Focus+context exploration of hierarchical embeddings

Citation for published version (APA):

DOI:
10.1111/cgf.13711

Document status and date:
Published: 01/06/2019

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Focus+Context Exploration of Hierarchical Embeddings

T. Höllt\textsuperscript{1,2}, A. Vilanova\textsuperscript{2}, N. Pezzotti\textsuperscript{2}, B.P.F. Lelieveldt\textsuperscript{3}, and H. Hauser\textsuperscript{2,4,5},
\textsuperscript{1}Leiden Computational Biology Center, Leiden University Medical Center, The Netherlands
\textsuperscript{2}Computer Graphics and Visualization Department, TU Delft, The Netherlands
\textsuperscript{3}Leiden University Medical Center, LKEB, Department of Radiology, Leiden, The Netherlands
\textsuperscript{4}Department of Informatics University of Bergen, Norway
\textsuperscript{5}Mohn Medical Imaging and Visualization Center, Bergen, Norway

Abstract
Hierarchical embeddings, such as HSNE, address critical visual and computational scalability issues of traditional techniques for dimensionality reduction. The improved scalability comes at the cost of the need for increased user interaction for exploration. In this paper, we provide a solution for the interactive visual Focus+Context exploration of such embeddings. We explain how to integrate embedding parts from different levels of detail, corresponding to focus and context groups, in a joint visualization. We devise an according interaction model that relates typical semantic operations on a Focus+Context visualization with the according changes in the level-of-detail-hierarchy of the embedding, including also a mode for comparative Focus+Context exploration and extend HSNE to incorporate the presented interaction model. In order to demonstrate the effectiveness of our approach, we present a use case based on the visual exploration of multi-dimensional images.

CCS Concepts

\begin{itemize}
\item Human-centered computing → Information visualization; Visualization theory, concepts and paradigms;
\end{itemize}

1. Introduction

In order to successfully benefit from the wealth of information in large and complex datasets, interactive visual data exploration and analysis is used in a variety of application areas such as text analysis [MCCD13], fraud detection [LGM\textsuperscript{18}], machine learning [PHG\textsuperscript{18}], and life sciences [OKB\textsuperscript{08}, LvUH\textsuperscript{18}]. Multi-dimensional data is often a core challenge in these processes and dimensionality reduction is regularly an essential part of the approach. Fortunately, a plethora of according techniques is available [BG06, vdMH08, EHH12]. However, with ever increasing data-sizes, visualizing a complete dataset in a single plot is often impossible or leads to a lack of detail or overview. Hierarchical techniques can mitigate those problems through an overview first, detail on demand approach and will likely become essential for visual analysis of large high-dimensional data.

A concept called Focus+Context [Mun14, Chapter 14] has been proven effective for multiple level-of-detail visualization in a single plot and has been used for many types of visualization [CKB09]. In essence, different visual encodings are used to separate semantic groups corresponding to an area of interest (Focus) and areas that provide Context. The original idea focused on transforming the visual space, such as for lens views [Fur99] or by so-called rubber-sheet warping [SSTR93]. Later on, the concept was generalized to use different visual channels besides space, such as opacity or frequency [Haut06] for separating focus and context. Even with extensive work in recent years, to the best of our knowledge, no Focus+Context concept for embeddings has been proposed yet.

In this paper, we now introduce and specify the concept of Focus+Context for the exploration of embeddings with multiple levels of detail (hierarchical embeddings) such as Hierarchical Stochastic Neighbor Embedding (HSNE) [PHL\textsuperscript{16}] or Hierarchical Point Placement (HiPP) [PM08]. We implement the proposed concept by extending HSNE and show its viability in a use case, showing the interactive exploration of multi-dimensional imaging data. The main contributions of this paper are twofold:

1. We specify the concept of Focus+Context for embeddings with multiple levels of detail, including the design of
   \begin{itemize}
   \item a set of interactions supporting the exploration, and
   \item a visual representation supporting the distinction of focus and context groups in the embedding.
   \end{itemize}
2. We extend HSNE to support Focus+Context exploration by
   \begin{itemize}
   \item adapting the creation of the HSNE hierarchy to fit a more fine grained exploration and by
   \item specifying multiple modes to define the similarity of points originating from different levels of the hierarchy.
   \end{itemize}

In the following, we first present a requirement analysis for Focus+Context for embeddings (Section 2) and give an overview of the related work (Section 3). In Section 4 we describe our interaction and visualization design, followed by the according extensions to HSNE (Section 5). Then, we present a use case in Section 6 and conclude in Section 7.
2. Problem Description

First, let us briefly define hierarchical embeddings. A hierarchical embedding is a special type of embedding of high-dimensional data in a low-dimensional (e.g., two-dimensional) space. Instead of limiting the embedding to a single mapping, a hierarchy, consisting of \( n \) levels \( \mathcal{L}^0, \ldots, \mathcal{L}^{n-1} \), is defined on the input data. Here, \( \mathcal{L}^0 \) contains the complete dataset, while every level \( \mathcal{L}^{k+1} \) is less detailed than the previous level \( \mathcal{L}^k \). An element \( L^i \in \mathcal{L}^{k+1} \), called a landmark, represents (→) a set of elements \( \{L^i \mid L^i \rightarrow L^{k+1}\} \subset \mathcal{L}^k \).

Similarly, we define \(-\rightarrow\) on sets, i.e., a set \( S^{k+1} \) represents a set on the more detailed level \( S^k \) by the union of the sets represented by its elements
\[
S^k = \bigcup_{S^{k+1}} \{L^i \mid L^i \rightarrow L^{k+1} \text{ and } L^{k+1} \in S^{k+1}\}.
\]

Typically, representation is achieved by aggregation or selection. With the hierarchy defined, the embedding is then defined as a set of mappings, one for each level of the hierarchy. Existing examples of hierarchical embeddings include HSNE [PHL*16] and HiPP [PNML08]. Other forms of hierarchical data representation (e.g., through hierarchical clustering) with mappings in levels (e.g., t-SNE per level) are certainly also viable.

Methods like HSNE [PHL*16] enable an interactive exploration of the hierarchical representation. Usually, the exploration starts on the highest level \( n - 1 \) of the hierarchy. The analyst can then select any subset \( S^{n-1} \subset \mathcal{L}^{n-1} \) and request a new plot that will contain all elements \( S^{n-2} \subset \mathcal{L}^{n-2} \), represented by the elements in \( S^{n-1} \). Typical exploration paths can roughly be divided into three groups, illustrated in Figure 1. In the most simple case, the analyst is interested in a specific, large part of the data, separated on the highest level of the hierarchy as \( S^{n-1} \). Once the analyst has identified this group they typically zoom into this group several times, until a desired level of detail \( S^{n-m} \) with \( 0 < m \leq n \) is reached (Figure 1a). If the part of the data that is of interest is small and not directly identifiable or separable at the top level of the hierarchy, the analyst zooms into a superset \( S^{n-1} \) of this group \( G^{n-1} \subset S^{n-1} \) and then recursively zooms into smaller and smaller subsets (Figure 1b) until \( G^{n-m} \) can be separated. In case the analyst wants to compare groups, they can use any combination of the aforementioned strategies to zoom into two groups, \( G^1 \) and \( G^2 \), and compare their structure on the same level \( k \) (Figure 1c).

Every such selection and zoom operation leads to a new plot, each limited to a single level of detail (LoD). As a result, multiple disconnected plots separating the data of interest from the context are created during a typical exploration session (compare Figure 2).

Figure 2: Illustration of an Exploration of a hierarchical embedding through a set of disconnected plots/views.

Figure 1: Illustration of Typical Exploration Paths for hierarchical embeddings. Paths indicated by blue nodes in the hierarchy. Repeated zooming into one subset, a), recursive subset selection and zooming, b), and zooming into two groups for comparison, c).

are created during a typical exploration session (compare Figure 2).

Method 1 request less detail for all data,
Method 2 request more detail for all data,
Method 3 request less detail for all data, and
Method 4 return to the initial state.

For zooming into areas of interest, the analyst needs to be able to

- define an area of interest (focus),
- change the focus to a subset of the current focus,
- change the focus to a different set of points,
- request more detail for the focus, and
- request less detail for the focus.

To support a comparative analysis, all of the above need to be implemented with the addition of the possibility to

- create a second focus for comparison and to
- resolve the second focus.

Fulfilling these requirements, we can support all of the above illustrated exploration paths. By adding requirements I2, I3, and I7,
enabling the reversal of other interactions, we provide a fluid traversal of the hierarchy in both directions and consequently multiple successive exploration paths. To support the analyst in the exploration, we consider the following characteristics as relevant for the visual design:

- **V1** the focus must use extended space in the visualization,
- **V2** the focus must be separated from the context,
- **V3** connections between focus and context should be maintained,
- **V4** different hierarchy levels must be identifiable.

Requirements V2 and V3 are competing. Sometimes the analyst might desire a clear separation between focus and context (V2), for example when focussing on an already separated region in the embedding (Figure 2, left). Providing a clear separation between focus and context will then further improve the separation of the cluster. In other cases, however, it might be desired that connections between data points of different levels of detail are maintained, at least to a certain degree (V3). For example, when there is no strong separation between data points that are to be assigned to separate groups (Figure 2, middle). In such a case a mapping that respects the connections between data points, while providing some separation, is desired.

3. Related Work

Munzner [Mun14, Chapter 14], as well as Cockburn et al. [CKB09] give an overview of Focus+Context visualization for data exploration. Focus+Context is an established concept to improve exploration of large and complex data. To the best of our knowledge this is the first approach that applies Focus+Context to the exploration of embeddings of high dimensional data. Sedlmayr et al. [SMT13] derive guidelines on the visualization of dimensionality reduced data from an empirical user study. Brehmer et al. [BSIM14] present a task analysis for dimensionality reduction based on interviews with analysts. Finally, Sacha et al. [SZS+17] provide an overview of typical interaction patterns for dimensionality reduction visualizations through an in-depth analysis of 58 papers on dimensional- ity reduction in typical visual workflows.

While single-level-of-detail dimensionality reduction techniques are ubiquitous, the number of hierarchical techniques is limited. HSNE [PHL+16] builds a hierarchy, by selecting representative data-points at different levels of detail, and represents the similarity at each level according to the underlying data. In its original conception, the hierarchy is then explored top to bottom, starting with a complete embedding of the lowest detail level. More and more detailed embeddings are then computed, based on user-selected subsets of the data, and visualized in disconnected views. Previously, we added a hierarchy view [HPvU+18] to the concept that collects all plots in a single visualization. HiPP [PMO8] uses Least-Square Projection [PNML08] to map the data to a low dimensional space. The mapped data is then hierarchically clustered and can be visualized at different levels of detail. While this method also allows for in-place expansion of selected groups it does not provide an importance driven assignment of the visual space. Instead the layout is largely identical for different levels of detail, driven by the point placement on the most detailed level. Sparse Multi-Dimensional Scaling (MDS) [SBT04] and MDSteer [WM04] are extension of MDS with a hierarchical data backng. While these techniques mostly aim at increasing computational performance of the traditional MDS, they could also be adapted to the Focus+Context technique presented in this paper. Approaches for hierarchical PCA exist [WK98, JPLL01, AEGLM07], however, they are not a good fit for our proposed technique and the introduced distortion, due to the linear nature of PCA. We have chosen HSNE to implement the Focus+Context concept presented here, as its non-linear nature and flexible and fast hierarchy computation make it a good fit.

4. Focus+Context for Hierarchical Embeddings

As outlined in Section 2, we aim to improve the interpretability of hierarchical embeddings and make their exploration easier through the use of Focus+Context concepts. Figure 3 illustrates the idea of a Focus+Context exploration of a hierarchical embedding. To realize our solution, we make use of two key concepts.

First, let’s assume that an embedding is a mapping from a high-dimensional space to a low-dimensional space (here 2D for visualization). Often, this mapping is non-linear. For example, similarity-based embeddings, such as t-SNE [vDM08], aim to preserve local neighborhoods rather than distances. The main optimization goal is that points that are neighbors in data space should be neighbors in the visualization. While relative distances can provide information about local structure they have little meaning for global structure. Applying Focus+Context techniques to such a plot is then quite natural. We can distort the space between groups without the risk to compromise its interpretation too much, for example, by assigning more visual space to an area of interest (focus). In fact, if we incorporate the notion of Focus+Context regions directly into the mapping, we do not even need to transform the resulting space—instead, the mapping will adapt automatically. As a result, relative local distances will be preserved within a level of detail (LoD) while the analyst only needs to be careful to not compare distances over multiple LoDs. To avoid this pitfall we provide a clear visual separation between different LoDs (requirement V4).

Second, while this idea can already improve traditional embeddings with a single LoD, we can mix multiple levels of detail in the same plot by adding a hierarchical representation of the data and the corresponding mappings. This allows a more fine-grained separation between focus and context. We can show the context with little detail while providing more detail for the focus.

Figure 3 illustrates both concepts through three stages of the same exploration as in Figure 2. Note that here the three panels correspond to different stages of the same view, while in Figure 2 they correspond to separate views. The exploration starts with a single LoD on level 2 of a given hierarchy (left). The analyst selects a part of the embedding (left, dashed blue line) as the focus. In the

![Figure 3: Illustration of a Focus+Context Exploration of a hierarchical embedding in a single view.](image-url)
next step (middle), the focus region has been expanded, more points have been added from level 1 of the hierarchy. While the number and hierarchy level of points in the context, $C^2$, did not change, the mapping was adjusted to move them slightly to the side and make the representation a bit more compact. The focus, $F^k$, takes more visual space. In the third step, a subset (middle, dashed blue line) of the previous focus was selected as the new focus $F^0$ and expanded with data from level 0 (right panel). We now have two context regions, $C^2$ and $C^1$, at different LoDs and a detailed focus, $F^0$.

4.1. Interaction Design
To enable the different exploration strategies presented in Section 2, we define a set of interactions fulfilling requirements 11–19. In general, these interactions work on and produce sets of representations of data points. We use the following sets: focus $F^k$ (comparative focus $F^k_{c}$), as defined on level $k$ of the hierarchy; $C^k$ stands for the context set on level $k$; $\lambda^k$ corresponds to a selection on level $k$; and $D$ denotes the union of the focus and context sets, active at the current state of the exploration. The new focus and context after the interaction are annotated as $F^{k'}$ and $C^{k'}$.

Figure 4 provides an overview of the interactions. Some of the requirements map directly to interactions. As requirements 11–13 refer to all data, $D$, no additional information is required. We define three interacting actions as (11) refine all data, $f(D)$, (12) simplify all data, $s(D)$, and (13) reset all data, $r(D)$.

Requirements 16 and 17 are similar in that they operate on a fixed data subset—here, the focus $F^k$ instead of all data $D$. We define two interactions, differentiate focus more, $d_{s}(F^k)$, and differentiate focus less, $d_{l}(F^k)$, to fulfill these requirements. As indicated by the term differentiate, these interactions are more general: the goal is to increase or decrease the difference between focus and context. By default, when executing these interactions, we simply request more or less detail for the focus, as required by 16 and 17. However, in some cases, for example when the focus is at the most detailed level, differentiate focus more decreases the LoD of the context.

To define or change the focus (requirements 14, 15a, and 15b), the part of the data $\lambda^k$ that shall become the focus needs to be specified. In practice, the analyst selects $\lambda^k$, for example by brushing. Based on contextual information, we can fulfill requirements 14, 15a, and 15b with a single interaction create focus & differentiate, $d_{s}(\lambda^k)$, that updates the focus and context sets on their corresponding levels in the hierarchy. In any case, we first update the context and then set the new focus to the selection.

If no focus is defined (14), for instance at the beginning of the exploration or after $r(D)$, we can simply define the context as

$$C^k := D \setminus \lambda^k$$

according to the hierarchy level $k$ on which the user interacted. To fulfill requirements 15a and 15b, we need to evaluate the selection with respect to the Focus+Context subdivision on which the user interacted and adjust the contexts, accordingly. While selections over multiple LoDs can be implemented by resolving and/or merging the involved sets, we propose to limit selections to a single level of detail for clarity. As a result, we have to consider two potential cases. If the new selection does not overlap with the old focus, $\lambda^k \cap F^l = \emptyset$ (15b), it must be part of a less detailed hierarchy level, $l < k$. In this case, the old focus $F^l$ needs to be reduced in detail to level $k$, and the resulting $F^k$ added to the old context without the selection $\lambda^k$ to create the new context:

$$C^k := (C^k \setminus \lambda^k) \cup F^k$$

If $\lambda^k \subset F^l$ (15a), it follows that $k = l$ and we can simply add a new context on level $k$:

$$C^k := F^k \setminus \lambda^k$$

Note that several context sets can exist on different hierarchy levels. After the context sets are updated, we update the focus in two steps: First, we set the selection as the new focus, $F^k := \lambda^k$. Then, we differentiate the focus by adding the represented data points from the next level, $k-1$, of the hierarchy:

$$F^{k-1} := \{l^{k-1} \mid l^{k-1} \rightarrow F^k\}$$

To enable the comparison of two groups (requirements 18 & 19), we define two interactions create comparative group, $c_{s}(\lambda^k)$, and resolve comparative group, $c_{l}(F^k)$. In principal, $c_{s}(\lambda^k)$ works very similar to $d_{s}(\lambda^k)$ with the distinction that the comparative focus $F_{c}$ must be disjoint from the existing primary focus $F$. The same strategy for creating/changing the focus as described above can then also be applied when the user executes $c_{s}(\lambda^k)$. To allow a proper structural comparison, we expand $F_{c}$ immediately to the same hierarchy level as $F$ instead of differentiating the new $F_{c}$ just once. Resolve comparative group maps directly to requirement 19. Executing it simply dissolves the comparative focus and merges it into the context group in the hierarchy level it was derived from.

4.1.1. Tree-based Interaction Data Structure
To implement the presented interactions, we propose a tree-based data structure, to track the complete exploration process. A node in the tree represents a semantic group in a given hierarchy level. All leaf nodes combined correspond to the complete data, $D$, shown in the Focus+Context plot. Edges in the tree have different interpretations, depending on the number of outgoing edges of a node. If a node has only one outgoing edge, this edge represents an increase in LoD. If a node has more than one outgoing edge the child nodes represent disjoint subgroups of the data points in the parent node (at the same LoD). With this structure, we can implement all (non-comparative) interactions described in Section 4.1.

Reset, $r(D)$, simply cuts all children from the root, returning to the initial state (Figure 5a). Operations that do not change the focus, i.e. $f(D)$, $s(D)$, $d_{s}(F^k)$, and $d_{l}(F^k)$, simply append a more detailed node to the impacted leaves or remove such leaves.
Setting a new focus appends a minimal tree such as the one shown in Figure 5b. As described in Equations 1 to 3, depending on which points are selected, \( d_r(X^k) \) behaves differently. These differences can be translated to slightly different strategies for appending the minimal tree as shown in Figures 5c and 5d. When the selection is a subset of a context group, we replace the parent node of this context with the minimal tree and define the nodes of the template according to Equations 2 and 4 (Figure 5c). When the selection is a subset of the root or current focus, we replace this node with the template and define the nodes of the template according to Equations 2 and 4 (Figure 5d). These template replacements work on any subgraph of arbitrarily complex state graphs of the exploration.

In the case of a comparative Focus+Context exploration, we need to make sure that the primary and the comparative foci are at the same level of detail. Accordingly, the difference between comparative and non-comparative interactions is mostly semantic in the proposed data structure. A comparative focus is created by the same operation on the tree as for creating the primary focus, but followed by as many refine operations as needed to move the comparative focus to the same level of detail as the primary focus. Resolving the secondary focus cuts off the tree at the original branching point.

4.2. Visual Design
As indicated in Section 4, requirements V1–V3 can be naturally met by directly adjusting the mapping of the embedding, instead of transforming the visual space of the embedding plot. Therefore, their execution depends on the choice of the embedding technique. In Section 5, we present how we extended HSNE to support Focus+Context exploration, as well as our extensions to support requirements V1–V3.

Here, we focus on requirement V4 (different hierarchy levels must be identifiable), which can be met universally by augmenting the embedding plot, independent of the type of the embedding. Typically, embeddings are visualized as a scatterplot. In principal, the hierarchy level is a property of each point. However, the visual channels available in a scatterplot, per point (position, color, size, shape) are usually used to show properties of the data itself. For example, in a typical use case of HSNE \([vUHP17]\), position is used to indicate the similarity, size to indicate the number of represented data points in the lower hierarchy levels, color to show metadata or the values of one of the original dimensions, and a halo is used to indicate selection status. Since the hierarchy level is the same for all data points within a semantic group and groups shall not intermix (compare requirement V2), we can assume that we can partition the visual space into connected areas of equal LoD. Thus, instead of indicating the LoD per point, we instead use the corresponding regions to indicate the hierarchy level. Particularly, by partitioning the complete visual space, we receive a discrete topographical map where height values correspond to hierarchy levels. We can then use standard methods for visualizing topographical information, such as iso-contours or color-coding to represent the LoD. As illustrated in Figure 3, we use increasingly lighter gray values for the background to indicate increasing LoD. We compute the background in real-time by rendering all points with their respective level mapped to the gray value, followed by an iterative region growing.

5. Focus+Context HSNE

We implemented a prototype of the proposed concept based on Hierarchical Stochastic Neighbor Embedding \([PHL+16]\). In the following, we give a brief introduction to HSNE (Section 5.1) and present the methodological extensions to the original HSNE in order to support Focus+Context exploration (Section 5.2).

5.1. Background - HSNE

HSNE is a hierarchical dimensionality reduction technique, based on the popular embedding technique t-SNE \([vdMH08]\). It constructs a hierarchy of so-called landmarks \(L_n^k\), essentially data-points that represent a local neighborhood in a level \(L^k\) of the hierarchy. The set of landmarks forming the most fine-grained level \(L^0\) equals the set of original data-points. Each subsequent level is then a subset of the previous level \((L^0 \supset \ldots \supset L^{n-1})\) with \(n\) corresponding to the number of levels. The hierarchy is explored through similarity embeddings, typically starting by embed-
Landmark Selection. The landmarks for level \( L^k \), \( k > 0 \), are selected based on their connectivity in level \( L^{k-1} \). In practice, the connectivity is defined by carrying out a set of random walks on the underlying neighborhood graph. In the original HSNE implementation, a threshold is defined on the number of terminated random walks, to identify the most important landmarks in a data-driven manner. While a threshold is a parameter that can be adjusted by the user, this is not very intuitive—it is hard to predict and control the number of landmarks on each level and the number of levels needed for a desired result. We thus present a modified selection criterion that provides better control over the reduction between hierarchy levels in Section 5.2.1.

Landmark Expansion. When zooming into or expanding a set of landmarks \( S^k \) to get more detail, HSNE makes use of a concept called Area of Influence (AoI). To define the AoI of a landmark \( L^k_i \), a second set of random walks is started from each node in \( L^{k-1} \). When a random walk reaches \( L^k_j \), the start node is added to the AoI of \( L^k_j \). The influence \( I_{L^k_i}(L^k_j) \) of a landmark \( L^k_i \) is defined by the fraction of random walks starting at \( L^k_j \) that end in \( L^k_i \).

Ultimately, to expand a selection of landmarks \( S^k \), we compute the combined influence of all landmarks on every landmark in \( L^{k-1} \). The set of landmarks \( S^{k-1} \) corresponding to the expansion of \( S^k \) is then the set of landmarks for which the combined influence of all landmarks in the selection is greater than a predefined threshold \( \gamma \):

\[
S^{k-1} = \left\{ L_{j}^{k-1} \mid \sum_{L_{j}^{k-1} \in S^{k}} I_{L^k_i}(L_{j}^{k-1}) > \gamma \right\}
\] (5)

Using \( \gamma \) this way means that the expansion of different sets can produce identical or overlapping results. For example, expanding either of the two sets \( \{L_1^0\} \) or \( \{L_1^0, L_2^0\} \) in Figure 6 with the default \( \gamma = 0.5 \) results in the identical set \( \{L_1^0, L_2^0, L_3^0\} \). Accordingly, when traversing the hierarchy upwards, we need to identify and select one of the given sets. We contribute a solution for computing such a set, presented in Section 5.2.2.

Landmark Similarity. Besides using the AoI for selecting the landmarks used for expanding, the pairwise degree of overlap of the respective AoIs of two landmarks in \( L^k \) with \( i > 0 \) defines the similarity between these two landmarks. By defining the similarity of the two landmarks in terms of the underlying neighborhood graph on the previous level the underlying manifold of the data is preserved even in the most abstract levels of the hierarchy. This notion of similarity is defined per level and thereby defines the mapping for the embedding per level. Therefore, the similarity cannot be directly computed when combining landmarks from different levels, as could be done for example by using Euclidean distances. We discuss ways to combine similarity sub-matrices from different levels in Section 5.2.3, fulfilling requirements V1–V3.

5.2. HSNE Extensions

To accommodate the integration of multiple HSNE levels into a single embedding and to allow a more fine-grained exploration we propose the following extensions to HSNE.

5.2.1. Landmark Selection

Here, instead of using a threshold on the number of terminated random walks, the user can directly specify the fraction of the landmarks on each level to proceed to the next level. In other words, based on a user-specified threshold \( p \) the top \( p \)-th percentile of landmarks, according to the terminated random walks is chosen, resulting in a fixed reduction of \( 1 - p/100 \) between two adjacent levels. Specifying the percentile is a more intuitive way of defining the granularity of the hierarchy and since the reduction is known at the time of computation the number of levels for the hierarchy can be computed automatically. In practice we typically observed reduction by approximately an order of magnitude in the original implementation, with the default hard threshold. Consequently, setting \( p = 90 \) yields similar results. To allow a more fine-grained exploration of the hierarchy, we set the default value to \( p = 75 \), meaning the number of landmarks is reduced to 25% between two adjacent levels. Even lower reduction rates are possible, however, they pose the risk of including uninformative landmarks.

5.2.2. Hierarchy Traversal

The proposed exploration of the HSNE hierarchy is much more fluent than the rather rigid original approach. Originally, every zoom operation results in an additional view and hierarchy traversal is strictly top-down. Here, a complete analysis session is carried out in a single embedding view. Typically, such a session will combine multiple instances of any of the workflows introduced in Section 2. As such, it will consist of subsequently setting the focus multiple times, adding and resolving a secondary focus, increasing and decreasing the LoD in different regions. Such a fluid interaction requires a more flexible handling of the traversal of the hierarchy.

In particular, HSNE does not provide means to zoom out. If less detail is required, one would need to find the corresponding previous view and continue exploration from there. As described in

\[
\begin{align*}
I_{L^1_i}(S^0) & = 0.3 & I_{L^1_i}(L^0) & = 0.3 \\
I_{L^1_i}(S^1) & = 0.3 & I_{L^1_i}(L^1) & = 0.3 \\
I_{L^1_i}(S^2) & = 0.3 & I_{L^1_i}(L^2) & = 0.3 \\
I_{L^1_i}(S^3) & = 0.3 & I_{L^1_i}(L^3) & = 0.3 \\
\end{align*}
\]

\[
\begin{align*}
S^0 & = \{L^0_1, L^0_2, L^0_3\} \\
S^1 & = \{S^0_1, L^0_1, L^0_2\} \\
S^2 & = \{S^1_1, L^0_1, L^0_2, L^0_3\} \\
S^3 & = \{L^0_1, L^0_2, L^0_3\} \\
\end{align*}
\]

**Figure 6: Example for Traversing the HSNE Hierarchy Upwards.** To find the landmarks best representing the selection \( S^0 \) we compute the fractional influence \( I_{L_i}(S^0) \) for all landmarks and select those above a user-defined threshold. For example using 0.5 would yield \( S^1 = \{L^1_1, L^1_2\} \). Note that this is not the only possible set \( S^1 \) as expanding \( L^1_3 \) alone would yield the same \( S^0 \).
Section 4.1.1 we implemented a tree structure, tracking the complete exploration. In most cases decreasing the LoD corresponds to reverting a previous increase of the LoD and therefore we can retrieve the less detailed representation directly by moving up in the existing tree structure. However, in rare cases it can be necessary to create a new group on a lower LoD. For example, when differentiating a focus that is already at the highest LoD, all contexts should be moved to a lower LoD that might not be in the tree.

As described in Section 5.1 there is not necessarily a unique solution to find the less detailed set \( S^{k+1} \) corresponding to \( S^k \). In principal a minimal set can be found by testing out permutations of all involved landmarks. Such an approach could become very costly. Instead, we propose to approximate \( S^{k+1} \) by computing the fraction of the influence of every landmark \( L_{i}^{k+1} \in L^{k+1} \) on the selected landmarks \( S^k \) compared to its total influence.

\[
I_{L_{i}^{k+1}}(S^k) = \frac{I_{L_{i}^{k+1}}(S^k)}{I_{L_{i}^{k+1}}(L^k)}
\]

(6)

Here, \( I_{L_{i}^{k+1}}(S^k) \) is the total influence of the landmark \( L_{i}^{k+1} \) on the selection \( S^k \) and \( I_{L_{i}^{k+1}}(L^k) \) the total influence of the same landmark on the complete level \( L^k \). We can then select the landmarks with a high relative influence on the selection, by thresholding on \( I_{L_{i}^{k+1}} \). We found experimentally that a threshold of 0.5, meaning more than half of the representation of a landmark corresponds to the current selection, creates small sets representing all of the input.

While the resulting set can be one of multiple possible solutions, in general it consists of the most important landmarks for the given input. We illustrate the example introduced in Section 5.1 in Figure 6. With \( I_L = 0.5 \), decreasing the LoD of \( S^0 = \{L_1^0, L_2^0, L_3^0\} \) results in the corresponding set \( S^1 = \{L_1^1, L_1^0\} \). \( L_2^1 \) is not strictly necessary in \( S^1 \), as expanding only \( L_1^1 \) would produce the same \( S^0 \). However, in such cases the additional landmarks are of low impact (here \( I_{L_2^1}(L^0) = 0.5 \)) and in practice would rarely be selected as landmarks during the construction of the hierarchy. Furthermore, in an explorative setting the user would be able to probe the resulting embedding and inspect the similarity of the landmarks allowing them to identify such outliers.

5.2.3. Landmark Similarity

To compute the similarity embedding of the data points, combined from different levels of detail (LoDs), we first need to define the similarities between these points. While this could be done by directly computing the distances in the high-dimensional space, the resulting similarities would not reflect the non-linear distances preserved in the HSNE hierarchy. HSNE does provide a similarity matrix \( L^k \) at each level of the hierarchy. However, it does not directly provide similarities for combinations of multiple LoDs. Here, we discuss ways to create a similarity matrix for combining multiple LoDs, underlining the similarity embedding.

One of the goals for the similarity embedding is that the focus must use more space in the visualization (requirement V1). As discussed in Section 5.1, HSNE defines the similarity of two points by the relative degree of overlap in their respective AoI. As the levels become smaller and the neighborhoods less detailed towards the higher levels of the hierarchy, the relative degree of overlap and consequently the values for the similarity become larger. We can use this property to fulfill requirement V1, by creating the similarity matrix as a combination of partial similarity matrices from different levels. For points that shall occupy more space in the embedding (the focus), we take the partial similarity matrix from a more detailed level, while for points that shall occupy less space (the context) we take the partial similarity matrix from a less detailed level. In principal the result is a mixed matrix consisting of four partitions, the matrices for the focus, \( F^k \), \( C^k \), and context, \( C^k \), as well as two blocks describing the similarities between focus and context \( F \cup C \) and between context and focus \( C \cup F \). Since the distances in HSNE are symmetric, \( F \cup C \) is a rotated version of \( C \cup F \). In case \( l = k \) the matrix is identical to the similarity matrix of the union \( F^k \cup C^k \). Using \( F \cup C \) and \( C \cup F \) from their respective levels guarantees that their structure in the combined embedding is as close as possible to the structure of the groups embedded separately.

Based on creating the similarity matrix through combination of partial matrices from different hierarchy levels, we propose two different modes, Simple Matrix Combination and Pull-Up to fulfill the competing requirements V2: the focus must be separated from the context and V3: connections between focus and context should be maintained.

Simple Matrix Combination. To achieve maximum separation between focus and context (requirement V2) we can simply set submatrices \( F \cup C \) and \( C \cup F \) to zero. This effectively cuts all inter-group connections and the embedding will separate the groups.

Figure 7: Distance Matrix Combination Modes for a basic Focus+Context graph shown in a). Simple Combination, b). Connections between focus and context are set to 0. Pull-Up, c). The focus is pulled up to the context level to extract the partial matrix between focus and context.

Figure 8: Distance Matrix Combination Modes with 3 levels of detail. A new focus was defined on \( F^3 \) from Figure 7, a), resulting in the respective nested distance matrices in b) and c).
Considering that we already have the similarity matrices per group, the most straightforward approach to do this simply concatenate these matrices. Figure 7a shows the basic Focus+Context tree with a single focus and context group, respectively. For this example the context $L_2$ is taken from hierarchy level 2, while the focus $F_1$ is one level more detailed. Figure 7b illustrates the combined similarity matrix for this approach.

**Pull-Up Approach.** To balance requirements $V_2$ and $V_3$, we need to sensibly fill $F_1 F_2^T$ and $C_1 F_2^T$. As described in Section 5.1 each less detailed level $L^k$ is a subset of the previous level $L^{k-1}$. This means, we can find all points contained in $L^k$ in $L^{k-1}$ and, vice versa, some points contained in $L^{k-1}$ in $L^k$. There are several ways to exploit this fact to construct a similarity matrix that considers interaction between focus and context.

Here, we propose to pull-up the focus to the context level and partially fill the empty part of the similarity matrix with those connections available in the context level. We start with the submatrices from the simple approach for the intra-group connections. Since we track the exploration in the Focus+Context tree, we can directly look-up $F_2$. With $F_2$, we can now extract the parts $F_2 C_2^T$ and $C_2^T F_2$ of the similarity matrix from level 2 of the HSNE hierarchy. However, the landmarks that were added when zooming into $L_1$ are not available in level 2 and therefore do not have connections to $C_2$. Figure 7c shows the resulting matrix.

For the examples in Figure 7 we used a simple graph with only one level difference between the focus and context. In practice the distance can vary, for example after applying multiple $d_s(F^3)$ operations. Independent of the number of levels between the two groups, the matrix is constructed in the same way. For explorations with multiple context sets (Figure 8) the process of combining the similarity matrices is repeated iteratively. Independent of the mode we can simply replace the similarity matrix for the previous focus (here, $F_1 F_1^T$) with a new sub-matrix that is constructed in the same way as described above.

**Discussion.** We found that the simple matrix combination method is very effective when the main goal is a strong separation between the focus and context groups while pull-up approach effectively balances requirements $V_2$ and $V_3$ while adhering to $V_1$. For cases where we separated a cluster in focus and context groups, the groups stay close together at their separation points, but separate enough to not disturb the intrinsic focus partition (Figure 9c). The generally stronger connections from the less detailed hierarchy level effectively balance the connections missed for points that exist in $F_2$ but not in $F_1$. For the pull-up approach we ad some links between the focus and context groups by pulling the focus to the more abstract level of the context. In principal we can also push the context landmarks to the focus level without adding additional detail. This would allow us to fill the complete intra-group sub-matrices. We expected that adding such a large amount of connections weakens the separation between the groups too much. We implemented two methods based on pulling down the context and could indeed observe undesired mixing in experiments. All methods introduce some distortion, as compared to embedding data from a single LoD, only. For the simple matrix combination this distortion is between groups and easy to identify. The introduction of links between groups in the pull-up approach can additionally lead to distortions within a group, as only some landmarks within a group will be connected to another group. If such landmarks are weakly connected within their specific groups, they could be pulled out of their respective group. In practice, however, we noticed that this rarely happens as the landmarks that are connected to other groups are also strongly connected within their respective groups, as this was a criteria to select them as landmarks initially. This can also be seen in the examples in Figure 9 where the structure of the foci in Figure 9b and Figure 9c is very similar to the original HSNE embedding of the focus only in Figure 9d.
5.3. Implementation

We extended the open source High Dimensional Inspector library to allow the described seamless traversal of the hierarchy in both directions. The prototype for illustrating the interaction design is implemented in Cytosplore [HPvU16, vUHP17]. The library and application are implemented in C++. Computation of the hierarchy and embeddings are performed with the original HSNE library. For a detailed analysis of the computational performance we refer to our previous work on the HSNE algorithm itself [PHL16].

Typically a Focus+Context embedding will converge slightly faster since we initialize it with the previous positions, whereas the original HSNE creates a new embedding that is initialized randomly.

On the other hand, some of this gain will be offset by the fact that we introduce additional landmarks to the embedding compared to the standard HSNE of only the focus selection. For an intuition, we provide computation times for the examples in Figure 9 and Figure 10. All measurements were taken using a quad core intel core i7 6820HQ at 2.7Ghz with all data in working memory. We use OpenGL for fast rendering of the embeddings and to derive the topographical maps for visualizing the level of detail on-the-fly, directly during the optimization.

6. Use Case

To illustrate the effectiveness of Focus+Context HSNE, we follow a use case of the original work by Pezzotti et al. [PHL16] on the exploration of hyperspectral images of the sun. The dataset consists of twelve dimensions, represented each by a gray-scale image, corresponding to different spectral regions (Figure 10a). The image resolution is $1,024 \times 1,024$. We consider every pixel a twelve-dimensional data point, resulting in roughly one million data points as input to HSNE. An illustration of the original exploration is shown in the original publication [PHL16, Figure 6]. As indicated in Section 2, originally, the low-detail overview embedding is explored first, by probing different regions of the embedding. Selecting a region in the embedding highlights the corresponding pixels in an image view. After identifying and zooming into two regions of interest, the more detailed plots are displayed separately in new views and further exploration and probing is limited to these sub-regions.

Here, in addition to using the presented Focus+Context approach for the exploration, we augment the visualization in image space by recoloring. Cheng et al. [CXM18] present recoloring of multi-dimensional images, based on an optimized 2D projection, here we follow a similar approach, but use the $(x,y)$ coordinates in the embedding, as shown by Abdelmoula et al. [APH18]. When inspecting the embedding and image views, the user needs to be able to identify the origin of pixels in the HSNE map and compare different pixels according to their origin. Furthermore, we do not want to steer attention with the colormap. Therefore, we picked several colormaps that provide a reasonable compromise between tasks ER2, ER3, SR2 presented by Bernard et al. [BSM15, Table 3]. For the example in Figure 10, we chose the Mittelstädt et al. colormap [SBM14]. In the implemented prototype we allow the user to choose between the Bremm et al. regular and stretched [BLBS11], TeulingFig3NoWhitening [TSS11], and the presented Mittelstädt et al. colormaps. We extracted the colormaps from ColorMap-Explorer [SBT15].
the color corresponding to their location in the plot. We can immediately identify two main groups in this plot. Comparing the spatial representation in Figure 10h we can see that the two groups correspond to the surface of the sun (pink) and the space (green). While some more fine-grained structure can already be identified at this LOD the main focus here is on the separation of the two groups.

Following the example of Pezzotti et al. [PHL⁺16], we first want to inspect the space cluster in the embedding. Therefore, we select this cluster and create and differentiate the focus, \(d_c(C)\). The resulting embedding is shown in Figure 10c and the corresponding image in Figure 10i. We can see that the embedding behaves as desired, leading to a more compact representation of the context, \(C\), while the focus, \(F\), becomes larger and more detailed. A curious detail shown in the theoretical original becomes visible here. A small group of points (arrow in Figure 10c) separates from the main group. Comparing the image view, we see that this cluster corresponds to the overlaid AIA-logo. We create a new focus, excluding this group, and differentiate, \(d_c(C)\), resulting in the embedding in Figure 10d. Now the focus strongly dominates the plot, while the context groups are again more compact. As a result the surface now hardly shows any structure in the image view Figure 10j, leading the attention to the increased structure in the focus. Here, we can now clearly see the layered structure of the corona by its color from deep purple to light blue. Two strong sunflares can be identified on the left side of the image in blue, while areas with reduced activity show up on the top right (arrow) and bottom.

In the next step we want to investigate the surface in more detail. Thanks to the proposed Focus+Context interaction model we can simply select the corresponding landmarks in Figure 10d and call set focus & differentiate, \(d_c(C)\). As described in Section 4 the selection now becomes the new focus, one level more detailed than its initial level, while the remaining data becomes the context and is moved back up to the initial LOD. The resulting embedding (Figure 10e) now assigns much more space to the surface group, while the space group collapses to a much smaller region. We can already see more detail on the sun surface (Figure 10k). Hotter regions are on the lower-right part in the embedding, resulting in a purple color in the image view, while orange parts correspond to lower temperature regions. In particular, three active regions on the surface start to appear (Figure 10k, white arrows). The same differentiation had been observed by Pezzotti et al. [PHL⁺16]. Finally, we select a small region on the bottom part as the new focus and differentiate, \(d_c(C)\). We can now clearly see that the purple area corresponds in large part to a large low-temperature area on the top right of the surface, corresponding to an area that also showed clear differentiation in the corona (black arrows in Figures 10j–10l). For computing the combined similarity matrix, we used the pull-up approach as described in Section 5.2.3. As indicated by the purple arrow in Figure 10f the points in the focus area are still in close proximity to the points they were separated from during selection. For comparison we show the same focus and context groups with the similarity matrix combined with the simple approach in Figure 10g. Here, the new focus is completely separated from its origin. While the recolored images (Figure 10l and 10m) show very similar structure, the connection between the surface context and focus groups is lost in the embedding with the simple approach. In most cases the pull-up approach, preserving this information, should be preferred.

7. Conclusion

We have presented a framework, including an interaction model and visual design for Focus+Context exploration of hierarchical embeddings. We extended the hierarchical dimensionality reduction technique HSNE to support the proposed model. We have demonstrated its effectiveness in an exemplary use case on hyperspectral images. In particular, incorporating the Focus+Context concept directly into the mapping of the dimensionality reduction by combining similarity matrices from different levels of detail, is a natural fit for non-linear embeddings. This approach can be tuned by selecting the type of similarity matrix combination.

While our extensions to the HSNE hierarchy allow for a much more fine-grained exploration compared to the original implementation, the depth of a zoom operation is left to be specified by the user. A future research direction could be to optimize the levels according to the available space. For example, skipping several levels during the differentiate operation while reducing detail of the context at the same time. We illustrate a brief case study for the exploration of multi-dimensional images in Section 6. The application of the presented techniques to real-world data exploration tasks alongside a structured evaluation for these tasks provide open questions for future work.

Acknowledgements. This work was partially funded through NWO projects 12720, VinPHRe, and 12721, Genes in Space, and the VIDI project (#813558) on Visual Data Science for Imaging Biomarker Discovery, supported by the Trond Mohn Foundation in Bergen.

References


© 2019 The Author(s)
Computer Graphics Forum © 2019 The Eurographics Association and John Wiley & Sons Ltd.