Supplemental Material for Hagrid — Gridify Scatterplots with Hilbert and Gosper Curves

Rene Cutura
rene.cutura@tuwien.ac.at
TU Wien
Austria

Cristina Morariu
cristina@morariu.ro
University of Stuttgart
Germany

Zhanglin Cheng
zhanglin.cheng@gmail.com
Shenzen Institutes of Advanced Technology
China

Yunhai Wang
cloudseawang@gmail.com
Shandong University
China

Daniel Weiskopf
Daniel.Weiskopf@visus.uni-stuttgart.de
University of Stuttgart
Germany

Michael Sedlmair
Michael.Sedlmair@visus.uni-stuttgart.de
University of Stuttgart
Germany

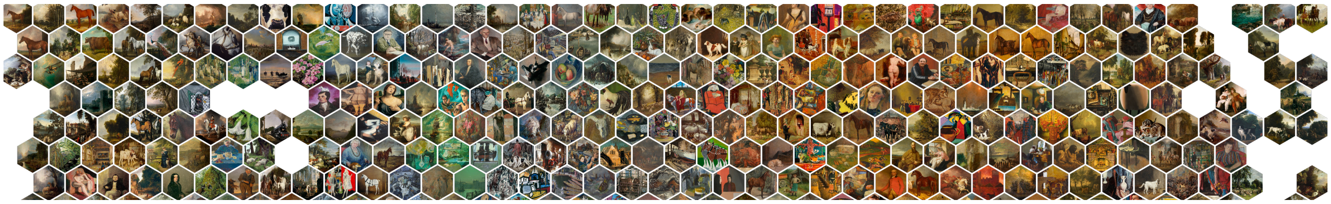


Figure 1: A detail view on a section of the t-SNE projection of the *Art UK Paintings* dataset. Gridified with HAGRID$_{\mathrm{GC}}$.

CCS CONCEPTS

- **Human-centered computing → Visualization techniques**.

KEYWORDS

Space-filling curve, Grid layout, Neighborhood-preserving.

ACM Reference Format:

Rene Cutura, Cristina Morariu, Zhanglin Cheng, Yunhai Wang, Daniel Weiskopf, and Michael Sedlmair. 2021. Supplemental Material for Hagrid — Gridify Scatterplots with Hilbert and Gosper Curves. In *The 14th International Symposium on Visual Information Communication and Interaction (VINCI '21), September 6–8, 2021, Potsdam, Germany*. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3481549.3481569

A INTRODUCTION

This supplemental material contains six parts: (B) Details on the datasets that we used for the quantitative evaluation in the paper, (C) details on the used metrics for the evaluation, (D) a more detailed representation of the results of this quantitative evaluation, (E) a qualitative illustration of HAGRID being applied to the Art UK Paintings dataset, (F) two use cases of HAGRID applied in interactive scenarios, and (G) a more detailed analysis of HAGRID's limitation w.r.t. datasets with extreme outliers.

B DATASETS

Table 1 shows the complete list of all datasets used in our evaluation. As already mentioned in Section 4 of our main document, we have used a total of 60 real and synthetic datasets to produce the scatterplots used in the evaluation. 54 of these datasets were proposed in the work by Sedlmair et al. [1, 22]. This data collection consists of both real and synthetic datasets. We have selected this dataset as it was previously used in a study relating to dimensionality reduction (DR) projections [22]. We have augmented this data with six additional collections of images including well known machine learning benchmarks [6, 16, 25], as well as other art and photography collections[5, 17, 20].

From the 60 datasets collected, we have generated 1695 dimensionality reduced projections using both linear and non-linear DR algorithms [2–4, 12, 15, 19, 21, 23, 24], as well as a parameter search for the parametric levels. We uniformly sampled 339 projections by taking the size of the dataset into account.

C EVALUATION METRICS

Neighborhood Preservation (NP, originally also called $AUC_{log}RNX$ [13]) is a metric that enhances the metric k-Neighborhood Preservation (NP_k) [18] by aggregating the measurements for all neighborhood sizes k. The metric NP_{k} is used by nearly all methods mentioned in the related work section [7, 8, 10, 14]. It calculates the average percentage of the k-nearest neighbors of each box that are preserved in the final layout. It takes a value between 0 and 1.

VINCI '21, September 6–8, 2021, Potsdam, Germany
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8647-0/21/09…$15.00
https://doi.org/10.1145/3481549.3481569

We calculate NP_k as follows:

$$NP_k = \frac{1}{n \cdot k} \sum_{i=1}^{n} \left| V_i^{(k)}(X) \cap V_i^{(k)}(Y) \right|,$$

where X is the original set of points, Y is the gridified set of points. $V_i^{(k)}$ returns the set of the k-nearest neighbors of the i-th point of the respective set of datapoints, and n is the total number of points in the data. These values are scaled and aggregated as follows:

$$\begin{aligned} \widetilde{NP}_k &= \frac{(n-1)NP_k - k}{n-1-k}, \\ NP &= \frac{\left(\sum_{k=1}^{n-2} \widetilde{NP}_k / k\right)}{\left(\sum_{k=1}^{n-2} 1/k\right)}. \end{aligned} \tag{1}$$

Values between 0 and 1 are possible, where 1 means perfect neighborhood preservation, and 0 means no neighborhood preservation.

Cross-Correlation (*CC*) measures the distance correlation between pairwise distances in the original layout compared to ones in the new layout. The distance can be interpreted as a measure of dissimilarity between the points; ideally, dissimilar points in the original layout remain as such in the new one. Hilasaca and Paulovich [10] also use this measure in their evaluation of DGrid. The *CC* measure is defined as:

$$CC = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\left(\delta(y_i, y_j) - \overline{\delta}_Y\right) \cdot \left(\delta(x_i, x_j) - \overline{\delta}_X\right)}{\sigma_X \cdot \sigma_Y}, \tag{2}$$

where x_i and x_j are points belonging to X (the original set of points), y_i and y_j are points belonging to Y (the gridified set of points), σ_X and σ_Y are the respective standard deviations, and $\overline{\delta}_X$ and $\overline{\delta}_Y$ are the respective mean distances between any pair of points.

This measure should be interpreted the same way as any correlation coefficient would be. If the value is close to -1, the pairwise distances are negatively correlated, i.e., points that used to be close together are now far away from each other. If the value is around 0, it means that there is no relationship between the pairwise distances of the original and new layouts. Ideally, this measure takes the value of 1: then, points that were close together stay together, and points further away from each other remain far away.

Euclidean Distance (*ED*) is the average distance between the original points and their gridified counterparts:

$$ED = \frac{1}{n} \sum_{i=1}^{n} \delta\left(x_i, y_i\right), \tag{3}$$

where x_i and y_i correspond again to the original and the final position of our data points. We use Euclidean distance as a measure for the global structure of the visualization. The main objective of methods such as NMap and DGrid is a space-filling visualization, rather than the preservation of global structure. Therefore, we expect them to perform poorly with respect to Euclidean distance. In our case, intuitively, in order for the global structure to change as little as possible, the points should be moved to a position as close as possible. The higher the average Euclidean distance is, the worse the global structure of the graph is preserved.

Size Increase (*SI*) is the ratio of the area of the convex hull of the initial scatterplot (C_X) and the area of the convex hull of the gridified version (C_Y):

$$SI = \frac{area(C_Y)}{area(C_X)}. \tag{4}$$

This measure is used for global structure preservation. Ideally, the resulting grid has a similar convex hull to the initial one. While the range for values of size increase is $]0, \infty[$, 1 is the optimal value. Size Increase has also been used in previous evaluations [7, 9]. Due to the space-filling aspect of DGrid, this metric will be expected to have worse results.

Run Time (*RT*) is the total time needed for one technique to compute the gridified version from the original scatterplot. All the techniques we evaluated against also examine the running performance of their methods in terms of time.

D ADDITIONAL RESULTS

Figure 4 - 6 show an alternative and more detailed representation of the results of quantitative evaluation. These figures show the metric results for each dataset separately, while the version in the paper shows the results aggregated by groups of datasets with similar number of points.

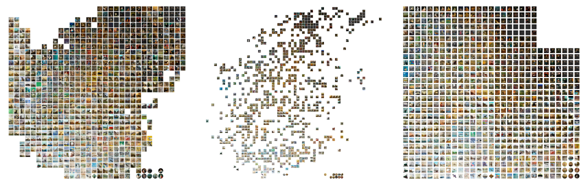
E EXAMPLE OF ART UK PAINTINGS

We include the full *Art UK Paintings* t-SNE projection that is used in the teaser image of the paper. The *Art UK Paintings* is a dataset consisting of 7792 images. Figure 7a is the original t-SNE layout, and Figure 7b is the full-scale figure of the resulting Gosper layout after using Hagrid.

F USE CASES

Our primaray goal with Hagrid was to find a good balance between removing overlap, preserving scatterplots structure, while having a fast runtime. To illustrate the value of this combination, we now provide two use cases: adjusting the curve level to create different types of layouts, and realizing an interactive lens view.

F.1 Interactively Adjusting Level and Point Size

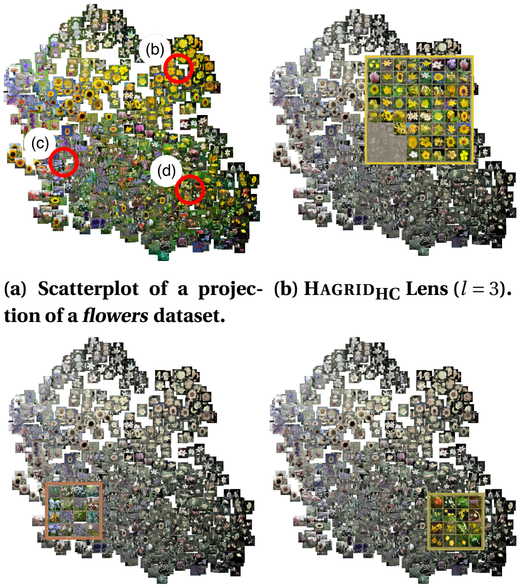


(a) Hagrid$_{HC}$ ($l = 5$) **(b) Hagrid$_{HC}$ ($l = 6$)** **(c) DGrid**

Figure 2: Gridified versions of the t-SNE projection of the *Art UK Paintings* dataset [5] for different settings of the Hagrid$_{HC}$ level parameters and the DGrid layout.

By altering the level parameter of our technique, Hagrid gives a natural handle to deal with the trade-off between space-fillingness (cell size and space efficiency) and point position accuracy (global structure preservation). With a low level setting, Hagrid can achieve a more squared layout, similar to DGrid. An example is given in Figure 2, where we can see the difference between a layout with the minimum HAGRID$_{HC}$ level (5), one level higher (6) that outputs the grid, and the DGRID layout for reference. In our situation, the white space is still scattered around the convex hull of the initial layout. Interactively transitioning between different levels allows to step-wise translate a scatterplot into a space-filling map with HAGRID.

F.2 Lens View



(a) Scatterplot of a projection of a *flowers* dataset. **(b) HAGRID$_{HC}$ Lens ($l = 3$).**

(c) HAGRID$_{HC}$ Lens ($l = 2$). **(d) HAGRID$_{HC}$ Lens ($l = 2$).**

Figure 3: Usage of HAGRID as lens. The circles indicate the mouse position for the screenshots (b), (c), and (d).

Due to its computational efficiency, our approach can also be used to support interactive lenses such as the one proposed in Glyphboard [11]. In Glyphboard, DR projections are plotted as normal scatterplots and as the user zooms in, the dots are replaced with circular glyphs. The lens view in their implementation is using a force-directed layout algorithm to handle overlap, but that does not guarantee neighborhood preservation. Based on our comparison to CMDS, which closely resembles a force-directed layout algorithm, we believe our technique could fit this type of use case better. In Figure 3, we show a possible way of using HAGRID as part of a lens view. Here, we used a t-SNE projection of photographs of flowers as the source dataset [17]. For the overall scatterplot, we maintain the original scatterplot, with overlaps. However, in the lens view we use a low-level HC, to dynamically align the photos. As our approach is computationally efficient, it lends itself toward supporting such interactive features.

Table 1: This table provides a list of all the datasets used in the evaluation, their sizes (#), and the sampled projections for each dataset. For parametrics DRs, such as UMAP or t-SNE, multiple projections of the same type were sometimes sampled from one dataset. The algorithms used were PCA (`P`), t-SNE (`T`), UMAP (`U`), Isomap(`I`), Spectral Embedding, Random Gaussian Projection, and LLE (all falling under the "Other" `O` category).

dataset	#	O	P	T	U	I
cereal	58	✓	✓	✓		
grid6-4d	59	✓	✓	✓		
hiv	61	✓	✓	✓		
italianwines	76	✓	✓	✓		
n100-d10-c5-spr0.2-out0	79	✓	✓	✓		
n100-d5-c3-spr0.1-out0	79	✓	✓	✓		
n100-d5-c3-spr0.2-out0	80	✓	✓	✓		
n100-d5-c5-spr0.2-out0	80	✓	✓	✓		
grid10-3d	80	✓	✓	✓		
n100-d5-c5-spr0.1-out0	80	✓	✓	✓		
n100-d10-c3-spr0.2-out0	80	✓	✓	✓		
n100-d10-c3-spr0.1-out0	80	✓	✓	✓		
n100-d10-c5-spr0.1-out0	80	✓	✓	✓		
fisheries-clusteredbyescapementtarget	93			✓		
fisheries-clusteredbyharvestrule	94			✓		
swanson	101	✓	✓	✓		
musicnetgroups	104	✓	✓	✓		
world-11d	114	✓	✓	✓		
iris	114	✓	✓	✓		
world-9d	117	✓	✓	✓		
boston	123	✓	✓	✓		
wine	141	✓	✓	✓		
worldmap	142	✓	✓	✓		
ms-interleaved-40	145	✓	✓	✓		
parkinsons-abs-croped	151	✓	✓			
bbdm13	159	✓	✓	✓		
tse300	194	✓	✓	✓		
mnis	200			✓		
ms-interleaved-60	231	✓	✓	✓		
ecoliproteins	263	✓	✓			
ionosphere	281			✓		
javiergenerateddata-3dinterleaved-4	312	✓	✓	✓		
unevendensity	339	✓	✓	✓		
breast	350	✓	✓	✓		
n500-d10-c3-spr0.1-out0	383	✓	✓	✓		
n500-d5-c5-spr0.1-out0	395	✓	✓	✓		
ms-interleaved-120	395	✓	✓	✓		
n500-d10-c5-spr0.1-out0	395					
n500-d10-c3-spr0.2-out0	399	✓	✓			
javiergenerateddata-3dinterleaved-5	399	✓	✓	✓		
n500-d5-c3-spr0.2-out0	399	✓	✓			
paintings	400			✓	✓	
n500-d5-c3-spr0.1-out0	401	✓	✓	✓		
n500-d10-c5-spr0.2-out0	401	✓	✓	✓		
n500-d5-c5-spr0.2-out0	403					
interleaved-100-200	428	✓	✓	✓		
olive	446	✓	✓	✓		
wdbc-class-1	449	✓	✓	✓		
javiergenerateddata-3dinterleaved-3	480	✓	✓	✓		
oxford-buildings-oxford	554		✓	✓	✓	✓
interleaved-100-500	678	✓	✓	✓		
twosquare	771	✓	✓	✓		
coil-100	792			✓	✓	
interleaved-250-500	871	✓	✓	✓		
flowers	880	✓	✓	✓	✓	✓
efashion	967	✓	✓			
yeast	1162					
ms-interleaved-400	1432	✓	✓	✓		
spambase	1657	✓	✓			
paris-buildings	1738	✓	✓	✓	✓	✓

G LIMITATIONS FOR SCATTERPLOTS WITH EXTREME OUTLIERS

Figure 8 - 11 show examples of scatterplots with extreme outliers. These types of scatterplots cannot be well handled by the current version of HAGRID as the grid is not adaptive to the extreme differences in point densities at the moment.

REFERENCES

[1] Michael Aupetit and Michael Sedlmair. 2016. SepMe: 2002 New visual separation measures. In *IEEE Pacific Vis. Symp. (PacificVis)*. IEEE, 1–8. https://doi.org/10.1109/PACIFICVIS.2016.7465244

[2] Mikhail Belkin and Partha Niyogi. 2003. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation* 15, 6 (2003),

1373–1396. https://doi.org/10.1162/089976603321780317

[3] Ella Bingham and Heikki Mannila. 2001. Random projection in dimensionality reduction: Applications to image and text data. In *Proc. ACM Intl. Conf. Knowledge Discovery and Data Mining (SIGKDD)*. 245–250. https://doi.org/10.1145/502512.502546

[4] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. 2011. Robust Principal Component Analysis? *J. ACM* 58, 3, Article 11 (June 2011), 37 pages. https://doi.org/10.1145/1970392.1970395

[5] Elliot Crowley and Andrew Zisserman. 2014. The State of the Art: Object Retrieval in Paintings using Discriminative Regions. In *Proc. British Machine Vision Conf.* BMVA Press.

[6] Li Deng. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine* 29, 6 (2012), 141–142. https://doi.org/10.1109/MSP.2012.2211477

[7] Felipe SLG Duarte, Fabio Sikansi, Francisco M Fatore, Samuel G Fadel, and Fernando V Paulovich. 2014. Nmap: A Novel Neighborhood Preservation Space-filling Algorithm. *IEEE Trans. Visualization & Computer Graphics (TVCG)* 20, 12 (2014), 2063–2071. https://doi.org/10.1109/TVCG.2014.2346276

[8] Erick Gomez-Nieto, Wallace Casaca, Luis Gustavo Nonato, and Gabriel Taubin. 2013. Mixed Integer Optimization for Layout Arrangement. In *Symp. Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 115–122. https://doi.org/10.1109/SIBGRAPI.2013.25

[9] Erick Gomez-Nieto, Frizzi San Roman, Paulo Pagliosa, Wallace Casaca, Elias S Helou, Maria Cristina F de Oliveira, and Luis Gustavo Nonato. 2013. Similarity Preserving Snippet-Based Visualization of Web Search Results. *IEEE Trans. Visualization & Computer Graphics (TVCG)* 20, 3 (2013), 457–470. https://doi.org/10.1109/TVCG.2013.242

[10] Gladys Hilasaca and Fernando V Paulovich. 2019. Distance Preserving Grid Layouts. *arXiv preprint* (2019). http://arxiv.org/abs/1903.06262

[11] Dietrich Kammer, Mandy Keck, Thomas Gründer, Alexander Maasch, Thomas Thom, Martin Kleinsteuber, and Rainer Groh. 2020. Glyphboard: Visual Exploration of High-Dimensional Data Combining Glyphs with Dimensionality Reduction. *IEEE Trans. Visualization & Computer Graphics (TVCG)* 26, 4 (2020), 1661–1671. https://doi.org/10.1109/TVCG.2020.2969060

[12] Joseph B Kruskal. 1964. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* 29, 1 (1964), 1–27. https://doi.org/10.1007/BF02289565

[13] John A Lee, Diego H Peluffo-Ordóñez, and Michel Verleysen. 2015. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing* 169 (2015), 246–261. https://doi.org/10.1016/j.neucom.2014.12.095

[14] Wilson E Marcílio-Jr, Danilo M Eler, Rogério E Garcia, and Ives R Venturini Pola. 2019. Evaluation of approaches proposed to avoid overlap of markers in visualizations based on multidimensional projection techniques. *Information Visualization* 18, 4 (2019), 426–438. https://doi.org/10.1177/1473871619845093

[15] Leland McInnes, John Healy, and James Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint arXiv:1802.03426* (2018). https://arxiv.org/abs/1802.03426

[16] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. 1996. Columbia object image library (coil-20). (1996).

[17] Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated Flower Classification over a Large Number of Classes. In *Conf. Computer Vision, Graphics & Image Processing*. IEEE, 722–729. https://doi.org/10.1109/ICVGIP.2008.47

[18] Fernando V Paulovich and Rosane Minghim. 2008. HiPP: A Novel Hierarchical Point Placement Strategy and its Application to the Exploration of Document Collections. *IEEE Trans. Visualization & Computer Graphics (TVCG)* 14, 6 (2008), 1229–1236. https://doi.org/10.1109/TVCG.2008.138

[19] Karl Pearson. 1901. LIII. On Lines and Planes of Closest Fit to Systems of Points in Space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2, 11 (1901), 559–572. https://doi.org/10.1080/14786440109462720

[20] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proc. IEEE Conf. on Comp. Vis. and Pat. Rec.* IEEE, 1–8. https://doi.org/10.1109/CVPR.2008.4587635

[21] Sam T Roweis and Lawrence K Saul. 2000. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290, 5500 (2000), 2323–2326. https://doi.org/10.1126/science.290.5500.2323

[22] Michael Sedlmair, Andrada Tatu, Tamara Munzner, and Melanie Tory. 2012. A Taxonomy of Visual Cluster Separation Factors. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 1335–1344. https://doi.org/10.1111/j.1467-8659.2012.03125.x

[23] Joshua B. Tenenbaum, Vin De Silva, and John C. Langford. 2000. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* 290, 5500 (2000), 2319–2323. https://doi.org/10.1126/science.290.5500.2319

[24] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. 9, Nov (2008), 2579–2605. http://jmlr.org/papers/v9/vandermaaten08a.html

[25] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms*. https://arxiv.org/abs/1708.07747

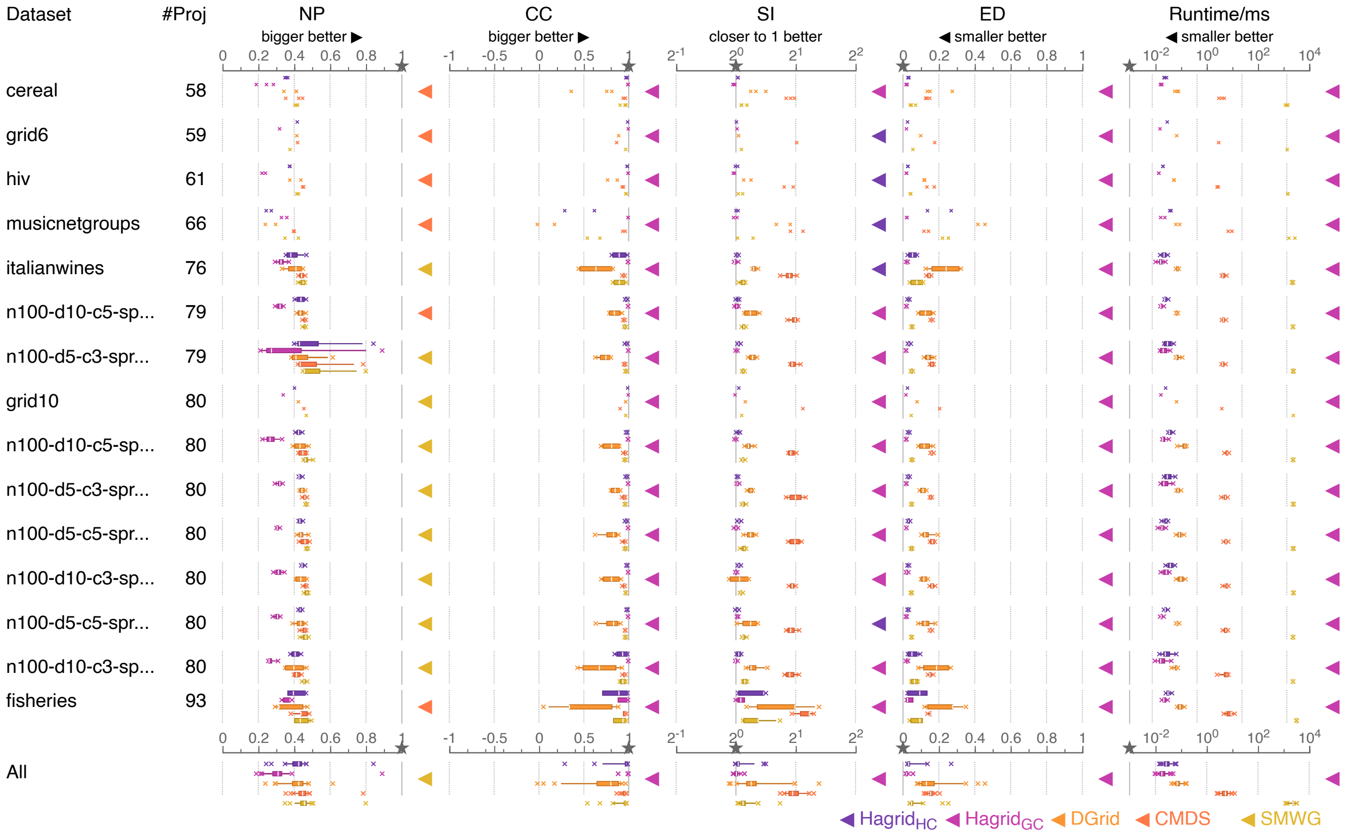


Figure 4: Results of quantitative evaluation by dataset, instead of by dataset size. The boxplots in this figure — and in the next two figures — show the median, the 25% and 75% quantiles, while the ends show the 5- and the 95 percentile of the respective data. The color of the triangles shows the respective best method, measured by the median of the particular metric. The results show that, apart from the *NP* metric, HAGRID performs better than the other techniques. Also the runtime ($\log_{10}$-scale) of HAGRID is better than of any other technique. The runtimes for CMDS are roughly 100 times higher, and for SMWG roughly 10.000 times higher.

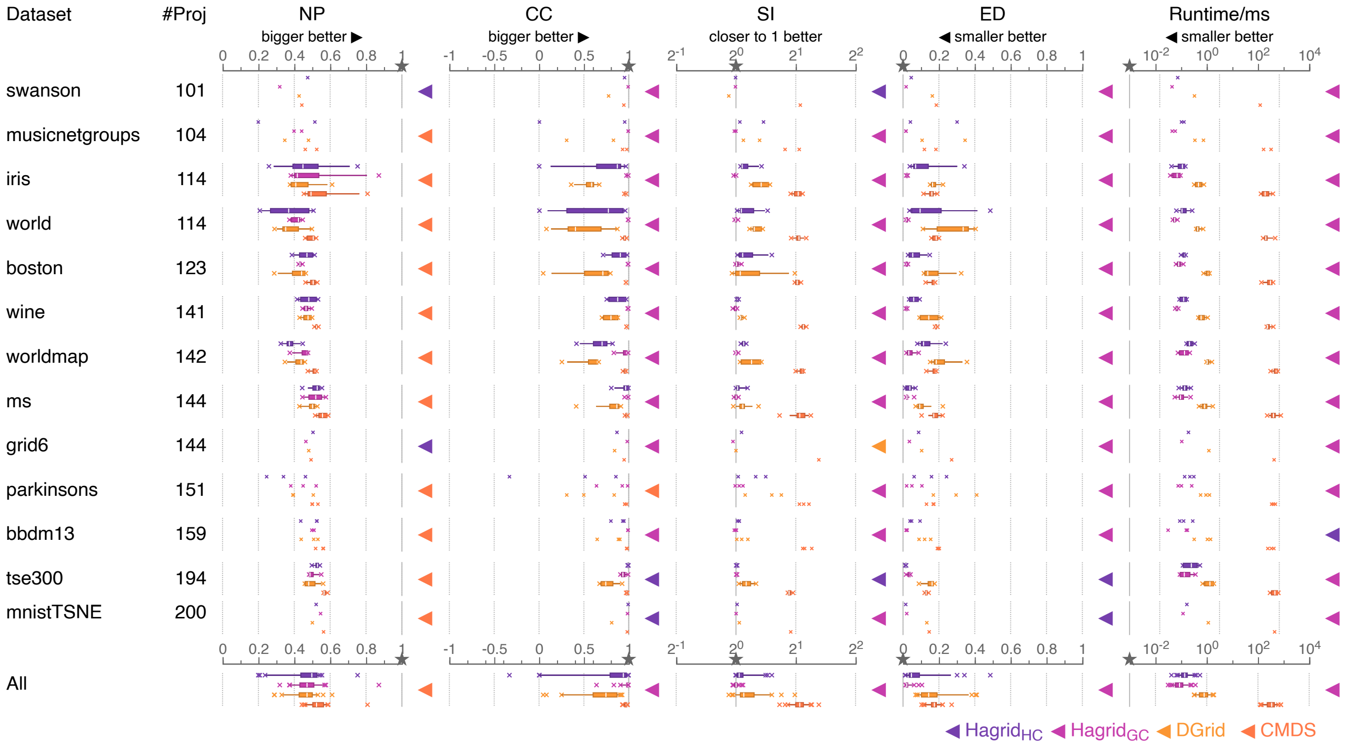


Figure 5: CMDS outperforms the other techniques in the *NP* metric, once in the *CC* metric, and once in *SI*, but needs at least 10 seconds. Else, HAGRID outperforms the other techniques.

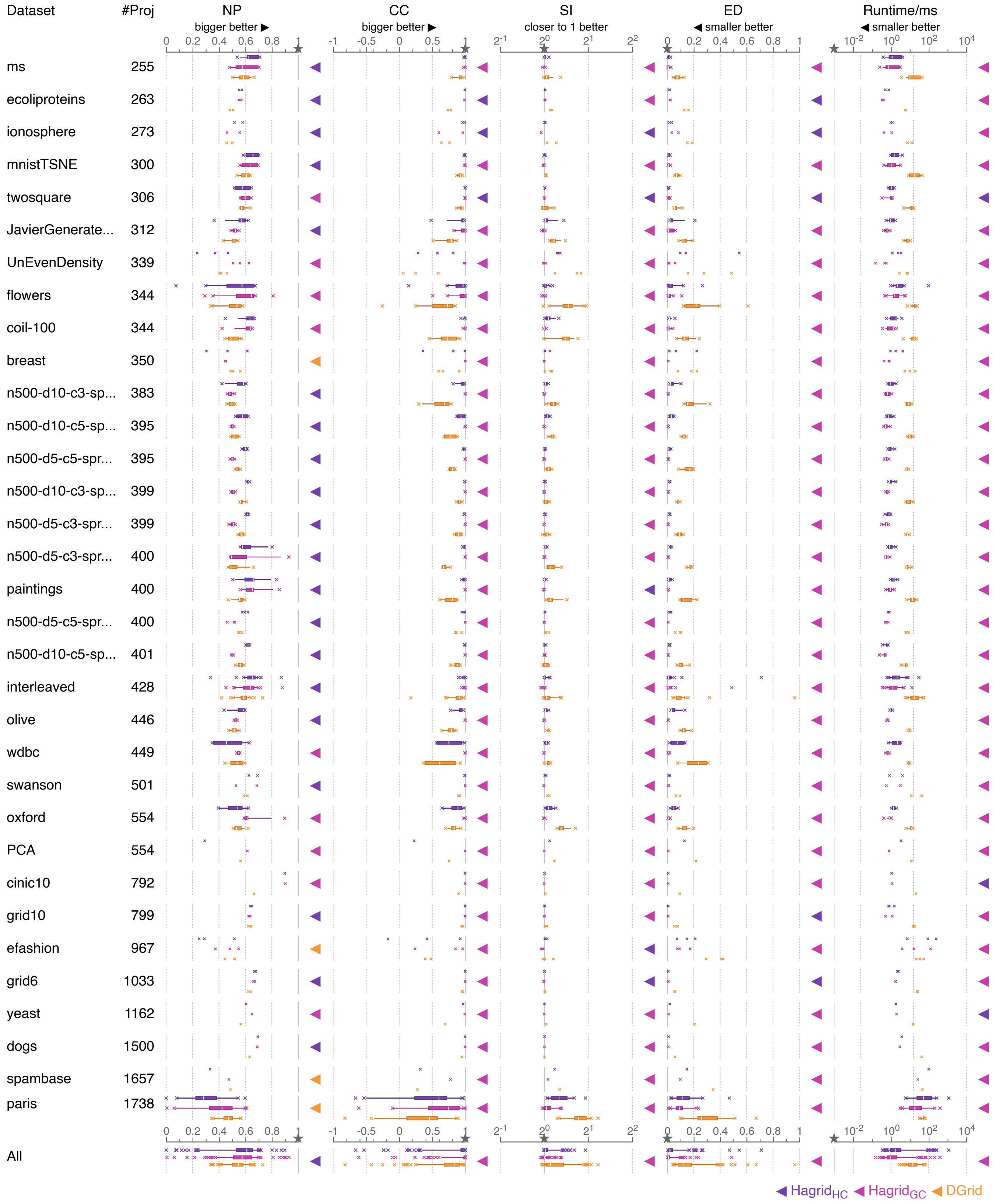
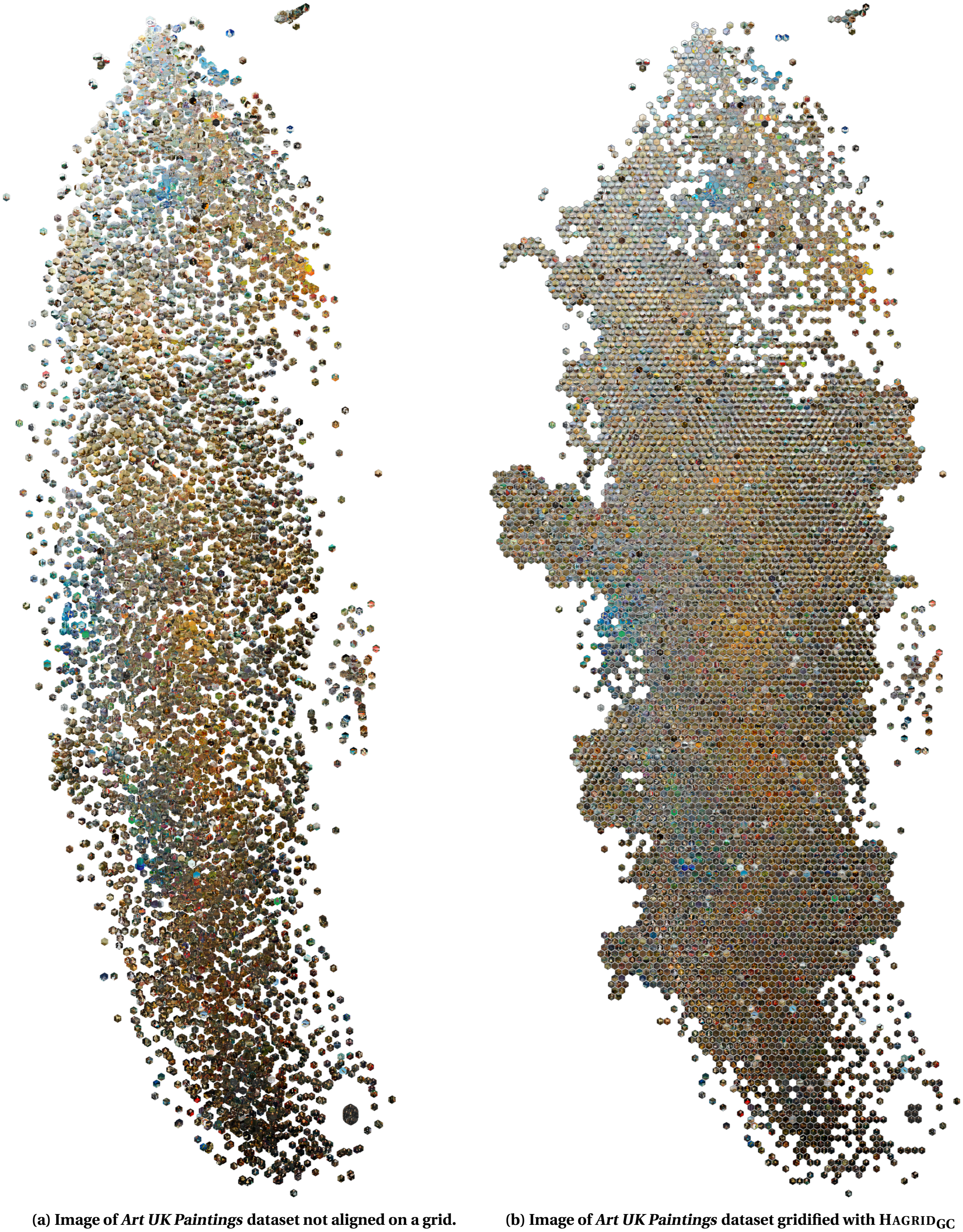


Figure 6: Results of datasets of size 250 up to 2000.Hagrid performs in all metrics better than DGrid with 4 exceptions, where DGrid performs better in the *NP* metric.



(a) Image of *Art UK Paintings* dataset not aligned on a grid.

(b) Image of *Art UK Paintings* dataset gridified with Hagrid$_{\text{GC}}$

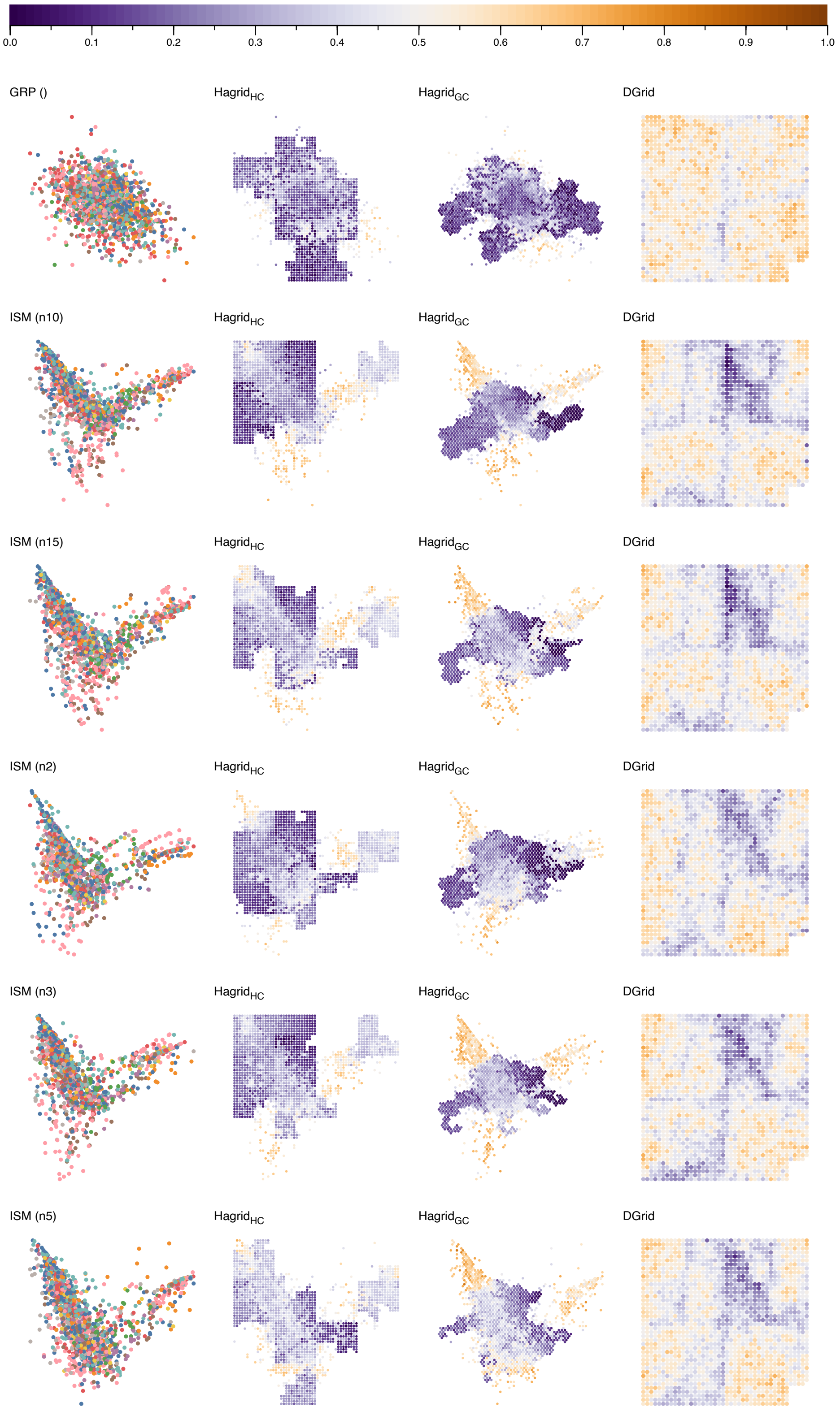


Figure 8: Scatterplots of projections of the biggest dataset, used in our evaluation (*paris buildings*. The points in the gridified version are color-coded by the *NP* metric. Both HAGRID$_{HC}$ and HAGRID$_{GC}$ are sensitive to outliers, often produced by projections of SE and LLE. DGRID is less sensitive to those outliers.

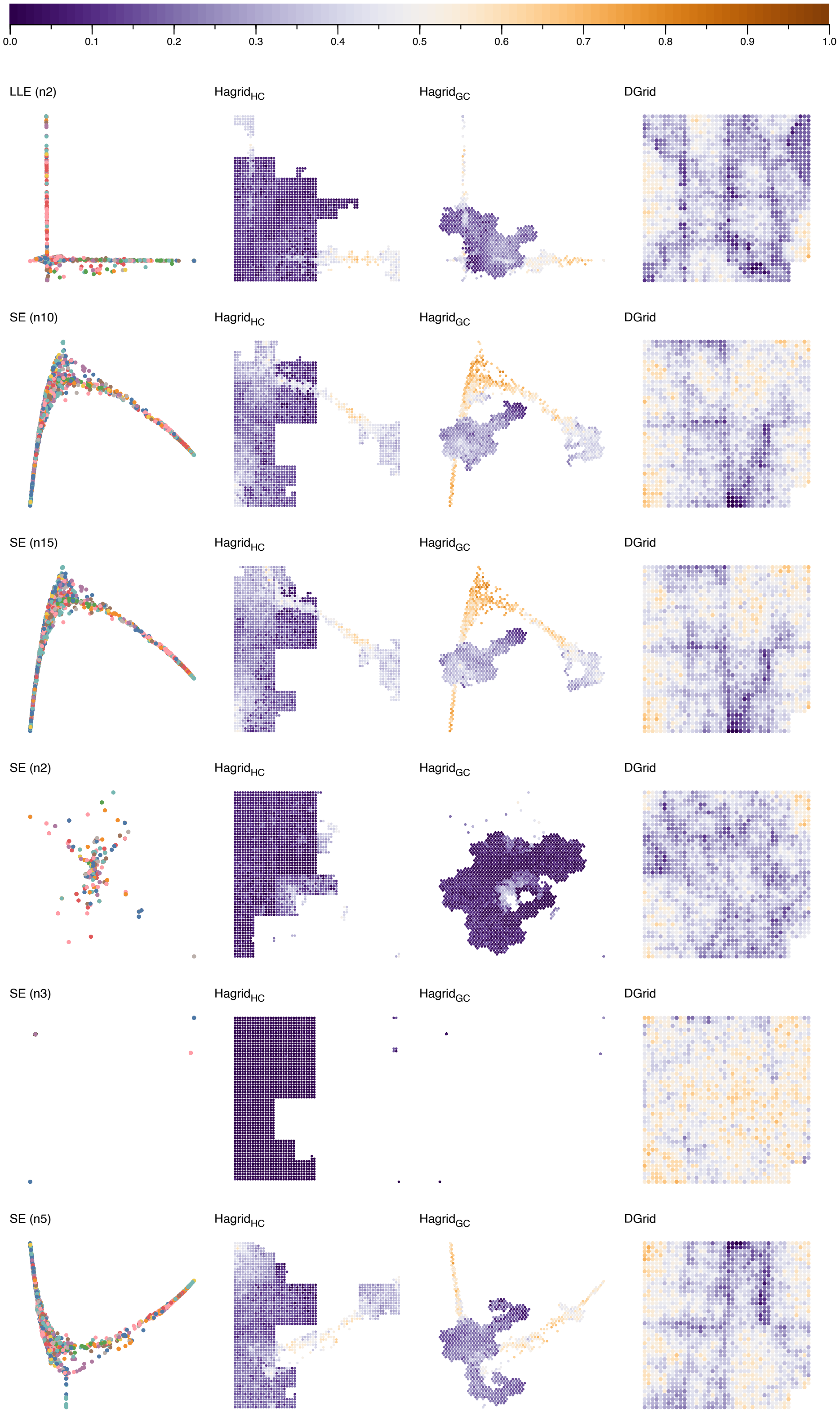


Figure 9: Figure 8 continued (1).

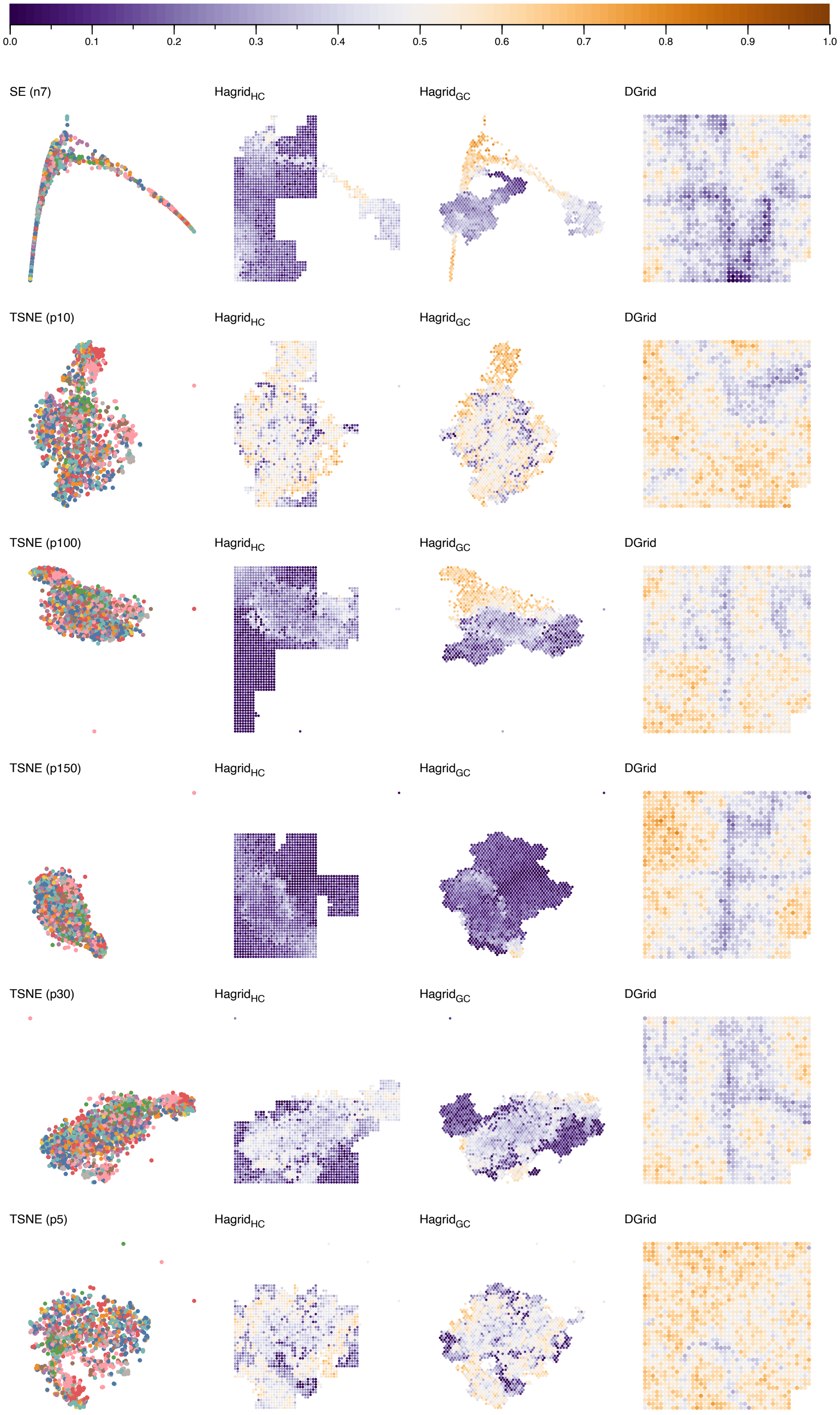


Figure 10: Figure 8 continued (2).

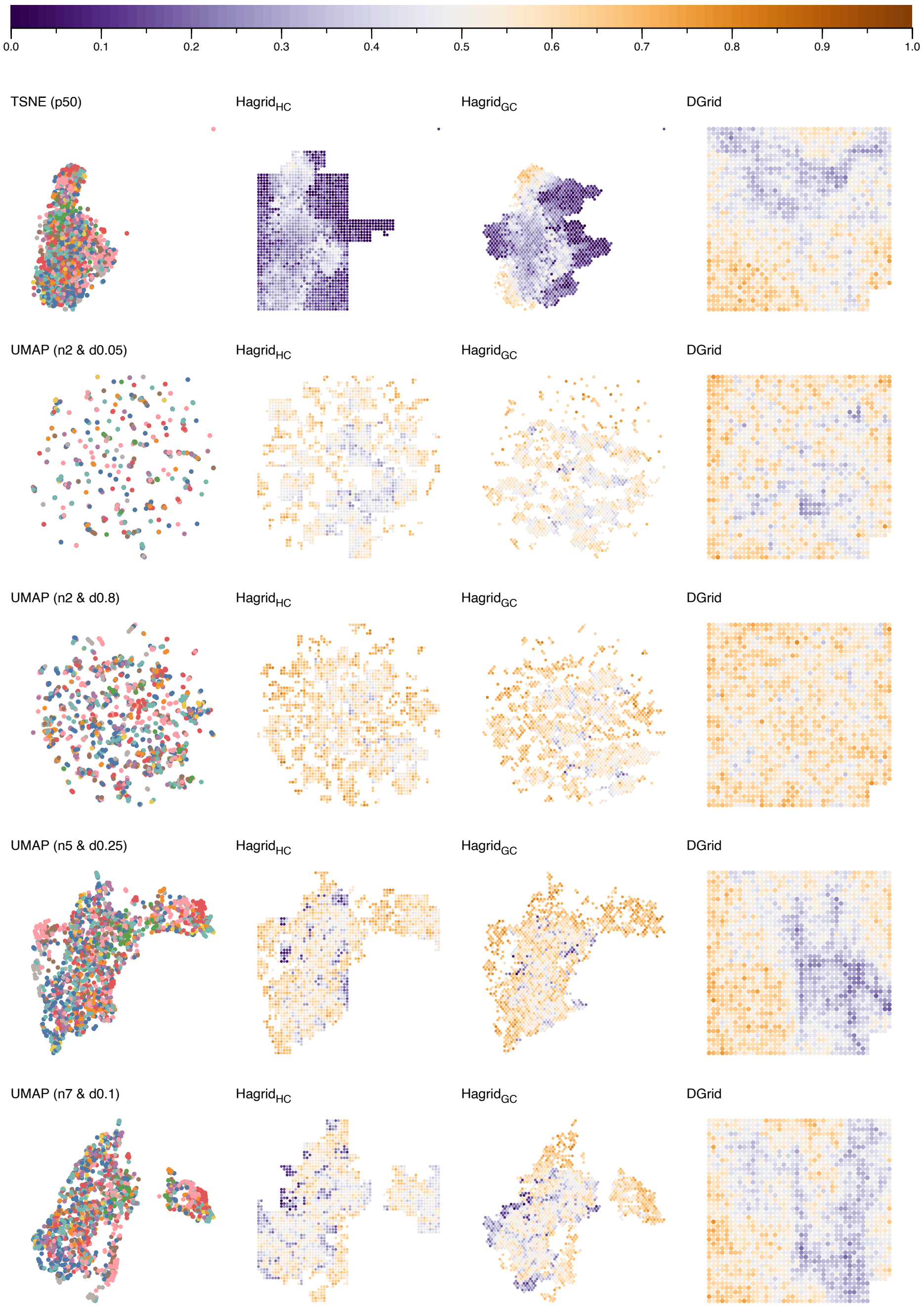


Figure 11: Scatterplots of projections of the biggest dataset, Figure 8 continued (3).