






Interactions for Seamlessly Coupled Exploration of High-Dimensional Images and Hierarchical Embeddings

Alexander Vieth¹ , Boudewijn Lelieveldt^{1,2} , Elmar Eisemann¹ , Anna Vilanova³ , Thomas Höllt¹ 

¹TU Delft, The Netherlands

²Leiden University Medical Center, The Netherlands

³TU Eindhoven, The Netherlands

Abstract

High-dimensional images (i.e., with many attributes per pixel) are commonly acquired in many domains, such as geosciences or systems biology. The spatial and attribute information of such data are typically explored separately, e.g., by using coordinated views of an image representation and a low-dimensional embedding of the high-dimensional attribute data. Facing ever growing image data sets, hierarchical dimensionality reduction techniques lend themselves to overcome scalability issues. However, current embedding methods do not provide suitable interactions to reflect image space exploration. Specifically, it is not possible to adjust the level of detail in the embedding hierarchy to reflect changing level of detail in image space stemming from navigation such as zooming and panning. In this paper, we propose such a mapping from image navigation interactions to embedding space adjustments. We show how our mapping applies the "overview first, details-on-demand" characteristic inherent to image exploration in the high-dimensional attribute space. We compare our strategy with regular hierarchical embedding technique interactions and demonstrate the advantages of linking image and embedding interactions through a representative use case.

CCS Concepts

• **Human-centered computing** → *Interaction design; Visual analytics*; • **Mathematics of computing** → *Dimensionality reduction*;

1. Introduction

High-dimensional images are a form of image data for which every pixel is associated with a high-dimensional attribute vector. Such images can be acquired by various means, e.g., hyperspectral cameras that acquire a large spectrum of the light, instead of only RGB, which produces hundreds of attributes per pixel. Another example is cyclic immunofluorescence (CyCIF), where the expression of tens of proteins in tissue is measured, each protein corresponding to one attribute. Such acquisition methods produce images with millions of pixels and up to hundreds of attributes. This abundance of data often necessitates an extensive exploration phase before specific hypothesis-driven analysis can be performed. Data exploration in these settings is concerned with two data spaces; the two-dimensional image and the high-dimensional attribute space.

A common exploration setup of high-dimensional images consists of multiple coordinated views showing an image representation and a low-dimensional embedding of the attribute data side-by-side. The image space is explored mainly by navigation, i.e., panning and zooming interactions to focus on a region of interest (ROI), since large images typically exhibit a higher resolution than a screen is able to display physically. Therefore, navigation in image space is commonly supported with image pyramids: Each attribute channel is repeatedly downsampled to yield smaller images at multiple scales of detail. Exploration starts at a lower resolution, matching the

viewport pixels closely, from where a user can then zoom into ROIs, which will automatically move down the pyramid into higher level-of-detail views.

Whereas scalar or three-dimensional data can be easily mapped to colors, high-dimensional attribute data cannot be directly shown in screen space without a mapping from the high-dimensional to a color space. This mapping is often achieved through a selection of attributes, clustering, or coloring of 2D/3D-projections of the attribute data. Typically, the attribute space exploration of high-dimensional images still cannot be performed well in image space alone but is augmented with views on the attribute data. Attribute vectors are often embedded with dimensionality reduction (DR) techniques like UMAP [MHM18] or t-SNE [van14] and subsequently explored in the resulting low-dimensional embedding spaces, e.g., in single-cell analysis with CyCIF images [KBJ*19], hyperspectral images of artworks [VSE*21], or remote sensing [HSH*20]. While image sizes in the order of a million pixels are common, data set sizes of over 100,000 points are considered very large for DR techniques like t-SNE [KB19]: the resulting embeddings usually cannot capture all desired detail and come with increased computational cost.

Hierarchical DR techniques, such as HiPP [PM08], HSNE [PHL*16] or HUMAP [MEPM21], have been developed to tackle issues that emerge from large amounts of data points.

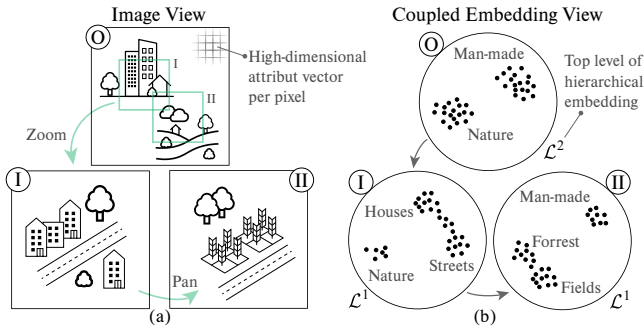


Figure 1: Coupling concept: (a) Image interactions with a high-dimensional image showing mostly man-made object (top) and nature (bottom) specifically zooming (I) and panning (II) to focus on more detailed views. (b) Coupled view of a hierarchical embedding corresponding to the overview (O), zoom (I) and pan (II) image views. Interacting with the image view triggers a corresponding change of detail level in the embedding, from hierarchy level 2 to 1.

They decrease the embedding size by using landmarks to create a hierarchical data structure, in which each level represents the original data set at a different level of abstraction. Hierarchical DR techniques follow the “overview first, zoom, filter, details-on-demand” approach [Shn96] for interactive data exploration. They start out presenting the user an overview embedding, which shows dominant data structures. From there, the user can request more refined embeddings by selecting clusters, which will show a subset of the data at a more detailed hierarchy level. This refinement interaction can be seen as analogous to zooming in image space, based on an image-pyramid, to achieve higher levels of detail.

Existing (hierarchical) DR methods largely target abstract high-dimensional data and thus lack interactions specific to exploration of high-dimensional images. Most importantly, there is no coupling between user interactions in image space and embedding space. E.g., zooming into a part of the image, i.e., requesting more detail for this part of the data, has no effect on the level-of-detail of the embedding view. This requires a set of interactions (i.e., selection and zoom) in the embedding space to achieve the desired detail. Ideally, navigation in image space comes with a desired adaptation of the view of the hierarchical embedding space. Figure 1 shows an abstract example of such coupled image and embedding views: Zooming into an image region Figure 1a triggers an update of the embedding view Figure 1b, which is set to display a higher detail embedding level, e.g., the roads that were not visible previously. A coupling between the image scale space and hierarchical embedding space interactions would thus enable a fully image-aware high-dimensional data exploration and analysis.

The main contribution of this paper is an interaction paradigm that couples interactions in image space to hierarchical-embedding actions, including

- a mapping from image navigation interactions to embedding space actions,
- an optimization strategy for level-of-detail adjustment based on ROIs in image space, and
- its implementation as an extension of HSNE, and an evaluation on a representative data set.

2. Related Work

DR techniques and image exploration images are broad topics. Here, we review previous work that is most relevant to our specific contribution in interaction paradigms for exploration of high-dimensional images and refer to more in-depth reviews regarding visual analysis and exploration of high-dimensional data [KH13, LMW*17].

2.1. High-dimensional Data Exploration

In this work, we focus on DR techniques for large data sets. Even though UMAP [MHM18] and modern implementations of t-SNE [vdRBE22] enable efficient embedding computation of data sets the size of single digit million data points, these DR techniques reach their limits when being applied to mega- or gigapixel images. The embeddings insufficiently display detailed data characteristics like intra-cluster dissimilarities, and thus small-scale structures will not be visible anymore, or larger structure is lost. Hierarchical DR techniques tackle this issue and come with a lower computational cost [NA19]. Recently, such hierarchical DR techniques like Hipp [PM08], HSNE [PHL*16], or HUMAP [MEPM21] that extend existing single-level techniques have emerged. These methods all share hierarchical aggregation to represent data points on various levels of abstraction and aim to facilitate exploration of high-dimensional data, but are based on different DR methods.

2.2. Large Image Data Exploration

Generally, visualization and interaction systems for large images work with image tiles of various resolutions taken from image pyramids [EMJ16, MGP*22, SPW20]. Image exploration might be performed solely based on zoom and pan operations in the image pyramid as presented by Jeong et al. [JST*10], or can be supplemented with additional information. Molin et al. [MBT*16], e.g., propagate low-level features to the current zoom level. These examples deal with grayscale or color images, but high-dimensional images cannot be displayed or explored following the same approaches: they do not trivially extend to more than three image channels.

2.3. High-dimensional Image Exploration and Interactions

To adequately explore high-dimensional images, both the high-dimensional attribute space and the image layout have to be taken into account. Ellsworth et al. [EHN17] discuss a holistic approach of showing multiple channels side by side using a wall of monitors. Toolboxes like PySpacell [RRT*20] provide various spatial statistics functions to analysis pre-segmented high-dimensional images. SquidPy [PSK*22] is a framework that brings together high-dimensional image viewers and image analysis tools.

State-of-the-art high-dimensional image analysis toolkits stress the importance of region-of-interest based exploration of large images. Scope2Screen [JKW*22] is a Focus+Context oriented application, which provides lens views on ROIs and lets the user define false RGB recolorings of the viewport, based on manually selected attribute channels. They mention the need for DR techniques and suitable visual representations of found features in image space in order to couple image and feature space more closely. Others, like histoCAT [SJR*17], ImaCytE [SVUK*19], or Facetto [KBJ*19] offer multiple coordinated views to analyze high-dimensional image data, including image viewers, parallel coordinate plots, and DR plots. Our interaction coupling between hierarchical embedding and high-dimensional image data allows for embedding the

entire image data and recoloring of image ROIs based on the entire high-dimensional attribute space.

Interacting is essential for the exploration of dimensionality-reduced data. There exist various classification approaches, e.g., Liu et al. [LMW*17] who divide these interactions into computation-centric, exploratory, and model manipulating. Sacha et al. [SZS*17] described common user interactions with DR methods more thoroughly. Past discussions of interactions with visualization techniques for high-dimensional data by Yang et al. [YWR03] and Sifer et al. [Sif06] focused on parallel coordinates and table-based approaches. Recently, Höllt et al. [HVP*19] proposed Focus+Context-based interaction techniques specifically for the exploration of hierarchical embeddings. Marcílio-Jr. et al. [MJEP*21] similarly specified an interaction technique for single-level embeddings. These prior works, however, focus on interactions solely with embeddings. In our work, we discuss how a hierarchical embedding should react to user interaction with an image representation of the data.

Elmqvist and Fekete [EF10] propose a generalized model for interactions with visualizations of hierarchically aggregated data. Their model assumes a single view of the data though, whereas we tackle the problem of interacting with two separate views: a spatial data layout (image view) and an embedding hierarchy (embedding view), coupled to the image view. We aim to specify suitable interactions with the image view and corresponding actions of the embedding view.

3. Tasks and Requirements

The main purpose of coupling image-space interactions to embedding-space actions is to enable a user to navigate in image space (**T1**), while simultaneously exploring the attribute space of the currently visible image region (**T2**). Two dimensional embeddings are a useful modality for exploring similarities in the attribute space. Albeit not a direct interaction with the image space, the user should still be able to coarsen and refine the level of detail in the embedding directly (**T3**), as is already possible in traditional approaches.

In summary, the user should be able to

- T1** navigate (zoom or pan) in image space,
- T2** explore the attribute space of an image ROI, and
- T3** request more or less detail for a ROI in attribute space.

A typical image exploration starts with the entire image in view (*overview*), followed by zoom and pan operations to different ROIs for *detail* inspection. The embedding space exploration should mirror this “*overview first, details-on-demand*” characteristic of the image navigation (**R1**) with the intention of providing analogous reactions in the attribute-space depiction to a single image-space interaction. This entails that, as the user focuses on a spatial region of interest, the embedding should be limited to a set of points which represents the ROI (**R2**, **R4**). This contrasts other conceivable approaches that might follow Focus+Context paradigms and would represent image areas outside of a ROI as well. Instead, **R1** ensures a maximal appropriate detail level for a given ROI and reduces computational costs by restricting both view and computation to a subset of all data point. Additionally, to minimize cognitive load on the user, the embedding should preserve coherence between updates when changing ROIs (**R3**). In order to allow for an interactive

data exploring, the embedding has to update fast, that is, any additional computational effort on top of the embedding procedure should be minimal (**R4**). Further, to enable linking of image and attribute spaces, e.g., through highlighting of representative embedding points and their represented pixels, a data mapping between arbitrary selections in either space has to be supported (**R5**).

Thus, to successfully accomplish the user tasks, the image-to-embedding coupling should:

- R1** follow the “*overview first, details-on-demand*” approach,
- R2** represent the ROI,
- R3** provide stable transitions between embeddings,
- R4** update at interactive exploration speeds, and
- R5** link selections between image and embedding space.

4. Coupling Image Navigation and Embedding Space

An intuitive way of coupling image-space navigation and high-dimensional attribute-space exploration is to create a new embedding for each new ROI in image space. However, this approach does not fulfill all our embedding-view requirements and user tasks. First and foremost, neighborhood-based DR techniques would have to recompute neighborhood graphs over and over, which severely limits interactivity, thus breaking **R4**. Gigapixel or larger images are not uncommon and already the first top-level embedding, which encompasses all data points, can be infeasible to compute in reasonable time. With such large images, even ROIs that cover only a small part of the image space can contain hundreds of thousands or even millions of data points. Further, the large number of points leads to indistinguishable similarity structures within clusters, which might obscure interesting data characteristics. Finally, this approach only enables a single level of detail per ROI since it is not possible to refine or coarsen a standard embedding, thus breaking **T3**.

Hierarchical DR techniques overcome these issues. They are better suited for embedding large data sets and do not require neighborhood re-computations for data subsets (**R4**). Their hierarchical structure also allows a user to coarsen or refine the level of detail shown for the attributes of an image ROI (**T3**). An additional benefit to exploring image patches using hierarchical embeddings is that the resulting data representations are informed by the entire data manifold and not just a subset of the data.

4.1. Interactions: Zoom and Pan, Drill-Down and Roll-Up

Large images are most commonly navigated by zooming and panning operations. These operations change the size (zooming) and position (panning) of a viewport over the image and determine the ROI shown to the user. We generally follow the mathematical notations for hierarchical embeddings and interactions with hierarchical data structures as used in the literature [HVP*19], and fully laid out in Supplement S0. This notation does not support the concept of linking two spaces, which motivates the introduction of two new symbols: We denote the set of all data points within an image viewport as \mathcal{V} and the set of all corresponding landmarks shown in the embedding view used to represent this viewport as \mathcal{E} .

Zooming and panning focus exclusively on the spatial data layout. Both operations modify \mathcal{V} by adding visible or removing invisible data points. In our coupled image and embedding setting, we want zooming and panning in the image view to update the embedding (**T1**, **T2**). Thus, we need a mapping of the image interactions to

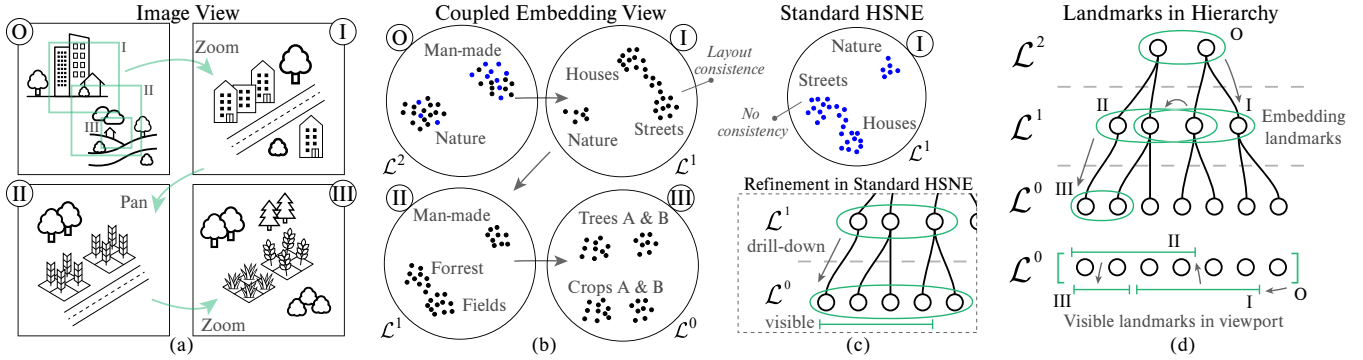


Figure 2: Interaction overview: Image interactions and embedding hierarchy reactions: (a) exemplary image space depicting a region of mostly man-made object (top) and mostly nature (bottom) with three viewports (I zoom from overview, II pan, III pan and zoom). (b) depicts the embeddings as shown after each interaction. The data points that are currently in the viewport and landmarks that are shown in the corresponding embedding views are shown in (d). In contrast to standard HSNE interactions, where a refinement of the landmarks that represent ROI I, marked with blue ●, leads to inconsistent cluster placement, our method keeps a consistent layout, (b, I) and (c, top). The scattered placement of those top-level landmarks renders manual selection practically impossible. Changing the level of detail for a given ROI (e.g., drilling down) can lead to embeddings containing invisible landmarks as seen in (c, bottom).

possible actions in the embedding. Whenever \mathcal{V} changes, we have to recompute \mathcal{E} so that every landmark $L_j^k \in \mathcal{E}$ on embedding hierarchy level k represents at least one data point $L_i^0 \in \mathcal{V}$ and all data points in \mathcal{V} are represented by a landmark in \mathcal{E} (R2). Given a single level of detail in the image viewport we define all landmarks in the embedding to be from a single hierarchy level as well. The selection of the hierarchy level will be discussed in Section 4.2.

Figure 2a showcases zoom and pan actions in an abstracted high-dimensional image and indicates which set of data landmarks \mathcal{L}^0 are currently in \mathcal{V} (Figure 2d). Each viewport change triggers an update of \mathcal{E} (Figure 2b). The updated embeddings aim to be consistent with their predecessors, i.e., clusters that represent similar data points remain in nearby embedding positions. Standard hierarchical embedding refinement do not feature such coherence (Figure 2c).

Starting an exploration, the viewport encompasses the entire image and thereby all \mathcal{L}^0 landmarks. The corresponding top-level embedding contains all top-level landmarks. Zooming-in only ever removes elements from \mathcal{V} and, in line with this, the updated embedding will either only contain a subset of the previous landmarks from the same abstraction level or landmarks from a finer abstraction level that are represented by the previous landmarks. Panning and zooming-out, however, may add previously unseen data points into the viewport and thereby might require the inclusion of previously unrepresented landmarks into the embedding. The pan, labeled II in Figure 2a, shows such an update of \mathcal{E} .

The set of data points in the viewport \mathcal{V} corresponds to a selection \mathcal{S}^0 of landmarks from the data level: $\mathcal{S}^0 = \mathcal{V} \subseteq \mathcal{L}^0$. To find a set of landmarks \mathcal{E} on a level k that represents \mathcal{S}^0 , we need a general mapping of landmark selections between levels from \mathcal{S}^k to \mathcal{S}^{k+1} .

4.2. Landmark Mapping

The importance of landmark mapping is twofold: defining which embedding landmarks $\{L_j^k\} \in \mathcal{E}$ best represent \mathcal{V} (R2), as well as linking selections between image and embedding views (R5).

Standard approaches use top-down mappings since users define selections in embedding space: starting at the top-level, refinement

actions should represent all landmarks contained in the selection; but in our image-driven scenario this yields many landmarks outside the image ROI, see Figure 2c (bottom). We use a **bottom-up mapping** approach to map \mathcal{S}^0 to $\mathcal{S}^k := \mathcal{E}$, which represent the image ROI (R2). For rolling-up one level we define a set \mathcal{S}^{k+1} that represents a set \mathcal{S}^k as the union of all parents of the landmarks in \mathcal{S}^k . To avoid traversing the hierarchy when rolling-up several levels, we can cache the representative landmark on each level for every data point when computing the embedding hierarchy in the first place. We use the same approach when drilling-down into the hierarchy, based on a given viewport \mathcal{V} . Instead of computing all children \mathcal{S}^{k-1} that are represented by \mathcal{S}^k we use the bottom-up mapping to find the minimal set $\mathcal{S}^{k-1} \subseteq \mathcal{S}^{k-1}$, that contains only the landmarks needed to represent \mathcal{V} . This means any set \mathcal{S}^k , representative for the data points \mathcal{S}^0 in the viewport, can immediately be computed as the union of the representative landmarks on level k .

For linking selections between a subset of the image viewport to the embedding we follow the same bottom-up approach, the only difference being that instead of rolling-up the entire viewport we start with the selected subset $\mathcal{S}^0 \subseteq \mathcal{V}$. Vice versa, for linking selections from the embedding to the image, we traverse the hierarchy downwards for all selected embedding landmarks $\mathcal{S}^k \subseteq \mathcal{E}$ to find the corresponding data point selection $\mathcal{S}^0 \subseteq \mathcal{L}^0$.

At this point, we need to define how to couple the zoom factor in the image, i.e., the ROI's fraction of the full image space, to the selection of the hierarchy level k in the embedding, according to the underlying goals and requirements. We define such a heuristic with the aim to keep the number of landmarks in the embedding space within a pre-defined budget, similar to the visual entity budget introduced by Elmqvist et al. [EF10]. Different from their approach, we propose to find the level with the number of landmarks closest to a target number t , instead of applying a hard maximum to accommodate hierarchies with large differences in the number of data points between neighboring levels. In our framework, the budget is user defined. According to our requirements, the budget should be chosen small enough to allow for interactive computation of the embedding (R4) but could, e.g., also be defined as to not overload

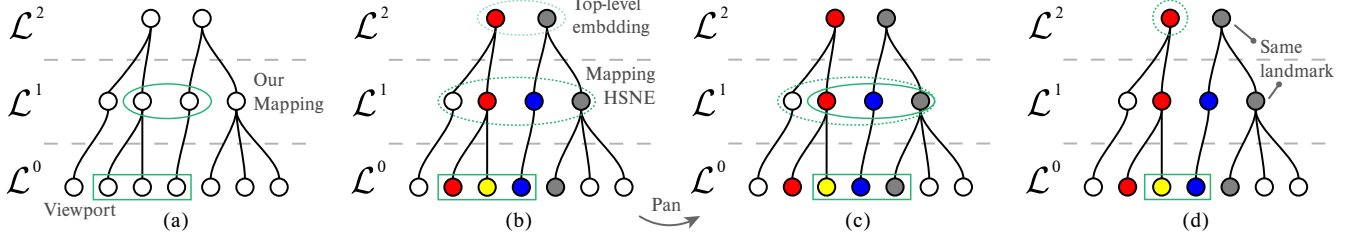


Figure 3: Mapping comparisons: (a) shows the \mathcal{E} in \circ resulting from our mapping. (b) depicts the HSNE landmarks in \circ resulting from the top-down approach. Colors indicates data level indices. (d) shows how a pan action updates landmarks in the coupled bottom-up approach whereas the top-down approach might not display any change at all and (c) depicts HSNE’s bottom-up, landmarks-in-viewport-only mapping.

the visual capacity of the linked embedding view. Given a budget t , we determine the hierarchy level k by calculating the set \mathcal{S}^k that represents \mathcal{S}^0 for all hierarchy levels. We pick the level k for which $|\mathcal{S}^k|$ is closest to t .

We also enable refining or coarsening the level of detail of the representative landmarks \mathcal{S}^k in the embedding view for a given viewport selection \mathcal{V} (T3). Knowing the current hierarchy level k and \mathcal{V} , we can immediately compute \mathcal{S}^{k-1} or \mathcal{S}^{k+1} with our bottom-up mapping. In contrast, simply rolling-up or drilling-down the current landmarks in the embedding hierarchy view without considering the viewport does not yield a satisfying result. As exemplified by Figure 2c (bottom), e.g., requesting more detail for an embedding by drilling-down all landmarks can yield landmarks outside the viewport (breaking R2)

4.3. Implementation using HSNE

Our landmark and interaction mapping is suitable for any tree-based hierarchical embedding method. As proof of concept, we implement our interactions with HSNE [PHL*16] in the visual analytics framework ManiVault [VKT*23] with code available at github.com/ManiVaultStudio/ImageEmbeddingCoupling. To do so, we had to adjust HSNE in some aspects. Most importantly, HSNE defines representation between levels probabilistically. HSNE defines an *area of influence* $I_{L_i^{k+1}}(L_i^k)$ that indicates the probability that a landmark $L_j^{k-1} \in \mathcal{L}^{k-1}$ is well represented by $L_i^k \in \mathcal{L}^k$. The resulting data structure is not a proper tree since each landmark in \mathcal{L}^k can be represented by multiple landmarks in \mathcal{L}^{k+1} . In order to convert this structure into a tree, we compute for every data point the landmark in each scale with the maximum influence exercised on the respective data point. Notably, HSNE landmarks are not aggregates but each landmark L_i^k corresponds directly to an actual data point L_j^0 .

In contrast to our method, regular HSNE employs a **top-down mapping** for selection linking. It only applies this mapping from embedding to the data level (image) and does not use a reverse mapping from image to embedding view. Particularly, when linking from a lower to a higher hierarchy level, selections are linked based on the landmarks that are contained in both levels only, but not their parents. That means, rolling-up some \mathcal{S}^0 in the image will result in $\mathcal{S}^k = \{L_i^k \mid L_i^k \in \mathcal{S}^0 \text{ and } L_i^k \in \mathcal{L}^k\}$. Figure 3 shows a simplified HSNE hierarchy with three levels, where each landmark is connected to its most representative landmark in the next abstract level. Figure 3a indicates embedding landmarks resulting from our mapping for the three data points in view. In contrast, as regular HSNE is top-down-oriented, it coarsens all top-level landmarks (or

those representing the viewport), resulting in an embedding that contains all level 1 landmarks, of which some do not represent the viewport at all (Figure 3b). Using a visual budget target of 2, HSNE would also not refine the top-level at all in this example. Selecting the data points \bullet and \bullet in the image viewport (Figure 3d), would only highlight \bullet in \mathcal{L}^1 and none in \mathcal{L}^2 , since neither point is a landmark in that level.

In the standard HSNE exploration this mapping is sensible. Typically, interactions with an HSNE embedding, like refining a selection, build on the assumption that the selection is continuous in the embedding space [HVP*19]. The likelihood that $\mathcal{S}^0 \cap \mathcal{L}^k$ contains all relevant landmarks is thus high. But when interacting with the image space, that is, selecting a spatially connected region in the image, the linked embedding landmarks usually do not correspond to similarly confined regions in the hierarchy or the embedding layout. Rather, representative landmarks will be scattered throughout the embedding, since neighboring image pixels might depict data points with vastly dissimilar attribute vectors (Figures 2b and 2c).

4.4. Initialization of Embedding Updates

When drilling-down or rolling-up in a hierarchical embedding a new embedding needs to be created. Initialization is a crucial step when calculating neighborhood-based embeddings. In the default implementations of HSNE these new embeddings are initialized randomly. In order to preserve the analysis coherence, we want to re-use landmark positions from the current embedding for the initialization of the updated embedding (R3). Similar to [HVP*19], we initialize all landmarks in \mathcal{E} that were preserved during drill-down or roll-up with their previous positions from before the interaction. Landmarks that were added during a drill-down are initialized with the position of their respective parents. When moving horizontally in the hierarchy, newly added landmarks are initialized based on their neighborhood in the hierarchy. Therefore, we query the existing neighborhood graph of the scale and interpolate the position using the three closest neighbors that were present in the corresponding embedding before the interaction. In cases, where \mathcal{E} changes strongly, it might not always be possible to find a neighbor for a given added landmark. In this case, the new landmark is initialized at a random position.

As a result, upon panning to a region that remains similar with respect to the attributes of the shown data points, the embedding is not expected to change strongly; our initialization ensures that there is consistency between embedding updates (R3). Nevertheless, when panning to a region containing points with rather different attribute vectors, there will be little overlap between the previous Current \mathcal{E} and the embedding will essentially be initialized randomly.

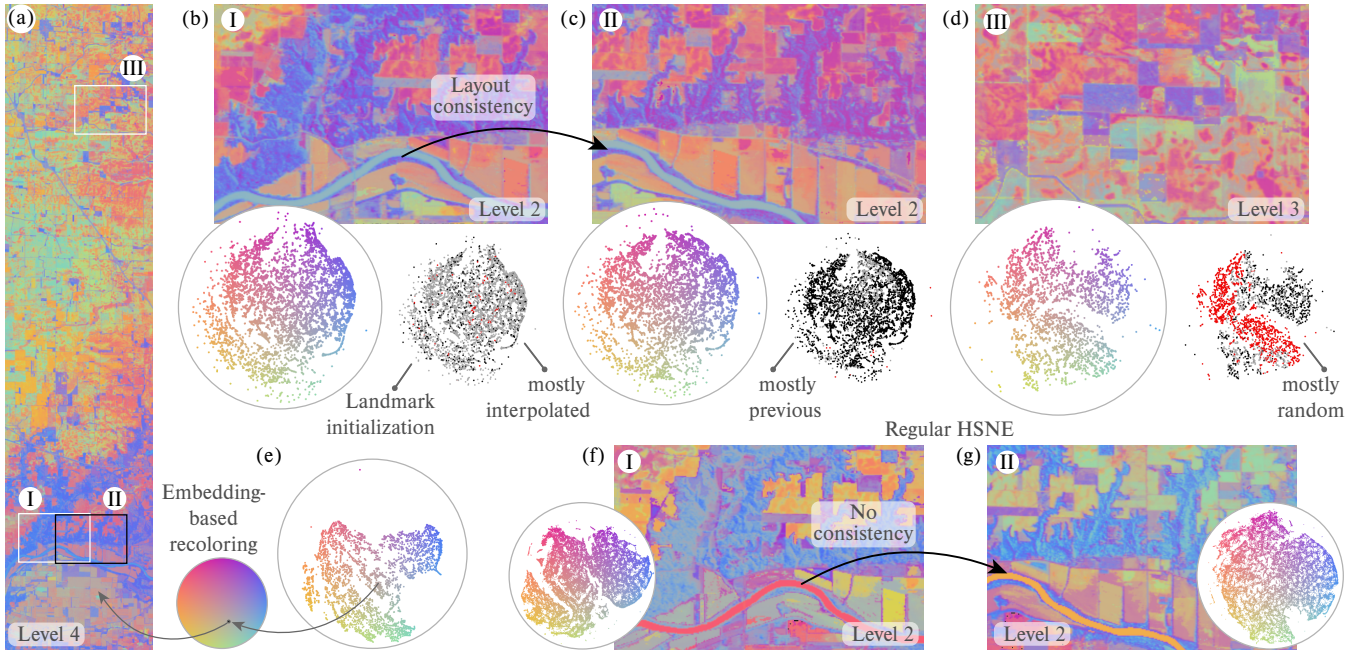


Figure 4: Indian Pines: (a) Recolored image based on top-level embedding as shown in (e), with three ROI viewports as obtained after a zoom (I) and pan (II, III). The top row (b, c, d) shows corresponding coupled embeddings as well as the current re-colored viewports. When changing the viewport to a region which is represented by a similar set of landmarks, e.g., spatially neighboring regions, to embedding layout stays consistent. The initialization mode of each landmark is indicated in a second scatterplot as based on previous ●, interpolated ●, and random ● positions. Standard HSNE refinements of landmark sets that are representative of a current viewport, in (f) and (g), do not enable consistent embedding exploration and contain more landmarks for the same level of detail.

Initializing t-SNE embeddings (or derivatives that follow the same optimization procedure) with small values is important for their convergence [KB19]. During the optimization process, the embedding’s extents grow due to the repulsing forces that are responsible for creating space between dissimilar points. To re-enable proper convergence behaviour, we utilize the embedding extents at a given moment of the first optimization and re-scale all points into this frame (shrink the embedding) during re-initialization of the embedding for each update.

5. Exemplary Use Case: Hyperspectral Image Exploration

Here, we show the application of our approach to a representative hyperspectral data set. We indicate the performance in comparison to exploration of the same data using standard HSNE and focus on embedding transitions and visual stability.

The Indian Pines Test Site 3 [BBL15] data sets contain a $614 \times 2,678 \approx 1.6\text{M}$ pixel large image with 200 dimensions attached to each pixel that depict the light spectrum reflected by the objects in view, specifically fields (e.g., corn and soy), forests, roads, rivers and houses (more information on the data set in Suppl. S2). A typical workflow during the exploration of such data using DR techniques starts with an *overview* of the image layout and attribute data, in our scenario given by the top-level HSNE embedding.

In the following, we use image re-colorings to indicate embedding structure — we map the 2D embedding positions to color using a 2D colormap by Mittelstädt et al. [SBM*14], as shown in previous work [VVL*22]. This color-coding allows to connect the embedding and image space while minimizing imposing additional structure

that comes with choosing a specific clustering algorithm. We computed a 5-level HSNE hierarchy by picking the 25% most important landmarks (with respect to their area of influence) at every level to go to the next abstraction level [HVP*19], leading to a top-level embedding with 4,205 landmarks (Figures 4a and 4e). This embedding is laid out over 2000 iterations. For better reproducibility and to improve global structure we initialize the top-level embedding with the first two PCA components of the top-level landmark attribute data [KB19]. Each following embedding is laid out over 500 iterations with otherwise default parameters as introduced by Pezzotti et al. [PHL*16]. Fewer iterations are sufficient due to the initial embedding structure given by our re-initialization scheme; additional iterations can be run on demand. The visual budget target is set to 10,000 landmarks to provide an appropriate balance between detail and performance, aiming for total update times of less than one second between changing the viewport and finished embedding. The embedding view is constantly updated during the gradient descent iterations starting about 300 ms after the user interaction, providing visual feedback and thus visual coherence. For detailed timings, we refer to Suppl. S1. The highlighted ROIs mostly depict three spectrally distinct top-level clusters (Suppl. S2 Fig. 4) with blue and violet hues corresponding to woods, and reddish and yellow-green hues corresponding to different types of fields.

Starting with the top-level embedding and image in full view, we will first focus on an area in the lower part of the image (T1) which is predominantly covered by pixels from two regions in the top-level embedding, woods and various fields (T2). We *zoom-in* to ROI (I) to focus on a subset of 57,424 pixels (Figure 4b). Driven

by the image-space zoom, we automatically drill-down the HSNE hierarchy in a single interaction step to level 2 with a representative landmark count of 9,738, closest to our visual budget target of 10,000 (R1, R2). The entire embedding update took a little under a second of which our hierarchy traversal only contributed 50 ms (R4). Compared to the overview recoloring Figure 4b, we see that the recolored patch in the new embedding shows a more detailed structure, within the forest areas and also between (sub)types of fields, due to the more detailed hierarchy level. Notably, since we initialize most of the landmarks in the updated embedding, based on the previous landmark positions (indicated in Figure 4b lower right), their general global positions and thus corresponding colors are preserved (R3). E.g., the forest region is still represented by landmarks in blue and violet hues, preserving a user’s mental frame. In contrast, following standard HSNE interactions we start with a selection of top-level landmarks that are representative for the ROI (obtained using our bottom-up mapping, since standard HSNE does not provide such correspondence) and drill-down twice to obtain the same level of detail. The resulting embedding contains 50,116 landmarks — almost as many as pixel in the viewport. This embedding represents larger image regions outside the current ROI than our embedding and fails at preserving coherent landmark positions between embedding updates. Our approach requires fewer interaction to yield a more detailed and ROI-specific embedding.

As a next exploration step, we are interested if the observed patterns can also be found in close vicinity of the current ROI. In Figure 4c we show the result of a *pan* to the partially overlapping ROI (II). Using our coupled interaction, we stay on the same abstraction level 2, now with 9,533 representative landmarks. Since the data points covered by the current ROI overlap substantially with the previous one, many representative landmarks remain the same during the embedding update. Landmark clusters that are represent the same image regions also remain in similar embedding regions, preserving coherence between the updates. In contrast, standard HSNE interactions would require drill-down recomputations, leading to inconsistencies between embedding updates (Figure 4g).

To see how the method behaves when moving to an unrelated region, we *pan* to ROI (III) (Figure 4d). The pan triggers with a roll-up in the embedding hierarchy to level 3 with 3,979 representative landmarks (level 2 had 16,404 landmarks for this viewport). Covering a very different set of data points, the new ROI is represented by many landmarks that have not been in the previous embedding, as visible by the abundance of randomly initialized landmarks. Since the number of representative landmarks in level 2 and 3 are almost equally distant to the visual target, a user might decide to request more detail by manually *drilling down* the entire ROI (T3). Figure 5 in Supplement S2 shows the result of this operation. We can observe that the refined level-3 embedding follows the layout of the level-2 embedding, indicating that the level 3 embedding was already capturing most of the variation.

6. Limitations

Our method currently restricts all embedding landmarks \mathcal{E} to be from the same hierarchy level k . For scenarios in which a ROI contains regions of varying homogeneity, an embedding view that reflects this with multiple levels of detail could be helpful. To address this, and further following the terminology of Elmqvist [EF10], our set of

image interactions might be extended with a new *local-aggregation* interaction. For this Focus+Context inspired action the user would define a focus selection $\mathcal{S}_f^0 \subset \mathcal{S}^0$, a subset of the current viewport. We would then need to find sets of focus $\mathcal{F} = \mathcal{S}_f^{k-1}$ and context landmarks $\mathcal{C} = \mathcal{S}^k$ such that the embedding $\mathcal{E} = \mathcal{C} \cup \mathcal{F}$ represents the current ROI well. However, even though Höllt et al. [HVP*19] already proposed such a Focus+Context framework for hierarchical embeddings, it is not straightforward to extend to our setting. Contiguous selections in the embedding space ensure that the represented data points on lower levels are well connected. As discussed in Section 4.3, these selections can be assumed when interacting with the embedding rather than the image. Selections of spatially connected regions in image space typically do not correspond to similarly confined regions in the embedding hierarchy since data points with vastly different neighbours might be spatially close. Rather, linked landmarks might be scattered throughout the embedding. Landmarks with few HSNE transition matrix connections experience lower attracting forces during the embedding optimization and are pushed to the periphery by dominating repulsive forces.

Further, our re-initialization scheme in Section 4.4 does not guarantee consistency between embeddings when moving back and forth between two ROI without any overlap in their corresponding \mathcal{E} . We consider approaching these limitations future work.

7. Conclusion and Future Work

We have presented an interaction strategy for the coupled exploration of high-dimensional images with hierarchical embeddings. Our method couples image navigation interactions (zooming and panning) with an embedding space that represents the attribute data. We showed how an “*overview-first, details-on-demand*” approach is well-suited for driving embedding detail based on user interactions with the image space.

Extending the embedding view from a single to multiple levels of detail by showing landmarks from several hierarchy levels, would be an interesting future work. Another possible direction would be exploring ways of providing a user with guidance on where to zoom and pan to. Finally, including information on the spatial image layout in the construction of the embedding hierarchy could be beneficial to further optimize the embedding set corresponding to the current viewport. However, already in this work, we showed the potential benefits of coupling image navigation with simultaneously updating embeddings for exploration analysis workflows.

Acknowledgements This work received funding from the NWO TTW project 3DOMICS (NWO: 17126). We thank Thomas Kroes, Julian Thijssen, Baldur van Lew and all authors of the ManiVault analytics framework.

References

- [BBL15] BAUMGARDNER M. F., BIEHL L. L., LANDGREBE D. A.: 220 Band AVIRIS Hyperspectral Image Data Set: June 12, 1992 Indian Pine Test Site 3, 2015. doi:10.4231/R7RX991C. 6
- [EF10] ELMQVIST N. FEKETE J.-D.: Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines. *IEEE Transactions on Visualization and Computer Graphics* 16, 3 (2010), 439–454. doi:10.1109/TVCG.2009.84. 3, 4, 7
- [EHN17] ELLSWORTH D. A., HENZE C. E., NELSON B. C.: Interactive Visualization of High-Dimensional Petascale Ocean Data. In *Proc. LDAV* (2017), pp. 36–44. doi:10.1109/LDAV.2017.8231849. 2

- [EMJ16] ELDAWY A., MOKBEL M. F., JONATHAN C.: HadoopViz: A MapReduce framework for extensible visualization of big spatial data. In *Proc. ICDE* (2016), pp. 601–612. doi:10.1109/ICDE.2016.7498274. 2
- [HSH*20] HUANG H., SHI G., HE H., DUAN Y., LUO F.: Dimensionality Reduction of Hyperspectral Imagery Based on Spatial-Spectral Manifold Learning. *IEEE Transactions on Cybernetics* 50, 6 (2020), 2604–2616. doi:10.1109/TCYB.2019.2905793. 1
- [HVP*19] HÖLLT T., VILANOVA A., PEZZOTTI N., LELIEVELDT B., HAUSER H.: Focus+Context Exploration of Hierarchical Embeddings. *Computer Graphics Forum* 38, 3 (2019), 569–579. doi:10.1111/cgf.13711. 3, 5, 6, 7
- [JKW*22] JESSUP J., KRUEGER R., WARCHOL S., ET AL.: Scope2Screen: Focus+Context Techniques for Pathology Tumor Assessment in Multivariate Image Data. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 259–269. doi:10.1109/TVCG.2021.3114786. 2
- [JST*10] JEONG W.-K., SCHNEIDER J., TURNEY S., ET AL.: Interactive Histology of Large-Scale Biomedical Image Stacks. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1386–1395. doi:10.1109/TVCG.2010.168. 2
- [KB19] KOBAK D., BERENS P.: The Art of Using t-SNE for Single-Cell Transcriptomics. *Nature Communications* 10, 1 (2019), 5416. doi:10.1038/s41467-019-13056-x. 1, 6
- [KBJ*19] KRUEGER R., BEYER J., JANG W.-D., ET AL.: Facetto: Combining Unsupervised and Supervised Learning for Hierarchical Phenotype Analysis in Multi-Channel Image Data. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2019), 227–237. doi:10.1109/tvcg.2019.2934547. 1, 2
- [KH13] KEHRER J., HAUSER H.: Visualization and Visual Analysis of Multifaceted Scientific Data: A Survey. *IEEE Transactions on Visualization and Computer Graphics* 19, 3 (2013), 495–513. doi:10.1109/TVCG.2012.110. 2
- [LMW*17] LIU S., MALJOVEC D., WANG B., BREMER P. T., PASCUCCI V.: Visualizing High-Dimensional Data: Advances in the Past Decade. *IEEE Transactions on Visualization and Computer Graphics* 23, 3 (2017), 1249–1268. doi:10.1109/TVCG.2016.2640960. 2, 3
- [MBT*16] MOLIN J., BODÉN A., TREANOR D., FJELD M., LUNDSTRÖM C.: Scale Stain: Multi-Resolution Feature Enhancement in Pathology Visualization, 2016. arXiv preprint. doi:10.48550/arXiv.1610.04141. 2
- [MEPM21] MARCÍLIO-JR W. E., ELER D. M., PAULOVICH F. V., MARTINS R. M.: HUMAP: Hierarchical Uniform Manifold Approximation and Projection, 2021. arXiv preprint. doi:10.48550/arXiv.2106.07718. 1, 2
- [MGP*22] MANZ T., GOLD I., PATTERSON N. H., ET AL.: Viv: Multiscale Visualization of High-Resolution Multiplexed Bioimaging Data on the Web. *Nature Methods* 19, 5 (2022), 515–516. doi:10.1038/s41592-022-01482-7. 2
- [MHM18] MCINNES L., HEALY J., MELVILLE J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, 2018. arXiv preprint. doi:10.48550/arXiv.1802.03426. 1, 2
- [MJEP*21] MARCÍLIO-JR W. E., ELER D. M., PAULOVICH F. V., RODRIGUES-JR J. F., ARTERO A. O.: ExplorerTree: A Focus+Context Exploration Approach for 2D Embeddings. *Big Data Research* 25 (2021). doi:10.1016/j.bdr.2021.100239. 3
- [NA19] NONATO L. G., AUPETIT M.: Multidimensional Projection for Visual Analytics: Linking Techniques with Distortions, Tasks, and Layout Enrichment. *IEEE Transactions on Visualization and Computer Graphics* 25, 8 (2019), 2650–2673. doi:10.1109/TVCG.2018.2846735. 2
- [PHL*16] PEZZOTTI N., HÖLLT T., LELIEVELDT B., EISEMANN E., VILANOVA A.: Hierarchical Stochastic Neighbor Embedding. *Computer Graphics Forum* 35, 3 (2016), 21–30. doi:10.1111/cgf.12878. 1, 2, 5, 6
- [PM08] PAULOVICH F., MINGHIM R.: HiPP: A Novel Hierarchical Point Placement Strategy and its Application to the Exploration of Document Collections. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1229–1236. doi:10.1109/tvcg.2008.138. 1, 2
- [PSK*22] PALLA G., SPITZER H., KLEIN M., ET AL.: Squidpy: A Scalable Framework for Spatial Omics Analysis. *Nature Methods* 2022 19:2 19, 2 (2022), 171–178. doi:10.1038/s41592-021-01358-2. 2
- [RRT*20] ROSE F., RAPPEZ L., TRIANA S. H., ALEXANDROV T., GENOVESIO A.: PySpacell: A Python Package for Spatial Analysis of Cell Images. *Cytometry Part A* 97, 3 (2020), 288–295. doi:10.1002/cyto.a.23955. 2
- [SBM*14] STEIGER M., BERNARD J., MITTELSTÄDT S., ET AL.: Visual Analysis of Time-Series Similarities for Anomaly Detection in Sensor Networks. *Computer Graphics Forum* 33, 3 (2014), 401–410. doi:10.1111/cgf.12396. 6
- [Shn96] SHNEIDERMAN B.: The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proc. Visual Languages* (1996), pp. 336–343. doi:10.1109/VL.1996.545307. 2
- [Sif06] SIFER M.: User Interfaces for the Exploration of Hierarchical Multi-dimensional Data. In *Proc. VAST* (2006), pp. 175–182. doi:10.1109/VAST.2006.261422. 3
- [SJR*17] SCHAPIRO D., JACKSON H. W., RAGHURAMAN S., ET AL.: histoCAT: analysis of cell phenotypes and interactions in multiplex image cytometry data. *Nature Methods* 14, 9 (2017), 873–876. doi:10.1038/nmeth.4391. 2
- [SPW20] SOLORZANO L., PARTEL G., WÄHLBY C.: TissUmaps: Interactive visualization of large-scale spatial gene expression and tissue morphology data. *Bioinformatics* 36, 15 (2020), 4363–4365. doi:10.1093/bioinformatics/btaa541. 2
- [SVUK*19] SOMARAKIS A., VAN UNEN V., KONING F., LELIEVELDT B. P., HOLLT T.: ImaCytE: Visual Exploration of Cellular Microenvironments for Imaging Mass Cytometry Data. *IEEE Transactions on Visualization and Computer Graphics* (2019), 1–1. doi:10.1109/TVCG.2019.2931299. 2
- [SZS*17] SACHA D., ZHANG L., SEDLMAIR M., ET AL.: Visual Interaction with Dimensionality Reduction: A Structured Literature Analysis. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 241–250. doi:10.1109/TVCG.2016.2598495. 3
- [van14] VAN DER MAATEN L.: Accelerating t-SNE using Tree-Based Algorithms. *Journal of Machine Learning Research* 15, 93 (2014), 3221–3245. URL: <http://jmlr.org/papers/v15/vandermaaten14a.html>. 1
- [vdRBE22] VAN DE RUIT M., BILLETER M., EISEMANN E.: An Efficient Dual-Hierarchy t-SNE Minimization. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 614–622. doi:10.1109/TVCG.2021.3114817. 2
- [VKT*23] VIETH A., KROES T., THIJSSSEN J., ET AL.: Manivault: A flexible and extensible visual analytics framework for high-dimensional data, 2023. arXiv preprint. doi:10.48550/arXiv.2308.01751. 5
- [VSE*21] VERMEULEN M., SMITH K., EREMIN K., RAYNER G., WALTON M.: Application of Uniform Manifold Approximation and Projection (UMAP) in spectral imaging of artworks. *Spectrochimica Acta - Part A: Molecular and Biomolecular Spectroscopy* 252 (2021), 119547. doi:10.1016/j.saa.2021.119547. 1
- [VVL*22] VIETH A., VILANOVA A., LELIEVELDT B., EISEMANN E., HÖLLT T.: Incorporating Texture Information into Dimensionality Reduction for High-Dimensional Images. In *Proc. PacificVis* (2022). doi:10.1109/pacificvis53943.2022.00010. 6
- [YWR03] YANG J., WARD M. O., RUNDENSTEINER E. A.: Interactive Hierarchical Displays: A General Framework for Visualization and Exploration of Large Multivariate Data Sets. *Computers & Graphics* 27, 2 (2003), 265–283. doi:10.1016/S0097-8493(02)00283-2. 3