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. Nat. Methods 15, 1053–1058 (2018).
[2] Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nat. Biotechnol. 36, 421–427 (2018).
[3] Lin, C., Jain, S., Kim, H. & Bar-Joseph, Z. Using neural networks for reducing the dimensions of single-cell RNA-seq

data. Nucleic Acids Res. 45, e156–e156 (2017).

(Remarks on code availability)

Version 4:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
The authors have addressed all my concerns, and I have no further questions.

(Remarks on code availability)
The codes are reproducible

NCOMMS-24-00513A

STAIG: Spatial Transcriptomics Analysis via Image-Aided Graph Contrastive Learning for Domain Exploration and Alignment-Free Integration

Yitao Yang, Yang Cui, Xin Zeng, Yubo Zhang, Martin Loza, Sung-Joon Park, Kenta Nakai

Responses to the reviewers

We sincerely appreciate the thoughtful suggestions and insights you have provided. Your feedback has significantly enhanced the quality and rigor of our work. In this revision, we have focused on addressing concerns regarding the stability of parameters, the ablation experiments for image feature extraction, and the integration capabilities. To bolster our findings, we have included additional comparative methods, improved evaluation metrics, and expanded our datasets for a more robust validation of our work. Corresponding revisions have been made to the manuscript, with all changes clearly marked.

Furthermore, in response to concerns about the reproducibility of our experimental results, we have set up an alternative configuration using cloud-based GPUs. We have uploaded supplementary materials that include the training ipynb files and a guide to accessing our cloud server. This allows you to verify the results presented in the paper within a pre-configured environment.

We hope that our amended manuscript fully addresses all the critiques and comments raised by the reviewers. Please find below our point-by-point responses to each comment.

General Response to Reviewer #1:

We sincerely appreciate your thorough review of our work and the constructive suggestions you have provided. We apologize for any issues on the implementation of STAIG. To address the reproducibility concerns regarding STAIG's results, we have rented additional GPUs and pre-configured the environment and training code. You can now follow the guidelines we have provided to operate the system and reproduce our findings. Moreover, we have significantly optimized the training time to facilitate the replication process.

To further demonstrate STAIG's integration capabilities, we have conducted a series of additional experiments and incorporated new evaluation metrics. We have also compared STAIG's performance with the algorithms you mentioned, providing a comprehensive assessment of our method. Additionally, we have performed extensive ablation studies to validate the effectiveness of our filtering process and the BYOL training framework. These experiments strongly support the robustness and efficiency of STAIG in various scenarios.

We hope that our revisions and the additional experiments adequately address your concerns. We genuinely appreciate the time and effort you have invested in reviewing our work, and we are grateful for your valuable feedback. Your suggestions have significantly contributed to the improvement of our manuscript and the overall quality of our research. Below, we respond to each comment

individually and describe the revisions made to the manuscript. We appreciate the positive feedback on the novelty and performance of our approach as well as the detailed suggestions for improvement.

Detailed Responses to Validity:

Comment: Using STAIG's example codes, we fail to obtain the spatial clustering results as good as those claimed in the manuscript: For the DLPFC 151673 dataset (Fig. 2c), the ARI for the spatial clustering results is 0.61, which is significantly lower than the reported 0.68 and the competing method GraphST (0.64).

Response: Thank you for taking the time to validate our code. To demonstrate the reproducibility of our experiments, we have set up an independent environment with a rented GPU on a cloud server for your verification. We have pre-configured the environment and provided the training code in the form of Jupyter Notebooks (ipynb). You can remotely connect to the server and perform the verification by following the detailed instructions we provide.

In the server, we have provided the training results for slice #151673 based on image features, achieving an ARI of 0.68, which is consistent with the results reported in our manuscript. Additionally, we have also included the training results for slice #151673 without image assistance (ARI = 0.65). To facilitate your initial verification, we provide you the ipynb files containing the output records of the training process.

We recognize the critical importance of reproducibility in scientific research. To this end, we have made our environment settings, training code, and comprehensive instructions publicly available. We believe this will enable future users to easily replicate and verify our results.

Comments: STAIG is essentially a method for single dataset rather than for multiple datasets, foundationally not very different from existing methods like SpaGCN. This can be told from the calculations of the block diagonal adjacency matrix A in equation (1) and the edge weight matrix W in equation (3). That is, spots from different datasets never interact with each other, with no information exchanged among them. Basically, the model should behave same as the case where the model is trained using the datasets one by one. Indeed, in our attempts to repeat the results for integrating horizontal slices, which have larger batch effects, using the provided example codes, we failed to reproduce the results in Fig.5c. Additionally, the UMAPs indicate suboptimal performance of the integration. As the two batches are well-separated while spots of same domains are dispersed.

Response: Thank you for your valuable comments. While it is true that the slices in STAIG are disjoint in the adjacency matrix during the initial construction stage, there is indeed information exchange between them during the training process.

Firstly, let's consider STAIG's neighbor contrastive loss function (9) (Line 460). In this equation, the numerator represents the computation of similarities between a sample and its immediate neighbors, treated as positive pairs within the augmented graph. Conversely, the denominator accounts for the

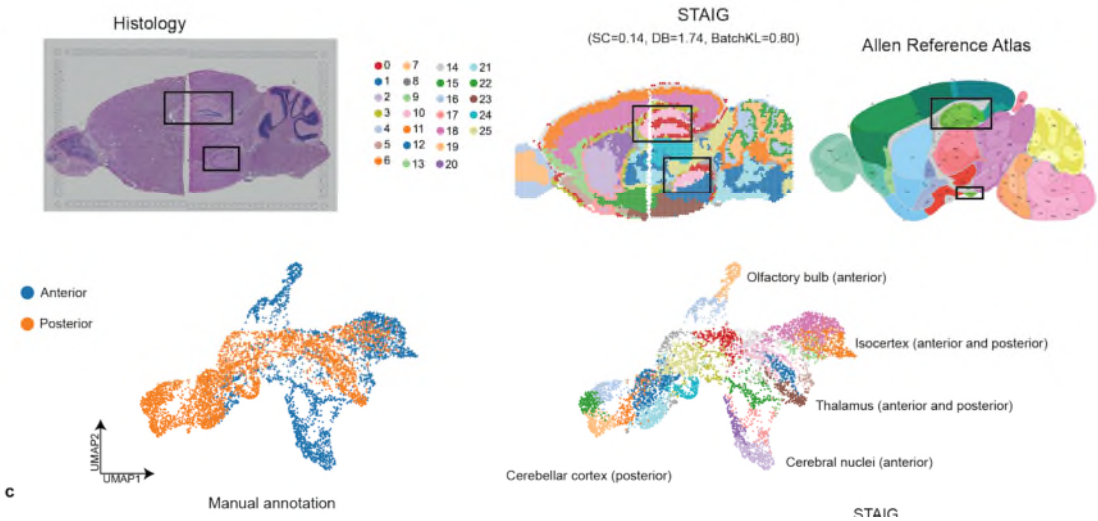
interactions with negative samples across the entire samples (The second term of the denominator). This global computation not only normalizes the impact of local interactions but also indirectly bridges different datasets by adjusting each data point's influence in relation to the overarching data characteristics. Therefore, although direct dataset interactions are not modeled by the matrices A and W, the denominator plays a crucial role in facilitating an indirect information exchange, thereby significantly enhancing the model's capacity to generalize across diverse datasets.

$$l\left(h_i^{(1)}\right) = -log\frac{\left(f^{1,2}(i,i) + \sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j) + \sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)\right)/\widehat{\mathcal{N}_i}}{f^{1,2}(i,i) + \sum_{(j \neq i) \cap \left(y_j \neq y_i\right)} \left(f^{1,1}(i,j) + f^{1,2}(i,j)\right)}, \quad (9)$$

Secondly, STAIG adopts a novel approach by concentrating on local patterns during edge perturbations. This strategy emphasizes subtle local dynamics over broad global changes to capture consistent expression patterns that are prevalent across different sections, even in the presence of varying batch effects. This allows STAIG's embeddings to neutralize batch effects effectively ensuring that the model's output remains biologically relevant and consistent, irrespective of variations in data processing or experimental conditions across different datasets.

Thirdly, the weights and dimensionality reduction of samples are shared by placing each slice diagonally in the adjacency matrix, allowing for information among exchange among samples during training.

Regarding the results in Figure 5, we have reproduced the experiments on a third-party cloud platform and provided the UMAP analysis results. The latest experimental results are updated in Fig. 5b(Line 193-204). It can be observed that the parts from same functional from the two slices overlap in the UMAP plot, such as Isocortex and Thalamus, while the slide-specific regions are separated, such as the Cerebellar cortex and Olfactory bulb. We compared our results with SpaGCN and GraphST (Supplementary Fig. S12), which manually align coordinates and integrate, and their UMAP distribution patterns are consistent with ours. Additionally, STAGATE's UMAP completely overlaps, but its clustering results are very poor, indicating that the integration of different functional areas cannot be judged solely based on whether they overlap.



The code and results for this integration part are available on the cloud server, where you can easily log in and use STAIG for verification. Furthermore, we have performed integration on additional slices from different individuals and on the Stereo-seq and Slide-seqV2 platforms (Line 217-229). We provide a detailed analysis of this part in the subsequent comments.

Comments: In Fig.3, authors claim the image features generated by BYOL can reflect the DLPFC stratifications. These results seem to be too good given the limitations of BYOL under this scenario. Therefore, the validity of Fig.3 needs further clarifications. Indeed, in our experiment using their codes to repeat Fig.3b, we failed to obtain a result as good as the one shown in the manuscript.

Response: Thank you for your valuable comments regarding the validity of the results presented in Fig. 3. To address your concerns, we have conducted comprehensive ablation experiments, comparing our BYOL framework with UNET and VIT, as well as evaluating the effect of the pre-filtering step. You may begin by reading the section on the ablation experiments in the supplementary materials (Lines 362-429), titled "BYOL ablation study for histology feature extraction." In summary, both image filtering and BYOL are crucial to the effectiveness of the STAIG model, enabling it to extract useful information from filtered images.

To further validate our results, we have reproduced the experiments on an independent server and updated the corresponding figures in the manuscript. The updated results for Fig. 3 can be verified on the cloud server we have provided.

Detailed Responses to Specific Comments

Comments: The endeavor to identify TME from tissue regions labeled as healthy. Despite these contributions, this study has limited significances in terms of its novelty in both targeted problems and the methodology:

• The work focuses on generating spot embeddings for spatial clustering and batch alignment tasks, both of which have been extensively studied in recent years. To name a few, STAGATE[1], GraphST[2], STAligner[3], SpaGCN[4] and conST[5].

Response: Thank you for your valuable comments. We acknowledge that since the emergence of spatial transcriptomics technology, numerous methods for spatial clustering and batch alignment have been proposed, and some have achieved impressive results. As you mentioned, examples include SpaGCN STAGATE, and more recently, GraphST and STAligner. Although all these methods employ graph convolution as their foundational framework, we believe that there is still significant room for improvement in terms of network structure enhancement and the extraction and integration of image features, which we have paid special attention to in the design of STAIG. Moreover, every step in STAIG's workflow was extensively optimized to function on diverse scenarios, allowing us to achieve the highest accuracy compared to the benchmarked algorithms.

This indicates that our method effectively optimizes the ST data structure. Next, we will introduce the innovative aspects of STAIG from three perspectives: the unique properties of ST data, image feature extraction and integration, and batch integration.

- Unique Properties of ST Data

ST data combines spatial and genetic information at each spot, posing unique challenges in its analysis. Often, methods transform ST data into graph structures, but unlike conventional graphs, where edges have inherent meanings like social interactions or chemical bonds, these graphs are typically formed by arbitrarily connecting points based on spatial proximity. This can lead to misrepresentations by linking biologically distinct regions, introducing biases that affect the analysis.

Current methods lack further consideration of the graph structure. STAligner directly utilizes the initially constructed graph. Although some methods consider this aspect during graph construction, such as SpaGCN simultaneously measuring image feature distance and spatial distance, and STAGATE preliminarily introducing the clustering results of dimensionality-reduced initial gene data as attention, they do not further perturb the existing structure during the training process, lacking a mechanism to optimize the graph structure.

Similarly, other methods such as GraphST and ConST utilize contrastive learning frameworks that preserve the original structure and involve random shuffling of nodes during training, leaving the structure intact throughout the process. Concurrently, these methods fail to consider the exclusion of potentially similar nodes from the negative samples during comparisons, resulting in biases from false negatives impacting the outcomes.

STAIG introduces an innovative approach by utilizing H&E image priors to dynamically optimize graph structures during training. It achieves this by adaptively adjusting edge weights, which helps to mitigate the effects of potentially incorrect edges and simultaneously exclude erroneous negative samples. Additionally, STAIG employs a unique contrastive learning objective function that concentrates on the local neighborhood structure of nodes. This focus significantly enhances clustering accuracy and effectively overcomes the limitations associated with traditional methods.

- Image Feature Extraction and Integration

Existing methods either directly use RGB information (SpaGCN) or employ pre-trained deep models to extract features (ConST). However, they struggle to handle uneven staining issues and lack the ability to capture high-level semantic features such as cell region boundaries. These limitations are more prominent in scenarios like brain tissue slices where textural features are less evident. Deep models designed for H&E image analysis, such as methods for nucleus segmentation, require additional manually labeled data for training and are difficult to generalize to ST data analysis.

To address these shortcomings, STAIG innovatively proposes extracting robust and discriminative feature representations using only a single H&E image through a band-pass filter and BYOL framework, adaptively reducing noise and staining issues while providing more prior knowledge for identifying different functional regions.

Moreover, unlike traditional methods that directly integrate image features into node features, we transform image features into graph structure perturbation probabilities through distance differences. Guided by the image information, the contrastive mechanisms in both structure and node aspects enable the model to further mine valuable information in the ST data.

- Batch Integration

Methods like GraphST, STAGATE, and SpaGCN require manual pre-alignment of coordinates for multiple slices before inputting them into the model, increasing the complexity of the analysis pipeline. They directly align slices on a two-dimensional plane, ignoring the differences in region proportions between slices, which affects the integration effect. Although STAligner can automatically perform alignment, its essence is to obtain embeddings of each group of slices, match the embeddings based on maximum mutual information neighbors (MMN), and then further integrate them. There is still room for exploration on how to integrate slices more conveniently and simply.

To exemplify the advantage of using STAIG over other popular methods we conducted an in-depth exploration of STAIG's improved results over benchmarked methods by attempting the integration of slices from different sources on both the Visium and MERFISH platforms. The results are shown in the updated Fig-5. Additionally, we have included the BatchKL and ILISI evaluation metric and added STAligner, which you mentioned, into the comparison. The results demonstrate that our integration effect is even better than STAligner's. Particularly, in the cross-platform integration (Slide-seqV2 and Stereo-seq), which is the same as STAligner's, our integration results outperform theirs across all evaluation metrics. We have reproduced their results as presented in their paper, ensuring a fair comparison.

Moreover, to highlight the unique contributions of our work over similar ones, we have revised the Introduction section of our manuscript (lines 19-63). The updated Introduction provides a clearer and more comprehensive overview of the limitations of existing methods, focusing on the challenges in graph structure construction and the training process. We discuss how current approaches often introduce biases by uniformly constructing edges between regions that belong to different biological

domains, and how they fail to modify the graph structure during training to eliminate these biases. The revised Introduction sets the stage for the novel solutions offered by STAIG, emphasizing its ability to dynamically optimize the graph structure, extract robust image features, and perform effective batch integration.

Comments: Although the authors claim STAIG is alignment-free, they do not consider batch effects among datasets but simply putting them together, which makes STAIG no different from methods specifically designed for single dataset (e.g., SpaGCN). So STAIG should not be considered as a method for handling multiple samples.

Response: Thank you for your comments. Initially, STAIG aggregates data by simply stacking them together rather than employing explicit pairing or network reformation. This approach avoids potential problems inherent in methods that force alignment post-training, e.g., introduce erroneous pairings. For instance, STAligner matches spots based on similarity in a reduced dimensional space, a process that may not reliably indicate true biological correspondence. Similarly, manual alignment methods risk could incorrectly overlay different functional areas due to imperfect overlaps.

STAIG introduces a novel contrastive framework that learns from local gene expression similarities consistent across different sections, allowing the constructions of embeddings with reduced batch effects. Let me explain why STAIG's principle of contrast allows for alignment-free integration:

Firstly, for our loss function: the contrast involves each node's corresponding augmented graph nodes as positive samples, and all other nodes as negative samples. During this process, there is no bias introduced by different batches.

Secondly, we acknowledge that different samples may exhibit batch effects, however, similar functional areas exhibit similar gene expression patterns. By learning the mapping relationships of these expression patterns, STAIG can produce embeddings that eliminate batch effects. STAIG's contrastive framework and target loss function enable this by maintaining node consistency between two augmented graphs in each iteration, only altering through random edge deletions. The deletion of edges results in the same nodes being aggregated differently due to some neighbors being absent, focusing on the discrepancies in local aggregation caused by these differences. This local information, despite being from different slices, tends to follow a consistent trend, a focus that is distinct from traditional contrastive learning which typically contrasts a node with its immediate neighbors. Moreover, by reinforcing consistency between a node and its first-order neighbors, STAIG emphasizes and clarifies local expressions.

Compared to SpaGCN, which lacks such a refined structure and consists merely of a GCN with clustering layers and an unsupervised function iterating based on sample category distribution, it cannot learn the generalization capabilities at STAIG's level when handling multiple samples. SpaGCN's convergence is based on predetermined clustering, adjusting all samples towards cluster centers based on spatial constraints, lacking dynamic error correction mechanisms and even lacking

a spatial distance when two samples are not aligned.

To date, STAIG's integration capability relies on its unique contrastive mechanism to learn and generalize local expression patterns' commonalities during iterations, achieving integration across multiple sections. Its design framework and adaptability are distinct from single-sample frameworks, making it broadly applicable.

From an experimental standpoint (line 182-229), STAIG has been rigorously tested across various integration metrics, including clustering accuracy, the BatchKL and ILISI index. Our results not only meet but exceed those of methods specifically designed for data integration, such as STAligner. This validation underscores STAIG's efficacy in handling multiple datasets without the need for pre-model alignment, affirming its utility in spatial transcriptomics where batch effects are a significant concern.

Comments: From the perspective methodological framework, it is very similar to conST in aspects including visual feature extraction and the idea of using contrastive graph learning.

Response: Thank you for your comments. Although both STAIG and conST employ image feature extraction and contrastive learning concepts, there are significant differences between the two methods.

Firstly, the method of image feature extraction in STAIG is fundamentally different from that of conST. ConST uses a pre-trained MAE model that, due to its large parameter size and transformer-based architecture, is not designed to be fine-tuned on single Visium HE images. This approach restricts its adaptability to the specific challenges of spatial transcriptomics, such as uneven staining and variable noise levels, leading to suboptimal feature representation as evidenced by the weak correlation of its extracted features with known annotations in the DLPFC dataset (Fig. 3a-c, Supplementary Fig. S29a-c). STAIG, however, employs a BYOL-based framework adapted for single HE images, which allows it to learn directly from the dataset it processes. This results in a more tailored and robust feature set that correlates better with biological annotations.

Secondly, the integration strategy of image features into the model also differs significantly. ConST merges these features directly with gene expression data, which may not always be optimal given the distinct nature of the data types involved, especially after evaluating the quality of their image features. After careful consideration, STAIG adopts a gene-centric approach, integrating image features into the graph structure itself during the training process. This integration is performed by adjusting edge weights based on image-derived information and utilizing image similarity to remove potential false negative samples. This method enhances the model's ability to identify and strengthen biologically relevant connections while minimizing noise-induced errors.

Thirdly, the contrastive learning frameworks employed by STAIG and ConST are fundamentally different, reflecting STAIG's advanced capabilities and nuanced approach in integrating and analyzing complex datasets:

- **ConST Approach**:
  - **Contrast Mechanism:** ConST uses the DGI[6] architecture, employing both local-local and global contrast. It retains a predefined graph structure throughout the training process, which may not adequately address the biases introduced by the initial graph construction. This rigidity in the structure can result in a partial loss of local information about nodes and their neighbors, as the contrasts are made primarily with nodes and their aggregated neighboring nodes.
  - **Bias and Sample Errors:** ConST does not account for biases arising from false samples during training, a factor known to impact contrastive learning outcomes significantly[7]. Additionally, its local-local contrast is designed to offset the lack of individual information consistency found in global contrasts, rather than to enhance the similarity between a node and its first-order neighbors. This often results in much more dispersed region clustering results, especially evident when handling non-Visium platform data (as shown in Supplementary Fig. S8,S9).
- **STAIG's Innovative Framework:**
  - **Dynamic Graph Adjustment:** Unlike ConST, STAIG dynamically adjusts its graph structure during training by updating edge connections based on the similarity of image features between nodes. This allows for the perturbation of edges to reflect more accurate biological connections and reduces the likelihood of forming erroneous links due to data noise or batch effects.
  - **Contrastive Goals:** The contrastive target in STAIG involves comparing the aggregation results of the same node under different neighborhood configurations due to edge deletions. By examining the differences in neighbor aggregations, STAIG learns the commonalities in local gene expression, which equips it with the capability to handle multi-slice integration.
  - **Bias Reduction:** During training, STAIG considers known image information to remove potential false negative samples, thus minimizing bias. This integration of image features into the decision-making process for edge retention or removal enhances the model's precision in forming biologically relevant clusters.
  - **Emphasis on Neighboring Consistency:** STAIG places significant emphasis on the consistency of neighboring nodes during the contrast process, aligning with the biological importance of spatial relationships in transcriptomics. This focus results in clustering outcomes with clearer boundaries for each functional area, outperforming ConST and other methods even on complex datasets like MERFISH (as evidenced in Supplementary Fig. S8,S9).

Fourthly, while conST relies on a complex setup involving pre-training and additional training steps, STAIG is designed as an end-to-end unsupervised model that learns directly from the data without the need for preliminary training phases. This makes STAIG not only more straightforward to deploy but potentially more adaptable to varying datasets without overfitting to preconceived features.

In conclusion, although both STAIG and conST utilize image extraction and contrastive learning, there are substantial differences in their approach, effectiveness, integration of image features,

contrastive learning framework, and overall model architecture. Finding a more suitable application of contrastive learning algorithms for spatial transcriptomics data remains a valuable research question, and STAIG presents a novel and effective solution to address the limitations of existing methods.

Comments: STAIG can hardly be recognized as a multimodal learning algorithm as the visual features are only used for graph augmentation rather than being incorporated into the final embeddings.

Response: Thank you for your comment. The definition of multimodal learning indeed encompasses a broad range of applications. In the field of spatial transcriptomics, several approaches integrate image information differently. For instance, stLearn[8] normalizes gene expression using image data, SpaGCN[4] constructs edges based on color information from images, and DeepST[9] uses CNN-derived features to reinforce graph structures. These methods integrate visual information but do not directly incorporate it into the embeddings. Direct integration of image features into embeddings is not the sole method to harness image data effectively.

STAIG uniquely transforms visual differences into edge weights, which significantly incorporates image information into the training process. To date, in the spatial clustering domain, only conST concatenates image information directly into node features after substantial preprocessing and pretraining. From our evaluation of conST's capability to extract relevant image features, it remains questionable whether features not fully aligned with known annotations truly aid in model training.

We opted to use images as weights rather than embedding them directly for several reasons:

1. Influence on Final Embeddings: Using images as weights in graph augmentation effectively influences the final embeddings. The graphical structure modified by these weights reflects more accurate biological relationships, indirectly embedding the visual information.

2. Enhanced Robustness and Generalization: Directly incorporating visual features into embeddings might introduce noise, especially considering the variation in staining quality and imaging conditions across different datasets. STAIG cleverly uses visual features to guide graph augmentation, avoiding direct noise incorporation into the embeddings. This approach significantly enhances the model's robustness and generalization capabilities across various datasets.

3. Empirical Support: Experimental results robustly validate the effectiveness of the STAIG approach. Indirect utilization of visual features through graph augmentation has shown significant improvements in clustering accuracy and annotation precision compared to methods that either ignore visual information or integrate it directly into embeddings.

In summary, the development of STAIG was aimed at leveraging the potential of image information to enhance the accuracy of clustering in spatial transcriptomics, without strictly adhering to the rigid categorizations of learning algorithms. Our empirical evidence demonstrates the effectiveness of this

approach.

Comments: STAIG is very slow compared to other similar methods like SpaGCN and STAGATE, as the clustering of a single dataset can take up to 4 hours.

Response: Thank you for your comment and feedback. We acknowledge the concern regarding the initial processing time of STAIG, which was indeed longer in its first versions due to extensive image preprocessing. In response, we have optimized the training epochs, significantly reducing the time required to extract image features to approximately 3-5 minutes.

Furthermore, we have developed an alternative version of STAIG that does not rely on image data but instead utilizes gene expression similarity for graph augmentation. While this version slightly underperforms compared to the image-based approach—achieving an ARI of 0.65 compared to 0.68—it still surpasses other comparable methods. Importantly, this non-image version can complete training and generate embeddings in about 10 seconds on a 3090 platform, aligning its performance time with that of SpaGCN and STAGATE.

We invite you to verify the improved performance and efficiency of both versions on our provided cloud platform, where the updates have been implemented to enhance user experience and processing speed.

Comments: The authors use BYOL to extract cell morphology features from H&E image. It may be unnecessary or even worsen the extracted features, as it is well-documented that a simple U-Net can effectively handle this task6. Authors need to justify their choice here.

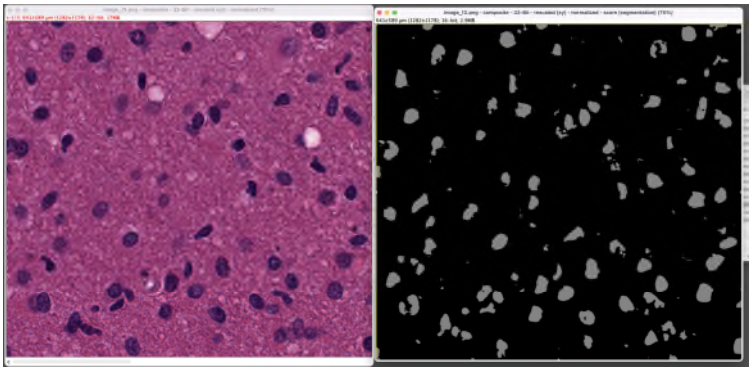
Response: Thank you for your insightful comment. We acknowledge that U-Net has been widely used in traditional H&E image analysis, particularly for tasks such as cell nucleus counting and segmentation. However, we believe that U-Net may not be the most suitable approach for handling H&E images in the context of spatial transcriptomics, and we have chosen BYOL for several reasons. Firstly, spatial transcriptomics datasets typically consist of a single H&E-stained image without corresponding pixel-level annotations or masks. U-Net, being a supervised model architecture, requires labeled masks for training. In the absence of such annotations, we would need to rely on pre-trained U-Net models to extract features from spatial transcriptomics H&E images, which may not be optimal.

Secondly, the training objective of U-Net models, which involves comparing the output with ground truth masks, primarily focuses on pixel-level classification tasks such as identifying cell nuclei. However, in the context of spatial transcriptomics, we aim to extract higher-level image features that align with human perception of different tissue regions. These features not only depend on the number of cell nuclei but also on their shape, size, density, and overall cellular morphology. The training paradigm of U-Net may not fully capture these high-level features.

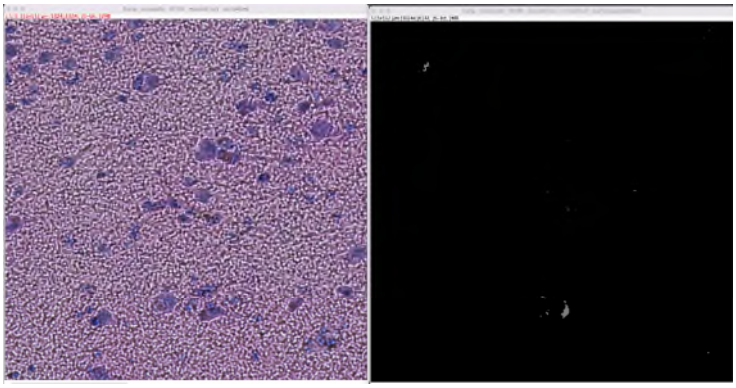
Thirdly, the architecture of U-Net poses challenges in directly extracting suitable embedding

representations. The output of U-Net is typically a pixel-level mask, while the intermediate layers have high-dimensional features that are difficult to directly use as embeddings.

To further illustrate these points, we have conducted experiments using a pre-trained U-Net model, as mentioned in your reference[10], on a region of the DLPFC 151673 slice. In the images below, the upper panel shows a classic H&E-stained image from a dataset used for training cell segmentation tasks. We can observe that the reference method you provided achieves good segmentation results on this image. However, when an image from the 151673 slice is inputted, as shown in the lower panel, we can see that hardly any cells are segmented. This is due to the presence of a large amount of useless noise in the spatial transcriptomics H&E image, which hinders the performance of the pre-trained U-Net model.



classic H&E-stained image by Unet



#151673 by Unet

In contrast, we have chosen BYOL as our self-supervised learning framework for the following reasons:

1. Self-supervised learning is a suitable approach for spatial transcriptomics H&E images, which often lack class labels. BYOL enables us to learn useful embedding representations from a single image for downstream classification tasks.

2. BYOL does not require negative samples, making it particularly suitable for scenarios where class

information is not available. In comparison, other self-supervised methods need to construct negative samples.

3. After evaluating various existing H&E image feature extraction methods, including RGB value clustering (stLearn), CNN (DeepST), and MAE (conST, another self-supervised model), BYOL achieved the best performance. It effectively mitigates the impact of color inconsistencies and noise through various image augmentation techniques, learning more robust feature representations.

4. Ablation experiments have demonstrated the effectiveness of BYOL's self-supervised training process. The embeddings output by the BYOL-trained model are more consistent with manual annotations compared to directly using ResNet50 features (Supplementary Fig. 29).

In summary, while U-Net has been successful in traditional H&E image analysis tasks, it may not be the most appropriate choice for handling spatial transcriptomics H&E images. BYOL, as a self-supervised learning method, is better suited to address the unique characteristics of these images and learn robust and discriminative feature representations. Our experimental results and ablation studies support the use of BYOL over U-Net in the context of STAIG (materials Lines 362-429, titled "BYOL ablation study for histology feature extraction).

We have also made changes in the corresponding places in the paper line 353-358: We utilized BYOL[11] with a ResNet50[12] backbone for advanced self-supervised image feature extraction due to its ability to learn robust features without the need for negative samples, ideal for spatial transcriptomics where class labels are absent. Unlike BYOL, U-Net, traditionally employed for tasks like cell nucleus counting and segmentation, requires supervised learning with annotated masks and is not designed to extract embeddings. This limitation makes U-Net[10] unsuitable for integrating complex biological information from HE images in spatial transcriptomics, which goes beyond simple cellular structures.

Comments: BYOL needs data augmentation. Unlike general images (e.g., ImageNet), the data augmentation techniques may not be applicable here, as histology images typically lack clear semantic segments (i.e., kind of fuzzy and blurred), potentially leading to significant biases. Authors need to clarify the usefulness of the data augmentation techniques.

Response: Thank you for your valuable comments. Regarding your concerns about the applicability of data augmentation techniques in processing histology images with BYOL, especially considering their fuzzy and blurred characteristics, we offer the following clarifications and evidence supporting our approach.

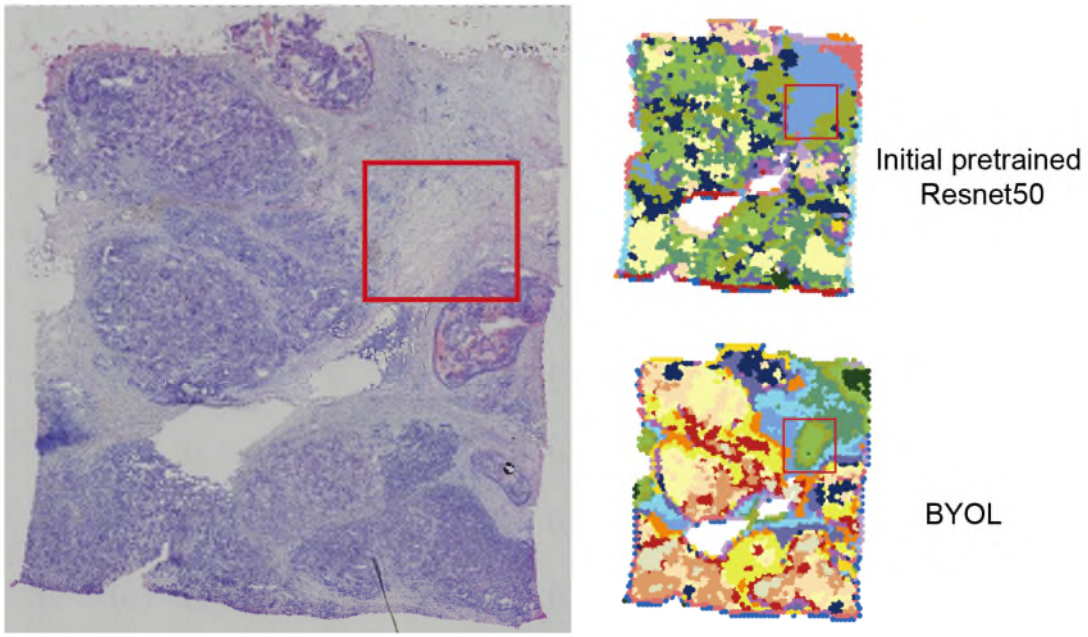
1. **Integration of Image Features with Genetic Information:** In the development of our STAIG model, we opted not to directly integrate image features but instead used gene-driven GCN computations. This method incorporates image features as weights to enhance the generalization of graph structures, effectively minimizing the influence of image noise. Thus,

even if biases arise in certain local areas, the overall clustering results remain reliable and unaffected by such distortions.

2. **Handling Spatial Transcriptomics Images:** Our methodology specifically addresses the challenges posed by blurred and fuzzy histology images by focusing on small patches rather than the entire image. These patches, similar in function to pixels in general imagery, represent small, consistent texture areas within the same tissue type. Our training on these patches ensures that the model captures common textural features essential for classification, thus mitigating the impact of boundary blurring.

3. **Data Augmentation and Texture Emphasis:** We have employed semantic-preserving image augmentation techniques such as rotation, scaling, and slight color adjustments. These augmentations enhance the model's focus on texture robustness, reducing interference from color and noise, which is crucial for extracting meaningful semantic information from histology images.

4. **Validation through Ablation Studies:** Our supplementary material (Fig. S29) presents an ablation study comparing the outputs of a baseline ResNet50 and the same network post-BYOL training. The results clearly show that BYOL has learned to discern contours effectively, thereby validating the utility of image augmentation in enhancing feature extraction capabilities. For example, the area of the red box in the image below, which was trained BYOL to successfully differentiate itself from the surrounding area, actually does look dissimilar to the naked eye when the texture is seen.



5. **Supporting Evidence from Related Work:** Although unsupervised methods are less common in HE image analysis, there have been promising attempts by researchers such as Punn et al., who developed an encoder-decoder architecture (BT-unet)[13] for medical segmentation. They

employed similar image augmentation techniques to train their encoder in an unsupervised manner before fine-tuning it in a supervised setting. Their results demonstrated significant improvements in cell nucleus segmentation accuracy and better feature capture in brain tumor segmentation tasks post data-augmented contrastive pre-training. We have cited this paper in line 362.

Comments: The parameters of the band-pass filter are not explicitly stated in the paper, which is a potential pitfall, given the different qualities between tissue slices or even within the same slice. An arbitrary setting may severely deteriorate the image (e.g., over-blurring of the image). Authors need to explain the standards for choosing these parameters.

Response: Thank you for your valuable comment. You are correct that arbitrarily setting the parameters of the band-pass filter can lead to a deterioration in the effectiveness of feature extraction. To address this concern, we have conducted additional experiments to test and optimize the filter parameters, which are now included in the supplementary materials, lines 338-359 (Bandpass filter parameters selection).

The default value is added in main text line 350-351: The default parameters for the band-pass were $245 - 275$ (see Supplementary Material, Bandpass filter parameters selection).

We have placed the supplementary material here for your convenience to read directly.

Before introducing images to the BYOL framework, it is necessary to apply a bandpass filter to the image input. This step is aimed at reducing the impacts of noise and uneven staining. Here, finer textures are removed, leaving behind coarser ones, effectively highlighting the features that the model should focus on before input. We ensure a uniform scaling of the image to 512x512 pixels so that the frequency range is (0,512). We evaluated the effect of the filter's frequency band on the final outcomes using slices #151507 and #151673 from the DLPFC dataset and a dataset from human breast cancer—these represent the two most common histology image scenarios: brain and cancer tissues.

From the result shown in supplementary Fig. S28c, we observed that broader frequency bands, such as 0-300, 200-300, or 200-512, did not effectively eliminate noise. The model predominantly identified vertical striations in brain images, and the boundaries in breast cancer slice images remained blurred. Upon narrowing the frequency band to 240-280, slice #151673 began to display clustering layers similar to the class labels, and the breast cancer image clustering also started to segregate the main cancerous areas. The classification effect on slice #151507 was still not very prominent. Further narrowing the band to 245-275, we noticed improved results; slice #151507 aligned with the distribution of class labels, with layer 1 being accurately captured. However, compressing the frequency band further to 250-270 caused the disappearance of detailed sub-clusters in all three images, suggesting excessive information filtering that could hinder the model's learning process.

From this parameter selection exercise, it is clear that information filtering is crucial for unsupervised feature extraction from images. Retaining too much information may distract the model with staining issues, preventing desired outcomes. Conversely, excessive filtering might reduce the model's learning performance. Therefore, we have determined that the optimal frequency band is 245-275. Given that we consistently fix the image size before input, this frequency band should offer a relatively consistent noise filtration and thus possess broad applicability.

Comments: The normalization of the aggregated raw gene expression matrix X on line 296 needs further explanation. This normalization potentially downweighs the importance of datasets with low sequencing depth, which yet may hold important information.

Response: Thank you for your valuable comment. We have refined our preprocessing steps before integration in the updated version of our manuscript. Specifically, following insights from your comments and methodologies described in STAligner and DeepST, we now implement an individual normalization for each slice separately. This approach preserves critical information within each dataset, ensuring that significant biological signals are not inadvertently diminished.

Furthermore, after normalizing each slice, we proceed by identifying and using the intersecting set of highly variable genes (HVGs) across the slices. This step is crucial for maintaining consistency and comparability across datasets, allowing for a more robust and integrated analysis.

These modifications have been elaborated upon in the manuscript from lines 373 to 377: Each slice was processed individually while dealing with multiple tissue slices. At the initial stage of integration, we adopted the same approach as STAligner to individually process each slice. The raw gene expression counts of each slice were log-transformed and normalized, and then the data were scaled to achieve a mean of zero and unit variance. We also selected a default of 5,000 highly variable genes (HVGs) for each slice. Subsequently, we took the intersection of all the HVGs as the final set of common F HVGs. After filtering for the common HVGs in each slice, we proceeded to the next step of integration.

Given m slices with the respective numbers of spots $N_1, N_2, \ldots, N_m$, the total spot count T is the sum of all N_i. We then obtained distinct adjacency matrices A^1 to A^m and image feature matrices C^1 to C^m for each slice. Batch integration commenced with the vertical concatenation of common gene expression into a single matrix $\mathcal{X} \in \mathbb{R}^{T \times F}$.

Comments: The slice integration seems unnecessary, as the way A constructed in equation (1) and W constructed in equation (3) means spots in different datasets would never interact with each other. So, this slice integration step is meaningless. Please comment on this.

Response: Thank you for your comments. Although in the initial input phase of STAIG the adjacency matrix is constructed diagonally, meaning there is no direct edge interaction between different slices, this setup allows multiple slices to share the same GCN weights and dimensionality reduction matrices during training, which is foundational in reducing batch effects. Additionally, our loss function involves nodes contrasting against all other nodes, which further reduces global discrepancies and

enables the model to learn commonalities in gene expression more effectively.

The neighbor contrastive loss function (9) is designed to incorporate global comparisons, as illustrated:

$$l\left(h_i^{(1)}\right) = -log\frac{\left(f^{1,2}(i,i) + \sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j) + \sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)\right)/\widehat{\mathcal{N}}_i}{f^{1,2}(i,i) + \sum_{(j \neq i) \cap \left(y_j \neq y_i\right)} \left(f^{1,1}(i,j) + f^{1,2}(i,j)\right)}, \quad (9)$$

This design underscores the importance of training nodes together to facilitate comprehensive comparisons in the final contrastive phase.

Indeed, arranging slices diagonally before integration is a common step for methods like DeepST and STALigner that combine heterogeneous slices. This step is also part of the STAGATE method when used with Harmony. The only difference is that we have explicitly mentioned this step in our methodology, while it is implicitly performed in these other methods.

Furthermore, our empirical evidence supports this approach. We compared the performance of STAIG in integrating slices from different origins against STAGATE, which constructs adjacency matrices diagonally without any information exchange, resulting in separation on UMAP visualizations. Our method, however, successfully integrates these slices, demonstrating its effectiveness in handling batch effects and enhancing integration across different datasets (Line 206-229).

Comments: The authors used the DS algorithm, which is essentially a K-means, to exclude negative samples as those within the same cluster. This step needs further clarification as the cluster size generated by K-means can have a large variance. This means many spots can end up in a single cluster and many cluster can have only one spot.

Response: Thank you for your insightful comments. In response to your concerns, we conducted a thorough evaluation of the impact of cluster numbers on the results, and the detailed findings are now included in the supplementary material (lines 148-175, titled "Cluster number in debiased strategy").

Additionally, we added the default value of cluster in the main text line457-458: The default parameter for Q is 40 (see Supplementary Materials, 'Cluster Number in Debiased Strategy')."

We have placed the supplementary material here for your convenience to read directly.

In the debiased strategy (DS), we initially assign pseudo labels to spots using the K-means algorithm. During the contrastive learning process, we remove samples with the same class labels from the negative samples. To evaluate the impact of different initial clustering numbers on the results, we assessed the average ARI and NMI values across 12 slices of the DLPFC dataset (Supplementary Fig. S23b) and the ARI and NMI values for a human breast cancer slice

(Supplementary Fig. S23c). We tested initial clustering numbers ranging from 10 to 100, with a step size of 10.

The results showed that when the number of clusters was less than 40, it led to a decline in performance. Although there are small number of inherent class labels, the heterogeneity of image textures is more diverse. Having too few clusters results in an excessive number of samples within each cluster, and these samples may not belong to the same class. Such pseudo labels can lead to the exclusion of true positive and negative sample comparisons. When the number of clusters exceeded 60, the performance slightly decreased and then stabilized. This is because finer-grained clustering reduces the number of samples in each class, and the comparison for some nodes shifts towards comparing with the entire set. Although the effectiveness of DS diminishes in this case, other aspects such as graph structure perturbation based on image features continue to play a role. Despite some performance loss, the model still maintains good results.

To further ensure that the K-means clustering results did not produce extreme distributions, where one class has a very large number of samples while the remaining classes have very few, we analyzed the number of samples in each class for the image embeddings of #151673 (Supplementary Fig. S23d) and the human breast cancer slice (Supplementary Fig. S23e) when the number of clusters was set to 40. We sorted the classes based on the number of samples in ascending order. The analysis revealed that most classes had a considerable number of samples, and extreme distributions did not occur. Although some classes had fewer samples, this does not affect the training process for two reasons. First, when a sample is compared with the global set, it effectively transforms into a traditional contrastive learning framework, which can still learn the differences among the entire set. Second, samples grouped into the same class are due to their similar image features, and they are likely to be adjacent and of the same class. Avoiding comparisons with nearby nodes of the same class in a small range helps the model learn local expression similarities and commonalities.

Based on the experimental results, we chose 40 as the default number of clusters for DS.

Comments: Please clarify the default value for the probability matrix (i.e., without labels) on line 328.

Response: Thank you for your insightful comments. We have added additional experiments using the DLPFC dataset to simulate the impact of different probabilities of random edge deletion, as well as the results of random edge deletion at different probabilities without images across various platforms. Additionally, we have introduced an adaptive edge deletion probability based on gene similarity. The metrics for measuring gene similarity have been tested using Euclidean distance, cosine distance, and KL distance. Our results are presented in the supplementary material on line 248, under the section "Comparing adaptive edge perturbation based on gene similarity with random perturbation."

We have also made corresponding modifications in the main text from lines 413-417, where we clarified the default parameters: In scenarios lacking image data, STAIG offers two methods of edge removal: fixed edge removal with a set probability, and adaptive edge removal based on gene similarity. For fixed edge removal, STAIG's default parameter p is set to 0.2. When performing adaptive edge removal based on gene similarity, we proceed as follows: PCA is used to reduce the preprocessed gene expression data of each spot to 64 dimensions, and this reduced data is then used to compute the cosine similarity between spots.

We have placed the supplementary material here for your convenience to read directly.

In the absence of imaging data, alongside random perturbation, we also tested an edge deletion strategy based on gene similarity to reduce parameter tuning. Specifically, we first reduced the dimensionality of the nodes' original gene data to 64 dimensions using PCA. As with image feature processing, we constructed a similarity matrix by calculating node-to-node similarities based on the reduced vectors and cosine similarity. This similarity was then translated into a probability of edge deletion, analogous to image feature distance, followed by adaptive neighbor-edge random deletion based on gene similarity.

We employed the average ARI and NMI from the DLPFC dataset to assess the effects of random versus gene-similarity-based adaptive edge deletion on spatial clustering when images are unavailable (Supplementary Fig. S27d). We set the random deletion probability, p, and investigated values ranging from 0.1 to 0.5. Our simulations revealed that $p = 0.2$ delivered optimal results. Increasing p to 0.4 led to a significant decline in both ARI and NMI averages, indicating that excessive disruption of the graph structure and information loss outweigh the benefits of perturbation.

The gene cosine similarity-based adaptive edge perturbation closely matched the results at $p = 0.2$ and outperformed other values. Although changing the similarity calculation method resulted in slight declines, these were minimal. These findings suggest that while gene similarity does not surpass the peak performance of random perturbation, its results are more stable, making it a preferable choice before finalizing p values.

Subsequent tests on the Stereo-seq platform and Slide-seqV2 mouse brain olfactory datasets (Supplementary Fig. S27e-f) again indicated that $p = 0.2$ performed best, achieving the highest SC and lowest DB. These outcomes closely paralleled those obtained with gene cosine similarity. However, from $p = 0.3$ onwards, both SC and DB metrics began to deteriorate, showing no significant differences at lower values. When comparing the clustering results of cosine similarity of genes to those obtained with $p = 0.2$, consistent clustering was observed in Stereo-seq's layers. In contrast, Slide-seqV2 exhibited variations, particularly clustering the accessory olfactory bulb (AOB) region and the glomerular layer (GL) together, likely due to shared gene expressions such as the CCK gene in the GL layer (Supplementary Fig. S27g). This suggests that gene similarity-based adaptive deletion may lead to biases from specific genes, although random perturbation helps

mitigate this by uniformly distributing edge deletion probabilities across iterations.

In conclusion, across different platforms, the optimal probability for random edge perturbation was found to be $p = 0.2$. Adaptive perturbation based on gene similarity, while potentially introducing biases on high-resolution platforms like Slide-seqV2, remains the preferable choice for initial experiments due to its adaptiveness and consistent performance across most platforms, reducing the need for extensive parameter tuning.

Comments: In equation (7), what are the parameters for the Bernoulli sampling?

Response: Thank you for your valuable comments. We have tested the parameters for Bernoulli sampling, which refers to the gene masking ratio, under both with and without image conditions. The specific results can be found in the supplementary material on lines 127-146 and 233-246.

Additionally, we have explicitly stated the default parameter as 10% in the main text on line 427: STAIG's default parameter for this is set at 10% (see Supplementary material, 'Masked ratio of features in graph augmentation').

Comments: The impacts of batch effects on slice integration should be analyzed more thoroughly, given the batch effects among vertical DLPFC slices are minor.

Response: Thank you for your valuable comments. In response, we have conducted additional experiments. First, we integrated slices from different donors of the DLPFC dataset. Second, we tested the integration of visual cortex (VIS) regions from two different mice using the MERFISH platform. Notably, these two samples differ in size; one includes the VIS region while the other is comprised only of VIS samples. We successfully achieved integration in these cases as well. Lastly, we compared our results with STAligner, achieving integration of the same Stereo-seq and Slide-seqV2 datasets, with our UMAP results showing superior performance. The corresponding findings are detailed in lines 182-229 of the main text.

Comments: The efficacy of image feature extraction needs more detailed analysis. For example, which part of BYOL is most critical for this effective feature extraction. Ablation studies or comparison with another computer vision representation learning framework may help on this point.

Response: Thank you for your comment. In efforts to clearly delineate the individual effects of image preprocessing and the BYOL framework on feature extraction from histopathological images, we performed a a set of ablation experiments. The result is shown in supplementary material line 357, "BYOL ablation study for histology feature extraction".

Comments: The choice of the significance threshold log2FC for DE analysis should be discussed

Response: Thank you for your valuable comments. In this study, we used 0.25 as the log2 fold change (log2FC) threshold for determining differentially expressed genes (DEGs). Firstly, during the analysis, we employed the Seurat package, where 0.25 is the default parameter in the FindMarkers() function. This default setting is widely used. Secondly, although in bulk RNA-seq DEG analysis (for example, using DESeq2 or limma), the log2FC threshold is often set at 1, a threshold of 0.25 is considered appropriate for spatial transcriptomics data. This is due to the particular characteristics of spatial transcriptomics, which typically involve lower sequencing depth and data that is accompanied by high noise and sparse features compared to bulk RNA-seq. These factors lead to a situation where setting the log2FC at 1 would make it difficult to detect significant differentially expressed genes in spatial transcriptomics data.

We have added the discussion of this content in the supplementary material lines 92-99 (Justification for the selection of log2fc threshold).

We also have made corresponding revisions in line 513 of the main text: A comprehensive discussion regarding the selection of the log2FC threshold is provided in the Supplementary Materials.

Comments: The statistical tests for the GO analysis (BH corrected test?) should be stated.

Response: Thank you for your valuable comments. We have employed the BH (Benjamini-Hochberg) corrected test for the GO analysis, and we have made the corresponding update in lines 510-511 of the main text: Gene ontology analysis was conducted using the clusterProfiler package, and the statistical tests were corrected using the Benjamini-Hochberg (BH) method.

Comments: In addition to ARI or SC, the results should also show one more metric (e.g., NMI).

Response: Thank you for your valuable comments. Thank you for your valuable comments. In the updated version of our manuscript, we have added the NMI (Normalized Mutual Information) metric for datasets with labels, and for datasets without labels, we have introduced the DB (Davies-Bouldin) index alongside the SC (Silhouette Coefficient) to evaluate the outcomes. We have updated the corresponding results sections and figures to reflect these additions.

Comments: In addition to the UMAPs, some metrics should be used to measure the batch mixing effects (e.g., BatchKL).

Response: Thank you for your valuable comments. We have incorporated the BatchKL metric and iLISI metric in all our integration experiments to measure batch mixing effects. The results indicate that our integration performance is generally better than STALigner.

Comments: In the paragraph (line 206-208), the authors claim differential expressions of muscle and tumor marker genes in the two clusters of slice A and B. However, it is really difficult to tell such differences from Fig.7f.

Response: Thank you for your valuable comments. We apologize for the color scheme in the original version, which made it difficult to discern the differences between the clusters. In the new version, we have updated the markers for muscle and tumor genes, which are still referenced in the original paper, and improved the color scheme for clearer differentiation. The original gene set included $atp2a1l$, $ckma$, $BRAF^{V600E}$, and $mt2$. The current gene set comprises $ckmb$, $ckma$, $BRAF^{V600E}$, and neb.
The corresponding modifications in the text are now on lines 273-274.

Comments: The study of TME in Fig.7 should include a competing method TESLA7 specifically designed for detecting TME using ST data.

Response: Thank you for your valuable comments. We have added a comparison with TESLA, the specific results of which can be found in the main text on lines 256-262 and in Supplementary Fig. S20.

We have placed the additional sections here for your convenience in directly accessing the information.

Furthermore, we incorporated TESLA[14], a tool designed to enhance gene expression matrices to the super-pixel level based on imaging, into our comparative framework for analyzing the tumor microenvironment. When cross-referencing cancer-associated gene lists from original research, TESLA was unable to identify interfaces with the same precision as STAIG. For instance, in slice A (Supplementary Fig. S20a), TESLA missed the interface area highlighted by the yellow frame, whereas STAIG successfully recognized it. In slice B (Supplementary Fig. S20b), STAIG also accurately identified the interface areas. Conversely, TESLA misclassified the interface locations as the core areas of the tumor (the white frame), incorrectly placing them away from the edges, which contradicts the actual observations.

Comments: The authors should add an experiment for discussing the use of different methods (e.g., kernels) other than Euclidean distance for constructing the image distance matrix D on line 311.

Response: Thank you for your valuable comments. We have conducted tests, the details of which are presented in the supplementary material at line 177, under "Methods for generating the image distance matrix." For convenience, we have included the content here for direct access.

To construct the image distance matrix, we tested three methods for calculating the similarity between nodes based on the features extracted from BYOL. We evaluated these methods using the average ARI and NMI values from the DLPFC dataset (Supplementary Fig. S23f). The methods we tested were Euclidean distance, cosine distance, and the RBF kernel. For the RBF kernel, we tested three different values for the gamma parameter: 0.01, 0.1, and 1.

The results showed that Euclidean distance and cosine distance performed similarly, indicating that the embedding differences can be distinguished to some extent based on either distance or direction. However, the performance of the RBF kernel was highly dependent on the choice of the gamma value. When gamma was set to 0.01, the performance was closest to that of Euclidean distance, but there was still a gap. When gamma was set to other values, the performance was worse.

Therefore, in the default construction of the image distance matrix, we chose to use Euclidean distance as the method for measuring the similarity between nodes based on their image features. This choice was based on its simplicity, effectiveness, and consistent performance compared to the other methods tested.

Comments: The authors can include an additional experiment for testing datasets without H&E images.

Response: Thank you for your valuable comments. In the ablation study of the previous version, we compared the ARI box plots for the DLPFC dataset under conditions without image data and with full use of image data. In the latest version, we have added NMI plots (supplementary material, line 283-284). Additionally, we simulated the impact of different edge perturbation probabilities under no-image conditions. Furthermore, we also assessed the effects of using an adaptive method based on gene similarity in the absence of image data (supplementary material, line 248).

Comments: The choice of masked ratio in equation (7) should be discussed, as the quality of the final embeddings hinges on this choice.

Response: Thank you for your valuable comments. We have already addressed this suggestion in a previous response.

Comments: The ability of STAIG in integrating non-vertical and non-horizontal slices (e.g., two slices of the same tissue but from different individuals in different studies, and cannot be horizontally stitched) should be analyzed.

Response: Thank you for your valuable comments. We have expanded our analysis to include integration across Stereo-seq and Slide-seqV2 platforms. Our methodology has been compared with STALigner, and details of these analyses can be found in the line 226-229.

Comments: On line 145, the “(Fig.4f and Supplementary Fig. S7)” appears twice in a row.

Response: Thank you for your valuable comments. We have corrected this duplication in the text in line 160.

Comments: On line 197, the traditional ST methods should be explicitly named.

Response: Thank you for your valuable comments. We have modified in line 250-254. Here is the revised section for your reference.

In simple slice configurations, such as slice A, all comparative algorithms identified the interface (Supplementary Fig. S18). However, when organ complexity increased, algorithms struggled to accurately demarcate these interfaces (Supplementary Fig. S19). Notably, GraphST, conST, DeepST, and Seurat inaccurately included normal muscle tissue within interface cluster boundaries, and while SpaGCN and STAGATE identified larger interface areas, they missed smaller scattered interfaces encircled by the tumor that STAIG accurately aligned with original study results (Fig. 7b).

Comments: In Fig.7f legend, it should be fine-grained instead of "fine-tuned".

Response: Thank you for your valuable comments. We have modified in line 780.

Comments: The H&E image patch segmentation idea is also very similar to HistToGene8, so this paper should be added to reference.

Response: Thank you for your valuable comments. You are correct that our approach to patch segmentation shares similarities with HistToGene. However, HistToGene directly generates gene expressions from images, lacking an intermediary embedding stage for comparative analysis, and is not directly comparable for gene enhancement and spatial clustering tasks. Therefore, we did not include it as a baseline in our initial comparisons. We have now added it to our references, cited on line 345 of the manuscript.

Comments: Another ST computational methods, TESLA7, for studying TME should be referred and compared.

Response: Thank you for your valuable comments. We have addressed this in a previous response. We have included TESLA in our comparisons and discussed it within the text.

Reference:

1. Dong, K. & Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nat Commun* **13**, 1739 (2022).
2. Long, Y. *et al.* Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat Commun* **14**, 1155 (2023).
3. Zhou, X., Dong, K. & Zhang, S. Integrating spatial transcriptomics data across different conditions, technologies and developmental stages. *Nat Comput Sci* **3**, 894–906 (2023).
4. Hu, J. *et al.* SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nat Methods* **18**, 1342–1351 (2021).
5. Zong, Y. *et al. conST: An Interpretable Multi-Modal Contrastive Learning Framework for Spatial Transcriptomics*. http://biorxiv.org/lookup/doi/10.1101/2022.01.14.476408 (2022) doi:10.1101/2022.01.14.476408.
6. Veličković, P. *et al.* Deep Graph Infomax. Preprint at http://arxiv.org/abs/1809.10341 (2018).

7. Zhao, H., Yang, X., Wang, Z., Yang, E. & Deng, C. Graph Debiased Contrastive Learning with Joint Representation Clustering. in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence* 3434–3440 (International Joint Conferences on Artificial Intelligence Organization, Montreal, Canada, 2021). doi:10.24963/ijcai.2021/473.
8. Pham, D. *et al. stLearn: Integrating Spatial Location, Tissue Morphology and Gene Expression to Find Cell Types, Cell-Cell Interactions and Spatial Trajectories within Undissociated Tissues*. http://biorxiv.org/lookup/doi/10.1101/2020.05.31.125658 (2020) doi:10.1101/2020.05.31.125658.
9. Xu, C. *et al.* DeepST: identifying spatial domains in spatial transcriptomics by deep learning. *Nucleic Acids Research* **50**, e131–e131 (2022).
10. Falk, T. *et al.* U-Net: deep learning for cell counting, detection, and morphometry. *Nat Methods* **16**, 67–70 (2019).
11. Grill, J.-B. *et al.* Bootstrap Your Own Latent a New Approach to Self-Supervised Learning. in *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Curran Associates Inc., Red Hook, NY, USA, 2020).
12. Wu, Z., Shen, C. & Van Den Hengel, A. Wider or Deeper: Revisiting the ResNet Model for Visual Recognition. *Pattern Recognition* **90**, 119–133 (2019).
13. Punn, N. S. & Agarwal, S. BT-Unet: A self-supervised learning framework for biomedical image segmentation using barlow twins with U-net models. *Mach Learn* **111**, 4585–4600 (2022).
14. Hu, J. *et al.* Deciphering tumor ecosystems at super resolution from spatial transcriptomics with TESLA. *Cell Systems* **14**, 404-417.e4 (2023).

General Response to Reviewer #2:

We sincerely appreciate your valuable suggestions and feedback. Many of your comments have greatly contributed to the overall evaluation of STAIG, especially the idea of using gene similarity for graph augmentation.

To further demonstrate STAIG's integration capabilities, we have conducted additional experiments and included the results in the revised manuscript. These experiments encompass heterogeneous samples, cross-platform datasets, and the use of the BatchKL and ILISI metrics to validate the elimination of batch effects. The results strongly support STAIG's ability to effectively integrate data from various sources and platforms.

Moreover, to facilitate the reproducibility of our work, we have set up a cloud server that allows you to further verify our results. We have provided detailed instructions on how to access the server and replicate our experiments. This ensures that our findings are transparent and can be easily validated by the research community.

We hope that our revisions and the additional experiments adequately address your concerns. We genuinely appreciate the time and effort you have invested in reviewing our work, and we are grateful for your valuable feedback. Your suggestions have significantly contributed to the improvement of our manuscript and the overall quality of our research. Below, we respond to each comment individually and describe the revisions made to the manuscript. We appreciate the positive feedback on the novelty and performance of our approach as well as the detailed suggestions for improvement.

Detailed Responses to Specific Comments

Comment: According to the method description, the overall spatial graph in STAIG is simply composed of disjoint subgraphs corresponding to individual slices, without any inter-slice connectivity. Consequently, all positive and negative pairs in contrastive learning come from the same slice only. Thus, it is not particularly clear why STAIG is able to achieve batch integration across slices. The examples showcased in the manuscript comprises are confined to slices from the same experiment, where batch effect is presumably minimal. Would STAIG work well with more substantial batch effects, e.g., those that may arise from different ST technologies?

Response: Thank you for your valuable comments. While it is true that the slices in STAIG are disjoint in the adjacency matrix during the initial construction stage, there is indeed information exchange between them during the training process.

Firstly, let's consider STAIG's neighbor contrastive loss function (9) (Line 460). In this equation, the numerator represents the computation of similarities between a sample and its immediate neighbors, treated as positive pairs within the augmented graph. Conversely, the denominator accounts for the interactions with negative samples across the entire samples (The second term of the denominator). This global computation not only normalizes the impact of local interactions but also indirectly bridges different datasets by adjusting each data point's influence in relation to the overarching data characteristics. Therefore, although direct dataset interactions are not modeled by the adjacency

matrix A, the denominator plays a crucial role in facilitating an indirect information exchange, thereby significantly enhancing the model's capacity to generalize across diverse datasets.

$$l\left(h_i^{(1)}\right) = -log\frac{\left(f^{1,2}(i,i) + \sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j) + \sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)\right)/\widehat{\mathcal{N}}_i}{f^{1,2}(i,i) + \sum_{(j \neq i) \cap \left(y_j \neq y_i\right)} \left(f^{1,1}(i,j) + f^{1,2}(i,j)\right)}, \quad (9)$$

Secondly, STAIG adopts a novel approach by concentrating on local patterns during edge perturbations. This strategy emphasizes subtle local dynamics over broad global changes to capture consistent expression patterns that are prevalent across different sections, even in the presence of varying batch effects. This allows STAIG's embeddings to neutralize batch effects effectively ensuring that the model's output remains biologically relevant and consistent, irrespective of variations in data processing or experimental conditions across different datasets.

Thirdly, the weights and dimensionality reduction of samples are shared by placing each slice diagonally in the adjacency matrix, allowing for information among exchange among samples during training.

From an experimental validation perspective, to further verify our integration capability, we conducted experiments on heterogeneous samples and cross-platform datasets, and we used the BatchKL and ILISI to validate whether batch effects were eliminated. First, we integrated slices from two different donors in the DLPFC dataset. Slices #151507 and #151508 came from one donor, while #151675 and #151676 came from another. The experimental results showed complete integration on the UMAP, and the BatchKL had no significant difference compared to other integration methods. Next, we experimented on the visual cortex datasets of two mice from MERFISH, and our method also achieved complete integration. Finally, we accomplished the integration of datasets from the Slide-seqV2 and Stereo-seq platforms. The specific experimental results are detailed in the "Integrating slices without prior spatial alignment" section starting from line 182 of the main text.

Comments: STAIG employs a band-pass filter to mitigate noise and uneven staining in the histological images. Is it necessary to adjust the band-pass filter when applied to different datasets? Furthermore, how much does this filter step contribute to the overall performance?

Response: Thank you for your valuable comments. The band-pass filter parameters are related to the size of the image. Since we rescale all images to the same size before input, there is no need to adjust the parameters on other datasets. We evaluated and found the optimal parameters on both brain tissue and cancer slices, and both can achieve good results under the same parameters. Our selection of band-pass filter parameters is detailed in the supplementary materials, line 338, "Bandpass filter parameters selection." Our parameters are applicable to both brain tissue and cancer slices. The band-pass filter actually serves to highlight and emphasize, helping the subsequent deep model reduce its focus on noise. Our ablation experiments on the filter step are in the supplementary materials, line 362 "BYOL ablation study for histology feature extraction." Here, we briefly describe the experimental setup and the results obtained. We also mentioned the

parameters in manuscript line 350: The default parameters for the band-pass were 245 – 275.

Experimental Setup

To assess the impact of BYOL and image preprocessing, we conducted experiments with seven different models and two types of input images. The models are as follows:

1. U-Net trained on the original CPM15 () dataset (supervised)
2. U-Net trained on preprocessed CPM15 images (supervised)
3. Initial imported ResNet50
4. ResNet50 after self-supervised training with BYOL on filtered images (complete STAIG framework)
5. ViT with weights from a pre-trained network
6. ViT with weights from a pre-trained network, self-trained on original images
7. ViT with weights from a pre-trained network, self-trained on preprocessed images

The two types of input images used in the experiments are:

1. Original histology images
2. Preprocessed histology images (filtered)

From the experimental results, the original images from the brain tissue DLPFC dataset could not be effectively recognized by any of the models. After preprocessing, the BYOL framework achieved the best recognition results. Our filter step also provided some help for other models; after filtering, the recognition effect of the U-Net model was slightly improved. From the ablation experiments, both the BYOL and filter steps contribute to the effective extraction of the final image features. You can directly refer to Fig.S29 to view the results.

In summary, the band-pass filter parameters do not need to be adjusted for different datasets due to our image preprocessing step. The filter step plays a crucial role in highlighting important information and reducing noise, which benefits not only STAIG but also other models. The ablation study demonstrates that both the BYOL framework and the filter step contribute to the superior performance of STAIG in extracting effective image features. Please refer to the supplementary materials for detailed experimental results and analysis.

Comments: It was mentioned in the method section that 512×512 pixel patches were extracted when partitioning the H&E images. Does that correspond to the physical dimension of the ST spots? How would STAIG adapt to ST data and H&E images with alternative spatial resolutions?

Response: Thank you for your valuable comments. In STAIG, the choice of patch size is meticulously calibrated to align with the inherent properties of the ST dataset being analyzed. Specifically, the size of the patches, which we initially mentioned as 512×512 pixels, is determined based on a dataset-inherent parameter known as the fiducial diameter fullres (denoted as *d*), which represents the pixel diameter of a fiducial spot in the full-resolution image.

We have updated the description in the method section of our paper (lines 345-348) to clarify that

the patch size is set to 3.5 times *d*, rather than a fixed 512×512 pixels: The size of the patches was based on a dataset-specific parameter, $fiducial\_diameter\_fullres$ (denoted as d), which is the pixel diameter of a fiducial spot in the full-resolution image. According to our tests, the default patch size was set to $3.5d$ (see Supplementary Material, Patch size selection).

We have explored the optimal multiple of *d* for patch size using two datasets: DLPFC and human breast cancer slices (Supplementary Materials, line 296, "Patch size selection").

We have placed the supplementary material here for your convenience to read directly.

We assessed the impact of patch size using slices #151507 and #151673 from the DLPFC dataset and a dataset from human breast cancer—covering the two most common scenarios: brain and cancer tissues. The patch sizes tested were multiples of *d*: 2*d*, 3*d*, 3.5*d*, and 4*d*.

With existing class labels, we overlaid contours on the results for a clearer comparison. As shown in Supplementary Fig. S28a, smaller patch sizes resulted in scattered clustering of image features with no distinct demarcation between tissue contours. For slices #151673 and #151507, the models fixated on color differences rather than textural ones, mainly dividing the images into left and right sections due to insufficient information from smaller patches, hindering the learning of classifiable features. At a 3*d* size, #151673 and the human breast cancer datasets began to discern textures, aligning with the reference labels; however, #151507 was still disrupted by staining artifacts. At 3.5*d*, the clustering of image features corresponded to manual labels across all three datasets, with #151507's layer 1 and #151673's layers 3 and 4, along with the WM sections, closely matching. The human breast cancer imagery adeptly separated different cancer regions. However, increasing to 4*d* improved the breast cancer clustering but blurred the brain imagery results due to limited textural details in brain slices that could not be further exploited with larger image sizes.

Considering the performance across both datasets, we decided to set the optimal patch size at 3.5 times the diameter *d*, balancing detail capture and clarity of clustering.

In summary, by setting the patch size as a multiple of d, STAIG automatically adapts to the specific spatial characteristics of each ST dataset.

Comment: When no image is available, STAIG conducts graph augmentation by removing edges at a fixed probability. Would it improve performance if the removal probabilities were determined based on transcriptome similarity between the spots?

Response: Thank you for your valuable suggestion. We have explored the idea of determining edge removal probabilities based on transcriptome similarity between spots and compared the results with the fixed probability approach. In the supplementary materials line 248, under the section "Comparing adaptive edge perturbation based on gene similarity with random perturbation," we present our findings.

We also revised in paper line 413-423: In scenarios lacking image data, STAIG offers two methods of edge removal: fixed edge removal with a set probability, and adaptive edge removal based on gene similarity. For fixed edge removal, STAIG's default parameter p is set to 0.2. When performing adaptive edge removal based on gene similarity, we proceed as follows: PCA is used to reduce the preprocessed gene expression data of each spot to 64 dimensions, and this reduced data is then used to compute the cosine similarity between spots.

The matrix of cosine similarities between spots is directly used as the edge weight matrix W in the computation of Equation (4). For a comparison of the effects of random edge deletion and adaptive edge deletion based on gene similarity, as well as the choice of distance formula for gene similarity, see the supplementary material 'Comparing Adaptive Edge Perturbation Based on Gene Similarity with Random Perturbation.' In the paper, we conducted tests using random edge deletion.

We have placed the supplementary material here for your convenience to read directly.

In the absence of imaging data, alongside random perturbation, we also tested an edge deletion strategy based on gene similarity to reduce parameter tuning. Specifically, we first reduced the dimensionality of the nodes' original gene data to 64 dimensions using PCA. As with image feature processing, we constructed a similarity matrix by calculating node-to-node similarities based on the reduced vectors and cosine similarity. This similarity was then translated into a probability of edge deletion, analogous to image feature distance, followed by adaptive neighbor-edge random deletion based on gene similarity.

We employed the average ARI and NMI from the DLPFC dataset to assess the effects of random versus gene-similarity-based adaptive edge deletion on spatial clustering when images are unavailable (Supplementary Fig. S27d). We set the random deletion probability, p, and investigated values ranging from 0.1 to 0.5. Our simulations revealed that $p = 0.2$ delivered optimal results. Increasing p to 0.4 led to a significant decline in both ARI and NMI averages, indicating that excessive disruption of the graph structure and information loss outweigh the benefits of perturbation.

The gene cosine similarity-based adaptive edge perturbation closely matched the results at $p = 0.2$ and outperformed other values. Although changing the similarity calculation method resulted in slight declines, these were minimal. These findings suggest that while gene similarity does not surpass the peak performance of random perturbation, its results are more stable, making it a preferable choice before finalizing p values.

Subsequent tests on the Stereo-seq platform and Slide-seqV2 mouse brain olfactory datasets (Supplementary Fig. S27e-f) again indicated that $p = 0.2$ performed best, achieving the highest SC and lowest DB. These outcomes closely paralleled those obtained with gene cosine similarity. However, from $p = 0.3$ onwards, both SC and DB metrics began to deteriorate, showing no significant differences at lower values. When comparing the clustering results of cosine similarity of

genes to those obtained with $p = 0.2$, consistent clustering was observed in Stereo-seq's layers. In contrast, Slide-seqV2 exhibited variations, particularly clustering the accessory olfactory bulb (AOB) region and the glomerular layer (GL) together, likely due to shared gene expressions such as the CCK gene in the GL layer (Supplementary Fig. S27g). This suggests that gene similarity-based adaptive deletion may lead to biases from specific genes, although random perturbation helps mitigate this by uniformly distributing edge deletion probabilities across iterations.

In conclusion, across different platforms, the optimal probability for random edge perturbation was found to be $p = 0.2$. Adaptive perturbation based on gene similarity, while potentially introducing biases on high-resolution platforms like Slide-seqV2, remains the preferable choice for initial experiments due to its adaptiveness and consistent performance across most platforms, reducing the need for extensive parameter tuning.

Comments: The hyperparameter optimization was conducted on the DLPFC dataset only, which raises questions regarding its generalizability. Does the optimal hyperparameter setting hold in other datasets, or when imaging data is not available?

Response: Thank you for your valuable comments. To address the generalizability of the hyperparameter optimization, we have conducted further tests on other platforms. We evaluated the performance using the Silhouette Coefficient (SC) and Davies-Bouldin Index (DB) clustering metrics (a new clustering evaluation metric we introduced) on the Stereo-seq and MERFISH platforms. Additionally, we tested the performance using the ARI and NMI on the STARmap platform. The results can be found in the supplementary materials, lines 194-231.

For each tested platform, we searched for an optimal set of hyperparameters. In general, the difference between the optimal parameters of each platform is mainly layer number, and the other parameters are relatively stable. The selection range of layer number is 1 or 2. In the actual use process, we recommend choosing according to the actual situation.

We have updated in the paper at line 489.

Comments: It was mentioned in the method section that the clustering results in DLPFC were additionally smoothed for the STAIG method. It is not particularly clear whether the same procedure was applied for other benchmarked methods, which could have noticeable impact on the evaluation metrics. Based on the visualization in Fig. 2e, it seems that the clusters are spatially scattered for stLearn and Seurat, suggesting that they might not have been smoothed?

Response: Thank you for your valuable comments. You are correct; in the initial version of our manuscript, we did not apply the refinement procedure to the clustering results of stLearn and Seurat, which could have affected the evaluation metrics. We acknowledge that this inconsistency in post-processing may have led to an unfair comparison between STAIG and the benchmarked methods.

To address this issue, we have now applied the same refinement procedure to the clustering results of all the compared algorithms, including stLearn and Seurat. The refinement process involves smoothing the cluster assignments based on spatial proximity, which helps to reduce the spatial scattering of clusters and improve the overall coherence of the results.

After applying the refinement procedure, we have updated the corresponding ARI and NMI scores, as well as the box plots comparing the performance of different methods. The updated results show that STAIG still achieves the highest accuracy among the compared methods, even after ensuring a fair comparison by applying consistent post-processing steps.

Specifically, we have updated the following figures and lines in the manuscript: Fig-2a,e, Line 105-106: GraphST achieved an ARI of 0.64 and an NMI of 0.73 but had discrepancies in the positioning of layers 4 and 5; other methods recorded ARIs between 0.25 and 0.55, with corresponding NMIs ranging from 0.42 to 0.69, owing to inaccuracies in layer proportions.

In Supplementary Meterials: FigS1, S2, Lines 18-19: To ensure fairness in the comparison results, we uniformly adopted the same refinement method for the DLPFC dataset.

Comment: In the "image feature extraction" section, the authors claimed that STAIG clusterings based on image features highly conforms with manually annotated regions. However, such conformity is not visually apparent in Fig. 3. E.g., the demarcation between WM and layer 6 doesn't seem clear (Fig. 3a). It would be helpful if the authors highlight the mentioned demarcation lines in each sub-panel of Fig. 3 to ease comparison across panels.

Response: Thank you for your valuable comments. To address this issue and facilitate easier comparison across panels, we have made the following improvements to Fig. 3:

For the DLPFC dataset (Fig. 3a-b), we have overlaid the demarcation lines between different cortical layers onto our clustering results. Additionally, we have explicitly labeled the identified layers in our results to allow for a more direct comparison with the manual annotations and histological images. This should help readers clearly see the correspondence between our clusterings and the cortical layers.
For the human breast cancer slides (Fig. 3c), we have overlaid the outlines of each manually annotated region onto our clustering results. This enables a more straightforward visual comparison of the effectiveness of our image features in capturing the different tissue regions.

Your comment correctly highlights a key issue: STAIG does indeed exhibit some blurriness at certain boundaries, which is attributed to the difficulty of brain histology. Despite these challenges, STAIG stands out among current mainstream comparative methods. According to the ablation experiments compared in our supplementary materials, STAIG's results uniquely align with the manually annotated layering trends (in the supplementary materials, line 362). Although some boundary areas

remain unclear, the overall clustering accuracy benefits significantly. This allows us to remove some false negatives during the comparison process—those samples that belong to the same class but are incorrectly treated as negatives—thus improving clustering outcomes. As shown in Figure 3c, when the texture of the image becomes richer, STAIG's edges are very sharp, and its clustering of regions is also remarkably precise compared to our added contour lines.

Furthermore, we have updated the text describing the results and their correspondence with the manual annotations to provide a clearer and more accurate interpretation of our findings: Line 122-138:

Comments: The authors identified a tumor-like interface and a muscle-like interface in zebrafish melanoma. As the spots in the ST slices correspond to multiple cells, is this partition due to different cell type proportions per spot or genuine difference in single-cell gene expression?

Response: Thank you for your valuable comments. In the interface area between the tumor and normal tissues, according to the results of the H&E staining described in the original article of dataset (Fig.1 a), the cellular morphology in this interface area is consistent with muscle cells, indistinguishable from the surrounding muscle cells, and no visible tumor cells are present in this area. However, the transcriptional profile of the interface area shows a stronger correlation with the tumor (R = 0.33) than with muscle (R = 0.06) (Fig.1 b).

We therefore suggest that the muscle-like and tumor-like clusters we identified in the interface area are not due to different proportions of cell types in each spot, as the cells in this region morphologically consistent with muscle cell rather than tumor cells. Instead, our results are due to differences in transcriptional profiles between these two clusters.

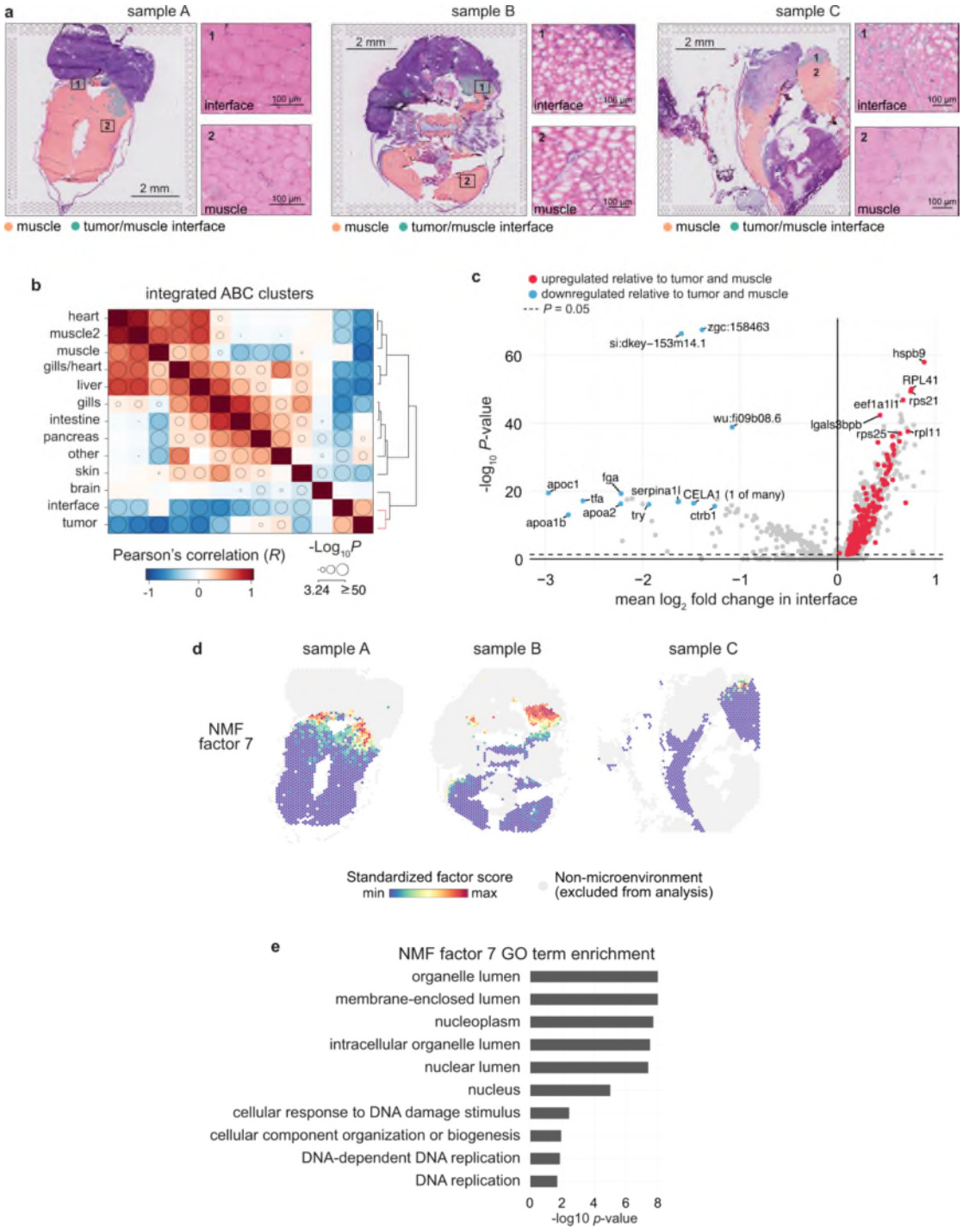


Fig. 1 Original study result from Hunter et al.[1]

***Comments**:* Most use cases in the manuscript were based on spatial barcoding technologies, which offer relatively high numbers of detected genes. Would STAIG work well for imaging-based technologies as well, e.g., MERFISH, seqFISH?

Response: Thank you for your valuable comments. We applied STAIG to two datasets from the MERFISH platform: one featuring 4000 genes in the human middle temporal gyrus (MTG) and another with 243 genes in the mouse visual cortex (VIS). The analysis incorporated both the manually annotated cellular classes and layer-specific labels from the original research (Fig. 4h). In the human MTG dataset, STAIG not only achieved the highest SC of 0.28 and the lowest DB of 1.26 compared to other algorithms but was also the only method that distinctly identified each layer (Fig. 4h and Supplementary Fig. S9). Other algorithms failed to clearly delineate the layers, with many clustering various layers together. Notably, although STAGATE performed relatively well, it still confused layers L4, L5, and L6. In the mouse VIS dataset, STAIG continued to outperform all other methods in terms of metric scores (SC:0.32, DB:1.09) (Fig. 4h and Supplementary Fig. S10). The clustering results from STAIG were nearly identical to the real manual annotations. Another algorithm that showed significant layer differentiation was GraphST; however, it incorrectly classified some samples from layer L1 as belonging to layer L4. (Paper line 171-180)

Comment: It was mentioned in the introduction section that "recent graph-based approaches for ST data are flawed" due to "artificially structured graphs". This is a rather bold claim, and worth more detailed explanation as to why the graphs can be "artificially structured"?

Response: Thank you for your valuable comments. Indeed, our original description was not accurate enough. We have updated the introduction of this part in the Introduction section.
(From line 28-41)
In contrast, emerging spatial clustering methods integrate genetic and spatial information using graph convolutional models. For instance, SEDR[2] employs a deep autoencoder alongside a variational graph autoencoder to map gene data. However, this class of methods may have potential pitfalls. Specifically, the transformation of ST data into graph structures primarily relies on artificially defined distance criteria, which can introduce biases. These biases arise from the uniform construction of edges between regions that belong to different biological domains but meet the arbitrary criteria.

Although some methods like STAGATE[3] address this issue by assigning weights to edges based on cell gene similarity, they do not modify the graph structure during training, which remains unchanged from its initial setup. This inflexibility prevents the elimination of biases introduced during the initial construction. Similarly, other methods such as GraphST[4] and ConST[5] utilize contrastive learning frameworks that preserve the original structure and involve random shuffling of nodes during training, leaving the structure intact throughout the process. Concurrently, these methods fail to consider the exclusion of potentially similar nodes from the negative samples during comparisons, resulting in biases from false negatives impacting the outcomes.

Comments: The training details of BYOL is not sufficiently explained. Was the BYOL model trained on a per-dataset basis, or pre-trained on a larger corpus of histological images?

Response: Thank you for your valuable comments. During the initialization phase, BYOL uses the

ResNet50 weights provided by the official PyTorch repository. In the subsequent stages, no additional pre-training is performed. After initialization, the model is directly used for task-oriented training.
We have updated the description in the article accordingly. (Line 360-361):
During the initialization phase of BYOL, we employed the official PyTorch-pretrained ResNet50 weights as the initial model, without any further fine-tuning on additional datasets.

Comments: Canonical BYOL uses the online network as the final network, but line 288 states that the "target network" was used after BYOL training. Is that a typo?

Response: Thank you for your valuable comments. You are correct; it was a typo in our manuscript. In line 368, it should state that the "online network" was used after BYOL training, not the "target network." We have now corrected this mistake in the revised version of the manuscript.

Comments: ARI or SC scores are not shown for Fig. S11, despite being shown for all other relevant figures.

Response: Thank you for your valuable comments. We apologize for the inconsistency in presenting the evaluation metrics across the supplementary figures. We have now added the ARI and SC scores to Fig. S12 (previously Fig. S11) to ensure consistency with the other relevant figures.

Comments: Fig. 2F and S3 differ by a 90 degree rotation, which is unnecessary.

Response: Thank you for your valuable comments. To maintain consistency and avoid unnecessary confusion, we have now adjusted the orientation of Fig. S3 to match that of Fig. 2F, ensuring that both figures are aligned in the same direction.

Comment: The superscript notations for augmented graphs are not consistent. Most occurrences contained no parentheses, but line 347–352 used parentheses.
Response: Thank you for your valuable comments. We apologize for the lack of consistency in the superscript notations for augmented graphs throughout the manuscript. To maintain clarity and uniformity, we have now removed the parentheses from the superscript notations in lines 347-352 (Now line 434-467), ensuring that all occurrences of the augmented graph notations are consistent and do not include parentheses.

Comments: The figure numbers in Table S1 are incorrect. E.g. there is no Fig. 3f (last row in Table S1) in the manuscript.

Response: Thank you for your valuable comments. We apologize for the inaccuracies in the figure references within the table. We have now thoroughly reviewed and corrected the table to ensure that all figure numbers mentioned in Table S1 correspond to the actual figures present in the manuscript.

Comments: Brightness in Fig. 6e bottom panels are not properly normalized.

Response: Thank you for your valuable comments. The revised Fig. 6e now features bottom panels with appropriately normalized brightness, allowing for clearer visualization and comparison of the results.

Comments: The authors have released `STAIG` as a Github repo, and provided necessary instructions on environment management and usage. The codes seem well structured and the tutorial runs without error, except that some comments and print messages are written in Chinese rather than English.

Response: Thank you for your valuable comments. We apologize for any issues on the implementation of STAIG. We have now updated the repository to ensure that all comments and print messages are written in English.

Furthermore, to facilitate the reproducibility of our experiments, we have rented a GPU on a cloud platform and replicated our experiments on this server. We have provided detailed instructions in the repository on how to remotely connect to the server and run our code using Jupyter Notebook (ipynb) to verify the results.

Reference:

1. Hunter, M. V., Moncada, R., Weiss, J. M., Yanai, I. & White, R. M. Spatially resolved transcriptomics reveals the architecture of the tumor-microenvironment interface. *Nat Commun* 12, 6278 (2021).
2. Fu, H. *et al. Unsupervised Spatially Embedded Deep Representation of Spatial Transcriptomics*. http://biorxiv.org/lookup/doi/10.1101/2021.06.15.448542 (2021) doi:10.1101/2021.06.15.448542.
3. Dong, K. & Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nat Commun* 13, 1739 (2022).
4. Long, Y. *et al.* Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat Commun* 14, 1155 (2023).
5. Zong, Y. *et al. conST: An Interpretable Multi-Modal Contrastive Learning Framework for Spatial Transcriptomics*. http://biorxiv.org/lookup/doi/10.1101/2022.01.14.476408 (2022) doi:10.1101/2022.01.14.476408.

General Response to Reviewer #3:

We greatly appreciate your thorough review and constructive suggestions for improving our manuscript. We have carefully considered each of your comments and have made significant revisions to address your concerns and enhance the overall quality of our work.

Specifically, we have made the following major changes:

(1) **Introduction:** We have expanded the introduction section to provide a more comprehensive overview of the current state-of-the-art methods and their limitations. We have also highlighted the challenging objectives that STAIG aims to achieve, emphasizing the novelty and significance of our approach.

(2) **Figures:** We have redesigned the layout of the figures to improve readability. The legends and the font sizes within the legends have been enlarged, and we have added numerical scales to the gene expression strips to facilitate quantitative comparisons.

(3) **Reproducibility:** To validate the authenticity and reliability of our experiments, we have taken two proactive measures. First, we have provided Jupyter Notebook (ipynb) files that document the training process and the corresponding outputs. These notebooks offer a detailed, step-by-step record of our experiments. Second, to facilitate the verification of our code, we have rented a GPU-equipped cloud platform. We have prepared a tutorial that guides you through the process of remotely accessing the server and reproducing our experimental results.

We hope that our revisions and the additional experiments adequately address your concerns. We genuinely appreciate the time and effort you have invested in reviewing our work, and we are grateful for your valuable feedback. Your suggestions have significantly contributed to the improvement of our manuscript and the overall quality of our research. Below, we respond to each comment individually and describe the revisions made to the manuscript. We appreciate the positive feedback on the novelty and performance of our approach as well as the detailed suggestions for improvement.

Detailed Responses to Specific Comments

Comment: The introduction quickly cites several competing approaches while providing minor information, being a bit distracting. It could be improved by providing additional details and how they differ from the proposed approach.

Response: Thank you for your valuable comments. We have revised and expanded the introduction based on your suggestions. In the updated Introduction, we have highlighted the limitations of current methods in the initial graph construction process, the challenges faced in image feature extraction, and the potential for improvement in batch integration. Finally, we have pointed out the specific issues that our proposed method aims to address. The revised content can be found in the main text, lines 20-63.

Comments: The methods and results are clearly described. Figures with small legends are hard to read, even digitally, due to the rasterization of text.

Response: Thank you for your valuable comments. To address the concern regarding the small legends in the figures, we have redesigned the layout of the figures and increased the size of the legends. This should improve the readability of the figures. Specifically, we increased the size of legend in Fig. 2g, Fig. 3c, Fig. 4d, 4g, Fig. 6a Fig.7a. In addition to these specific changes, we have also tried to enlarge the overall size of the figures where possible while maintaining the proper formatting of the manuscript. This should further enhance the visual clarity and readability of the figures.

Comments: Line 276: What are the parameters of the Gaussian filter and the band-pass filter? The authors suggest these filtering steps are very important. A study showing how they impact the algorithm's performance would be extremely helpful for researchers building upon this work and could be included in the ablation studies section.

Response: Thank you for your valuable comments. We have conducted additional experiments to address your concerns regarding the Gaussian filter and bandpass filter parameters and their impact on the algorithm's performance. The Gaussian filter kernel is 7x7 and the band-pass filter is 245-275. The results of these experiments are now included in the supplementary material. At the same time, we specify the default parameter in line 344 and 350 of the main text.

- Gaussian Blur Parameter Selection (Details in Supplementary Material, Line 320-336):

We tested different kernel sizes (3x3, 7x7, and 15x15) for the Gaussian blur on brain and breast cancer tissue images. The results showed that a 7x7 kernel size provided the best balance between detail preservation and noise reduction. Smaller kernel sizes (3x3) did not sufficiently smooth out the noise, while larger kernel sizes (15x15) led to excessive loss of detail. Therefore, we have set the default Gaussian blur kernel size to 7x7.

- Bandpass Filter Parameters Selection (Details in Supplementary Material, Line 338-359):

We evaluated the effect of the bandpass filter's frequency band on the final outcomes using brain and breast cancer tissue images. We found that broader frequency bands (e.g., 0-300, 200-300, or 200-512) did not effectively eliminate noise, while excessively narrow bands (e.g., 250-270) caused the disappearance of detailed sub-clusters. The optimal frequency band was determined to be 245-275, which provided a balance between noise filtration and information preservation. Since we consistently fix the image size before input, this frequency band offers relatively consistent noise filtration and broad applicability.

- BYOL Ablation Study for Histology Feature Extraction (Details in Supplementary Material, Line 362):

We conducted a comprehensive ablation study to delineate the individual effects of image preprocessing and the BYOL framework on feature extraction from histopathological images. We compared the performance of BYOL with other models, including pretrained ResNet50, U-Net (trained on original and bandpass-filtered images), and Vision Transformer (VIT) in various states (pretrained, fine-tuned on original images, and fine-tuned on bandpass-filtered images).

The results showed that BYOL demonstrated superior performance when using bandpass-filtered images, with sharper feature outlines compared to the baseline ResNet50. The bandpass filtering step effectively highlighted valuable information, enhancing feature extraction even for other models. The self-supervised training of BYOL proved to be efficacious, substantially elevating the quality of the extracted features compared to the initial pretrained ResNet50.

Comments: Line 314: Why is the log plus one transform applied to equation (3) if it's being transformed by e^x on equation (4)? And why is it necessary to do the element-wise product with A in equation (3) if equation (4) is conditioned to A_ij == 1

Response: Regarding the logarithmic transformation in Equation (3), we apply this transformation to the image spatial distance matrix D^{img} to modulate the distance values. By taking the logarithm of $\left(D^{img}+1\right)$, we compress the range of distance values, giving more emphasis to smaller distances and reducing the impact of larger distances. This transformation helps in capturing the local spatial relationships between spots in the image feature space.

The addition of 1 to D^{img} before taking the logarithm is to prevent taking the logarithm of zero, which is undefined. This is a common practice when dealing with non-negative values that might include zeros.

In our initial stages of development, we directly used the SoftMax function (Equation (4)) for probability transformation without applying the logarithmic transformation. However, upon investigation, we discovered that points with large distances caused significant probability imbalances. By applying the logarithmic transformation, we smooth out these large distances, mitigating the issue of probability imbalance.

Regarding your second point about the element-wise multiplication with the adjacency matrix A in Equation (3), you are absolutely correct. Since the denominator in Equation (4) explicitly ensures that only neighboring edges participate in the calculation, the element-wise multiplication with A in Equation (3) is indeed unnecessary.

We have made the necessary modifications to remove this redundant operation in Line 398.

Comments: A scale with numbers in Figures 4f, 6e, and 7f would improve the evaluation of the different gene expressions in the different regions.

Response: Thank you for your valuable comment. we have now added scales with numbers to Figures 4f, 6e, and 7f. These scales provide a clear indication of the range and magnitude of gene expression levels in the different regions depicted in the figures.

Comments: Figure 6e: It's unclear how appropriate it is to apply a statistical test to compare the

mean value of groups that, by definition, have different averages, since they represent a mixture of Gaussians. Moreover, some of the test assumptions are violated. For example, the cells are not independent because they have a spatial correlation. I suggest removing the p-values.

Response:
Thank you for your valuable comment. We agree that comparing means of groups representing a mixture of Gaussians and the violation of independence assumptions due to spatial correlation make the use of p-values inappropriate in this context.

We have now removed the p-values from Figure 6e to avoid potential misinterpretation.

Comments: Creating a Python package and additional documentation could improve the code repository. I did not run the code to confirm its reproducibility.

Response:
Thank you for your valuable comment. We appreciate your suggestion to create a Python package and provide additional documentation to improve the usability and accessibility of our code.

We acknowledge that installing and running the Python program locally can be complex. While we are currently developing a pip-based package, we have taken two alternative approaches to help you verify our program results:

We have provided the Jupyter Notebook (ipynb) files used during training, which clearly document the process of running our program. These notebooks offer a step-by-step guide to understand and reproduce our results.

For the duration of the review process, we have rented a GPU-equipped cloud platform with a pre-configured environment. Following the guidelines we have provided, you should be able to validate our results on this server.

We understand the importance of reproducibility and are committed to ensuring that our code and results can be easily verified. By providing the Jupyter Notebooks and access to a pre-configured cloud platform, we aim to facilitate the process of reproducing our findings.

NCOMMS-24-00513B

STAIG: Spatial Transcriptomics Analysis via Image-Aided Graph Contrastive Learning for Domain Exploration and Alignment-Free Integration

Yitao Yang, Yang Cui, Xin Zeng, Yubo Zhang, Martin Loza, Sung-Joon Park, Kenta Nakai

Responses to the reviewers

We sincerely appreciate the thoughtful suggestions and insights you have provided. Your feedback has significantly enhanced the quality and rigor of our work. In this revision, our main focus was to validate STAIG's integration capability through ablation experiments. Based on our experimental results, STAIG's integration ability primarily stems from its contrastive framework, which differs from DGI, and the random gene mask during training. Through ablation experiments, we discovered that restricting negative samples within the same slice leads to better integration effects. Consequently, we have updated the description of the loss function in the paper, providing more details. We have also updated the corresponding section with the latest integrated experimental results, showing improvements in all metrics compared to the previous version.

Furthermore, in response to concerns about the reproducibility of our experimental results, we have followed Reviewer 1's suggestion and improved the stability of our code. Our STAIG model achieves stable results across different GPU devices in our local test. We have also provided access to a cloud server to validate the experimental results presented in this revision.

We hope that our amended manuscript fully addresses all the critiques and comments raised by the reviewers. Please find below our point-by-point responses to each comment.

General Response to Reviewer #1:

We sincerely appreciate your continued guidance and insightful feedback.

In this revised version, we have optimized the algorithm and its stability. We have also addressed your concerns specifically. The effectiveness of STAIG has been validated through ablation studies, which have further enhanced its integration performance.

We hope that our revisions and the additional experiments sufficiently address your concerns. We are truly grateful for the time and effort you have invested in reviewing our work. Your feedback has been invaluable in improving our study.

Detailed Responses to Specific Comments:

Comment: Questionable reproducibility and local performance of STAIG. I appreciate the authors' provision of a pre-configured cloud server, which facilitates the verification of STAIG's performance. However, I remain concerned about its reproducibility and performance consistency across different environments. Our tests on a local server demonstrated significant discrepancies compared to the results reported in the manuscript (refer to the attached results in the reviewer's response email). This variation might stem from inherent limitations in STAIG's methodology, particularly the probabilistic perturbation in the graph's local structure during contrastive learning, which results in

highly variable augmented graphs G1 and G2 in each training instance. To mitigate this, I suggest resampling G1 and G2 multiple times and using their average structure, although this may increase training complexity and time. I recommend other reviewers also test STAIG on their local machines to verify STAIG's reproducibility at the user end.

Response: Thank you for your constructive feedback on our methodology. We have updated the STAIG code to optimize its stability. First, we have incorporated your suggestion of resampling G1 and G2 multiple times and using their average structure. Second, we have extensively tested STAIG on a variety of mainstream GPUs. Initially, our tests were limited to GPUs representing the Ampere architecture (sm_86), such as the RTX 3090, where we achieved reproducible and stable results. However, when we conducted tests on GPUs from the Ada (Lovelace) architecture, such as the 40-series, using our original setup instructions, we indeed observed a decrease in performance. This issue stemmed from an incompatibility with the CUDA version used; our original code utilized CUDA 11.7, while the 40-series GPUs require CUDA 11.8 or higher. This incompatibility led to a decrease in computational precision in PyTorch.

We have redesigned STAIG's installation process, employing updated versions of PyTorch and CUDA, which have improved computational precision. Although this update results in a slight decrease in computational speed, it ensures stable reproducibility across different GPU models.

Following the updated installation guide and code, you should be able to replicate our results locally as consistently as those obtained on our cloud server.

Comment: Theoretically suspicious batch correction
The approach to contrasting learning described in Equation 9 permits interactions across batches but may not effectively mitigate batch effects. Specifically, in equation 9, positive samples are all from the same batch, while samples from different batches only serve as negative samples in the denominator. Theoretically, batch effects can help minimize this loss function—its loss declines when the dissimilarity of such negative samples are increased, which the existence of batch effects can contribute to. In other words, rather than distancing cross-batch negative samples, batch effects are effectively addressed only when pulling closer cross-batch positive samples, which STAIG fails to achieve, at least theoretically.

Response: Thank you for your comment. To evaluate STAIG's capability for spatial integration, we conducted comparative studies with other contrastive learning frameworks and performed detailed ablation studies on various components of the loss function. The results of these experiments can be found in the supplementary material "Ablation Studies for Integration" (lines 435-495).

Based on these results, we can draw several conclusions. First, it's evident that samples from different individuals and platforms exhibit significant batch effects in their raw data. Second, after standard data preprocessing, the contrastive architectures of GraphST[1] and ConST[2] (representing DGI[3]) only achieved minor integration of data on the Visium platform and failed to integrate data

across platforms. In contrast, STAIG demonstrated robust integration capabilities both visually (UMAP) and through performance metrics, indicating that its strength primarily derives from a different contrastive framework. Further ablation studies on STAIG's loss function components showed that random gene masking and positive samples from first-order neighbors enhanced integration capability. However, gene masking was not the main factor, as DGI could not achieve cross-platform integration with or without gene masking, and was minimally influenced by the masking.

You are correct that negative samples across slices actually reduced integration capability. Restricting negative sample pairs to the same slice significantly improved integration. This does not imply that our training process is equivalent to inputting individual datasets into a GCN one by one, as the same process could not integrate data using DGI's architecture.

Based on our findings, we believe STAIG's integration capability stems from its node-by-node contrastive learning framework, which can effectively learn commonalities in gene expression. Interestingly, a recent study by Zhang et al[4]. supports our observations. They argue that while DGI relies solely on spatial dependencies, their InfoNCE[5] framework—a contrastive pattern similar to ours—can adaptively learn both spatial dependencies and common gene expression traits. Their conclusions align with our results, though they did not further explore integration. We have included their method in our comparative analysis.

Following the insights from the ablation studies, we restricted negative sample pairs to the same slice during batch integration, updating our original integration results (Line 180-227, Fig.5, Supplementary Fig.S11-15). Compared to the previous version, our current integration metrics have significantly improved. We have also updated the descriptions in the Methods section accordingly (Line 386-387, 464-469).

Comment: Insufficient task and methodological novelty

The novelty of STAIG is still questionable from both task-oriented and methodological perspectives. The tasks of spatial clustering and slice alignment are well-trodden areas, while the ideas of applying graph convolutional networks, histology image integration, and graph contrastive learning to spatial clustering have been previously explored in other studies. Moreover, the techniques used in STAIG, such as BYOL, local graph structure perturbation, GCN, are common in the field of deep learning.

Response: Thank you for your comments. While the domain of spatial transcriptomics has indeed seen significant research, existing methods still struggle with three critical issues: 1) ST data inherently contains spatial, genetic, and image information, but not graph structure. The optimal graph structure and processing method for ST datasets is still undetermined. 2) The effective extraction and utilization of image data. 3) Identifying the most suitable contrastive learning framework for ST datasets.

Firstly, most existing methods do not focus on the graph construction process, typically building graphs based directly on proximity, which can introduce erroneous edges and biases. STAGATE addresses this by adjusting edge weights based on genetic similarity, but it does not integrate image information. Methods like SpaGCN consider images by using RGB differences, which are severely affected by staining biases. STAIG pioneers in effectively integrating image data and implementing appropriate structural modifications in the spatial transcriptomics field.

Prior to STAIG, few image processing methods were specifically optimized for spatial transcriptomics (ST) datasets. Unsupervised learning has advanced significantly in medical imaging for modalities like CT, ultrasound, and X-rays, but HE images, often used in ST, have been less studied due to their lower commercial value and smaller dataset sizes. These images, especially those of brain tissue, present unique challenges due to subtle textures and uneven staining. Existing methods, largely focused on supervised and semi-supervised approaches, struggle to effectively extract features from these images. STAIG addresses these gaps by successfully applying self-supervised learning to extract meaningful features from brain tissue HE slides, which are notably noisy and differ significantly from traditional histology images.

Our ablation experiments clearly show that different contrastive learning frameworks extract valuable information from ST data differently. STAIG's unique framework achieves batch integration that prior methods could not.

With these innovations in processing ST data, STAIG achieves higher clustering accuracy and simpler integration methods. Additionally, our exploration of HE images from the Visium platform can inspire future research, providing significant reference value.

Spatial clustering and batch integration are foundational tasks in spatial transcriptomics research. Achieving higher clustering accuracy and simpler integration methods has always been a goal. Our baseline tests revealed significant flaws in current methods. For batch integration, most methods require additional tools like Harmony, which adjust embeddings but sometimes perform poorly. Compared to these, STAIG is significantly simpler and more effective, capable of both clustering and integrating with superior results to existing tools like STAlinger.

In our experiments, we believe the spatial transcriptomics field needs suitable, rather than novel computational methods. In exploring image extraction, we evaluated models and self-supervised methods including RGB, CNN, U-Net, and ViT. BYOL consistently outperformed others, demonstrating superior results on Visium images when combined with ResNet over ViT. Similarly, while local graph structure perturbation is not new in computer science, STAIG is the first to attempt it on ST data. Effectively extracting image features and utilizing them for adaptive graph structure perturbation still requires significant innovation.

In conclusion, we believe STAIG's contributions to the spatial transcriptomics field are substantial, and our findings and experiments provide valuable insights for future research.

Comment: Overly ad hoc parameter settings

The adjustments of critical parameters without detailed justification, such as the increase in the number of selected highly variable genes from 3000 to 5000 and the alteration in edge removal probabilities (from 0.3 to 0.2), raise concerns about STAIG's adaptability to diverse datasets. Other ad hoc parameters include the p_min, p_max on line 409, the Bernoulli masking ratio on line 427, the number of clusters in DS filtering on line 457, and many other hyperparameters in network architecture, image processing and loss functions. While a few ad hoc parameters are generally acceptable, too many such parameters undermine the model's robustness and generalizability.

Response: Thank you for your comment. We have not altered the count of highly variable genes (HVGs) used for clustering within a single spatial slice, which has consistently been set at 3,000 (Line 330). The 5,000 HVGs referenced on Line 374 pertain to settings adjusted for multi-batch integration, following your previous suggestions. Our initial approach involved concatenating multiple slices followed by normalization, selecting 3,000 genes thereafter. However, your feedback highlighted potential information loss with this method. In response, we revised our preprocessing workflow to normalize slices individually and then intersect the genes, adopting the 5,000 gene count to align with the methodologies described in STAlinger, the reference you provided. Since this process involves an intersection of genes, we believe it is reasonable to select a higher number than used for individual slices. Moreover, both the initial 3,000 and the revised 5,000 gene counts (different processes) have enabled STAIG to achieve robust integration.

Regarding the adjustment of edge removal probabilities. Initially, we did not perform sensitivity testing for the parameter p during our early experiments. However, in our previous version, we conducted sensitivity tests for all parameters across various platforms. From these experiments, we found that $p = 0.2$ slightly outperformed $p = 0.3$, prompting us to make this adjustment. Our findings indicate that variations in results are minimal when p is less than or equal to 0.3 (Supplementary material Fig. S27d-f), demonstrating stability within this range. Recognizing that the random perturbation probability p as a hyperparameter could impact user experience, we introduced an adaptive perturbation based on gene similarity in the last update, following Reviewer 2's suggestion. This adaptive approach eliminates the need to set a specific p value, especially on platforms without imaging capabilities.

Regarding the parameters p_{max} d p_{min} hese were initially designed to prevent extreme edge perturbation probabilities. Upon revisiting our calculations and assessing the range of adaptive perturbations based on image similarity, which typically falls between 0.1 and 0.4 (as shown in the probability distribution in Figure 1 of the subsequent response), we realized that the impact of this interval scaling was minimal. Throughout our testing, we consistently set p_{min} at 0.1 and p_{max} at 0.9 and never altered these values, which ultimately proved to have little effect. Your comment is quite insightful, and to avoid confusion and simplify the operation, we decided to remove these parameters. After re-testing, we found that the accuracy remained unchanged.

Except for the number of clusters in DS filtering, STAIG's hyperparameters are comparable to those

in ConST, essential for any contrastive learning framework. Compared to GraphST, STAIG only adds the Bernoulli masking ratio for gene perturbation and the number of DS clusters. The Bernoulli masking ratio was optimized through sensitivity testing across multiple platforms, enhancing model generalization and batch integration capabilities. DS filtering, which slightly improves performance by removing bias in negative samples, is an optional strategy, proven by ablation studies to be effective without fundamentally altering STAIG's performance (Supplementary material Fig. S27h).

For image feature extraction, key parameters include filter settings, Gaussian blur parameters, and image size. In partnership with Visium platform's specifications, we have implemented adaptive scaling that requires no user adjustments. This contrasts with methods like DeepST and ConST, which rely on CNNs and ViTs respectively, necessitating more parameters.

In summary, compared to other deep learning methods, STAIG does not introduce an excessive number of hyperparameters. We have conducted extensive hyperparameter testing to ensure the stability and robustness of the algorithm.

Comment: Overly casual clustering in DS filtering

Given the number of clusters in the DS filtering is key to the determination of negative samples and subsequent graph contrastive learning, its setting necessitates further investigation. The authors should provide original images of the clustering results to allow the interpretation of labeling samples in the same cluster as positives while in distinct clusters as negatives. Meanwhile, given the batch effects among H&E images of different slices, it is very likely that image patches of different tissue domain types from same slice can end up in the same cluster, while those of the same tissue type from different slices in different clusters. Such clustering results are not desirable as they are inconsistent with the images' semantic meaning.

Response: Thank you for your comments.

In STAIG, we first assign a class label to each node based on the k-means clustering results of the image features. When the DS filter is not applied, the negative samples are all nodes that are not adjacent to the given node. After applying the DS filter, among all non-adjacent nodes of the given node, a portion of nodes with the same class label are removed from the negative samples. Assuming the number of clusters is 40, except for the samples belonging to the same class as the node, all non-neighboring nodes of the remaining 39 classes are considered as negative samples and participate in the calculation.

At the same time, the DS filter does not affect the selection of positive samples; it only excludes a small portion of negative samples during the calculation. This exclusion of negative samples is an optimization technique, and it does not have a significant adverse impact on the overall results of the STAIG framework. STAIG employs a training approach that primarily relies on genes and uses image features as auxiliary information. In fact, in the ablation experiments without the DS strategy and without the image platform, the accuracy of STAIG already surpasses that of the compared algorithms.

Regarding your request for original images of the clustering results, we would like to clarify that there are two possible interpretations of your request. If you are referring to the visualization of the k-means clustering results based on the high-dimensional image features, it is challenging to present the clustering results in a two-dimensional format after obtaining the class labels. Alternatively, if you are asking for the direct presentation of the original spot patch images belonging to the same cluster, it is impractical to enumerate all the patch images within a cluster due to the large number of spots (typically over 3,000) in a single Visium slice.

To address your concerns and provide a better understanding of the effectiveness of our DS approach, we present an alternative method of explanation. We have included a stacked bar plot (Figure 1) showing the distribution of the 40 clusters based on the image features of slice #151673, along with the proportion of class labels within each cluster. We acknowledge that due to the inherent noise in the images and the shallow textures of brain images, it is challenging to achieve precise segmentation solely based on image information without prior knowledge. The presence of different classes within each cluster is inevitable.

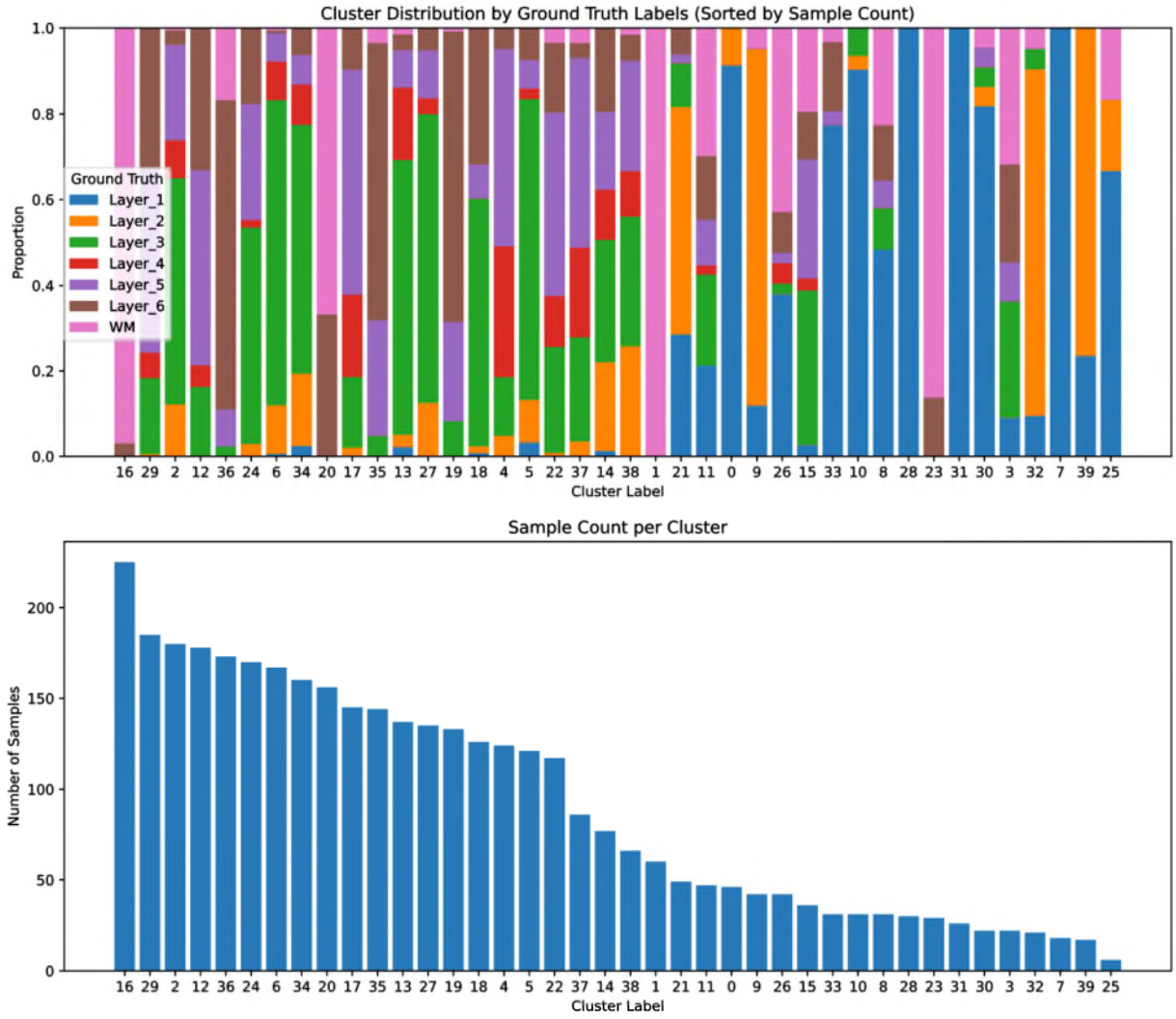


Figure 1 Distribution of the 40 clusters based on the image features of slice #151673

In the stacked bar plot, we have sorted the clusters in descending order based on their sample counts. We can observe that the clusters with the highest sample counts are predominantly composed of the WM layer. In the subsequent clusters with relatively high sample counts, there is usually one layer that occupies a significant proportion. Moreover, even when multiple classes are present within a cluster, they tend to be adjacent layers, such as Layer6 and Layer5, or Layer3 and Layer4.

When applying the DS filter, nodes with the same class label within a cluster are excluded from the negative samples, along with nodes of other classes that also belong to the same class. In most cases, the layer with the largest proportion benefits more from this approach. Compared to not using the DS strategy, for the nodes in the first bar, over 200 nodes of the same class are excluded from the negative samples during training.

Increasing the number of clusters can undoubtedly increase the dominant proportion within each cluster. However, this leads to a decrease in the sample count of each class, resulting in a smaller number of excluded samples per iteration and less significant benefits. In our assumed usage scenario, given an H&E stained slice without any prior knowledge, we aim to extract and utilize features effectively. Currently, there is no clustering algorithm that can operate without parameter settings, which means that the number of clusters in DS can only be provided as a hyperparameter. We have conducted tests on existing datasets and determined an empirical value (Supplementary Fig. S23b-c) that is considered appropriate.

Regarding the handling of multiple slices, we agree with your suggestion that batch effects exist among images from different slices. In the updated version of our approach, when dealing with multiple slices, we do not employ the DS strategy to mitigate the potential inconsistencies arising from batch effects.

Comment: 6. Insufficient mathematical solidity

STAIG falls short of mathematically solidity, as reflected not only by excessive ad hoc parameter settings but also the lack of succinct mathematics and efficient computations. For example, the use of a logarithmic operation in Equation 3 followed by an exponential in Equation 4 seems unnecessary and only incurs extra computational costs. This could be streamlined by skipping equation 3 and directly employing D^{img}+1 in Equation 4. Additionally, the computation of the distance matrix D^{img} could be optimized by leveraging the sparsity of the adjacency matrix, rather than using the computationally intensive function like "euclidean_distances = cdist(node_emb, node_emb, metric='euclidean')" on line 106 in adata_preprocessing.py. Not only this brutal-force calculation is not inefficient, but also potentially leads to memory issues in case of large-scale data and high-dimensional embeddings.

Response: Thank you for your comment. While it might seem straightforward to obtain normalized probabilities directly through Softmax, in practice, due to significant variations in Euclidean distances, not applying a logarithmic transformation would lead to extremization of edge perturbation probabilities. Figure 2 displays the distribution of adaptive edge deletion probabilities under different processing conditions. It clearly demonstrates the importance of applying a logarithm before the

Softmax operation, as it effectively prevents the extremization of probabilities.

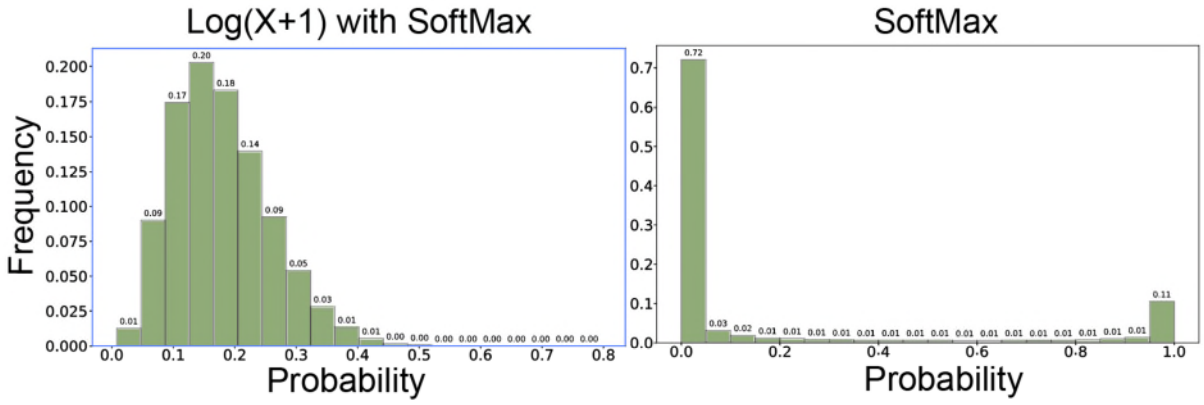


Figure 2. The distribution of edge deletion probabilities (log with SoftMax vs SoftMax).

Additionally, following Reviewer #3's suggestion, we found that when performing the logarithmic operation, the minimum values of D^{img} were all significantly greater than 1, rendering the "+1" in the original $\log(x+1)$ unnecessary, as it does not prevent extremely small values that could potentially cause issues in subsequent results. Therefore, based on Reviewer #3's advice, we changed $\log(D^{img}+1)$ to $\log(D^{img})$(Line 398). As shown in Figure 3, the probability distributions before and after this adjustment remain unchanged.

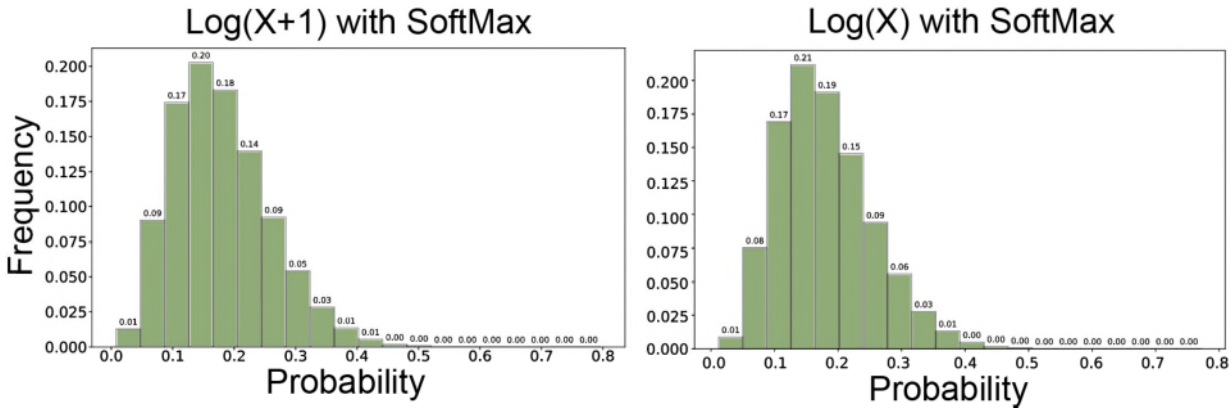


Figure 3. The distribution of edge deletion probabilities (log(X+1) vs log(X)).

In terms of code implementation, we modified the original computation of the distance matrix D^{img}. Instead of using *cdist*, we now only calculate the similarities between the necessary neighboring edges, avoiding unnecessary computations. Additionally, we have added a progress bar to visualize the progress of the computation, allowing users to track the program's advancement clearly.

Reference:

[1] Long Y, Ang K S, Li M, et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST[J]. Nature Communications, 2023, 14(1): 1155.

[2] Zong Y, Yu T, Wang X, et al. conST: an interpretable multi-modal contrastive learning framework for spatial transcriptomics[J]. BioRxiv, 2022: 2022.01. 14.476408.

[3] Velickovic P, Fedus W, Hamilton W L, et al. Deep graph infomax[J]. ICLR (Poster), 2019, 2(3): 4.

[4] Zhang L, Liang S, Wan L. A multi-view graph contrastive learning framework for deciphering spatially

resolved transcriptomics data[J]. Briefings in Bioinformatics, 2024, 25(4): bbae255.
[5] Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding[J]. arXiv preprint arXiv:1807.03748, 2018.

General Response to Reviewer #2:

We sincerely appreciate your continued guidance and insightful feedback.

In this revised version, we have updated the descriptions of the formulas to make them easier to understand. Additionally, we have demonstrated the effectiveness of STAIG through ablation experiments. Based on the findings from these experiments, we have updated the experimental results for batch integration.

We hope that our revisions and the additional experiments sufficiently address your concerns. We are truly grateful for the time and effort you have invested in reviewing our work. Your feedback has been invaluable in improving our study.

Detailed Responses to Specific Comments

Comment: - The authors explained cross-slice interaction in the STAIG model by referring to negative samples in the contrastive loss. If I understand it correctly, the negative samples include not only pre-existing edges that get removed based on image feature similarity, but also edges that do not exist in the original adjacency graph, such as those across slices. If that was the case, the second term in the denominator of Eq. 9 involving a sum over all such negative samples would be highly inefficient to compute. I suspect that some sort of negative sampling strategy was actually employed instead of computing the exhaustive sum, which was not clearly reflected in Eq. 9. Furthermore, the normalization of negative samples does not seem to align with that of positive samples in Eq. 9. Additionally, I am also unclear why f^1,2(i, i) is included in the denominator; should this not be a positive edge? Lastly, I still find it counterintuitive why batch effect correction works in STAIG, which obviously still lacks positive samples across slices. "Emphasizing on subtle local dynamics over broad global changes" is posited as an explanation, but remains a fuzzy intuition. Could ablation experiments help clarify the underlying mechanism of STAIG's batch effect correction, and verify the authors' intuitions provided here?

Response: Thank you for your valuable comments. Firstly, let we explain the loss function when not considering the DS filter strategy. The loss function is formulated as follows:

$$l(h_i^1) = -log\frac{\left(f^{1,2}(i,i) + \sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j) + \sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)\right)/\widehat{\mathcal{N}_i}}{f^{1,2}(i,i) + \sum_{(j \neq i)}\left(f^{1,1}(i,j) + f^{1,2}(i,j)\right)}, \tag{1}$$

The last two terms in the denominator of Equation (1) can be decomposed as:

$$\sum_{(j \neq i)} f^{1,1}(i,j) = \underbrace{\sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j)}_{intra-graph\ neighbor\ pos} + \underbrace{\sum_{v_j \notin \mathcal{N}_i^1} f^{1,1}(i,j)}_{intra-graph\ neg}, \tag{2}$$

$$\sum_{(j \neq i)} f^{1,2}(i,j) = \underbrace{\sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)}_{inter-graph\ neighbor\ pos} + \underbrace{\sum_{v_j \notin \mathcal{N}_i^2} f^{1,2}(i,j)}_{inter-graph\ neg} \tag{3}$$

The Equation (1) can be rewritten as:

$$l(h_i^1) = -log\frac{\left(f^{1,2}(i,i) + \sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j) + \sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)\right)/\widehat{\mathcal{N}_i}}{f^{1,2}(i,i) + \underbrace{\sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j)}_{intra-graph\ neighbor\ pos} + \underbrace{\sum_{v_j \in \mathcal{N}_i^1} f^{1,1}(i,j)}_{intra-graph\ neg} + \underbrace{\sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)}_{inter-graph\ neighbor\ pos} + \underbrace{\sum_{v_j \in \mathcal{N}_i^2} f^{1,2}(i,j)}_{inter-graph\ neg}}, \tag{4}$$

To show this equation more clearly, we color-code each term in the equation, corresponding to the schematic of Figure 1.

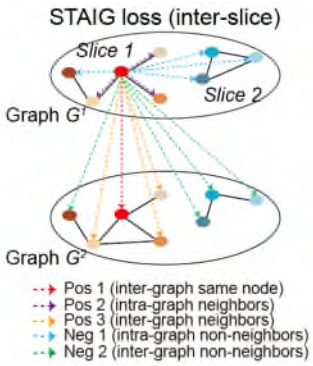


Figure 1. Schematic of the STAIG loss

As we can see, the training objective for the loss function is to maximize the numerator while minimizing the contribution of negative samples in the denominator. Typically, in contrastive learning, the denominator includes the sum of the exponential of the similarities for all sample pairs (both positive and negative), serving as a regularization term. This helps to prevent the numerator from disproportionately influencing the overall loss value. If only the similarities of negative samples were calculated, the denominator might be too small, leading to large or unstable loss values. Including positive samples ensures that even when the similarity of positive samples is high, the loss function remains within a reasonable range.

Furthermore, this setup is computationally advantageous, as the similarity calculations for all sample pairs can be performed directly through matrix operations (multiplying two vector matrices and summing across specified dimensions). This computation can be accelerated with GPU processing, which is faster than selecting several negative samples manually.

On the coding level, assume that after two augmentation graphs, the resulting normalized matrices are H^1 and H^2. The intra-graph similarity is calculated as $\exp(H^1(H^1)^T)$, and the inter-graph similarity is calculated as $\exp(H^1(H^2)^T)$. If it is necessary to exclude some negative samples, a mask matrix can be pre-built to set the corresponding similarity entries to zero.

The calculation for negative samples would then be the row-wise sum of $\exp(H^1(H^1)^T)$ plus the row-wise sum of $\exp(H^1(H^2)^T)$, subtracting the diagonal sum of $\exp(H^1(H^1)^T)$ (to remove the non-existent $f^{1,1}(i,i)$ from the formula).

Moreover, regarding various contrastive learning loss functions, when the first-order adjacency matrix is not considered, the loss function transitions to the one used by Zhu et al.[1]:

$$l(h_i^1) = -log\frac{f^{1,2}(i,i)}{f^{1,2}(i,i) + \sum_{(j\neq i)}\left(f^{1,1}(i,j) + f^{1,2}(i,j)\right)}, \tag{5}$$

When further removing the negative samples within the intra-graph, the loss function is transformed into the basic InfoNCE[2] loss:

$$l(h_i^1) = -log\frac{f^{1,2}(i,i)}{\sum f^{1,2}(i,j)}, \tag{6}$$

Thank you for suggesting ablation experiments. To evaluate STAIG's capability for spatial integration, we conducted comparative studies with other contrastive learning frameworks and performed ablation studies on various components of the loss function. The results of these experiments are detailed in the supplementary material "Ablation Studies for Integration" (lines 435-495), where we have included schematic diagrams of different loss functions for clarity.

Initially, we compared STAIG with other contrastive learning frameworks, such as DGI[3], which is represented by models like GraphST[4] and ConST[5]. These models employ different graph augmentation strategies. In the DGI architecture, the augmented graph does not alter the graph structure but instead randomly permutes the node positions. Their model defines positive samples as the node and its first-order neighbors in the original graph and negative samples as nodes in corresponding positions in the augmented graph.

After similar data preprocessing, DGI's contrastive architecture managed only minor integration of data from the Visium platform and failed to achieve cross-platform integration. In contrast, STAIG demonstrated its integration capability both visually (UMAP) and through performance metrics, indicating that its strength primarily derives from a different contrastive framework. Further ablation studies on STAIG's loss function components showed that random gene masking and positive samples from first-order neighbors enhanced integration capability. However, gene masking alone was not the primary factor, as DGI was unable to achieve cross-platform integration with or without gene masking and was minimally affected by the masking.

Our ablation experiments also revealed that negative samples across slices actually reduced integration capability. When we restricted negative sample pairs to the same slice, integration capability improved significantly. This does not imply that our training process simply inputs individual datasets into a GCN one by one, as the same process could not integrate data using DGI's architecture.

Based on these findings, we believe STAIG's integration capability stems from its node-by-node contrastive learning framework, which can effectively learn commonalities in gene expression. Interestingly, a recent study by Zhang et al.[6] supports our observations. They argue that while DGI relies solely on spatial dependencies, their InfoNCE framework—a contrastive pattern similar to ours—can adaptively learn both spatial dependencies and common gene expression traits. Their conclusions align with our results, though they did not further explore integration. We have included their method in our comparative analysis.

Following the insights from the ablation studies, we restricted negative sample pairs to the same slice during batch integration, updating our original integration results (Line 180-227, Fig.5, Supplementary Fig.S11-15). Compared to the previous version, our current integration metrics have significantly improved. We have also updated the descriptions in the Methods section accordingly (Line 386-387, 464-469).

Comment: - Regarding the "artificially structured graphs," I now understand its meaning, but still have a minor complaint regarding the assertion that the artificial graph structure in STAGATE does not change. My understanding is that the graph attention mechanism in STAGATE could effectively reduce the influence of unreasonable edges through low attention scores, which might mitigate this issue. A more distinct comparison between STAIG and STAGATE might focus on STAIG's integration of image features rather than solely relying on transcriptomic data.

Response: Thank you for your valuable comments. We agree with your points, and have updated our phrasing in the Introduction accordingly. At line 35, we have revised the statement to read: "Although some methods like STAGATE address this issue by assigning weights to edges based on cell gene similarity, they do not integrate the image information."

Comment: The colors representing "intra-view non-neighbors" and "inter-view non-neighbors" in Fig. 1e do not match the contrastive learning illustration, which may confuse readers.

Response: Thank you for your valuable comments. We have corrected the color errors in Fig. 1e. Additionally, we have included more detailed schematic diagrams in the supplementary material section to clarify this further (Fig. S30).

Comment: In line 463, the second highlighted term should be l(h_i^1).

Response: Thank you for your valuable comments. We have corrected the superscript of this term. The modification can now be found at line 470.

Comment: The panel order in Fig. 3c does not match those in Fig. 3a, b.

Response: Thank you for your valuable comments. We have corrected the order in Fig. 3c.

Reference:

[1] Zhu Y, Xu Y, Yu F, et al. Graph contrastive learning with adaptive augmentation[C]//Proceedings of the Web Conference 2021. 2021: 2069-2080.

[2] Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding[J]. arXiv preprint arXiv:1807.03748, 2018.

[3] Velickovic P, Fedus W, Hamilton W L, et al. Deep graph infomax[J]. ICLR (Poster), 2019, 2(3): 4.

[4] Long Y, Ang K S, Li M, et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST[J]. Nature Communications, 2023, 14(1): 1155.

[5] Zong Y, Yu T, Wang X, et al. conST: an interpretable multi-modal contrastive learning framework for spatial transcriptomics[J]. BioRxiv, 2022: 2022.01. 14.476408.

[6] Zhang L, Liang S, Wan L. A multi-view graph contrastive learning framework for deciphering spatially resolved transcriptomics data[J]. Briefings in Bioinformatics, 2024, 25(4): bbae255.

General Response to Reviewer #3:
We sincerely appreciate your continued guidance and insightful feedback.

Following your recommendations, we have made several important revisions to our manuscript. We refined the expressions in the paper's formulas for enhanced clarity and added a detailed section in the supplementary materials to describe the loss function and contrastive mechanisms, accompanied by corresponding ablation studies. These revisions underscore the effective integration capability of STAIG.

Responding further to your feedback, we conducted additional tests and removed the parameters you identified as unnecessary.

We hope that our revisions and the additional experiments sufficiently address your concerns. We are truly grateful for the time and effort you have invested in reviewing our work. Your feedback has been invaluable in improving our study.

Detailed Responses to Specific Comments
Comment: The authors have addressed my comments and most comments from the other reviews. The revised version of the manuscript clearly improves the original version, especially by the inclusion of results from imaging-based spatial transcriptomics.

However, as raised by reviewers #1 and #2, the claims of global interaction between multiple datasets could be excessive.
While the responses clearly show that equation 9 is a global computation, they are not connected through adjacencies, only through their embedding space --- which is the default for deep learning training with multiple samples --- it's unclear how much of that can be qualified as a novelty once the unnecessary terms are removed and the formulas simplified.

This doesn't disqualify the paper's claims and experiments that they obtained a better integration, but the description could be more transparent.

This extends to equation 4, as acknowledged during their response. Despite being originally motivated by softmax, once you apply "log" and then "exp," it would be clearer to show that "Pij" is a normalized distance, which raises the question of why the +1 is an important constant to this normalized distance.

Response: Thank you for your valuable comments. In response to the concerns raised by Reviewers 1 and 2, we have made concerted efforts in two main areas. First, we have added a more detailed description of the loss function in the main text from line 452-473. Additionally, we have conducted ablation experiments to compare the graph augmentation strategies of STAIG with those of other methods. We also performed ablation studies on various components of the loss function, the results of which are documented in the supplementary material "Ablation Studies for Integration," lines 435-

495. For clarity, this section now includes schematic diagrams of the various loss functions.

Our efforts focused on three main tests: 1) comparing the graph augmentation strategies of STAIG with those of DGI[1], 2) examining the effects of restricting negative samples within a single slice versus across slices, and 3) evaluating the impact of random gene masking on integration.

Our experimental results demonstrate that STAIG's integration capability primarily stems from its contrastive architecture, which is distinct from DGI. Under the same data preprocessing conditions, DGI is unable to integrate cross-platform data. Moreover, we found that restricting negative samples to a single slice can enhance batch integration, a finding that emerged from our ablation studies and admittedly contradicts our previous descriptions. Additionally, we observed that random gene masking plays a significant role in STAIG's integration capability, simulating batch effects to some extent and thereby enhancing the model's generalizability. However, this approach was not as beneficial for DGI.

In fact, a recent study by Zhang et al.[2] shares similar findings. They argue that while DGI can only utilize spatial dependencies, their InfoNCE[3] framework—a contrastive pattern akin to ours—can adaptively learn both spatial dependencies and common gene expression features. Their conclusions align with our experimental results, though they did not further explore integration. We have also included their method in our comparative analyses.

Given that restricting negative samples within slices leads to better aggregation results, we have updated the formula descriptions (Line 386-387, 464-469) and redone the related integration experiments (Line 180-227, Fig.5, Supplementary Fig.S11-15). The metrics have shown significant improvements over the previous version.

Regarding Formula 4, Figure 1 displays the distribution of adaptive edge deletion probabilities under different processing conditions in #151673. It clearly demonstrates the importance of applying a logarithm before the SoftMax operation, as it effectively prevents the extremization of probabilities.

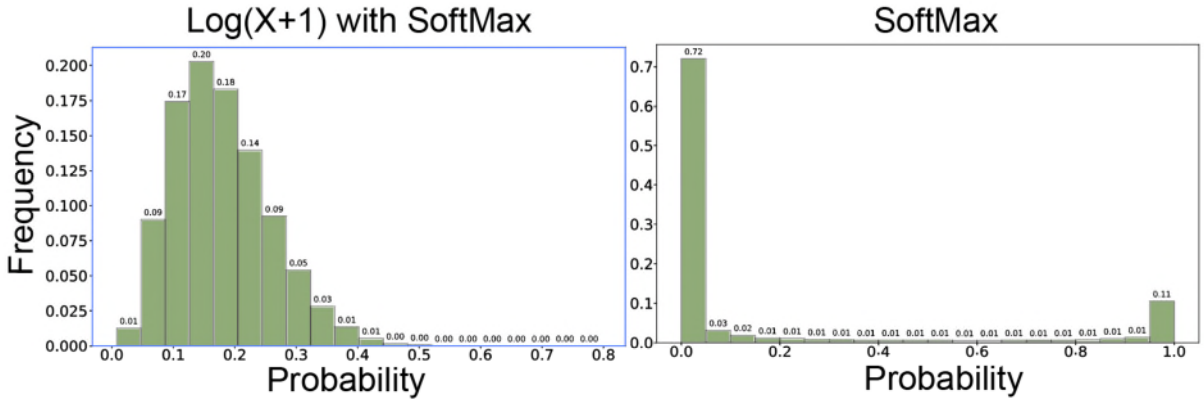


Figure 1. The distribution of edge deletion probabilities (log with SoftMax vs SoftMax).

Originally, we used $log(x+1)$ to leverage the curvilinear part of the logarithmic function when $x > 0$ to smooth the Euclidean distances. This approach was based on the consideration that if the

Euclidean distance between two points was very small, directly applying $log(x)$ would generate a large negative number. This could result in the corresponding edge's importance being almost ignored in subsequent processes, and potentially cause numerical underflow when applying the softmax function. By using $log(x+1)$, we avoided generating negative values, thus maintaining numerical stability and ensuring a smoother calculation of probability distributions.

However, upon revisiting the calculations based on your suggestion, we discovered that the minimum values in the initially obtained Euclidean distance matrix were all significantly greater than 1, indicating that our original scenario did not exist. We appreciate your pointing this out. Consequently, we have removed the $+1$ term, changing it to $W = log(D^{img})$(Line 398). As shown in Figure 2, the probability distributions before and after this adjustment remain virtually unchanged.

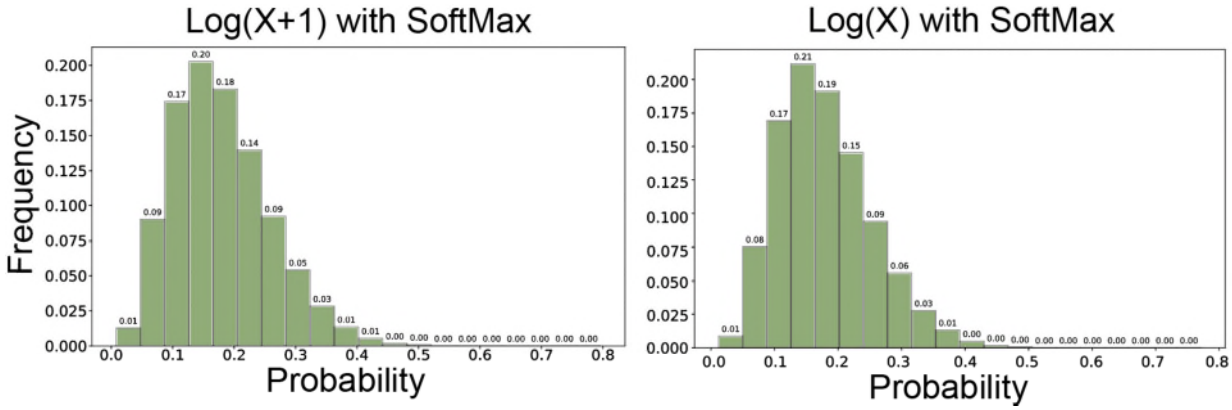


Figure 2. The distribution of edge deletion probabilities (log(X+1) vs log(X)).

Reference:

[1] Velickovic P, Fedus W, Hamilton W L, et al. Deep graph infomax[J]. ICLR (Poster), 2019, 2(3): 4.

[2] Zhang L, Liang S, Wan L. A multi-view graph contrastive learning framework for deciphering spatially resolved transcriptomics data[J]. Briefings in Bioinformatics, 2024, 25(4): bbae255.

[3] Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding[J]. arXiv preprint arXiv:1807.03748, 2018.

NCOMMS-24-00513C

STAIG: Spatial Transcriptomics Analysis via Image-Aided Graph Contrastive Learning for Domain Exploration and Alignment-Free Integration

Yitao Yang, Yang Cui, Xin Zeng, Yubo Zhang, Martin Loza, Sung-Joon Park, Kenta Nakai

Response to Reviewer #1:

Comment: 1. The authors claim that they have updated STAIG code and incorporated resampling process into the construction of G1 and G2 to improve STAIG's performance stability. However, I do not see the incorporation of resampling process in their method section. Moreover, the authors should show experimental results, including the ablation study on the resampling process, to prove the improved stability due to these efforts. Given the randomness inherent to STAIG, it is better to add a section in the manuscript, describing the stability issue and authors' efforts on addressing this issue. Also, given the authors' claim the update results in a slight decrease in computational speed as well as the extra costs associated with the resampling process, STAIG's scalability against different sample sizes after these updates should be demonstrated.

Response: Thank you for your comment. First, we would like to apologize for any confusion caused by our previous response. In the last version, we improved the code and resolved the compatibility issues between Torch and CUDA, ensuring stable results across the hardware we were able to test. Reviewer 2 successfully replicated our results in their local environment. However, as we cannot test on all possible hardware configurations, we have implemented your suggestion as an optional feature for users to address any potential stability issues.

We have added a new section in the supplementary materials titled "Stability and efficiency of the averaged structure" (Supplementary Material, Lines 504-526, Figures S31d-f) to address the stability issue specifically.

In this section, we detail the method and conducted an ablation study using the DLPFC dataset from the Visium platform and the STARmap platform, which lacks image data, to evaluate the impact of varying the number of resampled graphs on both accuracy and time cost. As shown in Supplementary Fig. S31d-e, the performance is nearly identical to that without resampling when the number of resampled graphs is less than 10. However, when the number of resampled graphs reaches 20, the time cost doubles.

Additionally, we have expanded the discussion on this topic in the main text under the Discussion section (Lines 313-319):

We acknowledge a potential stability issue in the sampling process of G^1 and G^2. To address this, we implemented a resampling method that generates n augmented graphs and averages their structures to create the final enhanced graph (see Supplementary Materials: Stability and efficiency of the averaged structure). While resampling increases algorithmic complexity, our ablation studies

show stable results when n was kept below 10 (see Supplementary Fig. S31d-f). Therefore, we believe that the current methodology is robust across various datasets and scenarios. Nonetheless, this approach may not completely eliminate stability issues in every test or dataset. To mitigate potential biases, we provide users with the option to apply the resampling process to their own datasets, which we anticipate will help reduce any sampling-related bias.

Comment: 2.The authors have updated STAIG to restrict negative sample pairs to the same slice during integration and claimed that it significantly improves the integration. However, this update fails to address my previous concern about shortening distances between inter-batch positive pairs, which can truly mitigate batch effects in theory. Moreover, this update seems to contradict the authors' statement in the previous response, where I quote:

"Firstly, let's consider STAIG's neighbor contrastive loss functon(9) (Line 460). In this equation, the numerator represents the computation of similarities between a sample and its immediate neighbors, treated as positive pairs within the augmented graph. Conversely, the denominator accounts for the interactions with negative samples across the entire samples (The second term of the denominator). This global computation not only normalizes the impact of local interactions but also indirectly bridges different datasets by adjusting each data point's influence in relation to the overarching data characteristics.
Therefore, although direct dataset interactions are not modeled by the matrices A and W, the denominator plays a crucial role in facilitating an indirect information exchange, thereby significantly enhancing the model's capacity to generalize across diverse datasets".

This statement clearly indicates the importance of using cross sample negative pairs to make STAIG as a multi-batch model, which is however denied in this response. Although the authors have conducted ablation studies, attempting to demonstrate this point empirically, the seemingly good results may be purely random, particularly considering its theoretical deficiency. Restricting negative sample pairs to the same slice during integration essentially forbids interactions between samples across batches and makes STAIG a method for single batch only. In summary, I do not think authors fully address my concern about STAIG's effectiveness in batch integration, at least theoretically.

Response: Thank you for your comment. We apologize for not providing a more detailed explanation in our previous response. In this revision, we have clarified the underlying principles of STAIG's integration capability from the perspective of latent space analysis.

First, it is important to acknowledge that, even in the theoretical field of computer science, there is currently no widely accepted explanation for the success of contrastive learning[1]. In areas like computer vision, where contrastive learning is more frequently applied, practice often precedes theory[2]. Recent findings suggest that contrastive learning frameworks exhibit a certain degree of generalization, but the source of this generalization, and why positive and negative sample comparisons at the sample level lead to clustering, is only beginning to be theorized in the context of contrastive learning for images[3]. In contrast, theoretical analysis in graph-based contrastive

learning remains limited[4]. Based on the analysis from Huang et al.[3] in the image domain, the reason for generalization and clustering in sample-level contrastive learning stems from the proximity of augmented positive samples of the same class in latent space. Inspired by their work, we investigated the behavior of samples from two cross-platform datasets in the latent space as the iterations progressed.

The principles behind STAIG's integration in latent space, along with detailed experimental results, have been documented in the supplementary materials under the section titled "The principle of STAIG integration in latent space."(Line 527-600, Fig. S32,S33)

A summarized version of these findings and conclusions has been included in the main text within the Discussion section (Lines 291-304):

The batch integration capability of STAIG stems from the generalization power of the contrastive learning framework, enabling the model to extract meaningful information across multiple similar datasets. While a comprehensive theoretical explanation for why sample-level contrastive learning leads to effective self-supervised clustering and generalization is still lacking, we investigated this phenomenon from the perspective of changes in the positions of samples in the latent space. Specifically, we observed that, in the early stages after dimensionality reduction through GNN and graph augmentation, the latent space exhibited a directional shift between samples from different slices due to batch effects. However, the angular differences between samples of different categories remained consistent across slices. That is, while all samples are initially very close to each other in the latent space, during each iteration, the contrastive learning at the node level ensures that the model focuses on the angular differences between categories rather than the differences introduced by batch effects when processing consecutive slices. This alignment enables samples from the same category, regardless of the slice, to move in similar directions and distances. In subsequent iterations, the model increasingly learns and extracts features relevant to category distinctions, further mitigating the impact of batch effects. Ultimately, samples of the same category, irrespective of their slice origin, cluster together, leading to embeddings that are largely free from batch effects.

Comment: 3.Graph construction: Admittedly, few existing methods augment graph using H&E image. I believe this is due to several reasons: Firstly, Only a limited ST platforms provide H&E image. Using gene similarity or proximity is for general applicability, at least better than the ad hoc parameter 0.2 used in the fixed edge removal in STAIG when H&E image is unavailable. Secondly, H&E image quality cannot be guaranteed, with different tissue types appearing similarly in the image. For example, a TME microenvironment may encompass many tumor subtypes but appear as a homogeneous tumor. Nonetheless, such subtle differences can be detected by gene expression data at the molecular level. Therefore, authors need to demonstrate H&E images' edge over ST data even when the H&E image quality is poor. Otherwise, authors need to provide corresponding solutions.

H&E image utilization: Spatial clustering methods in ST that effectively use the H&E image data are not limited. For example, iSTAR and conST are well-known such methods.

Batch integration: Since authors provide a claim contradictory to their previous claim about the using of cross-batch negative samples in the response to my question 2, as well as fail to address my concern about STAIG's batch integration from a theoretical perspective, I am not convinced about authors' claim "STAIG's unique framework achieves batch integration that prior methods could not".

Response: Thank you for your comments. We appreciate your thoughtful feedback and would like to address your concerns regarding graph construction, H&E image utilization, and batch integration in STAIG.

Regarding graph construction, we want to emphasize that STAIG's approach is not a mutually exclusive choice between using images, gene similarity, or fixed edge removal. Instead, it offers flexibility to users based on their specific needs and data availability. Our tests have shown that all three methods outperform current baselines. To the best of our knowledge, at the time of submission, few studies had approached ST optimization from the perspective of altering graph structure.

We acknowledge the potential unreliability of image quality, including noise and, as you mentioned, the inability to distinguish certain regions within tumors through imaging alone. Without prior knowledge, even experts might struggle to further segment tumors based on images. As a result, we have always used image features in a limited capacity rather than in a dominant role. Specifically, we do not incorporate image features directly into the node features but use them as a weighting factor in edge removal during graph augmentation. The node information remains gene-based. While images may not enable fine segmentation within a tumor, after our feature extraction process, we can distinctly identify the overall tumor boundary, which can be crucial in defining major functional regions. Compared to baselines like GraphST and conST, which do not alter the fixed graph structure, leveraging the overall boundary information provides an advantage. This approach reduces errors arising from incorrect neighbor edge aggregation. When the internal tumor regions have similar image characteristics, the removal probability between nodes becomes similar (like a type of fixed edge removal), effectively returning to a gene-level contrast for finer segmentation within the tumor.

It's important to note that even in traditional image processing, no single unsupervised algorithm can handle all types of images perfectly. Our method has shown broad effectiveness in our test experiments. The Visium platform provides images with consistent format and size, generally ensuring adequate quality. In cases where image quality is poor, users can opt for gene similarity-based clustering.

Regarding H&E image utilization, while conST's ViT-pretrained model demonstrated poor feature extraction results in brain regions (as shown in Fig. 3a-c) and significant discrepancies in cancer tissue contours compared to manual annotations, it raises questions about whether such image extraction quality can genuinely improve the final results. iSTAR is an excellent work that utilizes ViT for self-training, feature extraction, and segmentation, published on January 2, 2024. However, it was not publicly available during the development of STAIG. Additionally, our approach differs from theirs. We base our method on the ResNet framework, which requires fewer parameters and lower

hardware specifications compared to ViT, and can self-train on a single image without needing additional datasets. The goals of our methods also differ. This distinction does not undermine the innovation of our approach; rather, it highlights the potential for various image processing methods to inspire future researchers to explore new avenues.

Concerning batch integration, we appreciate your concern and have responded this in the previous comment.

Comment: 3. Thanks for the authors' detailed explanation about their choice of these hyperparameters, though I am a little surprised when reading authors' response "Regarding the adjustment of edge removal probabilities. Initially, we did not perform sensitivity testing for the parameter p during our early experiments.". Given the key role of p in the absence of H&E image for most ST platforms, the lack of sensitivity test leaves the choice of 0.3 in the initial submission purely based on speculation, not even on less rigorous empirical sensitivity test, let alone solid theoretical foundation.

Response: Thank you for your comment. Thank you for your comment. We sincerely apologize for the shortcomings in our initial work. We greatly appreciate the suggestions from you and the other reviewers, which have prompted us to conduct additional experiments to address the areas that were previously overlooked. As a result, we have now included sensitivity testing for the parameters, making the current version of STAIG significantly more robust and stable compared to the initial submission.

Comment: 4. Since the authors have removed the DS filtering in scenarios involving images from difference slices, corresponding results should be updated to reflect this change. For example, if Fig.7 involves DS filtering in the previous version, its results need to be updated.

Response: Thank you for your comments. In the previous version, we made the necessary adjustments for experiments involving multiple slices. Regarding Fig. 7, the results were based on experiments conducted on individual slices and did not involve integration analysis. We have carefully reviewed the figure and confirmed that no updates are required for this figure. In the previous version, all tasks involving multiple slices were addressed in Fig. 3 and the corresponding supplementary materials, and we have already made the necessary updates for these sections.

Comment: 5. The authors fail to understand my concern. Please see the simple math below:

$$W=\log\left(D^{img}\right)\rightarrow$$
$$e^{W_{i,j}}=e^{\log\left(D_{i,j}\right)}=D_{i,j}\rightarrow$$
$$P_{i,j}=\frac{D_{i,j}}{\sum D_{i,k}A_{i,k}}$$

Response: Thank you for your comment. We apologize for misunderstanding your concern in our previous response. We appreciate you pointing out the issue with our formula. We have made the necessary corrections in the corresponding section, and we have also further optimized the code to reflect these changes. The specific modifications can be found in Lines 415-419.

Reference

1. Parulekar, A., Collins, L., Shanmugam, K., Mokhtari, A. & Shakkottai, S. Infonce loss provably learns cluster-preserving representations. in *The Thirty Sixth Annual Conference on Learning Theory* 1914–1961 (PMLR, 2023).
2. Hu, T., Liu, Z., Zhou, F., Wang, W. & Huang, W. Your Contrastive Learning Is Secretly Doing Stochastic Neighbor Embedding. in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023* (OpenReview.net, 2023).
3. Huang, W., Yi, M., Zhao, X. & Jiang, Z. Towards the Generalization of Contrastive Self-Supervised Learning. in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023* (OpenReview.net, 2023).
4. Yuan, Y. *et al.* Towards generalizable Graph Contrastive Learning: An information theory perspective. *Neural Networks* **172**, 106125 (2024).

Response to Reviewer #2:

Comment: I appreciate the authors' efforts in addressing my previous questions. The clarity of the manuscript has significantly improved. Following the recommendation of reviewer 1, I attempted to install and run the software as an end user. I am glad to report that the obtained results were nearly identical to those presented in the manuscript when executed on a local server equipped with A100 GPUs.

Response: Thank you for your valuable comments. We greatly appreciate you taking the time to validate our experimental results.

Comment: I have one remaining request concerning the newly-added ablation experiments on batch integration. Typically, batch correction carries the risk of over-correction, which may result in the removal of genuine biological signals. Therefore, in addition to the iLISI and batchKL metrics that quantify the extent of batch mixing, I suggest that the authors also report metrics related to the preservation of biological signals in these ablation tests (such as NMI and ARI), as is already included in previous parts of the manuscript.

Response: Thank you for your valuable comments. In this round of revisions, we have added metrics related to the preservation of biological signals during the ablation experiments. Specifically, for the DLPFC dataset, we have included ARI and NMI metrics. For cross-platform dataset integration, we have supplemented the analysis with SC and DB indices. The corresponding results can be found in Supplementary Materials, Figures S30 and S31.

From these metrics, we observe that under the STAIG framework, ARI and NMI remain robust, indicating that the integration process does not compromise the preservation of meaningful biological information. In contrast, the DGI framework not only exhibits inferior integration capabilities but also results in embeddings that lose valuable biological information, leading to lower ARI and NMI scores. The SC and DB metrics for cross-platform integration similarly reflect this trend. Additionally, in the ablation study involving random gene masking, we found that random gene masking enhances the effectiveness of the final embeddings produced by the model. Overall, the effectiveness of biological signal preservation aligns closely with the integration capability, demonstrating that STAIG can achieve strong integration while maintaining high clustering performance.

The specific changes are detailed in the Supplementary Materials section "Ablation studies for integration" (Lines 435-502) and in Supplementary Figures S30a-i and S31a-c.

NCOMMS-24-00513D

STAIG: Spatial Transcriptomics Analysis via Image-Aided Graph Contrastive Learning for Domain Exploration and Alignment-Free Integration

Yitao Yang, Yang Cui, Xin Zeng, Yubo Zhang, Martin Loza, Sung-Joon Park, Kenta Nakai

Response to Reviewer #1:

Comment: 1. Time Costs and Hardware Configurations: While the authors have addressed the stability concern through testing, I would like to see the incurred time costs. Specifically, the authors should report STAIG's running time in minutes, relative to the number of datasets and total number of spots, along with the hardware configurations of the test environment.

Response: Thank you for your comment. Following your suggestion, we have described our hardware configurations in Lines 533-534 of the main text:

STAIG was tested on an NVIDIA A100 GPU installed in a Linux system with an AMD EPYC 7742 CPU and 64GB of memory. Detailed runtime costs are provided in Supplementary Table S3.

Additionally, we have reported the running times in Table S3 of the supplementary materials (supplementary materials Line 722).

Comment: 2. Batch Integration Clarity: I remain unclear about the batch integration process. Specifically, does any information exchange occur between batches during STAIG's batch alignment? If so, how is this exchange achieved without interactions between dataset-specific adjacency matrices in the $\mathcal{A}$ matrix and cross-sample negative samples? Additionally, I seek clarification on the difference between using an aggregated $\mathcal{A}$ matrix (similar to whole batch training) versus training each dataset separately with their own adjacency matrices (akin to mini-batch training). I would appreciate it if the authors could explain this distinction using straightforward mathematical language.

Response: Thank you for your comments. During batch integration, information exchange occurs through the updating of shared neural network parameters. Notably, batch integration does not necessarily require establishing direct connections or proximity between nodes from different slices to achieve integration. Based on our previous observations in the embedding space, this integration capability stems from the GCN model's generalization ability to learn common mapping patterns. At the beginning of iteration, nodes from different slices are initially clustered together in the latent space. Through self-supervised learning, the model gradually moves nodes of the same category from different slices in the same direction, thereby achieving integration and clustering.

Reviewer 2 found our explanation reasonable and suggested testing integration capability limits using synthesized data with incrementally increasing batch effects. Following their suggestion, we used the scDesign3[1] method to artificially create batch effects of varying magnitudes by adjusting the gene BATCH marginal between slices. Using scDesign3, we determined that the marginal

distributions between different donors' DLPFC datasets (151507 and 151675) approximated a normal distribution with mean=0.12 and SD=0.23. We gradually increased these mean and SD values in our experiments. The results showed that batch effects could be effectively eliminated when mean≤0.5 and SD<0.8. At mean=0.8, batch effects could be well-eliminated when SD<0.2. These test results demonstrate that our current model can integrate batch effects across a wide range. Furthermore, users can pre-evaluate batch effects between slices using scDesign3, ensuring that integration is not performed blindly.

Regarding your second question about the difference between whole batch and separate training:
For whole batch training, the GCN computation formula is:

$$H = \sigma(AXW),$$

where A is the merged adjacency matrix, X is the input data, W represents the GCN parameters, and σ is the activation function.

For separate training, the GCN computation formulas are:

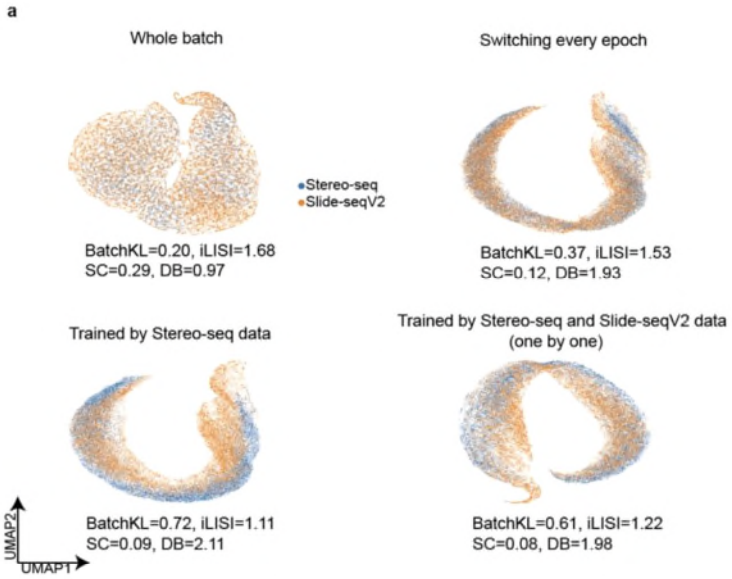
$$H_1 = \sigma(A_1X_1W_1), H_2 = \sigma(A_2X_2W_2),$$

where A_1, A_2 are adjacency matrices for the two datasets, X_1, X_2 are their respective input data, W_1 and W_2, are independent GCN parameters, and σ is the activation function.

From a formulaic perspective, the distinction between whole batch and separate training lies in the iteration process. In whole batch training, the loss is calculated as the average of all points across all slices, with W's gradient updates computed based on the total loss. In separate training, W_1 and W_2 are updated independently, with each backward gradient calculated based on the loss from corresponding slice nodes. This process amplifies differences during iteration, causing nodes from different slices to map to distinct locations in the latent space.

According to Hu et al.'s research[2], mini-batch performance is inferior to whole batch training. Additionally, their experiments involving dataset division into training and test sets for GCN training revealed that test set points did not completely align with training set points in UMAP visualization, further demonstrating the importance of shared parameters.

To further validate these findings, we conducted cross-platform slice integration experiments under four scenarios: First, whole batch training. Second, training using the Stereo-seq platform model and generating embeddings for both Stereo-seq and slide-seqV2 platform datasets. Third, sequential training - first with the Stereo-seq platform dataset, followed by the slide-seqV2 platform dataset. Fourth, alternating training between Stereo-seq and slide-seqV2 platform datasets (switching every epoch).



The UMAP visualization results are presented in Supplementary Figure S34a. The results clearly demonstrate that only whole batch training ensures proper slice integration. Alternating training ranks second in performance, though its SC and DB metrics are suboptimal. The other two approaches fail to achieve integration, exhibiting high BatchKL values and very low iLISI scores.

We have added this section to Fig. S34a and lines 644-675 of the Supplementary Material and to Lines 310-313 of the main text.

Comment: Systematic Hyperparameter Selection: I urge the authors to adopt a more systematic approach to hyperparameter selection. For instance, they should consider the four types of feature gene selection methods: selecting highly variable genes (HVGs) 1, minimizing redundancy in gene selection2, 3, utilizing co-expression gene networks4, 5, and selecting spatially variable genes (SVGs) 6, 7. At a minimum, the authors should cite these methods and, if feasible, conduct tests to demonstrate a rigorous and comprehensive approach to hyperparameter selection..

Response: Thank you for your comments. Following your suggestion, we have added citations for all the methods you mentioned in the main text (Lines 359-366) and implemented these four gene selection methods in our code. Considering Python environment compatibility, we chose mean.var.plot, which is also an empirical distribution-based method, as an alternative for selecting highly variable genes. The detailed implementation methods and performance evaluation can be found in the supplementary materials, Lines 678-695.

Reference

[1] Song D, Wang Q, Yan G, et al. scDesign3 generates realistic in silico data for multimodal single-cell and spatial omics[J]. Nature Biotechnology, 2024, 42(2): 247-252.

[2] Hu W, Fey M, Zitnik M, et al. Open graph benchmark: Datasets for machine learning on graphs[J]. Advances in

neural information processing systems, 2020, 33: 22118-22133.

Response to Reviewer #2:

Comment: The authors' rationale that model generalizability is the key to its batch correction capability makes sense. The challenge of batch effects in many existing embedding techniques (e.g., PCA or VAE) stems from their requirement to reconstruct the original expression data, thereby preserving any major data variation in the embeddings, including batch effects. Common strategy to mitigate this issue include incorporating "batch" as a covariate in the embedding process, allowing batch variation to be explained away[1], and explicitly pulling batches together with cross-batch anchors[2]. Alternatively, eliminating the need for embeddings to reconstruct the entire expression profile offers a third solution. Early attempts using supervised classification[3] or STAIG's use of self-supervised contrastive learning fall into this category. By eliminating the data reconstruction requirement, embeddings are less inclined to retain batch variations, and more likely to use up all explanation power for capturing within-batch variations, which leads to integration when combined with generalizability across batches.

Nevertheless, the absence of a theoretical guarantee means that the effectiveness of STAIG in batch correction could vary, particularly as "generalizability" of deep learning models can be unstable, and really depends on the extent of distribution shift between batches. It is conceivable that STAIG's batch correction capabilities might falter as batch effects intensify beyond a certain generalizable limit. Demonstrating this through synthetic augmentation of batch effects between slices could help identify STAIG's operational limits, and provide valuable guidelines for users regarding the reliability of STAIG in batch effect correction under varying conditions.

Response: Thank you for your valuable comments. Following your suggestion, we utilized the scDesign3 tool to simulate and generate spatial transcriptomics data. Since scDesign3 is one of the few tools that meet our testing requirements and can specifically generate simulated data for the Visium platform, we focused our tests on this platform. We tested STAIG's integration capability by artificially adjusting batch effects. The detailed content can be found in supplementary materials Lines 697-711 and Table S4:

The integration capacity of STAIG stems from its strong generalization capability. To further examine the limits of STAIG's integration performance, we used scDesign3[1] to artificially modify batch effects in the data, allowing us to systematically explore the model's integration threshold.

Specifically, we applied scDesign3 to data from two donors (#151507 and #151675) in the DLPFC dataset. Initially, we measured the original batch marginal distribution (Supplementary Fig. S34c), approximated a normal distribution with a mean of 0.12 and an SD of 0.23. We then simulated datasets with specified batch effect intensities by varying the mean and SD values. The mean values used were 0.1, 0.3, 0.5, and 0.8, while the SD values were set to 0.2, 0.5, and 0.8. STAIG's integration capability was tested on each combination. The results are presented in Supplementary Table S4, with UMAP visualizations of selected combinations shown in Supplementary Fig. S34d. Based on previous integration metrics, we set the success thresholds for integration at a BatchKL below 0.5 and iLISI above 1.5. As shown in the table, when the mean is below 0.3, STAIG achieves good integration across all SD values. For mean values of 0.5, integration remains effective with SDs

below 0.5, while for a mean of 0.8, integration is still effective when SD is 0.2.

These results demonstrate that STAIG maintains a high integration capacity even under substantial batch effects. Users can apply scDesign3 to test whether the batch effects in their data fall within STAIG's effective range, helping ensure optimal application of the model.

Additionally, we have summarized these results in the main text (Lines 306-310):

We further evaluated integration capabilities of STAIG by utilizing scDesign3 to generate simulated data with controllable batch effect magnitudes. The detailed results can be found in the supplementary materials under " Exploring the integration capacity of STAIG with artificially modified batch effects.". The results demonstrate that STAIG can effectively perform integration under varying degrees of batch effects. By leveraging scDesign3, users can assess whether their data falls within STAIG's integration capacity limits.

Reference

[1] Song D, Wang Q, Yan G, et al. scDesign3 generates realistic in silico data for multimodal single-cell and spatial omics[J]. Nature Biotechnology, 2024, 42(2): 247-252.

Key results

In this study, the authors employ a graph contrastive learning algorithm (STAIG) with the help of H&E images to generate embedded representations of spatial spots across multiple ST datasets. These embeddings are applicable to spatial clustering and batch integration. The visual feature embeddings from H&E images, not directly incorporated into the final embeddings, are mainly used to generate a probability matrix for augmenting the graph of each dataset. The authors conducted a series of experiments using various ST datasets to show the STAIG's superiority in discerning domains in tissues including human and mouse brains, human breast cancers, and zebrafish melanoma. They also attempt to demonstrate STAIG's exceling in integrate ST datasets derived from both vertical and horizontally alignable tissue slices. Finally, authors claims that STAIG can identify tumor microenvironments with elevated expressions of tumor-related marker genes.

Validity

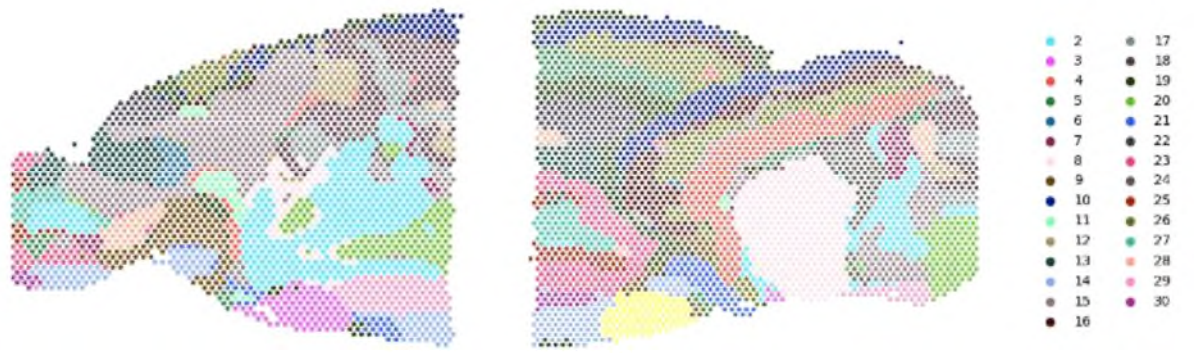
I have some concerns about the validity of the results:

- Using STAIG's example codes, we fail to obtain the spatial clustering results as good as those claimed in the manuscript:

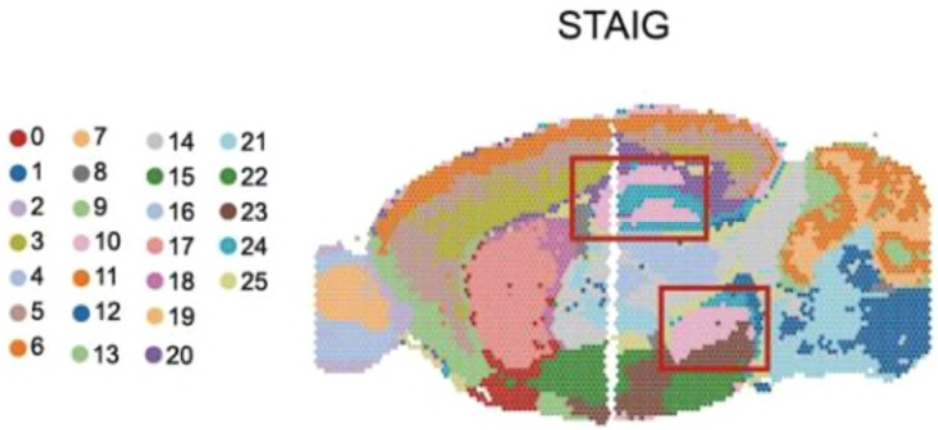
  For the DLPFC 151673 dataset (Fig.2c), the ARI for the spatial clustering results is 0.61, which is significantly lower than the reported 0.68 and the competing method GraphST (0.64).

- STAIG is essentially a method for single dataset rather than for multiple datasets, foundationally not very different from existing methods like SpaGCN. This can be told from the calculations of the block diagonal adjacency matrix A in equation (1) and the edge weight matrix W in equation (3). That is, spots from different datasets never interact with each other, with no information exchanged among them. Basically, the model should behave same as the case where the model is trained using the datasets one by one. Indeed, in our attempts to repeat the results for integrating horizontal slices, which have larger batch effects, using the provided example codes, we failed to reproduce the results in Fig.5c:

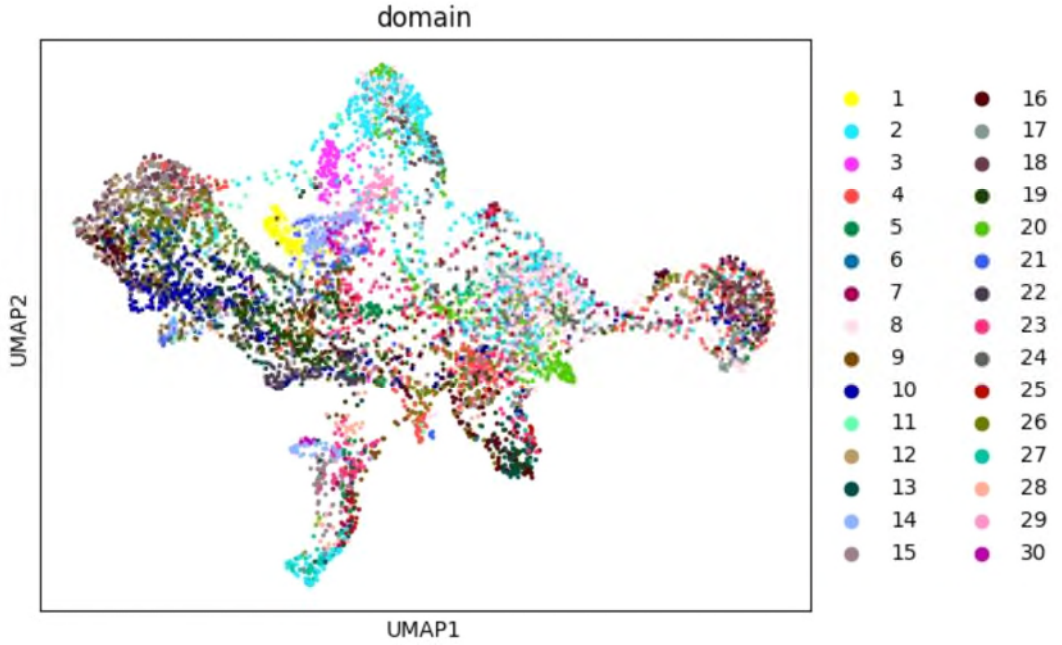
  Our results:

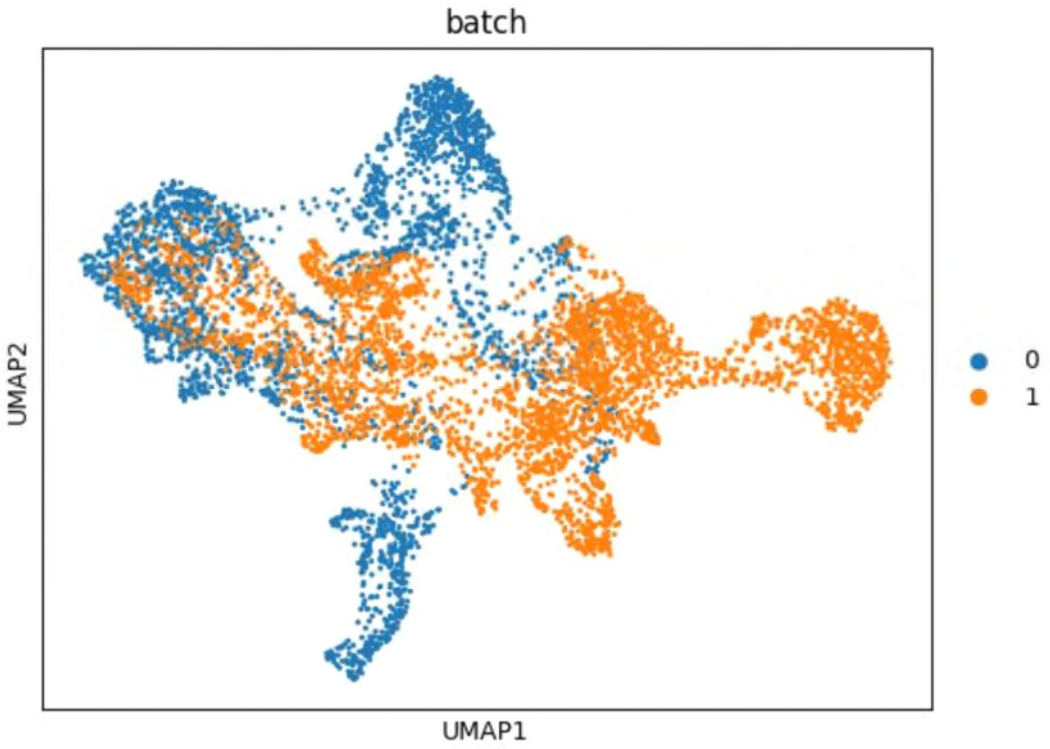


Results in Fig.5c:



Additionally, the UMAPs indicate suboptimal performance of the integration. As the two batches are well-separated while spots of same domains are dispersed. See below:





- In Fig.3, authors claim the image features generated by BYOL can reflect the DLPFC stratifications. These results seem to be too good given the limitations of BYOL under this scenario, as well as the fact that my naked eyes can hardly recognize such stratifications. Therefore, the validity of Fig.3 needs further clarifications. Indeed, in our experiment using their codes to repeat Fig.3b, we failed to obtain a result as good as the one shown in the manuscript as shown below:
  Our experiment:

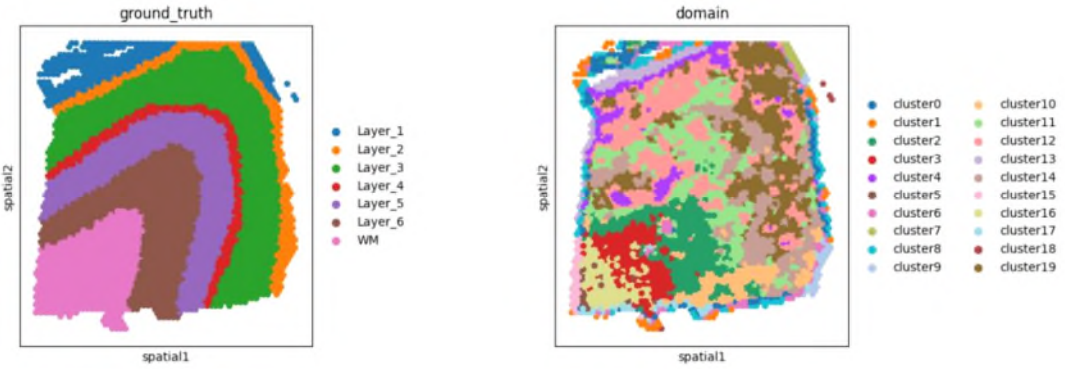
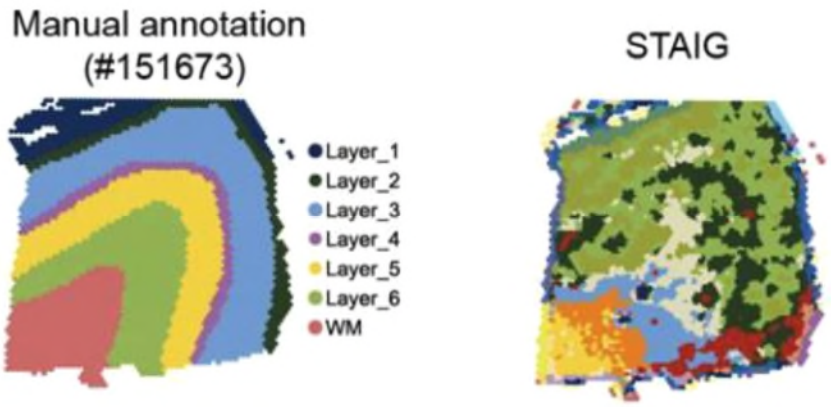


  Fig.3b of the manuscript



Significance

The contributions of this study include:

- Introduce a novel way of generating augmented ST graph for contrastive learning, based on H&E images.
- The endeavor to identify TME from tissue regions labeled as healthy.

Despite these contributions, this study has limited significances in terms of its novelty in both targeted problems and the methodology:

- The work focuses on generating spot embeddings for spatial clustering and batch alignment tasks, both of which have been extensively studied in recent years. To name a few, STAGATE[1], GraphST[2], STAligner[3], SpaGCN[4] and conST[5].
- Although the authors claim STAIG is alignment-free, they do not consider batch effects among datasets but simply putting them together, which makes STAIG no different from methods specifically designed for single dataset (e.g., SpaGCN). So STAIG should not be considered as a method for handling multiple samples.
- From the perspective methodological framework, it is very similar to conST in aspects including visual feature extraction and the idea of using contrastive graph learning.
- STAIG can hardly be recognized as a multimodal learning algorithm as the visual features are only used for graph augmentation rather than being incorporated into the final embeddings.
- STAIG is very slow compared to other similar methods like SpaGCN and STAGATE, as the clustering of a single dataset can take up to 4 hours.

Data and methodology

The data used in this study is public available, I have no particular concern about it. For the methodology, I have the following concerns:

- The authors use BYOL to extract cell morphology features from H&E image. It may be unnecessary or even worsen the extracted features, as it is well-documented that a simple U-Net can effectively handle this task[6]. Authors need to justify their choice here.
- BYOL needs data augmentation. Unlike general images (e.g., ImageNet), the data augmentation techniques may not be applicable here, as histology images typically lack clear semantic segments (i.e., kind of fuzzy and blurred), potentially leading to significant biases. Authors need to clarify the usefulness of the data augmentation techniques.
- The parameters of the band-pass filter are not explicitly stated in the paper, which is a potential pitfall, given the different qualities between tissue slices or even within the same slice. An arbitrary setting may severely deteriorate the image (e.g., over-blurring of the image). Authors need to explain the standards for choosing these parameters.
- The normalization of the aggregated raw gene expression matrix X on line 296 needs further explanation. This normalization potentially downweighs the importance of datasets with low sequencing depth, which yet may hold important information.
- The slice integration seems unnecessary, as the way A constructed in equation (1) and W constructed in equation (3) means spots in different datasets would never interact with each other. So, this slice integration step is meaningless. Please comment on this.
- The authors used the DS algorithm, which essentially a K-means, to exclude negative samples as those within the same cluster. This step needs further clarification as the cluster size generated by K-means can have a large variance. This means many spots can end up in a single cluster and many cluster can have only one spot.
- Please clarify the default value for the probability matrix (i.e., without labels) on line 328.
- In equation (7), what are the parameters for the Bernoulli sampling?

Analytical approach

I have the following concerns:

- The impacts of batch effects on slice integration should be analyzed more thoroughly, given the batch effects among vertical DLPFC slices are minor.
- The efficacy of image feature extraction needs more detailed analysis. For example, which part of BYOL is most critical for this effective feature extraction. Ablation studies or comparison with another computer vision representation learning framework may help on this point.
- The choice of the significance threshold log2FC for DE analysis should be discussed.
- The statistical tests for the GO analysis (BH corrected test?) should be stated.
- In addition to ARI or SC, the results should also show one more metric (e.g., NMI).
- In addition to the UMAPs, some metrics should be used to measure the batch mixing effects (e.g., BatchKL).

- In the paragraph (line 206-208), the authors claim differential expressions of muscle and tumor marker genes in the two clusters of slice A and B. However, it is really difficult to tell such differences from Fig.7f.
- The study of TME in Fig.7 should include a competing method TESLA[7] specifically designed for detecting TME using ST data.

Suggested improvements

The suggested improvements are:

- The authors should add an experiment for discussing the use of different methods (e.g., kernels) other than Euclidean distance for constructing the image distance matrix D on line 311.
- The authors can include an additional experiment for testing datasets without H&E images.
- The choice of masked ratio in equation (7) should be discussed, as the quality of the final embeddings hinges on this choice.
- The ability of STAIG in integrating non-vertical and non-horizontal slices (e.g., two slices of the same tissue but from different individuals in different studies, and cannot be horizontally stitched) should be analyzed.

Clarity and context

- On line 145, the "(Fig.4f and Supplementary Fig. S7)" appears twice in a row.
- On line 197, the traditional ST methods should be explicitly named.
- In Fig.7f legend, it should be fine-grained instead of "fine-tuned".

References

- The H&E image patch segmentation idea is also very similar to HistToGene[8], so this paper should be added to reference.
- Another ST computational methods, TESLA[7], for studying TME should be referred and compared.

Reports References

1. Dong, K. & Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nat Commun* **13**, 1739 (2022).
2. Long, Y. et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat Commun* **14**, 1155 (2023).
3. Zhou, X., Dong, K. & Zhang, S. Integrating spatial transcriptomics data across different conditions, technologies and developmental stages. *Nature Computational Science* **3**, 894-906 (2023).

4. Hu, J. et al. SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nat Methods* **18**, 1342-1351 (2021).
5. Zong, Y. et al. conST: an interpretable multi-modal contrastive learning framework for spatial transcriptomics. *bioRxiv*, 2022.2001. 2014.476408 (2022).
6. Falk, T. et al. U-Net: deep learning for cell counting, detection, and morphometry. *Nature methods* **16**, 67-70 (2019).
7. Hu, J. et al. Deciphering tumor ecosystems at super resolution from spatial transcriptomics with TESLA. *Cell systems* **14**, 404-417. e404 (2023).
8. Pang, M., Su, K. & Li, M. Leveraging information in spatial transcriptomics to predict super-resolution gene expression from histology images in tumors. *bioRxiv*, 2021.2011. 2028.470212 (2021).