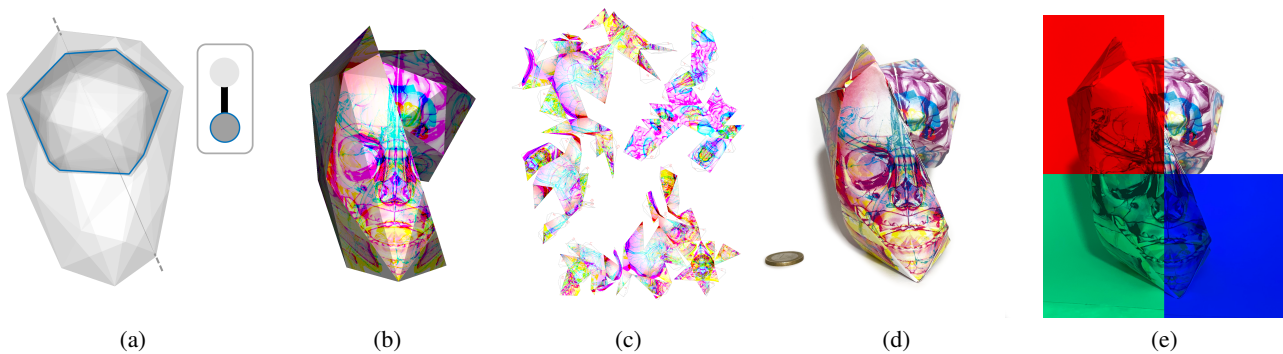


# Nested Papercrafts for Anatomical and Biological Edutainment

M. Schindler<sup>† 1</sup>, T. Korpitsch<sup>† 1</sup> , R. G. Raidou<sup>1</sup> , and H.-Y. Wu<sup>1,2</sup> <sup>1</sup>TU Wien, Austria <sup>2</sup>St. Pölten University of Applied Sciences, Austria

**Figure 1:** An example of a nested papercraft for the exploration of the structures of a human head dataset, with two levels. The outer level represents the outer, visible part of the head and the inner level represents the brain. We show (a) the calculation of the optimal cut for the nested configuration, (b) the texture rendering for the papermeshes of the two levels, (c) the unfolding of the textures onto single 2D patches, (d) the assembly of the nested papercraft, and (e) the use of colored filters to reveal distinct features on the textured papermeshes.

## Abstract

In this paper, we present a new workflow for the computer-aided generation of physicalizations, addressing nested configurations in anatomical and biological structures. Physicalizations are an important component of anatomical and biological education and edutainment. However, existing approaches have mainly revolved around creating data sculptures through digital fabrication. Only a few recent works proposed computer-aided pipelines for generating sculptures, such as papercrafts, with affordable and readily available materials. Papercraft generation remains a challenging topic by itself. Yet, anatomical and biological applications pose additional challenges, such as reconstruction complexity and insufficiency to account for multiple, nested structures—often present in anatomical and biological structures. Our workflow comprises the following steps: (i) define the nested configuration of the model and detect its levels, (ii) calculate the viewpoint that provides optimal, unobstructed views on inner levels, (iii) perform cuts on the outer levels to reveal the inner ones based on the viewpoint selection, (iv) estimate the stability of the cut papercraft to ensure a reliable outcome, (v) generate textures at each level, as a smart visibility mechanism that provides additional information on the inner structures, and (vi) unfold each textured mesh guaranteeing reconstruction. Our novel approach exploits the interactivity of nested papercraft models for edutainment purposes.

## CCS Concepts

• **Applied computing** → **Life and medical sciences**; • **Human-centered computing** → **Visualization techniques**; **Scientific visualization**; **Interaction paradigms**;

## 1. Introduction

Visualization is recognized as an essential component of learning and education [Gil05]. In the last few years, the impact of visualization in *edutainment* (i.e., educational entertainment) has

also been investigated—for example within the context of augmented [SWO\*14] and virtual reality [CLF\*06], or in comparison to serious games [Cha10]. *Physicalization* is a subdomain of visualization, which revolves around creating physical objects to represent and explore data [DSMA\*21]. The field has recently seen a resurgence, also for education and edutainment purposes [Per21]. Generated physical objects are tangible and employ more senses than plain vision, to enhance our cognition and perception about

<sup>†</sup> Equal contribution

displayed information [ZM08, JDI\*15]. Physicalizations provide a high degree of engagement, supporting understanding and memorability [JDF13, SSB15], while also making the entire process interactive and entertaining.

In education, physicalizations may be used either for *demonstration and presentation* purposes with pre-made data models or for *engaging students* in the learning processes by letting them handcraft their representations. In edutainment, physicalizations would rather focus on the latter. So far, the generation of physicalizations has mainly focused on *digital fabrication methods* (e.g., 3D printing, laser cutting, moulding, etc.) [BCMP18], *augmentations* (e.g., projections on passive props and augmented perspectives), and *active approaches* (e.g., data sculptures that use materials to communicate data, robots, and screens) [DSMA\*21].

In biology and anatomy education, physical models have always been used [MMŽ10], as opposed to on-screen educational applications that emerged only in the last few decades [BCK\*11, HDK17, PS18]. Nowadays, biological and anatomical physicalizations are mainly restricted to model kits or contextual 3D printed models. Although consumer-oriented 3D printers are becoming more and more popular, they can use a limited amount of colors and their maintenance remains costly and technically demanded for private usage [RMVTK\*10]. Recently, a series of works proposed computer-aided pipelines for the easy generation of data sculptures, using affordable and readily available materials [SB17, RGW20, SWR20, PWR21]. The core motivation around such approaches relates to the notions of *constructivism* [HRL10] and *embodiment* [JVJB17]. The former supports that active learning facilitates knowledge construction with less cognitive load, the latter claims that learning can benefit from the involvement of motorics and physical interaction.

Among all possible types of data sculptures, *papercrafts* are popular both as an art or recreational form [PDRK18] and as a tool serving educational purposes [EE98]. Papercrafts are 2D or 3D objects from paper or cardboard that have been created by a 3D mesh unfolding into 2D patches, which can be printed and reconstructed back to 3D. Papercraft generation has proven to be a difficult task, given the often occurring overlaps and distortions [HE12], as well as the complexity of the reconstruction process [TWS\*11]. A discussion with a professional papercraft designer [Cha] revealed to us that even “simple” models (i.e., with less than 100 polygon faces) require approximately a week for their creation and around ten days for the papercraft generation.

Creating a papercraft with nested structures would add further to this complexity. *Nested papercrafts* would, for example, be needed for the investigation of a biological or anatomical model, which often include multiple, nested structures. For example, a plant cell includes a nucleolus, which is nested within the nucleus. Subsequently, the nucleus is nested within the cytoplasm together with other structures, such as mitochondria, ribosomes, chloroplasts and the vacuole. Representing all these structures within one papercraft and ensuring their visibility, while maintaining the necessary level of interactivity and engagement without complicating the unfolding or the reconstruction, poses significant fabrication challenges.

In this work, we are investigating a new workflow for the computer-aided generation of nested papercraft physicalizations

that represent models from anatomy and biology. Figure 1 shows an example of our nested papercrafts. Our proposed workflow targets specifically the representation of nested substructures within a 3D model and ensures their visibility. In our approach, we employ hierarchical structuring to detect nested levels within a model, viewpoint calculation to ensure that outer levels can be cut for an unobstructed view on the inner levels, and a stability calculation at all levels (Figure 1 (a–b)). The final step is to unfold the nested levels of the model into 2D patches (Figure 1 (c)), which can be reconstructed and examined (Figure 1 (d)). As an additional visibility mechanism, we propose a smart strategy that takes advantage of the optical properties of our physical world by unveiling different channels of information through the use of different color channels (Figure 1 (e)), inspired by our previous work [SWR20]. Our approach supports 3D models of varying nesting complexity, by exploiting the interactivity of nested papercraft models, and the projection of additional structures on their surface.

The overall *contribution* of this work is the design and realization of a workflow for the computer-aided generation of nested papercrafts to represent complex models from anatomy and biology. Our *goal* is to support the generation of nested papercrafts in an easy (to manufacture and reconstruct), accessible (in resources and technologies) and affordable way. Our papercrafts can be used for tangible, interactive edutainment, where the user is engaged in the creation and assembly of the model’s papercraft. The main *components* of our approach include:

- The *automated detection and representation of nested configurations* in anatomical and biological models, as well as corresponding papermesh synthesis.
- A dual (i.e., topology- and texture-based) approach to guarantee *optimal visibility on the inner levels*.
- A strategy to ensure that the generated nested papercrafts are *realizable* (i.e., assemblable and stable) and *engaging*.

## 2. Related Work

**Anatomical and Biological Education and Edutainment:** Preim and Saalfeld [PS18] presented a survey on virtual anatomy education systems, where they review a big corpus of techniques targeting the education of medical students. These span from surface and volume visualization [PHP\*01, LSHL16, VGHN08] to animations [BPC\*14], and from anatomical labeling [BG05] to virtual and augmented reality [SSPOJ16, PPS19, JPC\*16, MWDE\*16]. Popular examples include VOXEL-MAN [PHP\*01] and open anatomy browser [HDK17]. Applications for the general public include the ZygoteBody [BCK\*11]. For biological data, learning biomolecular structures is achieved through gaming in VR [CLF\*06, CLZL06]. Other previous work focuses on teaching through hands-on graphics experiences that facilitate the exploration of structural aspects of macromolecular systems [Can01]. Animation is also a fundamental concept that is often assessed to determine its suitability for educational purposes in biology [MM21], as well as storytelling approaches [GDVZ21]. All the aforementioned approaches are on-screen solutions, which have proven to significantly contribute to anatomical and biological education, while also serving edutainment approaches either through gaming or storytelling. In our work, instead, we investigate how “hands-on” approaches involving physical models are applied to such scenarios.

**Anatomical and Biological Physicalizations:** Medical and biological data physicalizations are mainly 3D printed physical objects [KSV15, RMVTK\*10, ASS\*19]. Recently, Ang et al. [ASS\*19] developed a cardiac blood flow physicalization based on 3D printing that allows the user to explore 4D MRI data in a slice-based manner. On the biology side, Gillet et al. [GWW\*04] combine AR and 3D printing in an application for structural biology education. Approaches that do not involve 3D printing are limited. Historical approaches include the use of wax, wood, ivory, cardboard, and fabric models [MMŽ10]. Computer-assisted approaches involve the use of volvelles, i.e., interactive wheel charts of concentric, rotating disks that support the physical fine-tuning of transfer functions [SB17], or sliceform papercrafts that support the representation of volumetric or mesh data using an octree-based partitioning [RGW20]. Both approaches target the visual and physical representation of the entire volume of a structure. However, the former focuses more on transfer function definition. The latter provides a holistic slice-based view of the data, but for higher resolution, more slices are needed, which increases the complexity of the approach. Interaction is also not possible without compromising the stability of the sliceforms. Pahr et al. [PWR21] proposed an interactive slice-based alternative for the physicalization of medical data that resembles holograms of volumetric medical data. In our work, we also focus on largely available materials (i.e., paper), as this design choice can support the generation of physical models at home without the need for sophisticated solutions. Still, we focus on providing an illustrative physical representation of the data.

**Papercraft Generation:** The computer-assisted generation of papercrafts has been tackled before, for example for creating paper pop-ups from 3D meshes [RJLYL14, XZM\*18] or origami architecture papercrafts [LLN\*13], iris papercrafts [IIM16], cardboard-based papercrafts [ZGPR16], and animated pop-ups that show motions of articulated characters [RLL15]. This has also been investigated for large-scale models [SCS16, CS17] and entire 3D scenes through developable surfaces [PDRK18]. Mesh unfolding has different applications, such as papercraft models [TWS\*11, SP11] and models from self-folding materials [FTS\*13, ATG\*18]. Some techniques employ mesh deformation [CY17, MS04] to relax the unfolding problem. The simultaneous handling of unfolding and glue tabs has been explored in our previous work [KTGW20]. An optimization model is formulated to search for an unfolding solution with optimal number of glue tabs needed for papercraft reconstruction. However, to the best of our knowledge, none of the aforementioned approaches research generating physical models that contain nested structures or the stability of such composite papercrafts. This problem is not trivial, as it involves the ability to unfold individual meshes and the composition of the submeshes within a model. In our work, we investigate thoroughly this aspect, applying it in the context of anatomical and biological edutainment.

### 3. Requirements and Conceptual Choices

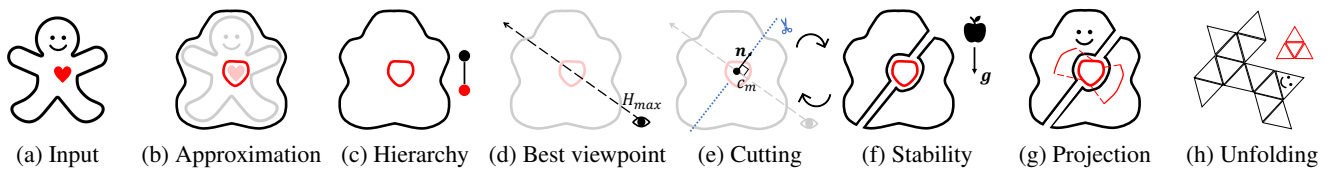
The basic *concept* of our approach is to provide a workflow for the cost-effective and easy generation of physical models that support nested structures, which are often present in biological and anatomical models. Building and exploring physical models aims to make the experience more engaging and, thus, more memorable or understandable [ZM08]. Although papercrafts are not a new ap-

proach for education or edutainment [EE98], the use of papercrafts to represent nested structures is novel and challenging. Our main *requirements* are:

- R1.** The user does not need to have any knowledge of the domain of application (i.e., anatomy or biology).
- R2.** The approach requires a 3D mesh model as input with multiple registered nested structures (submeshes). Examples could be a plant cell or an anatomical model of a human head.
- R3.** The user does not need to perform any complex interaction with our proposed pipeline (i.e., beyond few clicks) or to provide any other special input to it.
- R4.** The output of the workflow needs to be an easy-to-assemble and engaging physical twin of the virtual model.
- R5.** The assembly of the physical model needs to be intuitive, and time- and cost-efficient.
- R6.** The employed resources, i.e., materials and technologies, need to be easily affordable and accessible.

Based on these requirements, our *conceptual choices* are:

- C1. Supported data and structures:** Our data are meshes representing anatomical and biological structures, which contain nested substructures, as dictated by **R2**. These substructures are not ordered, i.e., there is no prior information about their topological hierarchy. Also, we do not restrict the number of nested levels and we do not pose any limits to the number of children at each nested level.
- C2. Nested papercrafts:** Our physicalizations are papercrafts with multiple nested levels, according to **R2**. The choice is made given the popularity of papercrafts, as well as the affordability and availability of materials (see **R6**). We abstract the input meshes, so the papercraft can be assembled in a reasonable time. Given that the substructures are not ordered, we also detect the nested configuration and employ viewpoint selection to cut the outer levels and reveal the inner ones. This approach poses several challenges concerning the unfolding and reconstruction of the physicalization. First of all, this step should not require the user to have prior knowledge of the domain (see **R1**), nor to heavily interact with the framework to create the papercraft (see **R3**). Furthermore, to ensure that our papercraft is always feasible, i.e., can be unfolded and stable in reconstruction (see **R4**), we abstract the structures of distinct levels to close-to-convex polygonal meshes.
- C3. Texture projections:** Conceptual choice **C2** does not account for three special cases: **(a)** the mesh abstraction might introduce distortions, if the shape of the structure is inherently concave; **(b)** the size of (inner) substructures might be prohibiting for printing and reconstruction, if these are too small; and **(c)** structures with genus higher than 0 or with complex topology, e.g., the surface of a Golgi apparatus might be problematic. For these cases, we choose to use a smart visibility mechanism that projects textures from inner structures as additional information on the surfaces of the papercraft (see **R2**). This allows us to show correctly the shape of the inner structures on the approximated surfaces, even for small or intricate objects, while it can serve as “X-ray vision” to reveal the underlying levels. Our projection is based on our previous work [SWR20] and reveals up to three different inner structures with the use of colored filters.



**Figure 2:** Conceptual diagram depicting our workflow. We start by (a) taking as input a set of 3D meshes and (b) simplifying each mesh to create a coarse approximation. (c) We create a hierarchical structure that represents the nested configuration and (d) we perform an entropy-based viewpoint calculation to obtain the optimal view on each inner level, which is used to (e) cut the model open. (f) Stability calculations ensure that the constructed model does not fall apart and stands, and (g) our projection method renders textures on the surface of each level. Finally, (h) we unfold each 3D model into a single 2D patch.

**C4. Mesh unfolding into single connected patch:** Unfolding a 3D mesh into a single patch facilitates the reconstruction, as opposed to using multiple patches, but dealing with overlaps and distortions is a difficult task [TWS\*11]. In our previous work [KTGW20], we proposed a method for unfolding a 3D mesh into a single connected 2D patch by searching the best unfolding variant, which is represented by a minimum spanning tree (MST). A simulated annealing optimization is used to find an optimal unfolding. The additional introduction of glue tabs supports the reconstruction process. We make use of this approach, and we adapt it to accommodate choices C2 and C3. This choice relates to R5.

**C5. Required materials and technologies:** The reason for selecting papercrafts, as opposed to other digital fabrication methods, is that they are easy to manufacture (R4) and reconstruct (R5), as they use affordable resources (paper and colored filters) and technologies (full-color 2D printers), according to R6.

## 4. Nested Papercrafts Pipeline

### 4.1. Pipeline Overview

Figure 2 depicts the workflow of our approach. Our approach takes as input a set of registered meshes that composes the 3D model of an anatomical or biological structure (Figure 2(a), C1). In a pre-processing step, we simplify each mesh to create a feasible papermesh, i.e., a coarse approximation, as a basis for the approach (Figure 2(b), Section 4.2, C2). In the next step, we use an iterative strategy to detect the nested levels within the model and to create a hierarchical structure that represents the nested configuration in the data (Figure 2(c), Section 4.3, C2). After obtaining the distinct nested levels, we perform an entropy-based viewpoint calculation step [VFSH01], which returns the optimal viewpoint that provides an unobstructed view of each inner level (Figure 2(d), Section 4.4, C2). Then, we cut each parent level based on the optimal viewpoint on the child level, to reveal at opening time the best possible view on the inner level (Figure 2(e), Section 4.5, C2). A stability estimation ensures that the cutting configuration at each level results in a feasible and stable physical model, i.e., the individual levels do not fall apart (Figure 2(f), Section 4.6, C2). Subsequently, a projection method creates textures to be rendered on the surface of the papermesh that represent a detailed version of the structure and/or additional inner levels or smaller structures, such as vessels (Figure 2(g), Section 4.7, C3). For this, we take advantage of the subtractive color printing properties of light [SWR20], i.e., we use cyan, magenta, and yellow inks to denote individual

structures. We can reveal individual structures on demand under appropriate colored filters (red, magenta, and blue, respectively). The final step is to unfold the multiple nesting levels of the 3D model with the additional texture projections into multiple single 2D patches [KTGW20] (C4), which we can later print and reconstruct (Figure 2(h), Section 4.8, C5). Our approach exploits the interactivity of nested papercrafts, as well as the projection of additional structures on their surface to allow the examination of 3D models of varying complexity.

### 4.2. Input and Data Approximation

Our input data are registered meshes from anatomical and biological models with a nested configuration R2. No other prior information is required about them, as discussed in Section 3, C1. Also, as discussed in C2 and C3, our fundamental idea is to use an approximation of the input mesh to create a papercraft model that represents the nested configuration, while we project additional structures as textures on the surfaces of the papercraft. In our previous work we already proved that projection can be an effective strategy [SWR20], while we also demonstrated the need for following more organic shapes for the crafts [RGW20]—even if these are simplified. We need two types of minimal user input (R1,3) for our papercraft physicalization: which meshes to group together onto a papercraft and which color to use for each. In our current implementation, users choose which structures are particularly interesting for them to use in the projections.

In this paper, we use the terms “papercraft” to denote the final output of the physicalization workflow and “papermesh” to denote the virtual twin of the papercraft. Each of our papermeshes is a manifold triangular mesh with genus 0 [Mun00]. Before we can create our nested papercraft, we need to approximate the fine input mesh to a reasonably foldable papermesh (R4,5). Takahashi et al. [TWS\*11] showed that the participants could construct meshes with around 150 faces, while meshes over 300 faces were less feasible. Although, theoretically, we support papermeshes with a high face number, we follow Takahashi et al. and limit the total face number to 150. The upcoming sections provide details on the achieved sizes for several anatomical and biological models.

In this preprocessing step, we create our papermeshes in an iterative manner. We start by initializing a bounding box that wraps each loaded structure, as shown in Figure 3. Subsequently, we subdivide the bounding box into a fine triangular mesh (e.g., 3,000 faces for a large anatomical model such as the head), and we move each vertex of the bounding box to the corresponding closest vertex on the

structure. This process allows us to create a compact wrapper that retains better the shape of the structure. Once we have such an approximating mesh, we apply the mesh simplification algorithm by Lindstrom and Turk [LT98] to significantly reduce the mesh face number and to create a papermesh that we can unfold later.

### 4.3. Hierarchy Detection

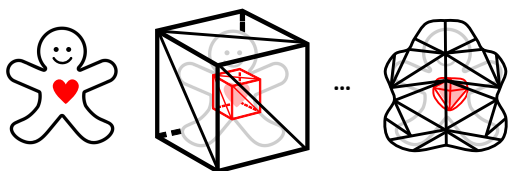
Once we create the papermesh approximations, we need to detect the hierarchy within the nested configuration of the entire model to ensure we can open up outer levels to show inner ones (C2, R2). We identify meshes that are encapsulated in others, and we use this information to build a tree  $T$  representing this relationship. Figure 4 shows an example of a tree constructed for a given nested model. We solve this by using a tree insertion algorithm. As input we consider an unsorted set of papermeshes  $M = m_a, m_b, \dots, m_n$ . The first mesh  $m_a$  is the root of  $T$ . We need to check all following meshes for their hierarchical relationship to the first one. Consider the second mesh to insert  $m_b$ . If  $m_a$  encapsulates  $m_b$ , we assign  $m_b$  as a child node of node  $m_a$  in  $T$ . Otherwise, if  $m_b$  encapsulates  $m_a$ , we assign  $m_b$  as the new root and  $m_a$  as a child of  $m_b$ . We continue this process and traverse  $T$  (Breadth-First Search, BFS or Depth-First Search, DFS) to find the right position of each papermesh.

We conduct the *mesh-inside test* in two steps. First, we test if all vertices on  $m_b$  are inside  $m_a$  [ATW21], and then we perform dD Iso-oriented box intersection tests [LRLTY21] on the edges to detect edge–edge intersections. If the first step is valid and we find no intersection, we settle the relationship between  $m_a$  and  $m_b$ . Note that the algorithm runs under the assumption that no papermeshes intersect, as this is the case for anatomical and biological structures.

### 4.4. Viewpoint Calculation

After capturing the hierarchy within the nested structure, we need to find a suitable configuration to cut it open (C2, R2). A suitable cutting plane must fulfill two conditions. First, when opening an outer nested level, we need to provide the *best view on the inner levels*. We define the best view, as a view that reveals the inner structure in the most recognizable manner, as soon as the outer structure is open. Second, cutting the outer structure should *not affect the stability* of the entire nested model. Although the best view needs to take the stability into account, we deal with this further together with the next two steps of our workflow (Sections 4.5–4.6).

Optimal viewpoint calculation is a heavily researched topic, relevant for several applications within computer graphics and visualization. We refer our readers to a recent comparative survey on viewpoint calculation methods by Bonaventura et al. [BFS\*18]. Our definition of the best view within our specific application is



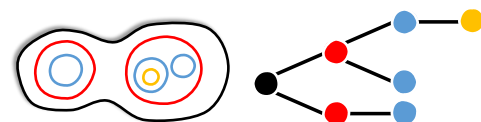
**Figure 3:** Conversion of input structures to a set of papermeshes.

the view that supports the recognizability of the inner levels when opening the outer level. Reviewing existing literature [BFS\*18, DCG10, SLF\*11, PPB\*05] points to entropy-based methods, such as the maximum viewpoint entropy calculation by Vazquez et al. [VFSH01]. This robust measure optimizes in a balanced way the projected area and the number of faces that are visible. Measures that target only area (or silhouette) optimization would not be sufficient, as these measures do not tell us about the amount of detail we can see in our scene. Measures that target only the number of faces are also insufficient, as scenes with a high number of small faces would be inadequately described. The maximum viewpoint entropy indicates that a certain viewpoint balances the number of visible faces with respect to the relative projected area. Although Sbert et al. [SPFG05] found that this method is sensitive to discretization, this is not an issue for our approach, while we discarded other approaches [VFSG06] due to high complexity.

Figure 5 shows the steps of our viewpoint calculation. We conduct a recursive calculation based on viewpoint entropy, as defined by Vazquez et al. [VFSH01] at each level of our nested model (except for the outermost level  $l = 0$ ). The viewpoint entropy  $H$  at each level  $l \geq 1$  assumes as probability distribution the relative area of the projected faces over a sphere of directions centered in the viewpoint, and is defined as:  $H_l = -\sum_{i=0}^{N_l} \frac{A_{i,l}}{A_t} \log \frac{A_{i,l}}{A_t}$ , where  $N_l$  is the number of faces of the scene at a specific level  $l$ ,  $A_{i,l}$  is the projected area of face  $i$  of level  $l$  over the sphere, and  $A_t$  is the total area of the sphere. The maximum viewpoint entropy  $H_l$  at each level  $l$  is used as input to cut the outer level  $l - 1$  (Figure 5), to reveal the structure of the inner level. For example, we use the viewpoint at level 1 that gives the maximum entropy  $H_1$  (Figure 5, see “1st iteration”, red) to cut the parent level (black). We cut the other levels similarly (Figure 5).

### 4.5. Optimal Cutting Plane

Figure 6 shows the steps of the optimal cutting based on the viewpoint calculation. To provide the optimal cutting plane (C2, R2), we need to conduct the following steps. First, we subtract the meshes of the inner level from the outer level to create an “envelope” mesh that carries the difference operation between the inner and outer level. Then, we compute the center of mass  $c_m$  of the meshes of the inner level and we place at this location the origin of our cutting plane. We retrieve the normal  $n$  from the viewpoint calculation, as described in the previous section (Figure 6 (a)). Subsequently, we clip the envelope mesh (between the outer and inner level) using the cutting plane as an implicit function. This returns two meshes with two boundary edge loops, each. We, then, stitch these together with triangles to ensure continuity.



**Figure 4:** Conceptual diagram of a hierarchical relationship in a nested model using a tree data structure.

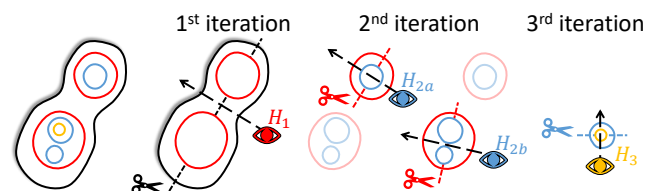
#### 4.6. Papercraft Stability Test

One important factor for a nested papercraft is the stability of the constructed model in the physical world (C2, R2,4). The assembled nested papercraft should not fall apart, and the subparts should stay connected with the entire physical model. We examine the papercraft stability using physical simulation. Note that the stability cannot account for errors introduced by the users during crafting, which cannot be easily controlled as they strongly depend on the user's experience. We conduct the stability test iteratively in conjunction with the viewpoint and the cutting plane calculation (Sections 4.4–4.5). Each cut should guarantee that the decomposed subparts will integrate well with each other. Otherwise, an unstable cut will cause the model to fall apart (Figure 6 (b), left column), while a slightly adapted cut ensures a stable composition based on gravitational and frictional forces (Figure 6 (b), right column). The articulated body algorithm by Featherstone [Fea87] for physics simulation is sufficient for our simple scenes. This is a forward dynamics approach with computational complexity  $O(n)$ . In our implementation, we use gravity as an external force, we assign a mass to each individual papermesh, and simulate if papermeshes will fall apart in a physical environment. If so, the algorithm will return an *unstable* flag back to the cutting plane algorithm. The cutting plane algorithm will look for the next best plane until we reach stability.

#### 4.7. Texture Projection

One papermesh can contain multiple substructures (e.g., an anatomical mesh contains bones, muscles and vessels) (C2–3, R2). To ensure the visibility of the different substructures, we use a strategy, adapted from our previous work [SWR20]. This strategy allows us to use up to three channels to render different substructures on the surface (C3), which we can bring forward through the use of red, green, and blue colored filters or light (C5). These colored filters isolate (groups of) structures rendered respectively in cyan, magenta, and yellow, taking advantage of the properties of visible light. We show an example of this in Figure 1 (e). For more details on the color filter mechanism, we refer to our previous work.

For each structure, we iterate over the triangles of the respective papermesh and we project on them a texture. Before starting the projection, the user can adapt the papermesh to fit better the individual structures using one of the following three modes: *inflation mode* wraps the papermesh around a single structure, projects and then, inflates it back to the original; *clipping mode* removes features that are more distant from the texture without changing



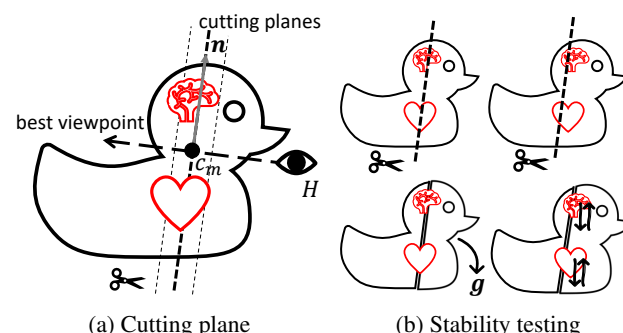
**Figure 5:** Example of the viewpoint calculation within the nested structure. The viewpoint at level  $l$ , where  $l = 1, 2, 3$  with maximum entropy  $H_l$  is used to cut the parent level  $l - 1$ .

the papermesh; and *cube mode* projects the papermesh faces onto the faces of the bounding cube along the face normals of the cube. Figure 7 summarizes these modes. For papermeshes that have multiple pieces, the bounding cube is also cut into multiple pieces, using the same cutting plane. To construct the texture for each triangle, we place a camera at the center of the triangle facing towards the structure. We map the values for each texture to a single hue (magenta, cyan, or yellow) sequential colormap, where higher values are mapped to higher luminance, i.e.,  $[t_{min}, t_{max}] \rightarrow (h, s, v) = (200, 240, 120 \dots 240)$ , as  $(h, s, v) = (200, 240, 120)$  corresponds to pure magenta and  $v = 240$  to white for all values of  $h$  and  $s$ . Subsequently, we normalize each texture to the range  $[0, 1]$  and we multiply with the other textures of this papermesh, if available.

#### 4.8. Papermesh Unfolding

The last step of our workflow generates manifold papermeshes to be unfolded individually. To follow up on (C4, R4–5), we do not allow mesh deformation to retain the shape approximation, as described in Section 4.2. We also aim to keep the unfolded patches as single connected components, so that the reader gains an overview on the number of individual papercrafts. To achieve this, we adapt our previous approach [KTGW20].

We handle the mesh unfolding problem using properties from graph theory, i.e., we use a spanning tree to represent an unfolding patch [TWS\*11]. We create the spanning tree by building a dual graph of the input papermesh. Then, we find the minimum spanning tree of this dual graph. We use the weight assigned to the edges in the spanning tree search algorithm to control the priority of a boundary edge to be cut. For example, if an edge  $e$  is a sharp mountain or valley fold, we avoid cutting this edge since it increases the complexity during construction. In this case, the edge  $e$  is likely to be added in the tree. For more details about the full criteria, we refer our readers to our previous work [KTGW20]. Mathematically, we use the spanning tree to describe variations of unfolded patches. The tree also serves as a backbone to control the unfolding process from 3D to 2D. In addition, we introduce glue tabs to facilitate stitching. This adds another complexity to the optimization, as the triangles of the input mesh should not intersect with the glue tabs when unfolded. For this, we use a simulated annealing optimization.



**Figure 6:** Example of (a) a cutting plane (perpendicular to the best viewpoint direction and passing through the center of mass  $c_m$  of the inner level) and (b) testing for a stable papercraft composition.

In our previous work, we did not consider textures on the meshes and used only indicators to guide the users during reconstruction. In the current approach, we consider textures as rendered from the previous projection step, but we still need the reconstruction indicators. For this, we take the advantage of double-sided printing. On one side of the paper we render the glue tabs, and on the other side we render the corresponding position to be glued on the back side of the patch. In this way, we can render additional indicators on the back of the paper without interfering with the texture of the structure projected on the front side. Figure 9 (c-1) shows an example of the front side and (c-2) an example of the back side.

#### 4.9. Nested Papercraft Construction and Interaction

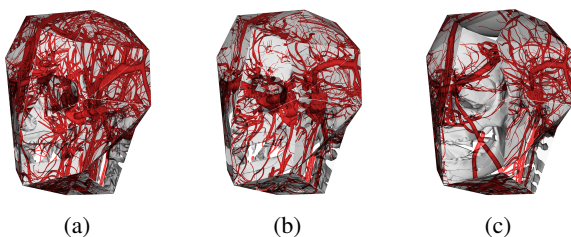
Once we generate the unfolded patches, we print out these on A4 or A3 papers and follow the glue tab instructions to create the papercraft physicalization (C5, R4–6). The assembly requires double-sided tape or glue. An advantage of unfolding all papermeshes to connected patches (one patch for each level of the papermesh) is to provide the users with an overview of the total number of papermeshes in this 3D model (C4, R4–5). Once we assemble the papercraft model, we can open up the subcomponents and explore the information projected on the surface with the use of colored filters (C5, R6) to bring forward different structures (Figure 1(e)).

#### 4.10. Implementation

The main source code is in C++17 and Python 3.9. Major libraries include VTK for fundamental computer graphics management, PyQt5 for the user interface, and CGAL for computational geometry. We compiled the code on a Windows 10 machine using MSVC and CMake. We used Pybullet [CB21] for the physical simulation, and OpenGL for rendering the final results to an off-screen frame buffer using a camera facing the center of the unfolding using parallel projection. Our implementation, along with examples and a demo of this work, is available at our repository [Nes].

### 5. Experimental Results

In this section, we demonstrate several nested papercraft examples generated with our approach. We first employ a set of synthetic models, where we experiment with nested structures of different topologies and hierarchies (Section 5.1). Then, we demonstrate several cases with anatomical and biological models (Section 5.2). Table 1 shows the number of vertices and faces for the input data and for the papermeshes (in parentheses), used in our examples.



**Figure 7:** Different modes of our projection technique [SWR20]: (a) inflation, (b) clipping, and (c) cube mode.

#### 5.1. Synthetic Models

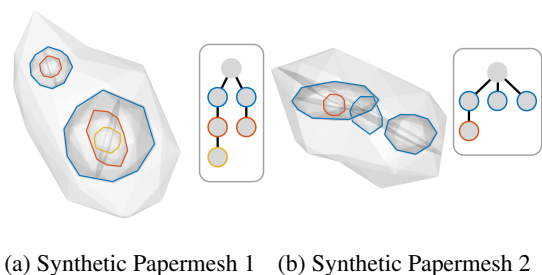
Synthetic meshes are used to showcase the theoretical possibilities of the pipeline, disregarding the possibility of reconstruction due to the limited paper size, when printing unfoldings. Figure 8 shows two different complex configurations of nested meshes. In these cases, we focus on the hierarchy detection, the viewpoint calculation, the identification of the optimal cutting plane, and the stability calculation (i.e., steps (a)–(f) in Figure 2). Our two examples demonstrate that the first part of the workflow is possible and gives solutions within reasonable times. The reconstruction of such complex models might not be possible, due to the limited size of printed patches which makes the nested structures too small to assemble.

#### 5.2. Anatomical and Biological Nested Models

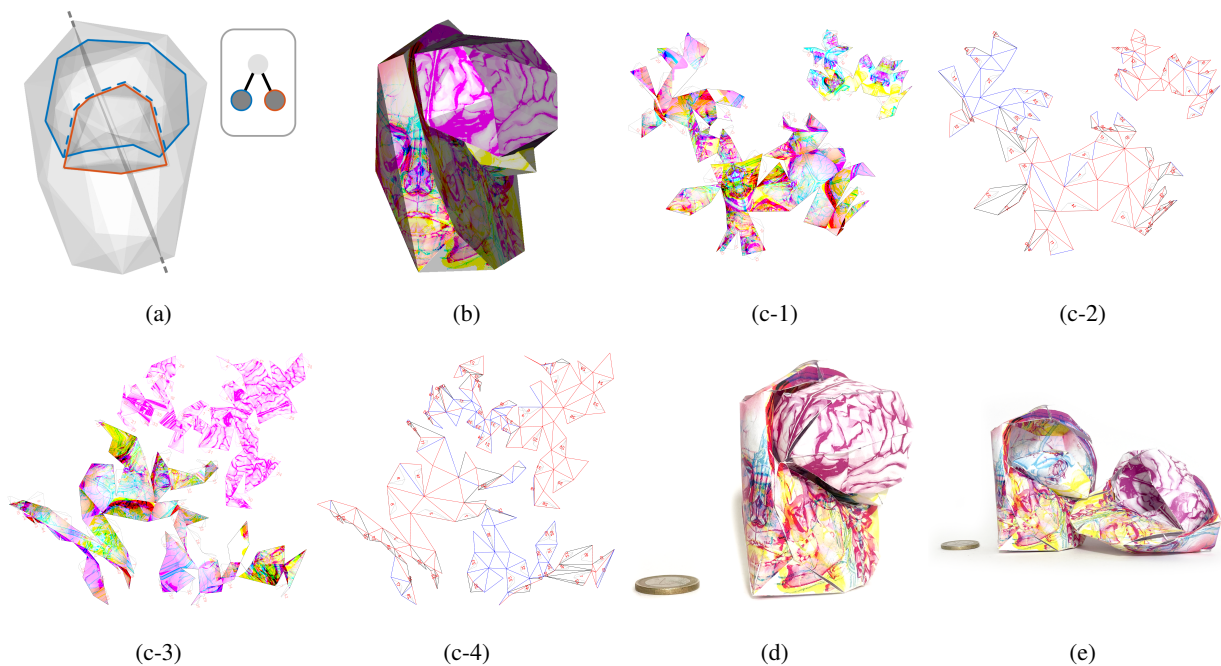
Figure 1 shows the workflow steps to obtain results of a nested papercraft for a *human head*, with only two nested structures, i.e., the outer head and the inner brain. Figure 9 shows a human head papercraft with a more complex nested configuration. Here, the inner level is split into two subparts, as shown in the tree and cutting configuration (Figure 9 (a)). The outer model consists of muscles (in magenta), bones (in cyan), and veins (in yellow), projected onto the papermesh (Figure 9 (b)), the unfolding (Figure 9 (c-1,3)) and papercraft (Figure 9 (d,e)) surface. The inner model consists of two meshes approximating the forebrain (in magenta), and the mid-brain together with the hindbrain as nested structures of the head. These are also projected on the papermesh (Figure 9 (b)), the unfolding (Figure 9 (c-1,3)) and papercraft (Figure 9 (d,e)) surface. This shows the typical use case for projecting structures that cannot be turned into papermeshes, while enabling the exploration of bigger structures and their relation to each other as papermeshes.

Figure 10 shows a nested papercraft example for a *plant cell*. The model is split into three levels in the tree configuration (Figure 10 (a)). Various small or complex structures, i.e., plasmodesmata, amyloplast, peroxisome and the Golgi apparatus (either in magenta or in cyan) are projected on the outer level that represents in yellow the cytoplasm (Figure 10 (b,c)). The nucleus and nucleolus can be unfolded as individual nested papermeshes (see respectively cyan and magenta in Figure 10 (b,c)), and assembled (Figure 10 (d)). Assembling the outer level, visible in Figure 10 (c) took 28 min 49 s, excluding the time for cutting out the template.

Figure 11 shows a nested papercraft of the *human upper body*.



**Figure 8:** Two synthetic models used to experiment with different nested configurations.



**Figure 9:** Our workflow applied on a nested head model. The model was obtained from BodyParts3D/Anatomography [Bod]. (a) The papermesh tree and cutting configuration. (b) Projected texture on papermeshes. (c-1)–(c-4) Unfolded patches for double-sided printing. (d) Assembled and (e) open papercraft, where the brain has been removed.

**Table 1:** The total vertex and face count for input meshes and approximate papermeshes (in parentheses) of each 3D model.

Models	Head (Fig. 1)	Synthetic 1 (Fig. 8(a))	Synthetic 2 (Fig. 8(b))	Head (Fig. 9)	Plant cell (Fig. 10)	Torso (Fig. 11)
Vertices	1,051,155 (102)	252 (109)	210 (152)	1,051,155 (84)	2,396 (60)	1,722,665 (98)
Faces	2,091,807 (196)	480 (210)	400 (296)	2,091,807 (156)	2,316 (112)	3,418,171 (188)

The tree configuration results into three levels for this model (Figure 11 (a)). The outer level shows the veins (in cyan), muscles (in magenta) and bones (in yellow) projected onto the surface (Figure 11 (b,c,f)). The papercraft for the outer level, visible in Figure 11 (g) took 41 min 58 s to assemble. The middle level contains the intestines (in yellow), bladder (in magenta) and kidneys (in cyan) together, and separately the lungs (in yellow) with the bronchial tree (in magenta) (Figure 11 (d,e)). The last level contains the heart (Figure 11 (e)). In this case the nested meshes also have multiple projected structures. Projecting all of the present structures onto the same surface would lead to visual clutter, making it harder to understand the distinct relation between different structures. Instead, generating multiple papermeshes for important organs leads to less visual clutter without losing information (Figure 11 (g)). All of the real world examples were printed on A3, and the relative scale is indicated on the figures for each of the models.

## 6. Evaluation and Discussion

In this section, we assess the feasibility of the presented approach. This includes a performance test of the components of the workflow (Section 6.1) and a feasibility test for papercraft reconstruction (Section 6.2). This is followed by an informal interview with users (Section 6.3). We conducted also a second study to evaluate the

impact of mountain-valley folds on the papercraft assembly (Section 6.4). We include also a discussion of the current limitations and intuitive solutions thereof (Section 6.5).

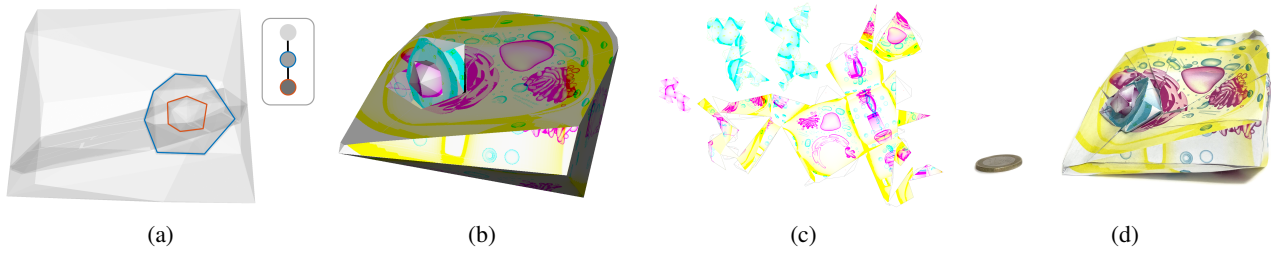
### 6.1. Quantitative Evaluation

Our system was tested on a Windows 10 machine with a 3.8 GHz, 64MB L3 Cache Processor and 32GB of RAM. Table 2 gives an overview of the running time in seconds of each component of our workflow. The first steps (components (a)–(f) of Figure 2, as discussed in Sections 4.2–4.6) do not take too much time. The primary computational bottlenecks come from the projection (Section 4.7, Figure 2(g)) and unfolding (Section 4.8, Figure 2(h)). The former is an iterative approach [SWR20] and the latter is an optimization approach [KTGW20], which justifies their performance times.

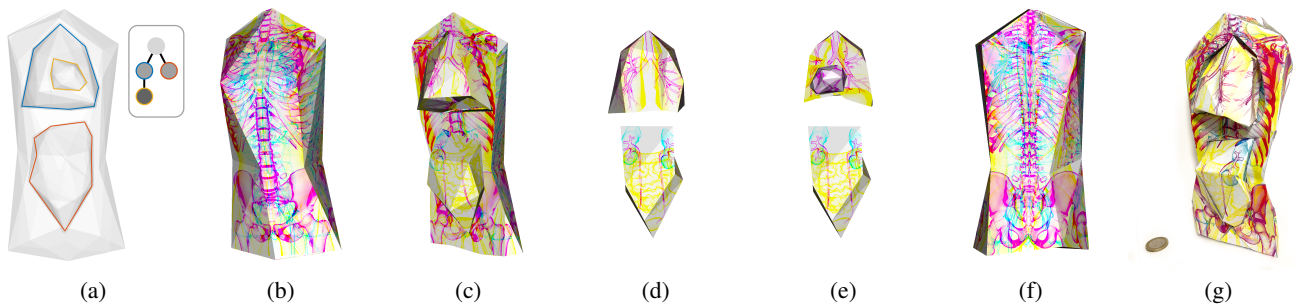
### 6.2. Feasibility of Mesh Construction

We conducted a user study to investigate the feasibility of constructing the nested papercrafts. We recruited 10 participants (3 females and 7 males) with ages ranging from 25 to 39. The participants were mostly colleagues and students. Only one participant (User 4, female) has no computer graphics background and one participant (User 7, male) is colorblind. We prepared the head model shown in Figure 1, being a model of adequate complexity





**Figure 10:** Our workflow applied on a nested plant cell model. (a) The papermesh tree and cutting configuration. (b) Projected texture on papermeshes. (c) Unfolded patches for double-sided printing (only front side visible). (d) Assembled papercraft.



**Figure 11:** Our workflow applied on a nested human torso model. The model was obtained from BodyParts3D/Anatomography [Bod]. (a) The papermesh tree and cutting configuration. (b) Projected texture on the papermesh of the outer level. (c) Outer level open and middle level revealed. (d) Projected texture on the middle level. (e) Middle level open and inner level revealed (on the upper model, inner level corresponds to the heart). (f) Projected texture on the papermesh of the outer level (back side). (g) Assembled, open papercraft.

**Table 2:** The running time (in s) of the results showcased in this paper. The last column indicates the total time (in s and in mm : ss).

Step	Approximation	Hierarchy	Best viewpoing	Cutting	Stability	Projection	Unfolding	Total time in s (and in mm : ss)
Fig. 1	2.04	0.05	0.95	0.3	0.484	461.1	6.69	471.614 (7:52)
Fig. 9	2.23	0.1	1.07	0.54	0.486	482.41	40.11	526.946 (8:47)
Fig. 10	9.85	0.05	1.22	0.56	0.502	727.25	19.78	759.212 (12:39)
Fig. 11	2.87	0.12	1.17	1.23	0.501	863.28	627.01	1496.181 (24:56)

and an interesting structure to learn anatomy. This model includes three components: the left and right half of the head, and the brain inside the head. It is adequately challenging to construct, due to the concavity of half head papercraft, which contains several sharp angles in the papermesh. The patches were printed double-sided on multiple A3 (220 GSM) papers. For each participant, we started by explaining the process of building individual papercrafts and pointed out that a single connected component corresponds to an enclosed papercraft. We moved on to explaining the meaning of the numbers on the glue tabs and that these glue tabs should be attached to the corresponding number on the back side using tape. We instructed the participants to start with the small papercraft of the brain (28 patches), as this is easier and could serve also as an introduction to the assembly of the other, more complex structures. The participants were allowed to observe the patches for as long as they needed and to ask questions regarding the renderings before reconstruction. This part was not included in the timings. We note that once the time measurement began, the participants could

use their own strategy to explore and construct the papercraft, as they would do in a real-life scenario. Table 3 summarizes the total construction time (excluding cutting time) of the nested papercraft for each participant. All participants managed to build the papercraft within less than two hours (average time was 90 min 41 s and standard deviation 21 min 22 s).

### 6.3. Interview with Evaluation Participants

After completing the reconstruction of the papercrafts, we asked the evaluation participants about their experience with the physicalization. When asked about whether the papercraft was easy to build or not, most of the participants indicated that they had difficulties—especially, when building the “negative” side of the head level, i.e., the head–brain interface. Reconstructing just the brain was considered easy by most, and all reconstructed it fast. Most people would have appreciated additional indicators for the reconstruction. One person commented that they “would like to build one with a single

**Table 3:** The assembly time (in mm:ss) of the head model (Figure 1) for each participant (F: female and M: male) for the two studies (A: without and B: with mountain-valley folds), excl. cutting time. We denote average (avg), standard deviation (sd), and no participation (N/A).

User	1 (M)	2 (M)	3 (F)	4 (F)	5 (F)	6 (M)	7 (M)	8 (M)	9 (M)	10 (M)	avg ± sd
Time A (mm:ss)	99:38	95:04	112:32	99:12	59:04	53:02	98:04	103:07	117:04	67:32	90:26 ± 22:23
Time B (mm:ss)	85:52	92:21	99:56	89:33	48:52	40:06	86:23	N/A	N/A	N/A	77:35 ± 23:13

texture, to see if it would be easier". We also asked the participants what they consider to be an upper limit in number of meshes for a feasible reconstruction. Most participants commented that 3–5 meshes would be the upper limit for them. We also asked them about a comparison to traditional on-screen visualizations. One participant commented that "it is interesting to be able to take the brain out with [their] own hands", and another one mentioned that "[...] the scope [i.e., between physicalization and visualization] is different". Two people were clearly in favor of an on-screen visualization. When asked about particular findings, one person commented that "you can see a very surprising scale of how big the brain is in comparison to the skull and that is more noticeable than a 3D rendering". We finally asked them about potential applications, and half of the participants commented about children education. One commented that the model is "very fragile to use for anything else than entertainment [and] would seem unprofessional to use in patient communication".

#### 6.4. Effect of Mountain-Valley Folds on Assembly

In a second study, we assess the effect of mountain-valley folds on the papercraft assembly. Two months after the first study, we asked the same people to participate in a second session. Participants 8–10 were not available, but participants 1–7 underwent the same process as described in Section 6.2. The difference in this case was that the papercrafts had an additional encoding of the mountain-valley fold [TWS\*11], supporting the users in folding the patch towards the "outside" or the "inside". Again, we measured the assembly time for all participants. The results are summarized in the last row of Table 3, and they confirm that the assembly time decreases with the use of mountain-valley folds by more than about 13 minutes (average time was 77 min 35 s and standard deviation 23 min 13 s). Other smart guidance techniques to support the efficient assembly of the papercrafts should be further investigated, while coupling with informative aids could make the process more engaging.

#### 6.5. Limitations

Our evaluation brought forward some limitations of our approach, opening opportunities for future improvements. Some limitations are inherited from the employed computer graphics solutions, while others were introduced by the specifics of our implementation. The papermesh after the mesh approximation should be *quasi-convex and manifold*. This implies that the papermeshes cannot self-intersect, as they need to fit together in a physicalization. If intersection happens after the mesh approximation, we have to slightly scale or translate the meshes to avoid this situation. Additionally, only *manifold meshes with genus 0* are currently supported. Whether a mesh with a genus higher than 0 can always be unfolded into a single connected patch remains a general open topic. As an alternative option, we can cut meshes with higher

genus values to a set of meshes with genus 0 and apply our approach. Note that the shapes of the approximated meshes influences whether an inner mesh can be *taken out of* its outer level. A precise shape analysis scheme could be used to optimally cut the branches of a mesh, instead. It remains to be investigated if a stable solution can be reached—and even, if a possible solution can be found—with an increasing number of constraints. For the same reason, *simple implicit functions* are used, and more complex configurations should be investigated in the future. Another straightforward solution would require to employ a user-defined cutting plane.

Our initial user study (Section 6.2) also showed that the combination of concave and convex patches on the meshes can lead to *difficulties during the folding process*. The assembly of the head papercraft employed in the user study was in average 1.5 hour, which can be a considerable amount of time of classroom education. The construction and assembly time of our papercraft is reasonable based on our discussion with a professional designer [Cha]. As also mentioned in Section 1, a simple 3D model with 66 polygonal faces requires a week for the creation and 10 working days for the generation of the corresponding papercraft, while assembling it requires one hour. Our early investigations on advanced guidance mechanisms (Section 6.4) indicate ways to simplify the construction, but we will pursue this further to identify adequate ways of reducing the user's overhead (e.g., assembly order, paper size).

#### 7. Conclusion and Future Work

Our paper presents a novel workflow to support the computer-aided generation of paper-based physicalizations. The nature of nested papercrafts allows us to map anatomical or biological models to submeshes that physically represent their hierarchical topology. The approach combines playfulness with insight, by supporting the easy design of papercrafts and data navigation through a combination of unfolding and texture projections that can be explored through colored filters or light. As shown in our user study, the constructed papercrafts can be used for edutainment purposes. It would be interesting to evaluate whether this could be also used outside of the domain of anatomy or biology. As most prominent future improvements, a constrained-based cutting algorithm can be investigated to minimize the number of cutting planes, while extending our input to non-manifold meshes remains an open and challenging problem. When thinking about human factors, an interesting direction would be to consider machine-assisted recommendations in the workflow in order to ease the papercraft construction process. Finally, an evaluation that covers the entire fabrication process (and potential ways of supporting the users better, e.g., with different papers or different scorings) needs to be conducted.

#### Acknowledgments

We thank Amao Chan [Cha] for sharing his knowledge on papercraft design and production, and his thoughts on future challenges.

## References

- [ASS\*19] ANG K. D., SAMAVATI F. F., SABOKROHIYEH S., GARCIA J., ELBAZ M. S.: Physicalizing cardiac blood flow data via 3D printing. *Computers & Graphics* 85 (2019), 42–54. doi:10.1016/j.cag.2019.09.004. 3
- [ATG\*18] AN B., TAO Y., GU J., CHENG T., CHEN X. A., ZHANG X., ZHAO W., DO Y., TAKAHASHI S., WU H.-Y., ET AL.: Thermorph: Democratizing 4D printing of self-folding materials and interfaces. In *Proc. of the 2018 CHI* (New York, NY, USA, 2018), pp. 1–12. doi:10.1145/3173574.3173834. 3
- [ATW21] ALLIEZ P., TAYEB S., WORMSER C.: 3D fast intersection and distance computation. In *CGAL User and Reference Manual*, 5.3 ed. CGAL Editorial Board, 2021. URL: <https://doc.cgal.org/5.3/Manual/packages.html#PkgAABBTree>. 5
- [BCK\*11] BLUME A., CHUN W., KOGAN D., KOKKEVIS V., WEBER N., PETTERSON R. W., ZEIGER R.: Google body: 3D human anatomy in the browser. In *ACM SIGGRAPH 2011 Talks* (New York, NY, USA, 2011), ACM, p. 19. doi:10.1145/2037826.2037852. 2
- [BCMP18] BICKEL B., CIGNONI P., MALOMO L., PIETRONI N.: State of the art on stylized fabrication. In *Computer Graphics Forum* (New York, NY, USA, 2018), vol. 37, Wiley Online Library, ACM, pp. 325–342. doi:10.1145/3355047.3359411. 2
- [BFS\*18] BONAVENTURA X., FEIXAS M., SBERT M., CHUANG L., WALLRAVEN C.: A survey of viewpoint selection methods for polygonal models. *Entropy* 20, 5 (2018), 370. doi:10.3390/e20050370. 5
- [BG05] BRUCKNER S., GRÖLLER M. E.: Volumeshop: An interactive system for direct volume illustration. In *Proc. of IEEE Vis* (Minneapolis, MN, USA, 2005), pp. 671–678. doi:10.1109/VISUAL.2005.1532856. 2
- [Bod] Bodyparts3d. <https://lifesciencedb.jp/bp3d/>. Accessed: 30-10-2021. 8, 9
- [BPC\*14] BAUER A., PACLET F., CAHOUE V., DICKO A. H., PALOMBI O., FAURE F., TROCCAZ J.: Interactive visualization of muscle activity during limb movements: Towards enhanced anatomy learning. In *Eurographics Workshop on Visual Computing for Biology and Medicine* (Vienna, Austria, 2014), Eurographics Association, pp. 191–198. doi:10.2312/vcbm.20141191. 2
- [Can01] CANNING D. R.: Teaching the structural nature of biological molecules: Molecular visualization in the classroom and in the hands of students. *Chemistry Education Research and Practice* 2, 2 (2001), 109–122. doi:10.1039/B1RP90013G. 2
- [CB21] COUMANS E., BAI Y.: Pybullet: A python module for physics simulation for games, robotics and machine learning. <http://pybullet.org>, 2016–2021. 7
- [Cha] CHAN A.: Amao Chan Art Studio. <https://amaochan.work/a>. Accessed: 08-02-2022. 2, 10
- [Cha10] CHARSKY D.: From edutainment to serious games: A change in the use of game characteristics. *Games and Culture* 5, 2 (2010), 177–198. doi:10.1177/1555412009354727. 1
- [CLF\*06] CAI Y., LU B., FAN Z., INDHUMATHI C., LIM K. T., CHAN C. W., JIANG Y., LI L.: Bio-edutainment: Learning life science through x gaming. *Computers & Graphics* 30, 1 (2006), 3–9. doi:10.1016/j.cag.2005.10.003. 1, 2
- [CLZL06] CAI Y., LU B., ZHENG J., LI L.: Immersive protein gaming for bioedutainment. *Simulation & Gaming* 37, 4 (2006), 466–475. doi:10.1177/1046878106293677. 2
- [CS17] CHEN L., SASS L.: Generative computer-aided design: Multimodality large-scale direct physical production. *Computer-Aided Design and Applications* 14, 1 (2017), 83–94. doi:10.1080/16864360.2016.1199751. 3
- [CY17] CHANG Y.-J., YEN H.-C.: Improved algorithms for grid-unfolding orthogonal polyhedra. *International Journal of Computational Geometry & Applications* 27, 01n02 (2017), 33–56. doi:10.1142/S0218195917600032. 3
- [DCG10] DUTAGACI H., CHEUNG C. P., GODIL A.: A benchmark for best view selection of 3D objects. In *Proc. of the ACM Workshop on 3D Object Retrieval* (New York, NY, USA, 2010), ACM, pp. 45–50. doi:doi.org/10.1145/1877808.1877819. 5
- [DSMA\*21] DJAVAHERPOUR H., SAMAVATI F., MAHDAVI-AMIRI A., YAZDANBAKHSH F., HURON S., LEVY R., JANSEN Y., OEHLBERG L.: Data to physicalization: A survey of the physical rendering process. *Computer Graphics Forum* 40, 3 (2021). doi:10.1111/cgf.14330. 1, 2
- [EE98] EISENBERG M., EISENBERG A.: Shop class for the next millennium: Education through computer-enriched handicrafts. *Journal of Interactive Media in Education* 98, 8 (1998). doi:10.1145/2702123.2702248. 2, 3
- [Fes87] FEATHERSTONE R.: *Forward Dynamics – The Articulated-Body Method*. Springer US, 1987, pp. 89–105. doi:10.1007/978-0-387-74315-8\_6. 6
- [FTS\*13] FELTON S. M., TOLLEY M. T., SHIN B., ONAL C. D., DEMAIN E. D., RUS D., WOOD R. J.: Self-folding with shape memory composites. *Soft Matter* 9, 32 (2013), 7688–7694. doi:10.1039/c3sm51003d. 3
- [GDVZ21] GOODSSELL D. S., DUTTA S., VOIGT M., ZARDECKI C.: Molecular storytelling for online structural biology outreach and education. *Structural Dynamics* 8, 2 (2021). doi:10.1063/4.0000077. 2
- [Gil05] GILBERT J. K.: Visualization: A metacognitive skill in science and science education. In *Visualization in Science Education*. Springer, 2005, pp. 9–27. doi:10.1007/1-4020-3613-2\_2. 1
- [GWW\*04] GILLET A., WEGHORST S., WINN W., STOFFLER D., SANNER M., GOODSSELL D., OLSON A.: Computer-linked auto-fabricated 3D models for teaching structural biology. In *ACM SIGGRAPH 2004 Sketches*. ACM, 2004, p. 28. doi:10.1145/1186223.1186258. 3
- [HDK17] HALLE M., DEMEUSY V., KIKINIS R.: The open anatomy browser: A collaborative web-based viewer for interoperable anatomy atlases. *Frontiers in Neuroinformatics* 11 (2017), 22. doi:10.3389/fninf.2017.00022. 2
- [HE12] HAENSELMANN T., EFFELSBERG W.: Optimal strategies for creating paper models from 3D objects. *Multimedia Systems* 18, 6 (2012), 519–532. doi:10.1007/s00530-012-0273-1. 2
- [HRL10] HUANG H.-M., RAUCH U., LIAW S.-S.: Investigating learners' attitudes toward virtual reality learning environments: Based on a constructivist approach. *Computers & Education* 55, 3 (2010), 1171–1182. doi:10.1016/j.compedu.2010.05.014. 2
- [IIM16] IGARASHI Y., IGARASHI T., MITANI J.: Computational design of iris folding patterns. *Computational Visual Media* 2, 4 (2016), 321–327. doi:10.1007/s41095-016-0062-4. 3
- [JDF13] JANSEN Y., DRAGICEVIC P., FEKETE J.-D.: Evaluating the efficiency of physical visualizations. In *Proc. SIGCHI* (New York, NY, USA, 2013), ACM, pp. 2593–2602. doi:10.1145/2470654.2481359. 2
- [JDI\*15] JANSEN Y., DRAGICEVIC P., ISENBERG P., ALEXANDER J., KARNIK A., KILDAL J., SUBRAMANIAN S., HORNBAEK K.: Opportunities and challenges for data physicalization. In *Proc. of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (New York, NY, USA, 2015), ACM, pp. 3227–3236. doi:10.1145/2702123.2702180. 2
- [JPC\*16] JOHN N. W., PHILLIPS N. I., CENYDD L. A., POP S. R., COOPE D., KAMALY-ASL I., DE SOUZA C., WATT S. J.: The use of stereoscopy in a neurosurgery training virtual environment. *Presence: Teleoperators and Virtual Environments* 24, 4 (2016), 289–298. doi:10.1162/PRES\_a\_00270. 2
- [JVJB17] JANG S., VITALE J. M., JYUNG R. W., BLACK J. B.: Direct manipulation is better than passive viewing for learning anatomy in a three-dimensional virtual reality environment. *Computers & Education* 106 (2017), 150–165. doi:10.1016/j.compedu.2016.12.009. 2

- [KSV15] KURUP H. K., SAMUEL B. P., VETTUKATTIL J. J.: Hybrid 3D printing: A game-changer in personalized cardiac medicine? *Expert Review of Cardiovascular Therapy* 13, 12 (2015), 1281–1284. doi: [10.1586/14779072.2015.1100076](https://doi.org/10.1586/14779072.2015.1100076). 3
- [KTGW20] KORPITSCH T., TAKAHASHI S., GRÖLLER E., WU H.-Y.: Simulated annealing to unfold 3D meshes and assign glue tabs. *Journal of World Society for Computer Graphics (WSCG)* 28, 1–2 (2020), 47–56. doi: [10.24132/JWSCG.2020.28.6](https://doi.org/10.24132/JWSCG.2020.28.6). 3, 4, 6, 8
- [LLL\*13] LE S. N., LEOW S.-J., LE-NGUYEN T.-V., RUIZ C., LOW K.-L.: Surface and contour-preserving origamic architecture paper pop-ups. *TVCG* 20, 2 (2013), 276–288. doi: [10.1109/TVCG.2013.108](https://doi.org/10.1109/TVCG.2013.108). 3
- [LRLTY21] LORIOT S., ROUXEL-LABBÉ M., TOURNOIS J., YAZ I. O.: Polygon mesh processing. In *CGAL User and Reference Manual*, 5.3 ed. CGAL Editorial Board, 2021. URL: <https://doc.cgal.org/5.3/Manual/packages.html#PkgPolygonMeshProcessing>. 5
- [LSHL16] LICHTENBERG N., SMIT N., HANSEN C., LAWONN K.: Sline: Seamless line illustration for interactive biomedical visualization. In *Eurographics Workshop on Visual Computing for Biology and Medicine* (Goslar, DEU, 2016), Eurographics Association, pp. 133–142. doi: [10.5555/3061507.3061529](https://doi.org/10.5555/3061507.3061529). 2
- [LT98] LINDSTROM P., TURK G.: Fast and memory efficient polygonal simplification. In *Proc. of the Conference on Visualization '98* (Research Triangle Park, NC, USA, 1998), IEEE Computer Society Press, pp. 279–286. doi: [10.1109/VISUAL.1998.745314](https://doi.org/10.1109/VISUAL.1998.745314). 5
- [MM21] MNGUNI L., MOYO D.: An assessment of the impact of an animation on biology students' visualization skills related to basic concepts of mitosis. *EURASIA Journal of Mathematics, Science and Technology Education* 17, 8 (2021), 1997. doi: [doi.org/10.29333/ejmste/11116](https://doi.org/10.29333/ejmste/11116). 2
- [MMŽ10] MARKOVIĆ D., MARKOVIĆ ŽIVKOVIĆ B.: Development of anatomical models-chronology. *Acta Medica Medianae* 49, 2 (2010). 2, 3
- [MS04] MITANI J., SUZUKI H.: Making papercraft toys from meshes using strip-based approximate unfolding. In *ACM TOG* (New York, NY, USA, 2004), vol. 23, ACM, pp. 259–263. doi: [10.1145/1015706.1015711](https://doi.org/10.1145/1015706.1015711). 3
- [Mun00] MUNKRES J. R.: *Topology*, vol. 2. Prentice Hall, Upper Saddle River, NJ, USA, 2000. 4
- [MWDE\*16] MESSIER E., WILCOX J., DAWSON-ELLI A., DIAZ G., LINTE C. A.: An interactive 3D virtual anatomy puzzle for learning and simulation - initial demonstration and evaluation. *Studies in Health Technology and Informatics* 220 (2016), 233–40. doi: [10.3233/978-1-61499-625-5-233](https://doi.org/10.3233/978-1-61499-625-5-233). 2
- [Nes] Nested papercrafts for anatomical and biological edutainment. <https://github.com/MarwinSc/Nested-Papercrafts-for-Anatomical-and-Biological-Edutainment>. 7
- [PDRK18] PACZKOWSKI P., DORSEY J., RUSHMEIER H., KIM M. H.: PaperCraft3D: Paper-based 3D modeling and scene fabrication. *TVCG* 25, 4 (2018), 1717–1731. doi: [10.1109/TVCG.2018.2820068](https://doi.org/10.1109/TVCG.2018.2820068). 2, 3
- [Per21] PERIN C.: What students learn with personal data physicalization. *IEEE Computer Graphics and Applications* (2021). doi: [10.1109/MCG.2021.3115417](https://doi.org/10.1109/MCG.2021.3115417). 1
- [PHP\*01] POMMERT A., HÖHNE K. H., PFLESSER B., RICHTER E., RIEMER M., SCHIEMANN T., SCHUBERT R., SCHUMACHER U., TIEDE U.: Creating a high-resolution spatial/symbolic model of the inner organs based on the visible human. *Medical Image Analysis* 5, 3 (2001), 221–228. doi: [10.1016/S1361-8415\(01\)00044-5](https://doi.org/10.1016/S1361-8415(01)00044-5). 2
- [PPB\*05] POLONSKY O., PATANÉ G., BIASOTTI S., GOTSMAN C., SPAGNUOLO M.: What's in an image? *The Visual Computer* 21, 8 (2005), 840–847. doi: [doi.org/10.1007/s00371-005-0326-y](https://doi.org/10.1007/s00371-005-0326-y). 5
- [PPS19] POHLANDT D., PREIM B., SAALFELD P.: Supporting anatomy education with a 3D puzzle in a VR environment-results from a pilot study. In *Proc of Mensch und Computer 2019*. ACM, 2019, pp. 91–102. doi: [10.1145/3340764.3340792](https://doi.org/10.1145/3340764.3340792). 2
- [PS18] PREIM B., SAALFELD P.: A survey of virtual human anatomy education systems. *Computers & Graphics* 71 (2018), 132–153. doi: [10.1016/j.cag.2018.01.005](https://doi.org/10.1016/j.cag.2018.01.005). 2
- [PWR21] PAHR D., WU H.-Y., RAIDOU R. G.: Vologram: An Educational Holographic Sculpture for Volumetric Medical Data Physicalization. In *Eurographics Workshop on Visual Computing for Biology and Medicine* (2021). doi: [10.34726/hss.2021.79540](https://doi.org/10.34726/hss.2021.79540). 2, 3
- [RGW20] RAIDOU R. G., GRÖLLER M. E., WU H.-Y.: Slice and dice: A physicalization workflow for anatomical edutainment. *Computer Graphics Forum* 39, 7 (2020), 623–634. doi: [10.1111/cgf.14173](https://doi.org/10.1111/cgf.14173). 2, 3, 4
- [RJLYL14] RUIZ JR C. R., LE S. N., YU J., LOW K.-L.: Multi-style paper pop-up designs from 3D models. *Computer Graphics Forum* 33, 2 (2014), 487–496. doi: [10.1111/cgf.12320](https://doi.org/10.1111/cgf.12320). 3
- [RLL15] RUIZ C., LE S. N., LOW K.-L.: Generating animated paper pop-ups from the motion of articulated characters. *The Visual Computer* 31, 6–8 (2015), 925–935. doi: [10.1007/s00371-015-1125-8](https://doi.org/10.1007/s00371-015-1125-8). 3
- [RMVTK\*10] RENGIER F., MEHNDIRATTA A., VON TENGG-KOBLIGK H., ZECHMANN C., UNTERHINNINGHOFEN R., KAUCZOR H.-U., GIESEL F.: 3D printing based on imaging data: Review of medical applications. *International Journal of Computer Assisted Radiology and Surgery* 5 (2010), 335–341. doi: [10.1007/s11548-010-0476-x](https://doi.org/10.1007/s11548-010-0476-x). 2, 3
- [SB17] STOPPEL S., BRUCKNER S.: Vol<sup>2</sup>velle: Printable interactive volume visualization. *TVCG* 23, 01 (2017), 861–870. doi: [10.1109/TVCG.2016.2599211](https://doi.org/10.1109/TVCG.2016.2599211). 2, 3
- [SCS16] SASS L., CHEN L., SUNG W. K.: Embodied prototyping: Exploration of a design-fabrication framework for large-scale model manufacturing. *Computer-Aided Design and Applications* 13, 1 (2016), 124–137. doi: [10.1080/16864360.2015.1059202](https://doi.org/10.1080/16864360.2015.1059202). 3
- [SLF\*11] SECORD A., LU J., FINKELSTEIN A., SINGH M., NEALEN A.: Perceptual models of viewpoint preference. *ACM Trans. Graph.* 30, 5 (2011), 1–12. doi: [10.1145/2019627.2019628](https://doi.org/10.1145/2019627.2019628). 5
- [SP11] STRAUB R., PRAUTZSCH H.: Creating optimized cut-out sheets for paper models from meshes. *Karlsruhe Reports in Informatics* 2011 36 (2011). doi: [10.5445/IR/1000025577](https://doi.org/10.5445/IR/1000025577). 3
- [SPFG05] SBERT M., PLEMENOS D., FEIXAS M., GONZALEZ F.: Viewpoint quality: Measures and applications. In *Proc. of the 1st Eurographics Workshop on Computational Aesthetics in Graphics, Visualization and Imaging* (2005), Computational Aesthetics'05, pp. 185–192. doi: <https://doi.org/10.5555/2381219.2381245>. 5
- [SSB15] STUSAK S., SCHWARZ J., BUTZ A.: Evaluating the memorability of physical visualizations. In *Proc. of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (New York, NY, USA, 2015), ACM, pp. 32470–3250. doi: [10.1145/2702123.2702248](https://doi.org/10.1145/2702123.2702248). 2
- [SSPOJ16] SAALFELD P., STOJNIC A., PREIM B., OELTZE-JAFRA S.: Semi-immersive 3D sketching of vascular structures for medical education. In *Eurographics Workshop on Visual Computing for Biology and Medicine* (Vienna, Austria, 2016), The Eurographics Association, pp. 123–132. doi: [10.2312/vcbm.20161280](https://doi.org/10.2312/vcbm.20161280). 2
- [SWO\*14] STEFAN P., WUCHERER P., OYAMADA Y., MA M., SCHOCH A., KANEGAE M., SHIMIZU N., KODERA T., CAHIER S., WEIGL M., ET AL.: An AR edutainment system supporting bone anatomy learning. In *2014 IEEE Virtual Reality (VR)* (Minneapolis, MN, USA, 2014), IEEE, pp. 113–114. doi: [10.1109/VR.2014.6802077](https://doi.org/10.1109/VR.2014.6802077). 1
- [SWR20] SCHINDLER M., WU H.-Y., RAIDOU R.: The anatomical edutainer. In *IEEE Vis Short Papers 2020* (Salt Lake City, UT, USA, October 2020), IEEE, pp. 1–5. doi: [10.1109/VIS47514.2020.00007](https://doi.org/10.1109/VIS47514.2020.00007). 2, 3, 4, 6, 7, 8

- [TWS\*11] TAKAHASHI S., WU H.-Y., SAW S. H., LIN C.-C., YEN H.-C.: Optimized topological surgery for unfolding 3D meshes. *Computer Graphics Forum* 30, 7 (2011), 2077–2086. doi:10.1111/j.1467-8659.2011.02053.x. 2, 3, 4, 6, 10
- [VFSG06] VIOLA I., FEIXAS M., SBERT M., GRÖLLER M. E.: Importance-driven focus of attention. *TVCG* 12, 5 (2006), 933–940. doi:https://doi.org/10.1109/TVCG.2006.152. 5
- [VFSH01] VÁZQUEZ P.-P., FEIXAS M., SBERT M., HEIDRICH W.: Viewpoint selection using viewpoint entropy. In *Proc. of the Vision Modeling and Visualization Conference 2001* (2001), pp. 273–280. doi:10.5555/647260.718491. 4, 5
- [VGHN08] VÁZQUEZ P.-P., GÖTZELMANN T., HARTMANN K., NÜRNBERGER A.: An interactive 3D framework for anatomical education. *International Journal of Computer Assisted Radiology and Surgery* 3, 6 (2008), 511–524. doi:10.1007/s11548-008-0251-4. 2
- [XZM\*18] XIAO N., ZHU Z., MARTIN R. R., XU K., LU J.-M., HU S.-M.: Computational design of transforming pop-up books. *TOG* 37, 1 (2018), 1–14. doi:10.1145/3156934. 3
- [ZGPR16] ZHANG Y., GAO W., PAREDES L., RAMANI K.: Card-boardizer: Creatively Customize, Articulate and Fold 3D Mesh Models. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), ACM, pp. 897–907. doi:10.1145/2858036.2858362. 3
- [ZM08] ZHAO J., MOERE A. V.: Embodiment in data sculpture: A model of the physical visualization of information. In *Proc. of the 3rd International Conference on Digital Interactive Media in Entertainment and Arts* (New York, NY, USA, 2008), ACM, pp. 343–350. doi:10.1145/1413634.1413696. 2, 3