This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3065443, IEEE Transactions on Visualization and Computer Graphics

Multiscale Unfolding: Illustratively Visualizing the Whole Genome at a Glance

Sarkis Halladjian, David Kouřil, Haichao Miao, M. Eduard Gröller, Ivan Viola, Tobias Isenberg

Abstract—We present Multiscale Unfolding, an interactive technique for illustratively visualizing multiple hierarchical scales of DNA in a single view, showing the genome at different scales and demonstrating how one scale spatially folds into the next. The DNA's extremely long sequential structure—arranged differently on several distinct scale levels—is often lost in traditional 3D depictions, mainly due to its multiple levels of dense spatial packing and the resulting occlusion. Furthermore, interactive exploration of this complex structure is cumbersome, requiring visibility management like cut-aways. In contrast to existing temporally controlled multiscale data exploration, we allow viewers to always see and interact with any of the involved scales. For this purpose we separate the depiction into constant-scale and scale transition zones. Constant-scale zones maintain a single-scale representation, while still linearly unfolding the DNA. Inspired by illustration, scale transition zones connect adjacent constant-scale zones via level unfolding, scaling, and transparency. We thus represent the spatial structure of the whole DNA macro-molecule, maintain its local organizational characteristics, linearize its higher-level organization, and use spatially controlled, understandable interpolation between neighboring scales. We also contribute interaction techniques that provide viewers with a coarse-to-fine control for navigating within our all-scales-in-one-view representations and visual aids to illustrate the size differences. Overall, Multiscale Unfolding allows viewers to grasp the DNA's structural composition from chromosomes to the atoms, with increasing levels of "unfoldedness," and can be applied in data-driven illustration and communication.

Index Terms—Multiscale visualization, spatially-controlled scale transition, visual abstraction, illustrative visualization, genome, DNA.

1 Introduction

Exploration of 3D data is essential for all physical sciences, from the smallest phenomena in particle physics to the largest known structures in astrophysics. In many of the sciences in this spectrum (including biology) and applied fields such as medicine and engineering, it is not only important to understand data at a single spatial scale. Instead, many phenomena require a multiscale data analysis. Within visualization, researchers have contributed to this analysis by providing several means for the exploration of multiscale data (e. g., [5], [9], [21], [25], [33], [34], [52], [55], [59]), for many of the mentioned application domains.

To date, most of the proposed data exploration strategies follow a temporal control in the data navigation. At any given time, the visual representation shows a subset of the 3D dataset at a single spatial scale factor, and this scale factor can be changed interactively, i. e., over time. This strategy has the benefit that it allows viewers to experience and explore the actual 3D spatial structure of the data. It requires them, however, to examine the data's scale changes sequentially and, thus, requires them to memorize the different spatial scale configurations for comparison. Moreover, if data has a complex spatial configuration at a given scale as the heavily intertwined DNA then it is difficult to make a comparison—even if both scales would be shown side-by-side.

An alternative way of exploring multiscale data is to use a spatial control of scale. The visual representation of different scale configurations is shown at different locations in a single image. Illustrators have used this strategy in the past to depict the different scale aspects of multiscale data such as the DNA, anatomy structures, and astrophysics (e. g., Fig. 2). This paradigm has the advantage that it allows us to explain—in particular to non-experts and in general education—the inter-connectedness of the scales as well as (local examples of) their spatial configurations, all in one view. This technique is particularly well suited for phenomena in which the spatial scales are, in fact, physically connected like in genetic macro-molecules (i. e., DNA, RNA).

We thus take inspiration from static illustrations and present a framework that allows us to interactively create and explore spatially-controlled multiscale visualizations based on captured (Hi-C) and reconstructed [53] genome data (Fig. 1). We demonstrate how to unfold the data according to their scales and depict them following chosen image-space paths. Next, we spatially transition from one unfolded scale to the next, with each transition depending on the characteristics of the two involved scales. We also introduce interaction techniques to navigate our new multiscale representation with spatially-controlled abstraction and relate our representations to temporally controlled multiscale visualizations. Finally, we report on feedback from professional illustrators and domain experts, which indicates that our illustrative visualization facilitates understanding and communication between experts and laypeople.

2 Related Work

While we focus on illustrating genome data, our approach relates to past work in several other fields. First, we briefly discuss hand-crafted multiscale visualizations. Next, we review how researchers have realized multiscale visualizations in the past in general, before focusing on the visualization of molecular and genome data. Finally, we relate our work to interactive multiscale approaches.

2.1 Hand-crafted illustrations

Artists *hand-craft* illustrations to depict linear structures at multiple scales in an unfolded manner. Example illustrations concern the

- *S. Halladjian and T. Isenberg are with Université Paris-Saclay, CNRS, Inria, LISN, France. E-mail: {sarkis.halladjian | tobias.isenberg}@inria.fr.*
- *D. Kouřil, H. Miao, and M. E. Gröller are with TU Wien, Austria. E-mail: {dvdkouril | miao | groeller}@cg.tuwien.ac.at.*
- *M. E. Gröller is also with the VRVis Research Center, Austria.*
- *I. Viola is with KAUST: King Abdullah University of Science and Technology, Kingdom of Saudi Arabia. E-mail: ivan.viola@kaust.edu.sa.*

Manuscript received September 11, 2020, revised January 28, 2021, accepted ????.

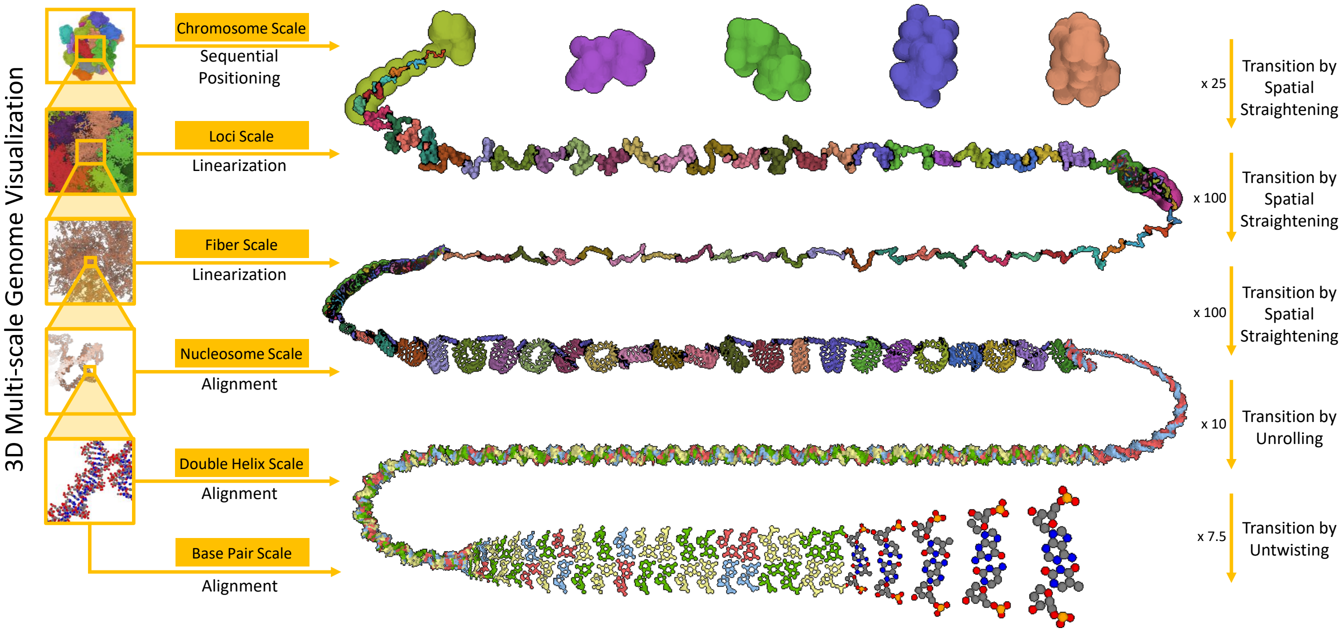


Fig. 1. Multiscale unfolding with continuous scale transitions of the 3D human genome, showing chromosomes, loci, fibers, nucleosomes, double helix, and flattened base pairs. Scales 1–4 are composed of details from the next-lower scale, and linearized based on the elements' own positions.

human genome (e. g., Fig. 2(a),(b)), muscle fibers (e. g., Fig. 2(c)), or astronomy data (e. g., Fig. 2(d)). These illustrations help a viewer to grasp the hierarchical structure, while global spatial characteristics are sacrificed. The artist uses the space as a resource to arrange the linear structure from its smallest to its largest scale representation. To the best of our knowledge, these illustrations are created manually and are usually not directly based on real data. These educational images are inspirational to us. With our work we provide means to automatically create similar illustrations based on real data and also make them interactively explorable.

2.2 Spatial transitions of multiscale representations

Multiscale phenomena are complex as they contain features that are only visible if viewed at a certain resolution. There are various strategies to deal with features at multiple scales. Glueck et al. [24] smoothly transition between coordinate grids for different scales to facilitate navigation and orientation in the multiscale environment. The method of Everts et al. [19] spatially contracts brain fiber tracts to represent the structure at a higher scale. Although brain fiber tracts have an elongated shape, this approach is not directly applicable to our case, as DNA strands do not share the necessary properties to apply the chosen aggregation strategies. In terms of rendering large complex scenes, Lindow et al. [42] propose a method to exploit the repetitive nature of molecules. Their approach bridges five orders of magnitude in scale. CellView [39] incorporates in a cell visualization a level-of-detail scheme that reduces the number of displayed atoms while the radii of aggregation spheres are increased for distant objects. The method is based on earlier work by Parulek et al. [55]. The CellView approach does not meet our needs, as we depict all levels in one image. Hsu et al. [27] cast non-linearly bent camera rays across several spatial scales. They create smooth transitions of views between different levels of detail and depict the multiscale nature of geometric and volumetric models. *Scalable Insets* are applied for the navigation of a large number of patterns in multiscale visualizations [40]. This technique uses magnified views to visualize small details that would not be visible either due to their size or location, which result from the multiscale nature of the data. Zhang [72] proposed a scale-space animation to visualize the spatial relationship between structures in multiple views. They argue to integrate cross-scale semantic information into an animation to convey the complex multiscale structure. Lueks et al. [43], based on an earlier system by Zwan et al. [65], proposed a dedicated control over the abstraction of molecular systems and thereby enable the creation of seamless spatial transitions. Their method does not deal with the spatial re-positioning of the structure. The DNA-specific approach by Miao et al. [47] provides the user with control over the level of abstraction through animated transitions in order to depict the correspondence between scale representations. Many multiscale visualization methods [19], [43], [47], [48], [65] apply continuous abstraction of the data for reducing the visual complexity. Visual abstraction has been formalized by Viola and Isenberg [67], with a recent update by Viola et al. [66], for the purpose of illustrative visualization. According to their definition of abstraction axes, our work is categorized into an abstraction along the geometric abstraction axis. Although these techniques offer solutions for certain types of multiscale phenomena, they are not applicable for spatially abstracting heavily tangled structures like the DNA. Spatial transitions are done for depicting multiscale phenomena and also for complexity reduction in general. In medicine, anatomical flattening [37] transforms a 3D object to a 2D representation for different analysis tasks. As an advantage, visual occlusion can be avoided in these 2D representations. Especially vascular structures are similar to DNA. There exist related approaches to spatially transform the elongated shape through straightening [10], flattening [62], and reformatting [51]. We are inspired by these techniques and advance them for genome data.

2.3 Visualization of genome data

Depending on the feature of interest, there are various techniques to represent individual molecules. Kozlíková et al. [35], [36] provide reviews on various molecular visualization techniques. Due to the massive number of DNA molecules, these techniques are only applicable to the atomistic scale in our data. For the

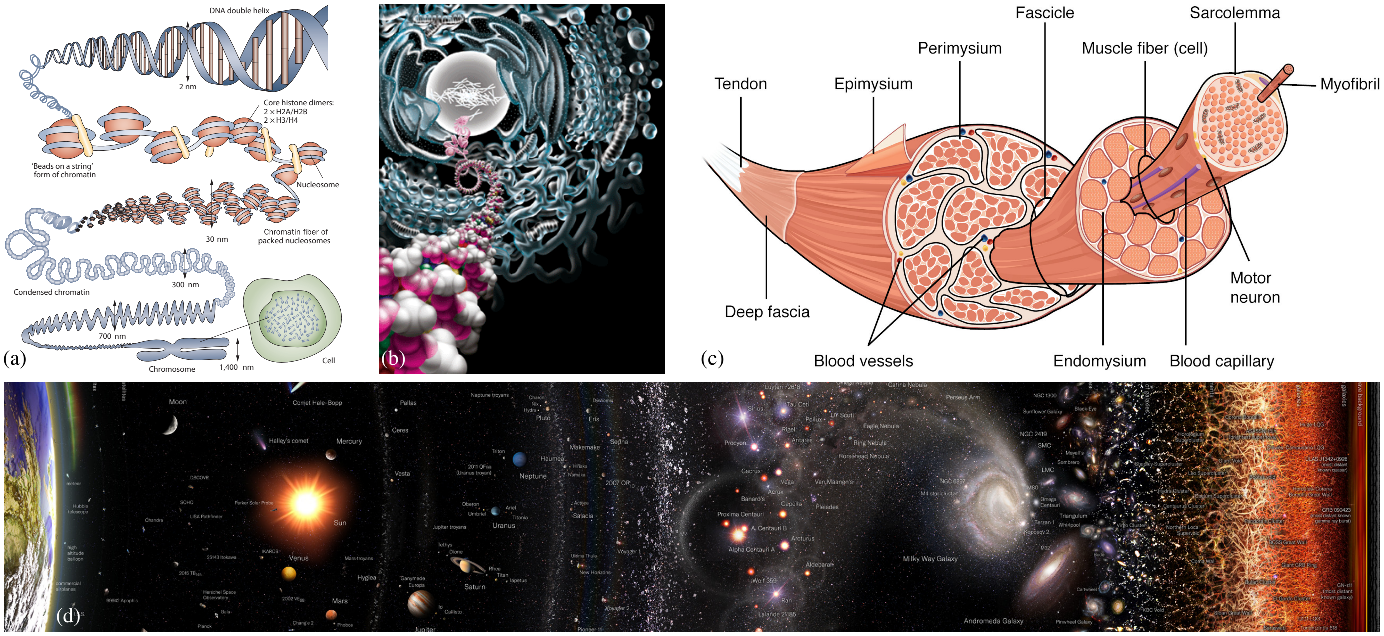


Fig. 2. Artistic depictions of image-space scale transitions for (a),(b) genome data, (c) anatomy, and (d) astrophysics. Images (a),(b) from [63]/[56] and © Springer Nature/The American Association for the Advancement of Science, respectively, used with permission. Image (c) by Leeah Whittier (illustrator) and Biga et al. [8] (authors), 🅭🅯🄎 CC BY-SA 4.0. Image (d) by Pablo Carlos Budassi (top & bottom cropped), 🅭🅯🄏 CC BY-NC 4.0.

specific application of genome data visualization, Nusrat et al. [54] provided taxonomies in this respect. The DNA's multiscale character presents numerous visualization challenges. Halladjian et al.'s [25] *ScaleTrotter*, e. g., bridges several orders of magnitude in scale and allows users to interactively explore the spatial structure. Their visual embedding approach, however, limits visibility to only one or two scales at a time. A linear arrangement of DNA molecules, in contrast, favors a spatially abstracted representation, where all scales are visible in one image (as in the traditional illustrations in Fig. 2). Miao et al. [46] proposed an *abstraction space*, where DNA origami strands can be continuously rearranged in space, going from a 3D to a 2D to a 1D representation. The user can adjust the abstraction level, but again only one scale is visible at a time. A steerable abstraction space [46], [52], [65] enables users to change the scale of a representation depending on interactions with a control panel. Connectomics datasets comprise structures of the nervous system and share similar properties with DNA, due to the tubular shape, large size, and given connections. In this domain, Mohammed et al. [52] continuously abstract the representation of astrocytes and neurites through navigating a specific abstraction space. In general, past visualization techniques for molecular structures and genome data do not offer means to spatially and simultaneously unfold the various scale representations.

2.4 Cross-scale interactions

While we can imitate hand-crafted static illustrations, a key argument for us is the interactive exploration in the spatially transformed representations. In our data-driven approach, we enable the viewer to access the features across different scales. In the past, researchers proposed various techniques for interacting and navigating in multiscale environments. Furnas and Bederson [22] deal with large information scale-space diagrams for understanding multiscale interfaces. In 2D visual spaces, pan, zoom, and bird's eye view are commonly used techniques. For objects of interests that are far apart these methods are ineffective. Javed et al. [29] present the PolyZoom technique, which enables users to build hierarchies of focus regions that can be stacked on each other to depict subsequent scales. Elmqvist et al. [18] proposes a space-folding technique for the same set of problems. Their distortion-based method folds the intermediary space to guarantee visibility for multiple focus regions. McCrae et al. [45] suggest an image-based environment representation to allow users a consistent navigation in multiscale datasets, such as the Earth. There are very different approaches to deal with multiscale phenomena and the data require specific interaction techniques. As our Multiscale Unfolding technique creates a multiscale linearized representation, we have to also provide appropriate interactions for it.

3 Concept And Overview

In our work we focus on DNA data with its unique structural properties. Specifically, DNA is extremely thin and long. If stretched out, the 3.2 Gbp (giga base pairs) [2], [60] of the human genome would be approximately 2 m long [2, p. 179], yet the DNA B-form's double-helix is only 2 nm in diameter [3], [68]—a ratio of 10^9:1. It is organized in different structural forms at different scale levels for which genome scientists have derived structural data [53]. The involved scales include: the *chromosome* scale, the *loci* scale, the *fiber* scale, the *nucleosome* scale, the *double-helix* scale, and the *nucleotide* scale. With the exception of the 46 chromosomes of the human genome, which are separate entities, the DNA is also physically connected through all the scales. To fit this extremely long chain into the 6 µm cell nucleus, the chain is tightly packed together [2, p. 187]. The result is a complex spatial arrangement of supercoiled [28] DNA strands at the different scale levels. To fully understand the activity and function of a genome, experts have to explore both its sequential chain structure and its spatial structure. Yet it is impossible to understand the sequential chain if looking at it in a traditional three-dimensional genome representation. In ScaleTrotter [25] (Fig. 3), for instance, only two scales are visible at a given time and user interaction is required to explore one scale and traverse to another one. It is obviously also impossible to understand the spatial structure with only a linear sequence of

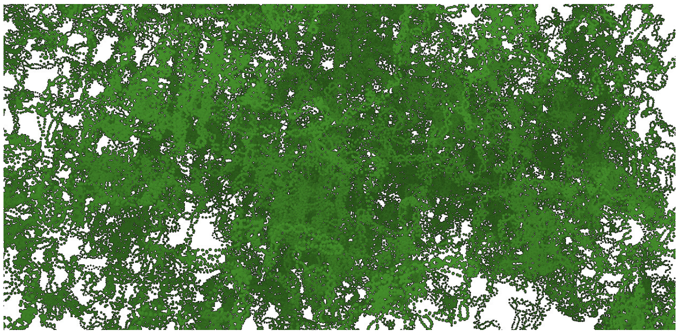


Fig. 3. Regular 3D visualization of the nucleosome positions (generated with ScaleTrotter [25]). The genome's linear structure is invisible.

elements at any scale. We thus have to partially sacrifice the spatial 3D structure to be able to explore both aspects of the genome. This is a common approach in visualization as we outlined in Sect. 2.2.

We thus face three major tasks: (1) making the sequential structure of the DNA visible and understandable across several scales with uncluttered views, (2) providing users with an understanding how the structure is linked to itself across different scale levels, and (3) making such a visualization interactively explorable. A major challenge of existing temporal exploration techniques for multiscale data is that they increase the viewers' memory load and that they are only effective with interaction. It would be necessary to show several or all scale levels in a single view. We take inspiration from professional illustrators (e. g., Fig. 2) and the few existing yet limited approaches from the visualization literature [27], [43]. Consequently, we control the depiction of scale based on the individual elements' location in image space.

To realize a spatialized control of the depiction scale (or "zoom level") we use four key concepts. The ***first key concept*** is the ***spatial straightening*** of the sequential structure. The straightening depends on the depicted scale level and is performed in relation to the neighboring scales. In visualization, reformatting has been applied, for example, to flatten 3D structures [37] to resolve visibility issues or to analyze the structures in a simplified way. We also use straightening for these purposes and adapt it to the tremendous scale differences in the genome data, which are of multiple orders of magnitude in size. Reformatting as an approach to normalize data and make them comparable has already been used, e. g., in geographic illustration more than 150 years ago (Fig. 4). Tufte [64, pp. 76–77] describes this illustration mainly as a means to compare river lengths that goes beyond "just another decorated bar chart" because it adds local detail and contains more (local) information about the rivers than their length alone. The straightening of the rivers is done in a scale-dependent way where large bends are removed and smaller ones are kept. This is analogous to a frequency-based or wavelet-based analysis where the various frequencies are selectively dampened or removed. In our case, straightening allows us to understand the local 3D structure at a given scale without the "wiggliness" of the genome sequence from a higher scale cluttering our 3D view. The spatial straightening serves as a *visibility management* technique, similar to Brosz et al.'s [12] *transmogrification* for 2D visualizations. We realize it by applying data from two neighboring scale levels, using Halladjian et al.'s [25] relationship between data and semantic hierarchies. As we depict, e. g., the semantic level of *fibers*, we render, in fact, the positions of the next-lower, more detailed data level of *nucleosome positions*. We preserve the local detail of the *nucleosome positions* at the lower level, while we straighten and remove the details from the path of the fiber positions at the higher level (Fig. 5). We perform straightening in a data-dependent semantic way relative to an upper level. It may involve, e. g., unbending, untwisting, unfolding, and untangling the corresponding lower level, and we describe its realization in Sect. 4.1.

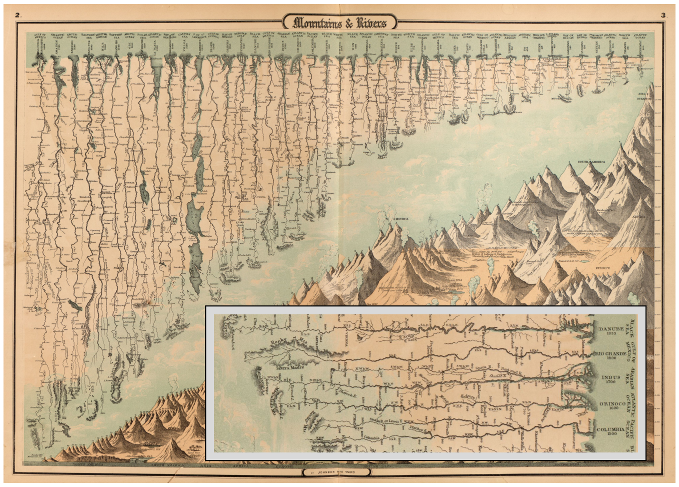


Fig. 4. Straightened river illustration. Image from [30, pp. 2–3], 🄏.

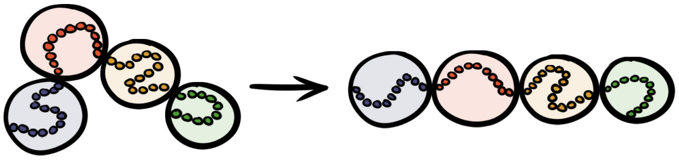


Fig. 5. Spatial straightening applied by re-arranging the spatial detail from a more detailed scale (the *straightenee*) to place the positions of the coarser scale (the *straightener*) along a chosen path.

Except for the chromosomes, the size of the DNA does not allow us to show all elements at a specific level after straightening in a single view. Due to the DNA's repetitive nature this is not a problem in many applications and showing a subset is sufficient to illustrate a specific level's characteristics. Nonetheless, we integrate all scales in a single view and need to convey how one scale relates to the next one as they are connected spatially and in scale space. We apply the straightening on several scales, but so far we treat each scale as independent from the others. The ***second key concept*** are thus ***continuous spatial transitions*** between scale representations that illustrate how the structure of a more detailed scale is packed and folded into the next-coarser one. We transition between scales via multiple linked views [6], [13], [70] of the spatial 3D data, yet without the need to separate the views from each other. We discuss how we realized the continuous transitions and how they relate to the straightened scale levels in Sect. 4.2.

With these transitions, we combine all scale levels together in a single view, depicting a representative detail section for each of them. We apply this straightening iteratively down the scale hierarchy. In each of the iterations the coarse-scale details are straightened out (i. e., removed) while the fine-scale details are kept. The ***third key concept*** is the ***role change of the scales*** as we move between the scales. For example, going down the hierarchy the fine-scale of the previous iteration becomes the coarse-scale of the current iteration. Our Multiscale Unfolding extends the two-scale strategy of spatial scale transitions as it was previously discussed in the visualization literature. We propose a multiscale strategy applicable as long as data for the spatial straightening and the transition depiction is available from more than two scales. One way of arranging the straightened information in the 2D image plane is a scanline layout as given in Fig. 1. To fit all levels into a single view, the scale changes from top to bottom with each row. The traversal direction flips as the levels are connected from one row to the next one. As a result we achieve a compact combined view across all scale levels. We are now able to display all scales of the DNA structure in a single view, with different sections depicting different scale levels. This has the benefit of no longer requiring interaction to understand how the different scales relate to each other. Nevertheless, as the ***fourth key concept*** we provide means to facilitate an ***interactive exploration of Multiscale Unfolding***. With interaction, viewers particularly benefit from the linked-view character of the visualization. For this purpose we reuse techniques from the HCI literature as we explain in Sect. 5.

4 Spatially-Controlled Multiscale Mapping

We base our approach on Nowotny et al.'s [53] reconstructed genome data, computationally derived from chromosome capture data such as Hi-C: realistic 3D locations for *chromosomes*, *loci*, *fibers*, and *nucleosomes* as well as the *nucleotide* positions for one nucleosome. This dataset is in an interphase configuration, not the often depicted mitotic or "H" chromosome. In this data, each chromosome contains 50–100 loci positions, a locus contains 100 fiber positions, a fiber has 100 nucleosome positions, and each nucleosome consists of 146 base pairs (bp). We derive the base pair locations from the 292 nucleotide positions in the nucleosome. We construct the linkers between two successive nucleosomes based on their orientation, with an average of 46 bp per linker.[1]

It is important to note that the six *data scales* (i. e., positions) are different from the *conceptual scales* that we perceive in a rendering [25]. The reason is that for each element of a given scale we only have a position in space, but no shape or size. We thus represent each element with a sphere. For example, if we render the positions of a sequence of 146 base pairs we perceive the shape of a nucleosome, even though we only render base pairs. So when we depict a conceptual scale, we actually render elements of the next-lower (more detailed) scale. Below we begin our discussion of Multiscale Unfolding by describing the spatial straightening for the scales so that we can control the placement of a scale independent from its detail (i. e., sub-elements).

4.1 Spatial straightening

The described data, at any given scale, is tightly packed, both in the genome's interphase configuration as we explore it and in the mitotic state as well. In a regular three-dimensional visualization such as ScaleTrotter, the actual linear structure of the genome is difficult or even impossible to see (e. g., Fig. 3). We thus spatially straighten the conceptual scale (i. e., the *straightener* level), for instance such that all nucleosomes we perceive can be arranged on a chosen path. We then adjust the positions of the rendered data level (i. e., the *straightenee* level)—in our example the nucleotide positions that make up a nucleosome—so that the connectivity of the straightenee level from before the spatial straightening is maintained also after the transformation. In our example the string of nucleotides that forms the nucleosomes needs to remain connected. As each scale has different requirements, we now describe each of them in turn.

Chromosome scale: Sequential positioning. We begin with the chromosome scale as the coarsest level of the data. This scale is different from all other scales because the chromosomes are independent structures, and are spatially not connected to each other. We render their local detail (in the form of *loci* positions) without any adjustments and arrange them sequentially in order along the chosen path as shown at the top of Fig. 1.

Loci and fiber scales: Linearization. The next scales are similar to each other—they are characterized by a tightly packed, intertwined chain—and we thus treat them the same way. We have to consider both the positions of the *straightener*—the conceptual scale which we straighten to follow a given path—and of the *straightenee*—the more detailed scale whose positions we actually render. Basically, we have the coarse-level path and fine-level details that are defined relative to the coarse-level path. Straightening now replaces the coarse-level path by a line, while we keep the fine-level details and apply them relatively to the line.

For this purpose we begin by computing displacement vectors for each of the elements of the straightenee scale (scale i). As we show in Fig. 6(a), we consider each straightenee element (small, filled green, blue, and brown circles) as belonging to a given, single element (large, non-filled circle; only blue shown) of the straightener scale (scale $i-1$). All N blue elements of the straightenee scale, e. g., belong to a single element of the straightener scale (the straightener's blue element: the large non-filled blue circle) whose position we depict with a red dot. We adjust the positions of the first $N/2$ blue straightenee elements based on their old locations and the new positions of straightener's green element and its blue element. Similarly, we reposition the second $N/2$ blue straightenee elements based on their old locations and the new positions of the straightener's blue and brown elements.

Let us consider the example of interpolating between the straightener's blue element and its brown element. We first determine the first and last elements e^i_1 and e^i_N in the straightenee sequence (at scale i), which belong to the same straightener element e^{i-1}_k (at scale $i-1$, with k being an iterator of this scale), as a base line with length l_i (Fig. 6(a)). We then subdivide this base line into $N-1$ sub-segments such that we have a reference position for each of the N elements of the sequence and center it on the position of the respective straightener element (shown in Fig. 6(b) for the blue sequence). For the second $N/2$ blue straightenee elements we then compute an offset vector $\Delta^i_{j,k}$ (j being an iterator of the straightenee scale i) from its corresponding position $f_{j,k} \in [0.5, 1]$ on the base line and its actual 3D position e^i_j before the interpolation (Fig. 6(b)), which represents the influence of the blue straightener element e^{i-1}_k (the influence factor being represented by $f_{j,k}$). However, we also need to consider the influence of the brown straightener element e^{i-1}_{k+1}. We thus use the same approach of deriving the base line for the brown straightenee elements, but we extend the baseline after centering by another $N/2-1$ sub-segments to represent the influence $f_{j,k+1} \in [0, 0.5]$ of the brown straightener element. As we show in Fig. 6(c), we then derive a second offset vector $\Delta^i_{j,k+1}$ for the same blue straightenee element e^i_j, but now for the influence $f_{j,k+1}$ of the brown straightener element e^{i-1}_{k+1}. The first $N/2$ brown straightenee elements are handled accordingly, and we repeat the process for all pairs of straightener elements. After we thus derived the offset vectors $\Delta^i_{j,k}$ and $\Delta^i_{j,k+1}$ for all k straightenee elements at scale i, we first transform them into the basis of the base lines as ${\Delta^i_{j,1}}'$ and ${\Delta^i_{j,k+1}}'$ and then compute, for the elements of the straightener scale $i-1$ after the straightening, the direction vectors

[1] Sources give various ranges for the linker DNA length such as "a few"–80 bp [2] or 20–90 bp [61]. We use 46 bp as the center of the 38–53 bp range cited on the respective Wikipedia article [69], but this number can be adjusted.

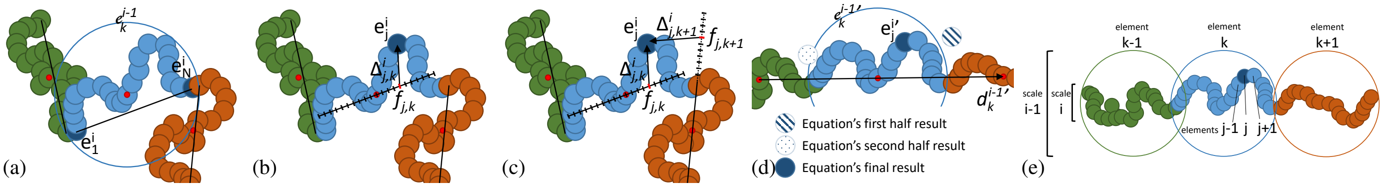


Fig. 6. Spatial straightening by linearization for the *loci* and *fiber* scales. In (e) we show a legend of the indices of the formalism we use.

$d_k^{i-1\prime} = e_{k+1}^{i-1\prime} - e_{k-1}^{i-1\prime}$. We then set $e_k^{i-1\prime}$ and $e_{k+1}^{i-1\prime}$ as the new straightener element positions and derive the new positions of the straightenee elements $e_j^{i\prime}$ as a straightforward interpolation between the offsets being applied to the new positions:

$$e_j^{i\prime} = \begin{cases} f_{j,k}\left(e_k^{i-1\prime} + d_k^{i-1\prime}(1-f_{j,k}) + \Delta_{j,k}^{i}{}'\right) + \\ f_{j,k+1}\left(e_{k+1}^{i-1\prime} - d_{k+1}^{i-1\prime}(1-f_{j,k+1}) + \Delta_{j,k+1}^{i}{}'\right) & : \; j > N/2 \\ \\ f_{j,k-1}\left(e_{k-1}^{i-1\prime} + d_{k-1}^{i-1\prime}(1-f_{j,k-1}) + \Delta_{j,k-1}^{i}{}'\right) + \\ f_{j,k}\left(e_k^{i-1\prime} - d_k^{i-1\prime}(1-f_{j,k}) + \Delta_{j,k}^{i}{}'\right) & : \; j < N/2 \end{cases}$$

We show the result schematically in Fig. 6(d). We apply this process to both the *loci* scale (as a straightener, with the *fiber* positions as straightenee) and to the *fiber* scale (as straightener, with the *nucleosome* positions as straightenee). We show the results of the application of this process in the second and third line in Fig. 1.

Nucleosome scale: Aligning the nucleosomes. The straightening of the *nucleosome* scale is conceptually simpler than the process for the previous scales because the shape of the nucleosome is known and does not have to be adjusted. The data do not contain information on the orientation of the nucleosomes, they only contain 3D positions. To align the nucleosomes on an arbitrary path we place the nucleosome positions in sequence, and then render the single included nucleosome model at these locations. We orient the model such that each donut-shaped nucleosome's "entrance" connects well to its predecessor and its "exit" connects well to its successor. We then connect each pair of consecutive nucleosomes with a section of double-helix that acts as linker DNA. As the data do not contain information about the length of this linker DNA we use an average of 46 bp per linker as explained above. We show the result in the fourth line from the top in Fig. 1.

Double-helix scale: Unrolling the nucleosome. At the conceptual double-helix scale we render positions of the atoms, which we derive from the *1AOI* PDB model. From this data we compute the positions of the base pairs. We later match these with the base pair positions in Nowotny et al.'s [53] nucleosome model for the transition between the scales. For the unrolling, we then fix the first base pair's position and then transform the remainder of the model with a rotation such that the connection between the first two base pairs is straigthened. We repeat this process until all 146 bp of the nucleosome have been "unrolled." We show the result of this process in Fig. 1 in the fifth line from the top.

Bases scale: Untwisting the double helix. In the final spatial straightening stage we "flatten" the double-helix. We calculate the positions of the individual bases from the *1AOI* model and derive vectors that encode the orientation of each base pair. Like with the nucleosome "unrolling," we start by fixing the first base pair and then, base pair by base pair, rotate the model such that the base pair centers all end up in the same plane. Each base's atom arrangement may still be difficult to understand or tell apart, so we introduce a second representation in which we also flatten the atom arrangement of each base, using manually adjusted models for *adenine*, *guanine*, *cytosine*, and *thymine*. Also based the *1AOI* model from the PDB, we created two models of each base with Samson Connect, one (A) with the flattened base geometry and one (B) in which also the backbone was rotated. We then interpolate between the stage after the rotation of the base center to the screen plane to the respective A-version and then to the B-version. As we start this transition, we also switch from coloring by bases to coloring by atoms and make base pairs bigger to better show the molecule structure, as depicted in the final row of Fig. 1.

4.2 Spatial transitions

Any two consecutive spatially straightened scales differ significantly in size and number of elements (with the exception of the last two). We need transition zones from one scale to the next one that show (a) the element size and number difference and (b) illustrate how the more detailed scale "folds" into the coarser one. In addition, we also use these transitions to (c) allow us to change direction such that we can arrange all scale representations in a zig-zag pattern (Fig. 7), to form a single multiscale unfolded representation. We connect one *constant-scale zone* with another one by means of a *transition zone*. Elements from the constant zones extend into the transition zone, in which their size is changed according to the respective change of spatial scale. We also need to show how the elements of the more detailed scale are located within the elements of the coarser scale. We generally place the elements in the transition zone along a curve that connects the coarser with the more detailed scale as shown in Fig. 7. Along this curve, we maintain the connectivity between the elements. We change size and rendering parameters so that we illustrate the containment relationship. Next we describe how we realized these transition zones, again starting from the coarsest representation.

Chromosomes-to-loci transition. In the chromosome scale we render loci positions (50–100 per chromosome), each represented by a sphere. In the next scale we have 100 fiber positions (again each represented by a sphere) for each locus. In each scale, for illustration, we render all spheres that make up one semantic unit using the same color; e. g., all 100 fiber positions are depicted in the same color for each locus as shown in Fig. 1. In the spatial transition between two consecutive scales, we fit each color group of a finer scale into a single sphere of the coarser scale. This also means that each color group of the semantic loci scale represents a single (locus) sphere as rendered in the chromosome scale.

Depending on the location of the DNA sequence of the depicted locus scale, a given number of locus spheres remain until the end of each chromosome. While we render the remainder of the locus spheres of the chromosome at the regular data positions in the chromosome scale, we place a fixed number (we use seven) along the upper arc of the transition zone curve. We evenly space and slowly enlarge them in size to illustrate the changing spatial scale (see upper part of Fig. 8(a)). For the second part of the transition, we place the same number of color groups from the loci scale on the lower arc of the curve, slowly decreasing in size (lower part of

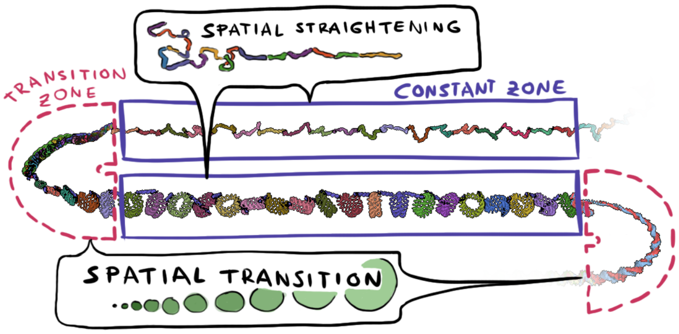


Fig. 7. Conceptual arrangement of constant and transition zones.

Fig. 8(a)). At the middle point of the transition curve, we render both a single locus sphere as well as the locus color group that represents the same data—so that their sizes match. To illustrate the "folding of one scale into the next," we then continue to render the loci color groups on top of the loci spheres for the upper arc of the transition zone such that the sizes match (i. e., the color groups are shrinking in size towards the top). To ensure that the elements of the more detailed loci scale remain connected, we use the same process as described in Sect. 4.1 for the spatial straightening of the loci scale—just using the transition curve as the straightening path.

Loci-to-fibers transition. In contrast to the first spatial transition, to change from loci to fibers we now need to connect two linearized sequences. We thus dedicate the loci color group that follows in the DNA sequence after the last linearized loci color group as the loci scale's transition group. This group represents a single locus and consists of 100 fiber positions. Each fiber position on the fiber scale, in turn, consists of 100 nucleosome positions which again form a color group on the fiber scale. Like before, we arrange the transition along a curve and place the loci scale's transition group on the upper arc of the curve, and five fiber color groups that from the DNA sequence before the depicted constant-zone loci groups. We also again use the basic approach from spatial straightening and adjust their scale such that the elements match when they meet. In the upper part of the transition, we continue to render the increasingly smaller fiber color groups until the start of the loci scale's transition group as shown in Fig. 8(b). As also can be seen in the figure, the end of this sequence is often hidden due to the spatial arrangement of the loci transition group. As we show in Fig. 8(b), we further adjust the rendering parameters like the outline transition so that the outline does not become over-emphasized for the loci elements as they are getting smaller towards the top.

Due to interaction, we often have the case that the DNA sequence is arranged such that the linearized part of the loci scale does not end exactly at the end of one of its color groups. In this case we use the remaining part of this last color group and proportionally add another section of the following loci color group to jointly serve as the loci scale's transition group as we show in Fig. 8(b) with the purple and beige color sections at the end of the loci scale. We thus create smooth animations in which the positions of the straightened scales can be continuously adjusted.

Fibers-to-nucleosomes transition. The next transition from the fiber scale to the nucleosome scale is similar to the previous one, with the difference that the nucleosome "color groups" are always full nucleosomes, each consisting of 146 base pairs. We follow the approach as before, only ensuring that we also appropriately render the linker sections between two consecutive nucleosomes in the lower part of the transition arcs. At the upper arcs the nucleosomes are placed so close to each other—one nucleosome per sphere of the fiber scale's transition group, which consists of 100 spheres—that we omit the linkers here as they would not be visible anyway. We show the result of this transition in Fig. 8(c).

Nucleosome-to-double-helix transition. The nucleosome's inherent structure prevents us from using the previous transition styles for switching from nucleosomes to the double-helix. Instead, we take inspiration from the conceptual unrolling of a nucleosome. We fix the end of the transitional nucleosome at the end of the constant-scale nucleosome zone, taking into account its spatial orientation. Depending on the needed amount of unrolling, we straighten this part and place it on the transition curve. We adjust the curve such that it always connects to the current point on the transitional nucleosome where the unfolding happens. We also use colors to indicate the two strands of the double-helix, as we show it in the snapshot of the transition in Fig. 8(d).

Double-helix-to-bases transition. The final transition between double-helix and bases no longer represents a change of spatial scale. Instead, with it we address the issue that in the double-helix we cannot easily see all nucleotides in a single view. Due to the helical twist, they are partially occluded by other nucleotides. We use a strand of the double helix with a complete rotation, extracted from our spatially straightend *1AOI* PDB model. In this model we identify the locations of the nucleotides (bases), find the centers between each pair to be the base pair locations, and derive the respective twist (rotation) at each base pair position. For the transition, we remove this rotation based on the position of the structure on the transition zone curve to completely flatten the helix at the end. We show the result of this transition in Fig. 8(e).

4.3 Illustrating scale differences

Our approach allows us to place the multiscale unfolding representation along an arbitrary curve. For most of our experiments, we decided to use a zig-zag layout with six rows. Fig. 7 illustrates this concept schematically and we show the result in Fig. 1. We depict one transitioning chromosome and four complete ones at the top, followed by 20 loci (i. e., 20%–40% of a chromosome), 20 fiber locations (20% of a locus), and 20 nucleosomes (20% of a fiber). Finally, we show two full nucleosomes (292 base pairs) and 39 base pairs for the flattened helix.

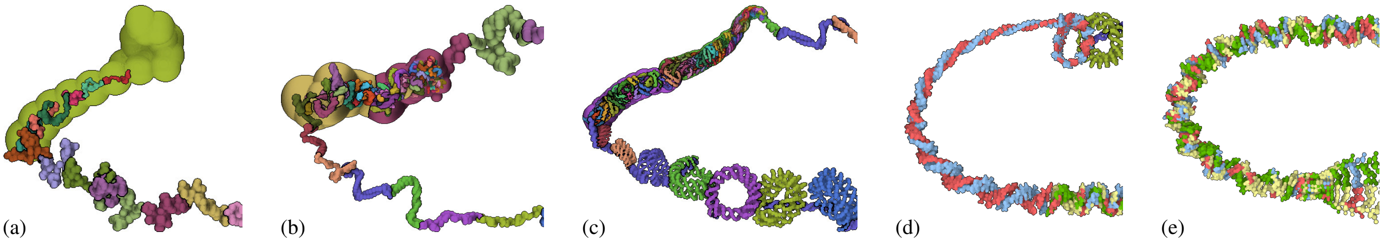


Fig. 8. Transitions between consecutive scales: (a) from chromosomes to loci, (b) from loci to fibers, (c) from fibers to nucleosomes, (d) from nucleosomes to double-helix, and (e) from double-helix to untwisted double-bases.

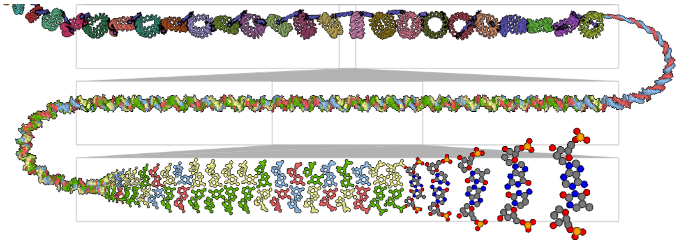


Fig. 9. Illustration of the spatial-scale ruler. The boxes only indicate the average length difference (based on the number of contained elements), not the actual location or correspondence between sections, because our visualization is sequentially continuous.

In addition, as none of the depicted elements have natural colors due to their small size, we assign random colors to each of the chromosomes from a pool. Next, for all conceptual scales from the loci to the nucleosomes, we assign random colors from the same pool such that each color group shows a single item of the conceptual scale. A single element of the loci scale, e. g., comprises 100 fiber positions, all of which belong to the same color group.

These color groups and our chosen number of elements already illustrate the scale differences with respect to the number of elements that are included into a single element of the next-coarser scale. For an even better illustration of the involved spatial sizes, we provide the option to add a spatial-scale ruler beneath the scale renderings as shown in Fig. 9. This ruler consists of boxes and gray trapezoids that schematically illustrate how much room the fully straightened, constant-scale section of a detailed scale takes in the next-coarser scale—based on the number of contained elements. The bottom-most base pair scale in Fig. 9, e. g., contains the equivalent of four full turns of the double-helix, while the double-helix scale contains 29.2 full turns. These 292 bp in the double-helix scale are the equivalent of two nucleosomes (with 10 bp in a full double-helix turn), which takes 1/10 of the nucleosome scale. For practical applications, all these settings can be adjusted as needed.

4.4 Implementation details and rendering performance

We built our implementation using the Marion framework's [50] molecular visualization functionality, augmented with functions to load Nowotny et al.'s [53] GSS genome data. The molecular rendering uses Le Muzic et al.'s [39] GPU-based dynamic atom injection. We inject the respective scales' basic colored rendering shapes as spheres. We use 2D sphere impostors instead of mesh models for the spheres [39]. As the sizes and positions of the depicted elements change in the transition zones, we adjust this data and reload it into the GPU in each frame. On an Intel Core™ PC (i7-7920HQ, 6 cores, 32 GB RAM, 3.10 GHz, nVidia Quadro M2200, on Windows 10 x64) we achieve approx. 6 fps when showing the visualization at a size of 1920 × 1080 pixels.

Typically, we load the chromosome data only at the highest scale (i. e., loci positions). Only for the currently unfolded chromosome, we also load the data for the remaining scales. It is possible to load all the data at once, but that would take approx. 3 minutes, instead of the mentioned 20 seconds for the details of a single chromosome. The long overall loading time results from the large amount of DNA data with 23,958,240 nucleosome positions for all chromosomes, plus the respective locations at the coarser scales.

5 Interaction for Multiscale Unfolding

Traditional multiscale visualization techniques are often designed to reduce the need for interaction by showing details for all involved scale levels in a single view. The essential benefit of Multiscale Unfolding is that we no longer need to interactively explore different scales and then mentally connect them. The logarithmic representation of the observable universe shown in Fig. 2(d), for instance, is a multiscale depiction of the massive size and distance differences, without the need to interact with a single part of it. This property also applies to our Multiscale Unfolding of DNA data. It works due to the repetitive nature of DNA, despite the huge amount of data in most of its scales. While our approach can produce such still-image illustrations of the genome, below we discuss suitable multiscale navigation techniques that also allow viewers to interactively explore our representation. We treat the first scale with its discrete chromosomes differently from the remaining scales which are spatially continuous through the scale levels.

5.1 Chromosome selection and panning

Most useful for the navigation of our one-dimensional data arrangement is panning along the centerline of our visualization. We enable this for the chromosome level and it works as expected as our visual mapping facilitates a smooth re-positioning. More important, however, for the straightened chromosome scale is to be able to select a new chromosome to "drill into," which we facilitate through a click-based selection. The major challenge here is the amount of data that has to be reloaded due to this single interaction. It takes approx. 20 seconds to load the data for all scales of the chromosome onto the GPU. This waiting time of more than 10 seconds leads to losing a user's attention [14], [49]. As the chromosomes are arranged sequentially, a panning of the scale or multiple successive clicks on the end to get to a particular chromosome are thus practically not feasible. We allow users to pan the chromosome scale, but without affecting the first chromosome that is spatially connected to the following scales. As soon as a user has found a chromosome to explore further, he or she can select it and only has to wait for the data to be loaded once.

5.2 Multiscale Zliding for spatially-specific panning

Similar to the chromosome scale, we support panning for all of the following scales. We realize this control with a mouse click-and drag interaction, mapping the horizontal offset to the respective translation of the scale on which the interaction was started. While this mapping is linear for the constant-scale zones, it would be non-linear for the transitions. For the sake of intuitiveness, we thus restrict the panning to starting points in constant-scale zones.

We adjust the applied translation speed to the respective scales such that the mouse movement is directly mapped to the chosen scale. We want to avoid having to start a new panning interaction each time we change scales. For this, we take inspiration from the HCI literature and combine ideas from Ramos and Balakrishnan's [57] TLslider widget with the same authors' [58] Zliding technique. We adapt their approaches to the context of our Multiscale Unfolding. The authors used the TLslider to navigate details of a video stream. They arranged video snapshots in an S-shaped curve away from the horizontal to make more room for seeing individual images. Their Zliding technique later generalized this idea and added zoom control based on pen pressure to the 2D positional input. In our case, we only need one-dimensional input (in the x-direction) for panning, freeing the second positional input (the y-direction) for the zoom factor, for which we use the spatial offset.

We suggest a novel *Multiscale Zliding* (see Fig. 10) that uses the y-distance from the initial click point to control the panning speed,

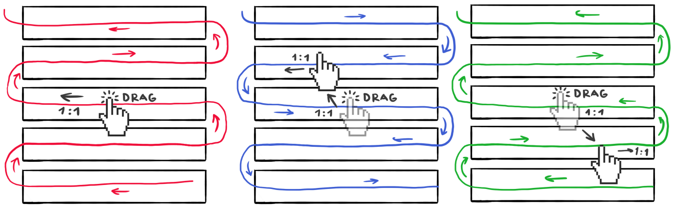


Fig. 10. Scale control with Multiscale Zliding. The black annotations indicate mouse movements. The colored paths and arrows indicate the motion of the elements during panning. We determine the gain factor from the hovered mouse position and appropriately increased and decreased through propagation to adjacent scales.

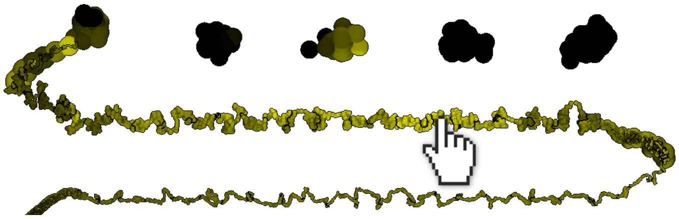


Fig. 11. Showing all elements' 3D-spatial proximity (brighter means closer) to a selected point in the sequence (enlarged pointer in the 2nd row in this example). Notice that we need to change the color mapping in transition zones to adjust it to the spatial scopes/ranges of the scales.

depending on the direction and scale the user hovers while dragging. If we move toward coarser scales, we increase the gain factor [44], while we decrease the gain factor when dragging toward more detailed scales. Regardless of the dragging direction, we adjust the gain factor to match the speed of the hovered scale, if the mouse pointer is within a given distance to its center line, and linearly interpolate in-between these constant-speed zones. An additional challenge is caused by our zig-zag arrangement of scales which results in opposite motion directions for consecutive scales. We flip the direction of the motion as we pass the half-way line between two scales. The ruler as discussed in Sect. 4.3 visually assists this interaction by illustrating the respective interaction zones.

Multiscale Zliding is a one-dimensional multiscale navigation technique that allows users to fluidly control the panning within the long DNA sequence, without having to re-start a click-and-drag interaction in-between. Because of the Multiscale Unfolding's inherent 1D nature, Multiscale Zliding allows us to navigate the data with a standard input device such as a mouse. It could easily be adjusted to a two-dimensional technique by employing, like Ramos and Balakrishnan [57], [58], pressure sensing to be able to use the input's y-component also for spatial control.

5.3 Visual mapping adjustments due to interaction

Multiscale Zliding now allows users to re-position the data in all scales. As a consequence, we had to make two major changes to the previously described visual mappings.The first change affects the transition between the double-helix and its flattened version. The previously described realization of the untwisting works only in a static view. As soon as we interactively move the double helix, fewer or more turns have to be removed depending on how many turns are included in the flattened section (which virtually extends in its flattened representation until the very end of the DNA sequence of at least the current gene). We fix the orientation of the flattened helix at the lower end of the transition zone so that it always meets the flat constant-scale region of our bottom scale. Conversely, as we pan this bottom scale, the rotation of the helix accumulates and we rotate the double-helix scale appropriately during interaction. We propagate this axial rotation further through the transition zone to the nucleosome scale. At the end of this transition, the double helix is so small in size that we no longer see the turning, so we do not have to propagate it any further.

We make a second major change also at the nucleosome unrolling, which we cannot express as a "normal" unrolling (like that of a garden hose from a rotating spool). The reason is that, similar to the double-helix, one end of the "nucleosome spool" has to be fixed to the remaining DNA chain. To emphasize the conceptual unrolling, we thus add a twisting transformation to the construction of the nucleosome spiral. While placed at their correct positions when the nucleosome is fully assembled, we space the double bases further apart from each other as the nucleosome is unrolled. This leads to the described animated twisting of the nucleosome which better illustrates its unrolling. Ultimately we "hide" two forms of accumulated twists due to the need to unroll linear structures with fixed ends.

5.4 Visually encoding the distance to a clicked item

While we remove the global 3D spatial configuration to reveal the DNA's linear structure, the spatial configuration still is important. The 3D distances between basepairs, in addition to their sequence, also have an effect on the genome's function (gene expression). To compensate for the loss of the spatial structure of the genome, we thus allow viewers to reveal these distances interactively as shown in Fig. 11. By clicking on a given element, we color-map the distances to other elements of the same scale or to elements in other scales. As the scales have vastly different sizes, we adjust the colors to which the distances are mapped to the given scales.

6 Discussion, Feedback, Limitations

Initially, we set out to solve an illustrative visualization problem: how to show all scales of a highly multiscale dataset in one view, without completely separating the scales. As our Multiscale Unfolding results (e. g., Fig. 1) show, we achieved this goal. Furthermore, the new visual representation also leads to several new insights. Below we review our results, report on feedback from illustrators and domain experts, and discuss limitations.

6.1 Visual abstraction and generalization

For the first time we can show the various local spatial arrangements of the different scales of the DNA in its interphase configuration—based on real data[2] and in a single view.

Despite using the same straightening method for both the loci and the fiber scales, for example, their configuration is visually different, which also can be seen in the transitions between the scales. Furthermore, through the interactive navigation we can experience and appreciate the huge spatial scale differences in the DNA structure. If we pan a given scale the next-coarser scale barely moves due to the 1:100 relationship of element containment for some of the scales. Similarly, if we move a coarse scale, the lower scales move so fast that this motion is beyond the animation capabilities even of the fastest computer graphics hardware.

Interestingly, with Multiscale Unfolding we perceive all scales without interaction. Also perspective is not necessary. Instead, our explicit spatial transitions demonstrate how one scale relates to

[2] To the best of our knowledge, current imaging technologies cannot capture the precise spatial molecule positions in the chromosome. We thus rely on computationally reconstructed data (i. e., Asbury et al. [4]), yet we can exchange the reconstructed data for more accurate positions as they become available.

and integrates into the next one, similar to traditional illustrations as shown in Fig. 2(a). Perspective can serve a similar purpose as shown in Fig. 2(b) or, likewise, by "emphasized perspective scale transitions" from the VIS literature such as Parulek et al.'s work [55]—here the scale change due to the perspective is emphasized through illustrative visualization mappings. A remaining question concerns the combination of the two approaches and how we would then control the abstraction in the scale transitions. Moreover, the multiscale aspect of our unfolding can be considered as a half of an extremely long one-dimensional fish-eye lens with stair-wise defined *optics* of detail. Of course, it would be possible to define another fish-eye lens profile—a symmetric one, for instance, that would generate effects closer to a traditional fish-eye metaphor. Moving such a lens over the unfolded genome could then interactively reveal the details at an arbitrary place in a sequence, while the rest of the structure is represented with less detail. Ultimately a user-adjustable scale-factor function could assign a specific scale level to each location along the straightened centerline, which would support an arbitrary arrangement of several focus and context regions simultaneously.

Another important question is also how and to which other domains we can generalize our approach, beyond the domain of illustrative genome visualization that we demonstrated. As depicted in Fig. 2, multiscale phenomena do not only exist in genome data but also, for example, in other scientific fields like anatomy and astronomy. Some of these domains, such as the muscle visualization of Fig. 2(c), probably only benefit to a limited degree from our approach. They do not comprise large scale differences. Several other data domains exist in which the data is linearly organized as 1D paths, with various levels of scale. For example, routes (e. g., path of the *Tour de France*) and GPS-based driving directions could similarly be subjected to Multiscale Unfolding. For the latter, existing abstraction techniques (e. g., [1]) could be employed, as well as the current level of interest for the control of the unfolding. For other domains, like astronomy (Fig. 2(d)), we could adjust Multiscale Unfolding to use specified paths and a selection of the shown physical property for each scale. If applied to two- or three-dimensional data, not only our linear interactions (one-dimensional panning) will have to be converted to two-or three-dimensional panning. Also, we will have to use different straightening strategies. For instance, concepts from image morphing (e. g., [7]) could assist the goal, but potentially also the concepts of axial or volumetric deformations [15], [16], [38], [41] and multiresolution curves [20].

The application to 2D or 3D data might also necessitate a globally linear arrangement as illustrated in Fig. 2(d), contrary to our current zig-zag structure. Moreover, astronomy data is characterized by huge empty spaces between elements and between different scales [25]. We could address this problem and the issue of the three-dimensional character by employing a spatial straightening to 3D astronomy data, similar to our aligning of non-connected chromosomes. Planets could be aligned on a path like pearls on a necklace. Further scales would align stars, galaxies, and so on. A remaining problem of domains like astronomy concerns the uneven density of available data [25]. We know much more about regions close to Earth than we know about the far ends of the observable universe. An interactive exploration of the entire space, as we can facilitate it, is thus difficult in these domains.

6.2 Feedback, application, limitations, extensions

To evaluate our approach with experts in this field, we consulted a number of professional illustrators and biologists. Three illustrators from the field of molecular biology (9, 33, and 45 years of post-PhD experience, respectively) specifically commented about our chosen representations. Their feedback allowed us to correct structures, in particular, at the base and double-helix scales. They also recommended to add color-coding based on nucleobases as we now show in Fig. 1. They criticized that we do not show histones at the nucleosome scale, which could easily be addressed by adding this element to the used nucleosome model. We did not yet do since we concentrated on the aspect of controlling the abstraction spatially. They also noted that the transitions lack perspective and are not informative enough about the scale difference. We do not see this as a problem as we are able to introduce perspective at will. Yet it has to be used with care: sufficient perspective to convey correct scale differences may imply that only some of the scales can be shown—Fig. 2(b) shows only two or three scales between atoms and chromosomes with perspective transitions. With respect to the application of our approach, they saw potential both for communicating the multiscale aspect of the DNA to laypeople or even biologists and for supporting expert tasks as long as additional data is overlayed on top of the existing visualization.

We also conducted a semi-structured interview with a certified illustrator with three years experience and a PhD in Bioengineering. Considering that she had seen ScaleTrotter before, she preferred the clarity and simplicity of Multiscale Unfolding because, without the 3D structure, she could see all the scales in one view. She stated that this fact would allow her to grasp the differences between non-consecutive scales. She was very interested in the interaction and was keen on exploring the tool herself. Overall, she foresees that our tool can produce aesthetic illustrations and animations for teaching a lay audience. She noted that our tool is based on actual scientific data, which does not only saves her research time, but can be also more truthful to nature. Ultimately, to produce illustrations that fit exactly her purposes, our approach would need to provide interactive capabilities to create custom layouts that can deal with different spatial constraints. To create a magazine cover that considers other visual elements such as text and other images as well, for instance, she would prefer to create custom paths in 2D and 3D space and to adjust the size of both constant and transition zones. Our system already allows for alternate layouts. Yet we still have to add interactions to position control points to allow manual control of the layout and transitions. She also noted that, "if vector images could be exported from the visualization," she could sketch over them and add her own artistic style to get a final result. If the goal is not to create a publishable illustration, she also sees this tool as a way for domain experts to create illustrations themselves without the need of an illustrator.

Finally, we conducted a semi-structured interview with an expert in molecular biology with 23 years of post-PhD experience. He specializes in genetics and studies the composition, architecture, and function of SMC complexes. He immediately made it clear that this tool can be used to educate first-year students. For it to be used by domain experts, however, additional data has to be overlaid on top of the existing structures (noting different types of data for each scale). He found it useful to have a multiscale visualization where, e. g., he can look at a gene on one scale, and at the cluster of genes that the gene belongs to on a higher scale, in one view. He imagined that a version of our tool with additional data would be much easier to use than current tools that rely on diagrams, because it offers more context. He also said that "it would be a dream" to have a tool that combines all of that data. Adding such data requires further work to solve respective data mapping and representation issues as well as to create dedicated interaction techniques. We take these ideas as inspiration for future work to investigate if and how the additional genome data at the different scales (e. g., gene locations, epigenomic signals, genomic conservation information, regulatory regions) can be combined with aspects of our current illustrative visualization. To create such an illustrative multiscale genome data browser for domain experts, however, requires more research. In particular, we would need to overcome the limitation that we currently rely on partially reconstructed (as opposed to truly 3D-captured) genome data—which is not a problem for our demonstrated use case of illustrative visualization.

The molecular biology expert also saw complementarity between this tool and ScaleTrotter, both multiscale genome visualization tools with different approaches. He imagined two windows side by side, with each showing one of both tools. Users could navigate in the full 3D structure and then switch to the multiscale straightened view, or use the latter to guide their way in the 3D model. The currently available interactions, however, are limited to a local navigation. A more comprehensive set of interactions would thus be desireable [17], [31], [32], [71], like a global navigation (e. g., "go to *this* location") or data analysis functionality (e. g., "compare *these* two sections").

With respect to technical limitations, while our current rendering speeds allow us to interactively explore the visualization, they are still relatively slow. During interaction we permanently need to update on the GPU the actively shown elements in the constant-scale and transition zones. This requires us to transfer for each frame a lot of data between CPU and GPU. Computing the active sections directly on the GPU could alleviate this problem. For public dissemination it would also be useful to realize the approach in a browser environment. In this case our current GPU rendering pipeline with a particle-based representation is likely unfeasible. Remote rendering could thus off-load the rendering to a powerful server (as with Marion [50]), maintaining the same Multiscale Unfolding visuals and interaction techniques.

7 Conclusion

On a conceptual level, our work on spatially controlled Multiscale Unfolding extends the state of the art of using abstraction in visualization and provides a rich tool set for the depiction and exploration of multiscale data. Beyond common interactively (i. e., temporally) controlled multiscale visualization techniques, this approach allows us to create convincing still-image visualizations of multiscale data. More importantly for the visualization field, our approach does not require viewers to memorize different scale representations for comparison. It *inherently supports one of the most essential visualization tasks* [11], [23]: data comparison—in our case across scales. Essentially our visual mapping is *a type of multiple linked views* [6], [13], [70], with the fundamental difference to their typical use that our Multiscale Unfolding sections are not *only linked in their response to interactive input but also linked continuously in space*. Our combination of multiscale visualization with linked views extends the visualization technique of *contextualization* [26] to the heavily intertwined DNA data and shows how it can be applied through multiple connected spatial scales. We are advancing the central visualization approach of *unfolding and flattening* [37] into the multiscale domain.

Mulitscale Unfolding also raises many new questions and opens up several new paths for research. With the basic framework in place we now want to explore, e. g., further image-space arrangements apart from the zig-zag layout. Spirals, space-filling lines, or arbitrarily defined paths from artists could be investigated. Similarly, alternative layouts that not only zoom into the data but also back out could avoid "cutting off" the data at the bottom.

Our work so far has concentrated primarily on the technical matters of illustrative multiscale visualization and on how to control multiscale abstraction. A next step should incorporate further feedback from professional illustrators and genome scientists to improve data representation, artistic control, usability, etc.

Another important direction for future work is to further explore interaction possibilities with Multiscale Unfolding to support other fundamental visualization tasks. This concerns the comparisons between different DNA sections on the same or different scales, maybe through an adjustable scale-factor function. Such a function might provide the possibility to control the scope of a given scale and to compare DNA parts that are far apart. Also comparing parts from different chromosomes could develop further beyond depicting two Multiscale Unfoldings side-by-side. Also the interaction with currently non-visible data parts can be addressed. Image-space arrangements other than the zig-zag layout pose questions on how to adjust interaction and navigation, and if Multiscale Zliding can still be used. Further adapting the visual mapping to the specifics of scale interactions can be done. An example is to motion blur fast-moving scales.

A final, overarching goal is to combine and integrate spatially-controlled Multiscale Unfolding with interactively- or temporally-controlled, traditional multiscale visualization. As we already showed in Fig. 1, the traditional equivalent of our spatially-controlled approach is ScaleTrotter [25], and it works essentially perpendicular to our new visual mapping. Our vision is thus to be able to *pivot* between any selected point, for a given location on the sequence and a given scale (e. g., selected by a double-click), to the exact same data element shown at the same scale and with the same orientation in ScaleTrotter. The resulting drastic view changes (except for the pivot element) likely require some new animation and visualization techniques to be developed. Yet together, the two techniques would provide viewers with a powerful tool to explore both the spatial and the multiscale aspects of DNA data.

Acknowledgments

We thank all experts who provided their feedback and were available for interviews. Part of this work was funded under the ILLUSTRARE grant by both the Austrian Science Fund (FWF): I 2953-N31 and the French National Research Agency (ANR): ANR-16-CE91-0011-01. The research was further supported by funding from King Abdullah University of Science and Technology (KAUST), under award number BAS/1/1680-01-01 and by funding from the ILLVISATION grant by WWTF (VRG11-010). The authors also thank Nanographics GmbH (nanographics.at) for providing the Marion Software Framework. This paper was partly written in collaboration with the VRVis Competence Center. VRVis is funded by BMVIT, BMWFW, Styria, SFG and Vienna Business Agency in the scope of COMET – Competence Centers for Excellent Technologies (854174), which is managed by FFG.

References

[1] M. Agrawala and C. Stolte, "Rendering effective route maps: Improving usability through generalization," in *Proc. SIGGRAPH*. New York: ACM, 2001, pp. 241–249. doi: 10.1145/383259.383286

[2] B. Alberts, A. D. Johnson, J. Lewis, D. Morgan, M. Raff, K. Roberts, and P. Walter, *Molecular Biology of the Cell*, sixth ed. New York: Garland Science, 2015.

[3] A. T. Annunziato, "DNA packaging: Nucleosomes and chromatin," *Nature Education*, vol. 1, no. 1, p. 26:1, 2008.

[4] T. M. Asbury, M. Mitman, J. Tang, and W. J. Zheng, "Genome3D: A viewer-model framework for integrating and visualizing multi-scale epigenomic information within a three-dimensional genome," *BMC Bioinform.*, vol. 11, no. 1, pp. 444:1–7, 2010. doi: 10.1186/1471-2105-11-444

[5] E. Axelsson, J. Costa, C. Silva, C. Emmart, A. Bock, and A. Ynnerman, "Dynamic scene graph: Enabling scaling, positioning, and navigation in the universe," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 459–468, 2017. doi: 10.1111/cgf.13202

[6] R. A. Becker and W. S. Cleveland, "Brushing scatterplots," *Technometrics*, vol. 29, no. 2, pp. 127–142, 1987. doi: 10.1080/00401706.1987.10488204

[7] T. Beier and S. Neely, "Feature-based image metamorphosis," *ACM Comput. Graph.*, vol. 26, no. 2, pp. 35–42, 1992. doi: 10.1145/142920.134003

[8] L. M. Biga, S. Dawson, A. Harwell, R. Hopkins, J. Kaufmann, M. LeMaster, P. Matern, K. Morrison-Graham, D. Quick, and J. Runyeon, *Anatomy & Physiology*. USA: OpenStax/Oregon State University, 2019.

[9] K. Bladin, E. Axelsson, E. Broberg, C. Emmart, P. Ljung, A. Bock, and A. Ynnerman, "Globe browsing: Contextualized spatio-temporal planetary surface visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 802–811, 2018. doi: 10.1109/TVCG.2017.2743958

[10] M. A. Borkin, K. Z. Gajos, A. Peters, D. Mitsouras, S. Melchionna, F. J. Rybicki, C. L. Feldman, and H. Pfister, "Evaluation of artery visualizations for heart disease diagnosis," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2479–2488, 2011. doi: 10.1109/tvcg.2011.192

[11] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2376–2385, 2013. doi: 10.1109/TVCG.2013.124

[12] J. Brosz, M. A. Nacenta, R. Pusch, S. Carpendale, and C. Hurter, "Transmogrification: Causal manipulation of visualizations," in *Proc. UIST*. New York: ACM, 2013, pp. 97–106. doi: 10.1145/2501988.2502046

[13] A. Buja, J. A. McDonald, J. Michalak, and W. Stuetzle, "Interactive data visualization using focusing and linking," in *Proc. VIS*. Los Alamitos: IEEE CS, 1991, pp. 156–163. doi: 10.1109/VISUAL.1991.175794

[14] S. K. Card, G. G. Robertson, and J. D. Mackinlay, "The information visualizer, an information workspace," in *Proc. CHI*. New York: ACM, 1991, pp. 181–186. doi: 10.1145/108844.108874

[15] S. Coquillart, "Extended free-form deformation: A sculpturing tool for 3D geometric modeling," *ACM Comput. Graph.*, vol. 24, no. 4, pp. 187–196, 1990. doi: 10.1145/97880.97900

[16] S. Coquillart and P. Jancène, "Animated free-form deformation: An interactive animation technique," *ACM Comput. Graph.*, vol. 25, no. 4, pp. 23–26, 1991. doi: 10.1145/127719.122720

[17] E. Dimara and C. Perin, "What is interaction for data visualization?" *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 119–129, 2020. doi: 10.1109/TVCG.2019.2934283

[18] N. Elmqvist, Y. Riche, N. Henry Riche, and J.-D. Fekete, "Mélange: Space folding for visual exploration," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 3, pp. 468–483, 2010. doi: 10.1109/TVCG.2009.86

[19] M. H. Everts, E. Begue, H. Bekker, J. B. T. M. Roerdink, and T. Isenberg, "Exploration of the brain's white matter structure through visual abstraction and multi-scale local fiber tract contraction," *IEEE Trans. Vis. Comput. Graph.*, vol. 21, no. 7, pp. 808–821, 2015. doi: 10.1109/TVCG.2015.2403323

[20] A. Finkelstein and D. H. Salesin, "Multiresolution curves," in *Proc. SIGGRAPH*. New York: ACM, 1994, pp. 261–268. doi: 10.1145/192161.192223

[21] C.-W. Fu and A. J. Hanson, "A transparently scalable visualization architecture for exploring the universe," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 1, pp. 108–121, 2007. doi: 10.1109/TVCG.2007.2

[22] G. W. Furnas and B. B. Bederson, "Space-scale diagrams: Understanding multiscale interfaces," in *Proc. CHI*. New York: ACM, 1995, pp. 234–241. doi: 10.1145/223904.223934

[23] M. Gleicher, "Considerations for visualizing comparison," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 413–423, 2018. doi: 10.1109/TVCG.2017.2744199

[24] M. Glueck, K. Crane, S. Anderson, A. Rutnik, and A. Khan, "Multiscale 3D reference visualization," in *Proc. I3D*. New York: ACM, 2009, pp. 225–232. doi: 10.1145/1507149.1507186

[25] S. Halladjian, H. Miao, D. Kouřil, M. E. Gröller, I. Viola, and T. Isenberg, "ScaleTrotter: Illustrative visual travels across negative scales," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 654–664, 2020. doi: 10.1109/TVCG.2019.2934334

[26] H. Hauser, "Generalizing focus+context visualization," in *Scientific Visualization: The Visual Extraction of Knowledge from Data*, G.-P. Bonneau, T. Ertl, and G. M. Nielson, Eds. Berlin: Springer, 2006, pp. 305–327. doi: 10.1007/3-540-30790-7_18

[27] W.-H. Hsu, K.-L. Ma, and C. Correa, "A rendering framework for multiscale views of 3D models," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 131:1–10, 2011. doi: 10.1145/2070781.2024165

[28] R. N. Irobalieva, J. M. Fogg, D. J. Catanese, T. Sutthibutpong, M. Chen, A. K. Barker, S. J. Ludtke, S. A. Harris, M. F. Schmid, W. Chiu, and L. Zechiedrich, "Structural diversity of supercoiled DNA," *Nat. Commun.*, vol. 6, pp. 8440:1–10, 2015. doi: 10.1038/ncomms9440

[29] W. Javed, S. Ghani, and N. Elmqvist, "Polyzoom: Multiscale and multifocus exploration in 2D visual spaces," in *Proc. CHI*. New York: ACM, 2012, pp. 287–296. doi: 10.1145/2207676.2207716

[30] A. J. Johnson and J. H. Colton, *Johnson's New Illustrated (Steel Plate) Family Atlas, with Descriptions, Geographical, Statistical, and Historical*. New York: Johnson and Ward, 1862.

[31] D. F. Keefe, "Integrating visualization and interaction research to improve scientific workflows," *IEEE Comput. Graph. Appl.*, vol. 30, no. 2, pp. 8–13, 2010. doi: 10.1109/MCG.2010.30

[32] D. F. Keefe and T. Isenberg, "Reimagining the scientific visualization interaction paradigm," *IEEE Computer*, vol. 46, no. 5, pp. 51–57, 2013. doi: 10.1109/MC.2013.178

[33] S. Klashed, P. Hemingsson, C. Emmart, M. Cooper, and A. Ynnerman, "Uniview – Visualizing the universe," in *Eurographics Areas Papers*. Goslar: Eurographics Assoc., 2010, pp. 37–43. doi: 10.2312/ega.20101005

[34] D. Kouřil, T. Isenberg, B. Kozlíková, M. Meyer, M. E. Gröller, and I. Viola, "HyperLabels—Browsing of dense and hierarchical molecular 3D models," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, 2021, to appear. doi: 10.1109/TVCG.2020.2975583

[35] B. Kozlíková, M. Krone, M. Falk, N. Lindow, M. Baaden, D. Baum, I. Viola, J. Parulek, and H.-C. Hege, "Visualization of biomolecular structures: State of the art revisited," *Comput. Graph. Forum*, vol. 36, no. 8, pp. 178–204, 2015. doi: 10.1111/cgf.13072

[36] B. Kozlíková, M. Krone, N. Lindow, M. Falk, M. Baaden, D. Baum, I. Viola, J. Parulek, and H.-C. Hege, "Visualization of biomolecular structures: State of the art," in *EuroVis STARs*. Goslar: Eurographics Assoc., 2015, pp. 61–81. doi: 10.2312/eurovisstar.20151112.061-081

[37] J. Kreiser, M. Meuschke, G. Mistelbauer, B. Preim, and T. Ropinski, "A survey of flattening-based medical visualization techniques," *Comput. Graph. Forum*, vol. 37, no. 3, pp. 597–624, 2018. doi: 10.1111/cgf.13445

[38] F. Lazarus, S. Coquillart, and P. Jancène, "Axial deformations: An intuitive deformation technique," *Comput. Aided Des.*, vol. 26, no. 8, pp. 607–613, 1994. doi: 10.1016/0010-4485(94)90103-1

[39] M. Le Muzic, L. Autin, J. Parulek, and I. Viola, "cellVIEW: A tool for illustrative and multi-scale rendering of large biomolecular datasets," in *Proc. VCBM*. Goslar: Eurographics Assoc., 2015, pp. 61–70. doi: 10.2312/vcbm.20151209

[40] F. Lekschas, M. Behrisch, B. Bach, P. Kerpedjiev, N. Gehlenborg, and H. Pfister, "Pattern-driven navigation in 2D multiscale visualizations with scalable insets," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 611–621, 2019. doi: 10.1109/TVCG.2019.2934555

[41] A. Lerios, C. D. Garfinkle, and M. Levoy, "Feature-based volume metamorphosis," in *Proc. SIGGRAPH*. New York: ACM, 1995, p. 449–456. doi: 10.1145/218380.218502

[42] N. Lindow, D. Baum, and H.-C. Hege, "Interactive rendering of materials and biological structures on atomic and nanoscopic scale," *Comput. Graph. Forum*, vol. 31, no. 3pt4, pp. 1325–1334, 2012. doi: 10.1111/j.1467-8659.2012.03128.x

[43] W. Lueks, I. Viola, M. van der Zwan, H. Bekker, and T. Isenberg, "Spatially continuous change of abstraction in molecular visualization," in *Abstracts of BioVis*. Los Alamitos: IEEE CS, 2011.

[44] S. I. MacKenzie and S. Riddersma, "Effects of output display and control-display gain on human performance in interactive systems," *Behav. Inf. Technol.*, vol. 13, no. 5, pp. 328–337, 1994. doi: 10.1080/01449299408914613

[45] J. McCrae, I. Mordatch, M. Glueck, and A. Khan, "Multiscale 3D navigation," in *Proc. I3D*. New York: ACM, 2009, pp. 7–14. doi: 10.1145/1507149.1507151

[46] H. Miao, E. De Llano, T. Isenberg, M. E. Gröller, I. Barišić, and I. Viola, "DimSUM: Dimension and scale unifying maps for visual abstraction of DNA origami structures," *Comput. Graph. Forum*, vol. 37, no. 3, pp. 403–413, 2018. doi: 10.1111/cgf.13429

[47] H. Miao, E. De Llano, J. Sorger, Y. Ahmadi, T. Kekic, T. Isenberg, M. E. Gröller, I. Barišić, and I. Viola, "Multiscale visualization and scale-adaptive modification of DNA nanostructures," *IEEE Trans. Vis. Comput.*

Graph., vol. 24, no. 1, pp. 1014–1024, 2018. doi: 10.1109/TVCG.2017.2743981

[48] H. Miao, T. Klein, D. Kouřil, P. Mindek, K. Schatz, M. E. Gröller, B. Kozlíková, T. Isenberg, and I. Viola, “Multiscale molecular visualization,” *J. Mol. Biol.*, vol. 431, no. 6, pp. 1049–1070, 2019. doi: 10.1016/j.jmb.2018.09.004

[49] R. B. Miller, “Response time in man-computer conversational transactions,” in *Proc. AFIPS*. New York: ACM, 1968, pp. 267–277. doi: 10.1145/1476589.1476628

[50] P. Mindek, D. Kouřil, J. Sorger, D. Toloudis, B. Lyons, G. Johnson, M. E. Gröller, and I. Viola, “Visualization multi-pipeline for communicating biology,” *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 883–892, 2018. doi: 10.1109/TVCG.2017.2744518

[51] G. Mistelbauer, A. Varchola, H. Bouzari, J. Starinsky, A. Köchl, R. Schernthaner, D. Fleischmann, M. E. Gröller, and M. Šrámek, “Centerline reformations of complex vascular structures,” in *Proc. PacificVis*. Los Alamitos: IEEE CS, 2012, pp. 233–240. doi: 10.1109/PacificVis.2012.6183596

[52] H. Mohammed, A. K. Al-Awami, J. Beyer, C. Cali, P. Magistretti, H. Pfister, and M. Hadwiger, “Abstractocyte: A visual tool for exploring nanoscale astroglial cells,” *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 853–861, 2018. doi: 10.1109/TVCG.2017.2744278

[53] J. Nowotny, A. Wells, L. Xu, R. Cao, T. Trieu, C. He, J. Cheng, and O. Oluwadare, “GMOL: An interactive tool for 3D genome structure visualization,” *Sci. Rep.*, vol. 6, pp. 20 802:1–8, 2016. doi: 10.1038/srep20802

[54] S. Nusrat, T. Harbig, and N. Gehlenborg, “Tasks, techniques, and tools for genomic data visualization,” *Comput. Graph. Forum*, vol. 38, no. 3, pp. 781–805, 2019. doi: 10.1111/cgf.13727

[55] J. Parulek, D. Jönsson, T. Ropinski, S. Bruckner, A. Ynnerman, and I. Viola, “Continuous levels-of-detail and visual abstraction for seamless molecular visualization,” *Comput. Graph. Forum*, vol. 33, no. 6, pp. 276–287, 2014. doi: 10.1111/cgf.12349

[56] E. Pennisi, “The human genome,” *Science*, vol. 291, no. 5507, pp. 1177–1180, 2001. doi: 10.1126/science.291.5507.1177

[57] G. Ramos and R. Balakrishnan, “Fluid interaction techniques for the control and annotation of digital video,” in *Proc. UIST*. New York: ACM, 2003, pp. 105–114. doi: 10.1145/964696.964708

[58] G. Ramos and R. Balakrishnan, “Zliding: Fluid zooming and sliding for high precision parameter manipulation,” in *Proc. UIST*. New York: ACM, 2005, pp. 143–152. doi: 10.1145/1095034.1095059

[59] K. Schatz, C. Müller, M. Krone, J. Schneider, G. Reina, and T. Ertl, “Interactive visual exploration of a trillion particles,” in *Proc. LDAV*. Los Alamitos: IEEE CS, 2016, pp. 56–64. doi: 10.1109/LDAV.2016.7874310

[60] V. A. Schneider, T. Graves-Lindsay, K. Howe, N. Bouk, H.-C. Chen, P. A. Kitts, T. D. Murphy, K. D. Pruitt, F. Thibaud-Nissen, D. Albracht, R. S. Fulton, M. Kremitzki, V. Magrini, C. Markovic, S. McGrath, K. M. Steinberg, K. Auger, W. Chow, J. Collins, G. Harden, T. Hubbard, S. Pelan, J. T. Simpson, G. Threadgold, J. Torrance, J. M. Wood, L. Clarke, S. Koren, M. Boitano, P. Peluso, H. Li, C.-S. Chin, A. M. Phillippy, R. Durbin, R. K. Wilson, P. Flicek, E. E. Eichler, and D. M. Church, “Evaluation of GRCh38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly,” *Genome Res.*, vol. 27, no. 5, pp. 849–864, 2017. doi: 10.1101/gr.213611.116

[61] H. J. Szerlong and J. C. Hansen, “Nucleosome distribution and linker DNA: connecting nuclear function to dynamic chromatin structure,” *Biochem. Cell Biol.*, vol. 89, no. 1, pp. 24–34, 2011. doi: 10.1139/O10-139

[62] M. Termeer, J. O. Bescós, M. Breeuwer, A. Vilanova, F. Gerritsen, and M. E. Gröller, “CoViCAD: Comprehensive visualization of coronary artery disease,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, 2007. doi: 10.1109/TVCG.2007.70550

[63] S. Tonna, A. El-Osta, M. E. Cooper, and C. Tikellis, “Metabolic memory and diabetic nephropathy: Potential role for epigenetic mechanisms,” *Nat. Rev. Nephrol.*, vol. 6, no. 6, pp. 332–341, 2010. doi: 10.1038/nrneph.2010.55

[64] E. R. Tufte, *Envisioning Information*. Cheshire: Graphics Press, 1990.

[65] M. van der Zwan, W. Lueks, H. Bekker, and T. Isenberg, “Illustrative molecular visualization with continuous abstraction,” *Comput. Graph. Forum*, vol. 30, no. 3, pp. 683–690, 2011. doi: 10.1111/j.1467-8659.2011.01917.x

[66] I. Viola, M. Chen, and T. Isenberg, “Visual abstraction,” in *Foundations of Data Visualization*, M. Chen, H. Hauser, P. Rheingans, and G. Scheuermann, Eds. Berlin: Springer, 2020, ch. 2, pp. 15–37. doi: 10.1007/978-3-030-34444-3_2

[67] I. Viola and T. Isenberg, “Pondering the concept of abstraction in (illustrative) visualization,” *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 9, pp. 2573–2588, 2018. doi: 10.1109/TVCG.2017.2747545

[68] J. Watson and F. Crick, “Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid,” *Nature*, vol. 171, no. 4356, pp. 737–738, 1953. doi: 10.1038/171737a0

[69] Wikipedia, “Linker DNA,” Web site: https://en.wikipedia.org/wiki/Linker_DNA, 2021, visited January 2021.

[70] G. Wills, “Linked data views,” in *Handbook of Data Visualization*, C.-h. Chen, W. Härdle, and A. Unwin, Eds. Berlin: Springer, 2008, ch. II.9, pp. 217–241. doi: 10.1007/978-3-540-33037-0_10

[71] J. S. Yi, Y. ah Kang, J. Stasko, and J. A. Jacko, “Toward a deeper understanding of the role of interaction in information visualization,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1224–1231, 2007. doi: 10.1109/TVCG.2007.70515

[72] X. Zhang, “Space-scale animation: Enhancing cross-scale understanding of multiscale structures in multiple views,” in *Proc. CMV*. Los Alamitos: IEEE CS, 2005, pp. 109–120. doi: 10.1109/CMV.2005.16

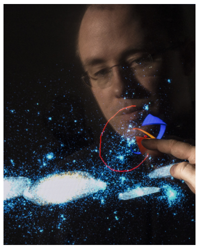
Sarkis Halladjian has recently defended his PhD thesis on “Spatially Integrated Abstraction of Genetic Molecules” at Université Paris-Saclay, France, as a member of the Aviz research team of Inria, France. His research topic is visual abstraction in the context of multi-scale molecular visualization.

David Kouřil is a doctoral student at TU Wien, Austria. He received his Master’s degree (Mgr.) in 2017 from Masaryk University in Brno, Czech Republic. His research topic lies in scientific visualization, where he focuses on three-dimensional data coming from structural biology and designs novel visualization and interaction methods that support exploration and understanding of the environments that this data represents.

Haichao Miao is a postdoc at TU Wien, Austria. His research interests include interactive multiscale visualization and modeling in DNA nanotechnology. He received his PhD in 2019 from TU Wien, Austria, where he focused on the abstract visualization of complex biological and medical data.

M. Eduard Gröller is professor at TU Wien, Austria, and adjunct professor of computer science at the University of Bergen, Norway. His research interests include computer graphics, visualization, and visual computing. He became a fellow of the Eurographics Association in 2009. He is the recipient of the Eurographics 2015 Outstanding Technical Contributions Award and of the IEEE VGTC 2019 Technical Achievement Award.

Ivan Viola is associate professor at King Abdullah University of Science and Technology (KAUST), Saudi Arabia. He graduated from TU Wien, Austria, in 2005 and moved for a postdoc position to the University of Bergen, Norway, where he was gradually promoted to the professor rank. In 2013 he received a WWTF grant to establish a research group at TU Wien. Viola co-founded the startup Nanographics to commercialize nanovisualization technologies.



Tobias Isenberg is a senior research scientist at Inria, France. Previously he held positions as post-doctoral fellow at the University of Calgary, Canada, and as assistant professor at the University of Groningen, the Netherlands. His research interests include scientific visualization, illustrative and non-photorealistic rendering, and interactive visualization techniques. He is particularly interested in the benefit, use, and control of abstraction for illustrative visualization.