3D4DT: AN APPROACH TO EXPLORE DECISION TREES FOR THEMATIC MAP CREATION AS AN INTERACTIVE 3D SCENE

J. Bagert[1], A. Degbelo[1,2]*

[1]Institute for Geoinformatics, University of Münster, Germany - joshua.bagert@web.de
[2]Chair of Geoinformatics, TU Dresden, Germany - auriol.degbelo@tu-dresden.de

KEY WORDS: decision trees, 3D visualization, automated thematic map creation, explainable (geo)software, Three.js

ABSTRACT:

Several tools have been proposed to automatically create thematic maps. A common drawback of these tools is that they rarely provide means for users to explore their decision trees. Users may thus create maps or get some maps proposed, but not know why some suggestions of maps were made. To address this gap, this work introduces 3D4DT, an approach to explore decision trees as a three-dimensional interactive scene. The 3D4DT approach uses JSON as a machine-readable format to represent decision trees and maps JSON elements to user interface elements (e.g. radio buttons and cubes). A preliminary evaluation of 3D4DT has shown effectiveness gains in comparison to information displayed as text+picture, for a decision tree with a simple hierarchical structure. The contributions of this work are relevant to the design of more transparent software for automated thematic map creation.

1. INTRODUCTION

Thematic maps are useful to portray geographic patterns of phenomena over a spatial region. They have become a medium of choice for the visualization of open geographic data related to the world's biggest challenges (see for example their use as a visualization medium by 'Our World in Data'[1]). Creating thematic maps may require familiarity with a software library, a programming language, or a user interface. For instance, creating maps through programming can be done using D3.js (Bostock et al., 2011), Vega-Lite (Satyanarayan et al., 2017), Florence (Poorthuis et al., 2020), Unfolding (Nagel et al., 2013), the mapmap.js API (Ledermann and Gartner, 2015), or programming languages such as R or Python. Creating maps through a user interface that suggests meaningful defaults can be done through tools/toolkits such as Descartes (Andrienko and Andrienko, 1999), SDG Viz (Gong, 2019), AdaptiveMaps (Degbelo et al., 2020), the GAV Toolkit (Van Ho et al., 2012), the Geoviz Toolkit (Hardisty and Robinson, 2011), and commercial solutions such as ArcGis Online[2] or Carto[3].

Research Gap: An important drawback of current state-of-the-art tools is that the expertise encapsulated in these software (e.g. enabling one to choose a type of map or visual variables depending on the characteristics of the data contained in the map), is often not well communicated to the user. That is, users can use these tools to create meaningful maps for their open geographic datasets but are offered little support regarding why some suggestions of thematic map types were made (e.g. why a dot map is proposed by a toolkit instead of a choropleth map). Put simply, users get little insight into the decision processes of current tools/toolkits for thematic web creation.

Contributions & Target audience: To help users learn about the decision processes of software for automatic map creation, this work introduces the 3D4DT approach. The approach uses JSON (JavaScript Object Notation) as a machine-readable format to represent decision trees and subsequently maps JSON elements to user interface elements for an interactive 3D scene. The contributions are twofold: 1) a controlled vocabulary to support the creation of machine-readable descriptions for decision trees of the Cartography literature; and 2) an approach to navigate these decision trees as interactive scenes in 3D. The approach is implemented as an open-source prototype. It is relevant to both developers and users of software for automatic thematic map creation. The controlled vocabulary can be used by developers to encode the decision trees underlying their software as machine-readable data, and make the 'brain' of their software available for reuse in multiple use cases. The exploration of the decision trees as an interactive scene is relevant to users who can retrieve information about the inner workings of software for map creation in an interactive format. The expected benefits for these latter users will be similar to the customary benefits of explainable software (Nunes and Jannach, 2017), namely: *transparency* (explain how the system works), *education* (allow users to learn something from the system) and *scrutability* (allow users to tell the system it is wrong). At last, since decision trees are often produced during research endeavours, the 3D4DT approach is one way of facilitating the reuse of research outcomes, in line with an ongoing initiative to foster the reuse of research products in the Earth System Sciences (NDFI4Earth, see Bernard et al. (2021)). Throughout the article, a decision tree is defined in line with (Nauta et al., 2023) as a rooted graph with a conditional statement at each node.

2. BACKGROUND

2.1 Exemplar decision trees from Cartography

In the Cartography literature, decision trees are useful to synthesize theoretical knowledge and best practices. These decision trees may be explicit (i.e. the tree-like model of decisions and their possible consequences is explicit) or implicit (i.e. the knowledge can be encoded into a tree-like model via a transformation). An example of *explicit* decision tree is the decision tree to guide the choice of thematic map types by (Kraak et al.,

* Corresponding author

[1] https://ourworldindata.org/ (accessed: April 06, 2023).

[2] https://www.arcgis.com/ (accessed: April 06, 2023).

[3] https://carto.com/ (accessed: April 06, 2023).

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W7-2023-3-2023 | © Author(s) 2023. CC BY 4.0 License.

2020), while the decision tree used in the AdaptiveMaps prototype to automate the creation of web maps (Degbelo et al., 2020) is *implicit*. Variable syntactics are also a good example of *implicit* decision trees. As discussed in (Roth, 2017), syntactics is a term that denotes the relationship of sign vehicles to one another in semiotics. In Cartography, 'syntactics prescribes the use of a (visual) variable given the level of measurement of the attribute information' (Roth, 2017). MacEachren (1995) synthesized a few variable syntactics based on a review of the literature: visual variable syntactics (to relate visual variables to Stevens' levels of measurement), dynamic/sound variable syntactics (to relate dynamic and sound variables to the levels of measurement), and colour syntactics (to relate colour type selection to the kind of data represented: qualitative, binary, quantitative sequential, and qualitative diverging). Figure 1b shows an example of an explicit decision tree. An example of an implicit decision tree is the variable syntactics in Figure 1c. This variable syntactics can, in principle, be brought into a tree-like model through a series of if/else statements for a rule-based geographic information system, for instance: **if** data_type == 'ordinal' **and** purpose == 'good effectiveness' **then** visual variable = 'size' **or** 'colour value' **or** 'colour saturation' **or** 'transparency' **or** 'crispness' **or** 'resolution'.

Though explicit and implicit decision trees are produced during research projects, they are often available in a textual and/or pictorial form. That is, they are rarely stored in a machine-readable form, for knowledge exchange across projects. We assert here that this is a missed opportunity for their reuse.

2.2 Visualizing decision trees

Several techniques have been proposed to visualize decision trees. As of this writing, treeviz.net (Schulz, 2011) lists no less than 333 techniques[4]. Filtering by dimensionality shows 67 techniques that cover the visualization of trees in 3D, but none addresses the visualization as an interactive scene aimed for in this work. In addition, filtering these techniques using the keyword 'interactive' returns 17 techniques from which only the Strata Treemap is a 3D technique. Furthermore, several libraries are useful to visualize decision trees. These include notably libraries to understand the parameters of a decision tree trained on a classification task, for example, Scikit-learn (Pedregosa et al., 2011), Matplotlib[5] or Graphviz (Ellson et al., 2004). An example of a visualization of a trained decision tree in 3D can be found in (Mrva et al., 2019). A commonality of these libraries is that they require knowledge of a programming language to generate the visualization. By contrast, the 3D4DT approach does not require such knowledge. The only prerequisite for 3D4DT is the ability to encode a decision tree in JSON.

3. THE 3D4DT APPROACH

As mentioned in Section 1, the representation of decision trees in a machine-readable format and their exploration as an interactive 3D scene are addressed through the 3D4DT approach. This necessitates two steps: i) the choice of an approach to store the decision trees computationally, and ii) the mapping of machine-readable elements to user interface (UI) elements.

As for i), given that a decision tree is a *directed* graph, it can be stored computationally as a *matrix* or as *structured data* (with or without formal semantics). We were interested in an approach that is easy to learn by users and produces results that can be also easily written/read. Hence, we decided against storing as a matrix. In addition, we have not chosen a language with formal semantics because of its relatively steeper learning curve. We could have gone for XML, YAML and JSON. XML is comparatively more verbose than JSON. The distinctive advantages of YAML and JSON are more challenging to pinpoint, but the best summary we have come across is from a StackOverflow user: "For settings/config files, YAML is better. For machine/machine interoperability use JSON. In other words: if your target audience is human, YAML is better. If the target is another program (but you still want the data to be human-readable), use JSON" (Stack Overflow, 2018). A study comparing YAML and JSON observed that 'JSON generation and parsing is faster for all sets of data tested' (Eriksson and Hallberg, 2011). Since parsing is needed to produce 3D interactive scenes, we have opted for JSON at this stage: this seems to provide the best compromise between machine/machine interoperability and human readability/learnability.

As for ii), the creation of the interactive scene necessitates the choice of an appropriate metaphor. Metaphors are useful here because they help understand a (usually abstract) domain in terms of a very different domain (more familiar) of experience, see Lakoff (1992). Using the basic conceptual building blocks presented in (Mandler and Cánovas, 2014), the following statements about the theoretical foundations of the approach can be made: nodes are *containers* and edges are *paths*. Furthermore, Matlock et al. (2014) reported that web users relied on two conceptual metaphors to structure their thinking about the web: "WEB SPACE IS PHYSICAL SPACE" and "OBTAINING INFORMATION IS MOVING THROUGH PHYSICAL SPACE". That is, browsing through nodes and edges to get meaningful options for a given set of constraints can be conceptualized as a *movement* between containers through paths. The translation of these three concepts to UI elements is discussed in Section 3.2.

3.1 A controlled vocabulary for machine-readable descriptions of decision trees

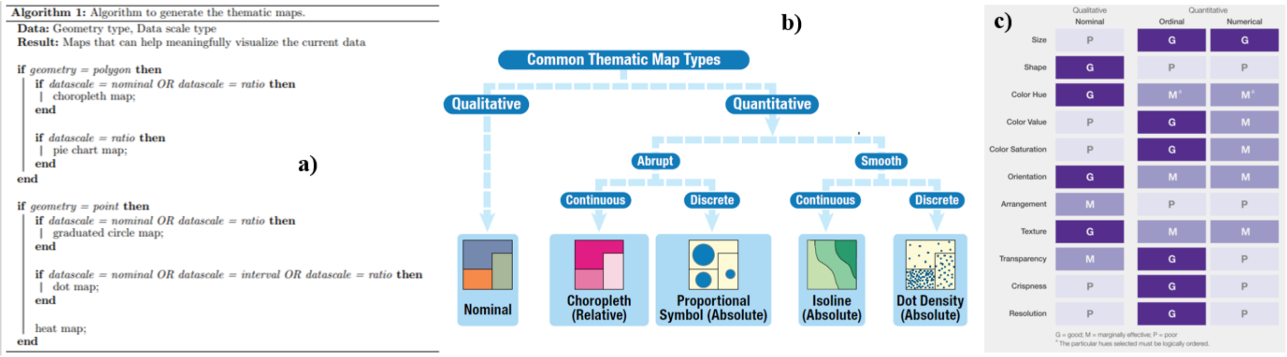
There are several ways of defining controlled vocabulary in the literature. The term is used here after Cardoso (2006) to denote a list of terms that have been enumerated explicitly. Controlled vocabularies limit choices to an agreed-upon, unambiguous and non-redundant set of terms. The main objective of a controlled vocabulary is to prevent users from defining their own terms which can be ambiguous, meaningless, or misspelt.

We were inspired by JSON Graph (JSON Graph Format, 2022) but aimed at keeping things as simple as possible. Hence, some elements of the JSON Graph specification (e.g. hyperedges) were left out in this first version. Our controlled vocabulary for representing decision trees features seven types of entities at the moment. The *graph* is the basic entity. A graph consists of nodes and edges. Furthermore, our critical analysis of the decision trees from Figure 1a,b led to the identification of three types of nodes and three types of edges that are relevant to the conversion of decision trees into a machine-readable format.

Nodes

- root: this is the starting point of the decision tree. A decision tree has only one root.
- leaf: this is a node without children. These are the thematic map types (see Figure 1b for examples).

[4] https://treevis.net/ (accessed: April 06, 2023).
[5] https://matplotlib.org/ (accessed: April 06, 2023).



```
Algorithm 1: Algorithm to generate the thematic maps.
Data: Geometry type, Data scale type
Result: Maps that can help meaningfully visualize the current data

if $geometry = polygon$ then
    if $datascale = nominal$ OR $datascale = ratio$ then
        choropleth map;
    end

    if $datascale = ratio$ then
        pie chart map;
    end
end

if $geometry = point$ then
    if $datascale = nominal$ OR $datascale = ratio$ then
        graduated circle map;
    end

    if $datascale = nominal$ OR $datascale = interval$ OR $datascale = ratio$ then
        dot map;
    end

    heat map;
end
```

	Qualitative	Quantitative	
	Nominal	Ordinal	Numerical
Size	P	G	G
Shape	G	P	P
Color Hue	G	M*	M*
Color Value	P	G	M
Color Saturation	P	G	M
Orientation	G	M	M
Arrangement	M	P	P
Texture	G	M	M
Transparency	M	G	P
Crispness	P	G	P
Resolution	P	G	P

G = good; M = marginally effective; P = poor
* The particular hues selected must be logically ordered.

Figure 1. Three examples of decision trees (DT) from the Cartography literature: a) decision tree of AdaptiveMaps (Degbelo et al., 2020) as pseudocode [implicit DT]; b) decision tree for the choice of thematic map types (Kraak et al., 2020) as a diagram [explicit DT]; c) an example of visual variable syntactics (White, 2017) that can be converted to a decision tree [implicit DT].

- inner-node: this is a node that does not belong to either of the former two categories.

Edges

- parameter: this is an edge that connects a root node and an inner node.
- superclass: this is an edge that connects two inner nodes.
- meaningful: this is an edge that connects an inner node and a leaf. It is useful to express that a thematic map is meaningful for a given set of constraints.

As we shall see, the controlled vocabulary with the set of terms above is enough to describe decision trees so that the machine-readable descriptions can be used to generate interactive 3D scenes. The JSON-based schema of this work also contains a description attribute for every leaf (see Figure 2). The decision trees from Figure 1a,b encoded using the controlled vocabulary, the specification of the JSON schema and the template to get started with the description of further decision trees are available on GitHub (see supplementary material, Section 7).

3.2 Generating 3D Interactive Scenes

The whole decision tree is a scene. The mapping of decision tree elements to UI elements is done as follows:

- root: not mapped to any UI element.
- leaf: these are mapped to *cubes* in the scene (Figure 3). Cubes have the advantage that pictures can be easily imported and are displayed in suitable proportions. The image just needs to be in 1:1 format (aspect ratio) and it can be used for every side of the cube. A mapping to spheres was implemented at first and this showed that images in a square format are distorted, which reduces the clarity of the presentation. Hence, we decided to stick with cubes.
- inner-node: these are mapped to *radio buttons* in the scene or *sections with headlines* for the radio buttons (Figure 3).
- parameter: is useful to create sections as HTML divs. We create one HTML div per parameter.
- superclass: are used to generate hierarchical dependencies between the radio buttons. Each target node of a superclass edge is mapped to a radio button. All radio buttons are grouped in the corresponding div for a parameter.
- meaningful: are useful to specify when (i.e. the conditions that must be fulfilled by the radio buttons) for a leaf to be highlighted.

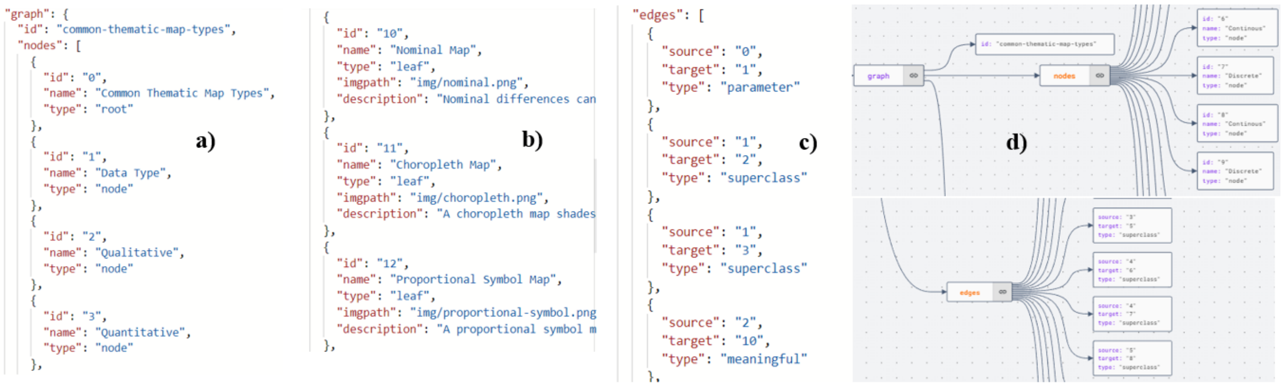
As mentioned above, nodes are containers, edges are paths and browsing through these elements is a movement, from the viewpoint of metaphor theory. Regarding the translation of these theoretical concepts to UI concepts, nodes (leaves and inner-nodes) are mapped to three types of *UI containers* (cubes, radio buttons, sections with headlines); the first two types of edges (parameter, superclass) are mapped to *menu elements for the navigation* (paths, see Figure 3, top left); the last type of edge (meaningful) specifies the cubes the user should navigate to next, based on their input condition, and hence *directs to the user to the next meaningful destination during their movement*.

An important interaction feature is the *highlighting* of the meaningful thematic map type, given the constraints selected by users. There are several approaches to realize emphasis in information visualization (see Hall et al. (2016); Robinson (2011) for a review). Here, we have used a combined variation of the visual variables of size and colour hue to achieve the desired effect. To distinctly highlight an object, its size is enlarged to more than the double size of the not highlighted ones. Also, the name of the object, displayed as text geometry above the cube, is enlarged and the colour is changed from grey to black (Figure 3, bottom left). The highlighting happens in real-time, which means that as soon as a change is made to the radio buttons, it is applied directly to the scene.

3.3 Prototype

The prototype is available as a web-based application on GitHub. The server is run using *Node.js*. To speed up the development of the frontend, we have used *Vitejs*. The 3D interactive scene is implemented using the JavaScript library *Three.js*. The choice of Three.js was motivated by the fact that it is 1) open source, 2) expressive enough to create a variety of 3D scenes in the browser (see examples at https://threejs.org/examples/), and 3) is actively maintained by a community of contributors.

The website consists of four areas. The first is the *decision tree selection area*: this is where the user can choose a decision tree to be visualized (the selection area is not shown in Figure 3). As of this writing, the selection area provides two decision trees from Figure 1 as examples to start with: (Kraak et al., 2020) and (Degbelo et al., 2020). The user is also given the possibility of uploading additional decision trees, using the JSON-based



a)

```
"graph": {
  "id": "common-thematic-map-types",
  "nodes": [
    {
      "id": "0",
      "name": "Common Thematic Map Types",
      "type": "root"
    },
    {
      "id": "1",
      "name": "Data Type",
      "type": "node"
    },
    {
      "id": "2",
      "name": "Qualitative",
      "type": "node"
    },
    {
      "id": "3",
      "name": "Quantitative",
      "type": "node"
    },
```

b)

```
    {
      "id": "10",
      "name": "Nominal Map",
      "type": "leaf",
      "imgpath": "img/nominal.png",
      "description": "Nominal differences can
    },
    {
      "id": "11",
      "name": "Choropleth Map",
      "type": "leaf",
      "imgpath": "img/choropleth.png",
      "description": "A choropleth map shades
    },
    {
      "id": "12",
      "name": "Proportional Symbol Map",
      "type": "leaf",
      "imgpath": "img/proportional-symbol.png
      "description": "A proportional symbol m
    },
```

c)

```
  "edges": [
    {
      "source": "0",
      "target": "1",
      "type": "parameter"
    },
    {
      "source": "1",
      "target": "2",
      "type": "superclass"
    },
    {
      "source": "1",
      "target": "3",
      "type": "superclass"
    },
    {
      "source": "2",
      "target": "10",
      "type": "meaningful"
    },
```

d)
graph
id: "common-thematic-map-types"
nodes
id: "6" name: "Continous" type: "node"
id: "7" name: "Discrete" type: "node"
id: "8" name: "Continous" type: "node"
id: "9" name: "Discrete" type: "node"
edges
source: "3" target: "5" type: "superclass"
source: "4" target: "6" type: "superclass"
source: "4" target: "7" type: "superclass"
source: "5" target: "8" type: "superclass"

Figure 2. Encoding of the decision tree from (Kraak et al., 2020) using the controlled vocabulary and the JSON format: a) examples of nodes; b) examples of leaves (including their description and images to use for their preview); c) examples of edges; d) the choice of JSON enables the use of existing visualization tools for JSON. The screenshots were created using https://jsoncrack.com/.

format introduced in Section 3.1. The second area of the website is the *filtering panel* where radio buttons can be selected to navigate the decision tree. This area is displayed after the user selects a decision tree. The third area shows the *interactive 3D scene*, which visualizes leaves from the decision tree. The leaves are converted into cubes arranged in an arc from the camera's starting position. *Three.js* offers the possibility of creating a PerspectiveCamera (projection mode that mimics the way the human eye sees) or an OrthographicCamera (that enables rendering in a way that an object's size stays constant regardless of its distance from the camera). The interactive 3D scene uses a perspective camera. The camera can be moved by using the "W"-, "A"-, "S"- and "D"-keys of the keyboard, as is common in interactive games. The fourth area comprises the descriptions of the leaves. The fourth area is useful for *additional descriptions* of a leaf that is currently highlighted. This is the area that provides introductory information to 'Choropleth Map' in Figure 3. It is only displayed if the user moves close to an object in the scene and enables the display of the description of the closest leaf. This can be done by clicking the key "E" on the keyboard. A function to determine closeness to a cube was implemented. It checks the current distance of the camera to all objects. If that distance is lower than a given threshold, the possibility to display additional information about a cube is shown to the user. Figure 3 shows a screenshot of the prototype.

4. EVALUATION

Since the approach touches on the representational and communicative aspects of decision trees, the evaluation uses two criteria, one for each aspect: expressiveness (representation) and usability (communication).

As for expressiveness, the controlled vocabulary presented in Section 3.1 was used to describe all three decision trees introduced in Figure 1. This demonstrates the pertinence of the approach to decision trees of the Cartography literature in this form. Decision trees a) and b) could be encoded using the vocabulary right away. As for decision tree c), we transformed it first into a pseudocode (available as supplementary material, Section 7) before encoding it in JSON using the controlled vocabulary. The transformation does not happen without loss of information though. Indeed, the meaningfulness of an option (expressed at the moment through the *meaningful* edge) is *binary* in the case of decision trees a) and b). For decision tree c), the meaningfulness is *graded* (poor/marginal/good). We discuss options to incorporate graded meaningfulness in Section 6.

As for usability, the prototype was evaluated through a lab-based study, with the aim of learning about the advantages and disadvantages of the interactive 3D scene in comparison to a presentation as text+images. The rest of this section presents background information about the user study and the results.

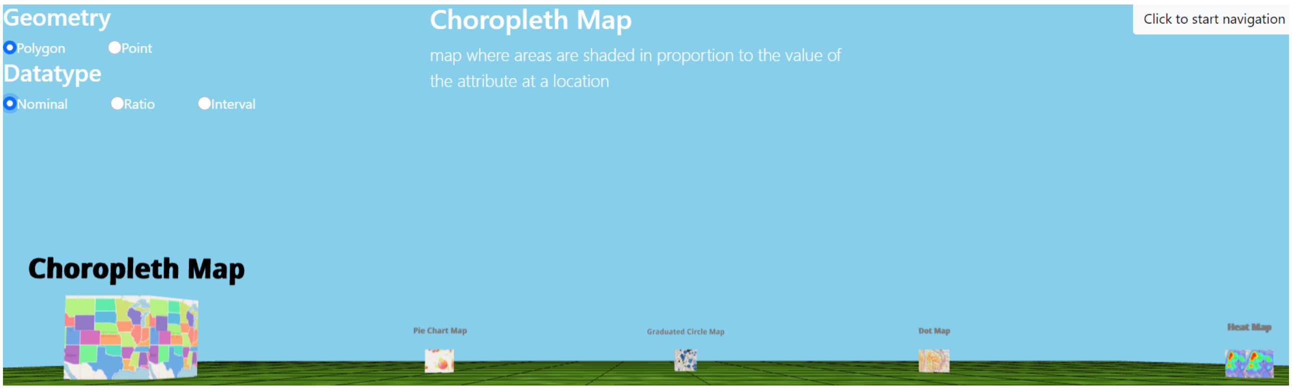
4.1 Variables and study design

The independent variables of the study were the two different decision trees (extracted from Kraak et al. (2020) and Degbelo et al. (2020)) and the two formats for communicating these trees to users: interactive three-dimensional scene (henceforth called the 3D condition) and the use of text and picture (henceforth called the traditional condition). The dependent variables of the user study were effectiveness (number of correct answers), efficiency (time taken to solve tasks), and the explicit memorability of the transferred knowledge. In line with Stusak et al. (2015), explicit memorization denotes the fact that the users were told to memorize the facts because they will be asked about them again later. This stands in contrast to implicit memorization when this warning is not given (Stusak et al., 2015).

The study followed a within-group design. That is, the participants were exposed to both the 3D and traditional conditions. The order of the conditions was counterbalanced, and so was the order of the decision trees used within these conditions. This resulted in four different groups of participants. Henceforth, the decision tree for the "adaptive maps algorithm" from (Degbelo et al., 2020) is referred to as DecisionTree A. The "common thematic map types"-decision tree from (Kraak et al., 2020) is referred to as DecisionTree B. Group 1 first worked through decision tree A in the interactive format and then example B in the traditional format. Group 2 had the same order of the examples, but the formats were interchanged. Group 3 processed example B in the traditional format before working on example A in the interactive format. The order of examples was the same for Group 4, however, the formats were exchanged. The allocation of the participants to the groups was determined by a random number generator.

4.2 Procedure

The study was created via LimeSurvey and executed remotely. The users received a link to a Zoom meeting via e-mail along with the consent form. After joining the meeting and reading and accepting the consent form, they received another e-mail with the link to the survey and were asked to share their screen.



Geometry
◉ Polygon ○ Point
Datatype
◉ Nominal ○ Ratio ○ Interval

Choropleth Map
map where areas are shaded in proportion to the value of the attribute at a location

Figure 3. A screenshot of the prototype. When filtering options are selected (top left), the meaningful maps for these options are highlighted to users (bottom left). Users can also display text messages about meaningful maps for their filtering options (top middle).

To ensure the correct execution and order, and the understanding of the controls, each participant stayed in the Zoom meeting with the researcher and shared their screen during the whole study. The first questionnaire of the study was about demographic questions. Afterwards, they started with the main tasks.

The user was told to open the website via a hyperlink in the survey. Then the researcher told them the format to choose, either the “interactive” or “traditional”, determined by the randomly assigned group. (If the choice was the interactive format, the controls of the camera were explained and practised shortly with the user.) For both formats, the participant was informed that the second questionnaire can be answered while exploring the selected decision tree (*effectiveness task*). These were nine questions, which could be answered with ‘yes’, ‘no’ or ‘not applicable’ (see supplementary material, Section 7) and were useful to assess the effectiveness of a condition. The participant was also informed that they need to answer questions about what they remembered from the website afterwards, with the website closed. They were not informed that time was measured. After completing these tasks, they closed the website and were then handed the third and last questionnaire (*memorability task*). This third questionnaire was used to assess what participants just recalled. The users were handed over five/six questions depending on the decision tree interacted with. The questions were a mix of questions with the forms “please write down everything you remember about...” and “what do you remember about...”. After completing the tasks for the first condition, the participants went again through the effectiveness task and the memorability task, with the second condition and a different decision tree. The study was approved by the institutional ethics board and pilot-tested, after which the wording of some questions was adjusted.

4.3 Participants

12 participants were recruited via personal messages and e-mail, three of these were female and nine were male. The average age was between 25 and 26 and the ages ranged from 23 to 28. Three participants had a high-school diploma as the highest degree of education, three have completed their apprenticeship and the other six have achieved a bachelor’s degree. Six participants stated to have no experience at all in the field of Geoinformatics, four claimed to be slightly experienced and two considered themselves very experienced. None of the participants had read any of the literature which was used for the examples (Kraak et al., 2020; Degbelo et al., 2020). Furthermore, a moderate level of English was required, as well as a desktop computer or laptop with access to the Internet.

4.4 Results

Effectiveness was assessed by the number of right answers in the questionnaires filled out during the interaction with the information. Efficiency was assessed by the time the participants used to answer these and memorability by the questions answered after closing the website. Since the questions to assess memorability required answers in free text, each answer was evaluated using keywords. That is, for each answer it was checked if different keywords were mentioned. Similar expressions (for example point map instead of dot map, were also accepted). The same applied to misspelt, but clearly assignable names and meaningful descriptions of the maps. False statements did not give a point deduction. The results are shown in Figure 4.

Efficiency. The average efficiency values observed for the 3D condition were: DecisionTree A: 480 seconds (sd: 276s); DecisionTree B: 587 seconds (sd: 243s). For the traditional condition, the average values obtained were: DecisionTree A: 436 seconds (sd: 164s); DecisionTree B: 392 seconds (sd: 198s). The efficiency values averaged over the two decision trees were about 534 seconds for the interactive 3D condition and about 414 seconds for the traditional condition. That is, completing the tasks using the traditional condition took about 2 minutes less than completing them using the 3D condition. None of these differences was statistically significant though.

Effectiveness. The average effectiveness values observed for the 3D condition were: DecisionTree A: 78% (sd: 27%); DecisionTree B: 100% (sd: 0%). For the traditional condition, the average effectiveness values obtained were: DecisionTree A: 93% (sd: 8%); DecisionTree B: 70% (sd:24 %). The effectiveness values averaged over the two decision trees were 89% for the 3D condition and 82% for the traditional condition respectively. Though the differences across the two decision trees were not statistically significant, users performed significantly better when using the 3D interactive scenes instead of picture+text when answering questions about DecisionTree B.

Memorability. The average memorability values observed for the 3D condition were: DecisionTree A: 40% (sd: 11%); DecisionTree B: 44% (sd: 30%). For the traditional condition, the average values obtained were: DecisionTree A: 51% (sd: 27%); DecisionTree B: 24% (sd: 22%). The effectiveness values averaged over the two decision trees were 42% for the 3D condition and 38% for the traditional condition. That is, answers given by participants about facts they recalled from their interaction with the two conditions were, on average, more accurate in the

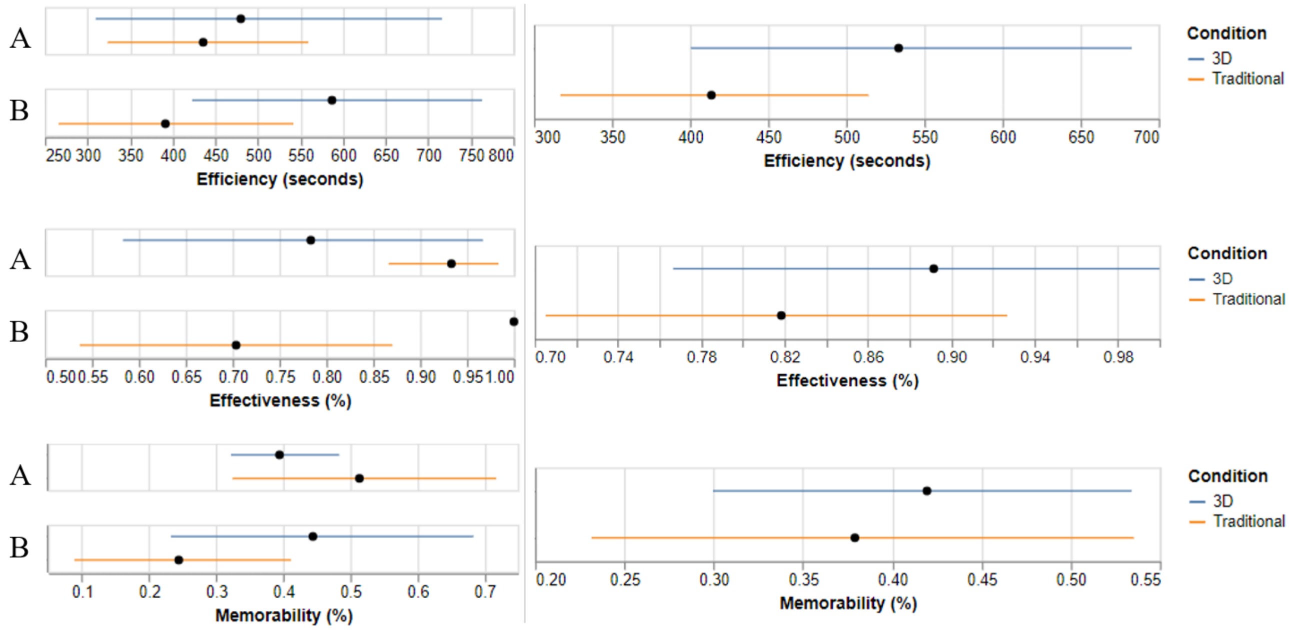


Figure 4. Interactive 3D vs. picture+text. Left: performance per decision tree; right: performance averaged for the two decision trees.

3D condition by 4%. This difference was also not statistically significant. Finally, there was a drop in accuracy of about 45% (Confidence interval: [35% - 55%]) between the answers users gave during their interaction with a condition (3D or traditional) and the answers they gave recalling their interaction. The drops in accuracy were 47% for the 3D condition and 44% for the traditional condition, but these were not statistically significant.

Participants' background. We also checked if the background of the participant (gender, education, experience) had an impact on the results. The differences were significant in two cases: efficiency (participants who reported being very experienced in Geoinformatics were significantly faster in the traditional condition) and memorability (the group who reported having completed their apprenticeship has a significantly higher memorability score in the 3D condition: 39% vs 8%). Other background characteristics did not have a significant impact on the results.

5. DISCUSSION

The evaluation did not aim to answer the question whether or not the 3D condition is better/worse than the picture+text condition. Instead, results from previous work on learning with 3D models (Keehner et al., 2004; Huk, 2006; Korakakis et al., 2009) suggest that it may be more interesting to find out *when* it is better/worse and *for whom*. As for the when, decision tree B was less complex than decision tree A because the hierarchical structure between its elements (displayed as radio buttons) was easier to recognize. The graph density values were 0.13 and 0.07 for decision trees A and B respectively. We conjecture that 3D4DT could lead to effectiveness gains for (sparse) decision trees with simple hierarchical structures for the parameters, similar to Figure 1b. As for the whom, despite a few dissimilarities, the differences observed are not conclusive at this point overall and need further investigation. For now, a safe takeaway is that an interactive 3D scene could be used as a complementary means to help users understand how thematic maps are created, especially when designers wish to convey this information most accurately. We were however surprised by how little users could remember, immediately after the interaction with the trees, in both conditions.

5.1 Implications

Representational aspects. The key value of the 3D4DT approach is to foster the reuse of decision trees for geosoftware design. In particular, a key notion of map/geovisualization reuse across contexts is transferability (see e.g. Griffin et al. (2017)). Because the 3D4DT approach relies on storage in a machine-readable format, it contributes to that goal. For instance, we have shown in Figure 2 that storing the decision trees in JSON enables their reuse not only to create interactive 3D scenes but also as input for existing tools for the visualization of decision trees as graphs. Thus, the 3D4DT is a 'store once, use many times' approach (e.g. for the creation of a 2D visualization, and 3D visualization, or visualization in virtual reality, visualization on multiple devices), in line with the vision of 'map plasticity' (Kray and Degbelo, 2019).

Communication aspects. Decision Tree A is the 'brain' of the AdaptiveMaps prototype that helps create web maps semi-automatically (Degbelo et al., 2020). Hence, by helping users visually explore that decision tree, the 3D4DT approach offers one way of realizing the requirement of 'algorithmic transparency' for intelligent geovisualizations from (Degbelo and Kray, 2018). In addition, as Gahegan (2005) pointed out in his call for visual support for the entire process of GIScience: "Although it is still difficult to share data between researchers and systems, it is even more difficult to share functionality. [...] As a result, a lot of research funding and effort results in outcomes that have reduced practical impact". Through the encapsulation of the conceptual structure underlying a software system into a decision tree, the encoding/sharing of that structure as machine-readable data, and the visualization of that conceptual structure as an interactive scene, the 3D4DT approach is one step in the direction of *functionality sharing across systems*. Hence, it can support the reuse of systems for geovisualization creation and contribute to the vision, outlined in (Degbelo, 2022), of reusable geovisualizations across multiple datasets and scenarios. Furthermore, the work examined memorability and one of the facets of memorability is the storage of information in short-term memory or long-term memory (see Camina and Güell (2017)). The fact that users were able to correctly answer questions related to the trees during the interaction, but their effectiveness at doing so dropped substantially after the interaction is an indication that the information conveyed was stored in the short-term memory quite well, but not in the long-term memory. Since both conditions were equally affected, this calls for further investigations that explore *designing for memorability* during the exploration of the decision trees.

5.2 Limitations

There are limitations related to the implementation. The mapping from the inner nodes to radio buttons is an easy and intuitive way to display the inner nodes. However, these inner nodes can also be mapped to the scene itself for more interactivity (this could have induced a more cluttered interactive scene though). Another possible improvement of the prototype would be to hide radio buttons that are currently disabled, instead of just showing them as disabled. The arrangement of the radio buttons could also be changed to mimic a tree visualisation, to communicate in a more direct way that users are interacting with a decision tree. Finally, the prototype supports the visualization of two decision trees extracted from the literature (Figures 1a,b) because these were readily available. While we anticipate that the JSON-based modelling language suggested is generic enough to model decision trees similar to these, more examples are needed to confirm this empirically.

The user study also had some limitations, notably the relatively small sample size and the relatively homogeneous group of participants. Furthermore, since the study was executed remotely on different devices, the differences in screens (number and size of the screens used) and input devices (e.g. touchpad and mouse) could have affected the results (especially the interactive 3D condition), but these were not controlled for explicitly.

6. CONCLUSION AND FUTURE WORK

The representation of decision trees in a machine-readable format and their exploration as an interactive 3D scene were addressed in this work through the 3D4DT approach. The results obtained from testing the approach on exemplar decision trees suggest that it is promising. There are several directions in which it could be extended in future work.

Representational aspects. As mentioned in Section 4, a useful extension to the current work could be a means of modelling graded (instead of binary) meaningfulness. A simple option could be to remove the current 'meaningful' edge type and replace it with new types of edges instead. For the example in Figure 1c, this would mean three new edges: 'poor_effectiveness', 'marginal_effectiveness' and 'good_effectiveness'. Realizing this could happen through the metadata property of the JSON Graph specification (e.g. {"metadata": { "type": "poor_effectiveness", "super-type": "effectiveness"})}. Another option could be the use of hyperedges (JSON Graph Format, 2022). Another direction for future work is the evaluation of the learnability of the approach, especially for non-computer scientists (e.g. how long do they need to accurately translate their own decision trees in machine-readable formats using the controlled vocabulary?).

Communication aspects. The evaluation could focus on identifying the upper bound of the approach (how many tree elements maximum until cluttering becomes unbearable) and learning about the scalability of the approach (how does user experience evolve as a function of the number of nodes/edges/leaves). Controlling more systematically for characteristics that could impact the ease of manipulation of the 3D interactive scenes (e.g. experience with computer games/virtual reality, size of the device) to learn about user groups that benefit the most/least from it, is also an interesting direction for future work.

Beyond thematic cartography and map creation. It would be interesting to explore how the lessons learned about the exploration of decision trees for meaningful map creation could transfer to other areas of GIScience, notably the exploration of decision trees for spatial analysis (e.g. meaningful spatial prediction and aggregation discussed in (Stasch et al., 2014)).

7. DATA AND SOFTWARE AVAILABILITY

The data collected during the user study, the analysis script and all questionnaires are available at https://doi.org/10.6084/m9.figshare.21723512. The code of the prototype, the JSON schemas, and the encoding of the decision trees as JSON are available at https://github.com/aurioldegbelo/3D4DT.

ACKNOWLEDGMENTS

Auriol Degbelo was funded by the German Research Foundation through the project NFDI4Earth (DFG project no. 460036893, https://www.nfdi4earth.de/) within the German National Research Data Infrastructure (NFDI, https://www.nfdi.de/).

REFERENCES

Andrienko, G., Andrienko, N., 1999. Interactive maps for visual data exploration. *International Journal of Geographical Information Science*, 13(4), 355–374.

Bernard, L., Braesicke, P., Bertelmann, R., Frickenhaus, S., Gödde, H., Keßler, C., Lorenz, S., Mahecha, M., Marschall, H., Hezel, D., Nagel, W. E., Reichstein, M., Sester, M., Thiemann, H., Weiland, C., Wytzisk-Arens, A., 2021. NFDI Consortium Earth System Sciences - Proposal 2020 revised. https://zenodo.org/record/5718944.

Bostock, M., Ogievetsky, V., Heer, J., 2011. D³ Data-Driven Documents. *IEEE TVCG*, 17(12), 2301–2309.

Camina, E., Güell, F., 2017. The neuroanatomical, neurophysiological and psychological basis of memory: current models and their origins. *Frontiers in Pharmacology*, 8.

Cardoso, J., 2006. Approaches to developing semantic web services. *International Journal of Computer Science (IJCS)*, 1(1).

Degbelo, A., 2022. FAIR geovisualizations: definitions, challenges, and the road ahead. *International Journal of Geographical Information Science*, 36(6), 1059–1099.

Degbelo, A., Kray, C., 2018. Intelligent geovisualizations for open government data (vision paper). F. Banaei-Kashani, E. G. Hoel, R. H. Güting, R. Tamassia, L. Xiong (eds), *ACM SIGSPATIAL 2018*, ACM Press, Seattle, Washington, USA, 77–80.

Degbelo, A., Sarfraz, S., Kray, C., 2020. Data scale as Cartography: a semi-automatic approach for thematic web map creation. *Cartography and Geographic Information Science*, 47(2), 153–170.

Ellson, J., Gansner, E. R., Koutsofios, E., North, S. C., Woodhull, G., 2004. Graphviz and Dynagraph - static and dynamic graph drawing tools. M. Jünger, P. Mutzel (eds), *Graph drawing software*, Springer, 127–148.

Eriksson, M., Hallberg, V., 2011. Comparison between JSON and YAML for data serialization. Bachelor's Thesis, The School of Computer Science and Engineering, KTH.

Gahegan, M., 2005. Beyond tools: visual support for the entire process of GIScience. J. Dykes, A. M. Maceachren, M.-J. Kraak (eds), *Exploring Geovisualization*, Elsevier, 83–99.

Gong, X., 2019. SDG Viz: A web-based system for visualizing sustainable development indicators. *Proceedings of the ICA*, 2, 1–8.

Griffin, A. L., White, T., Fish, C., Tomio, B., Huang, H., Sluter, C. R., Bravo, J. V. M., Fabrikant, S. I., Bleisch, S., Yamada, M., Picanço, P., 2017. Designing across map use contexts: a research agenda. *International Journal of Cartography*, 90–114.

Hall, K. W., Perin, C., Kusalik, P. G., Gutwin, C., Carpendale, S., 2016. Formalizing emphasis in information visualization. *Computer Graphics Forum*, 35(3), 717–737.

Hardisty, F., Robinson, A. C., 2011. The GeoViz toolkit: using component-oriented coordination methods for geographic visualization and analysis. *International Journal of Geographical Information Science*, 25(2), 191–210.

Huk, T., 2006. Who benefits from learning with 3D models? the case of spatial ability: 3D-models and spatial ability. *Journal of Computer Assisted Learning*, 22(6), 392–404.

JSON Graph Format, 2022. JSON Graph Format Specification Website. accessed: November 26, 2022.

Keehner, M., Montello, D. R., Hegarty, M., Cohen, C., 2004. Effects of interactivity and spatial ability on the comprehension of spatial relations in a 3D computer visualization. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 26.

Korakakis, G., Pavlatou, E., Palyvos, J., Spyrellis, N., 2009. 3D visualization types in multimedia applications for science learning: A case study for 8th grade students in Greece. *Computers & Education*, 52(2), 390–401.

Kraak, M.-J., Roth, R. E., Ricker, B., Kagawa, A., Sourd, G. L., 2020. *Mapping for a sustainable world*. The United Nations.

Kray, C., Degbelo, A., 2019. Map plasticity. *Poster Session at COSIT'19*, Regensburg, Germany.

Lakoff, G., 1992. The contemporary theory of metaphor. A. Ortony (ed.), *Metaphor and Thought*, Cambridge University Press, 202–251.

Ledermann, F., Gartner, G., 2015. Mapmap. js: a data-driven web mapping API for thematic cartography. *Brazilian Journal of Cartography*, 67(5), 1043–1053.

MacEachren, A. M., 1995. *How maps work: representation, visualization, and design*. Guilford Press, New York, USA.

Mandler, J., Cánovas, C., 2014. On defining image schemas. *Language and Cognition*, 6(4), 510–532.

Matlock, T., Castro, S. C., Fleming, M., Gann, T. M., Maglio, P. P., 2014. Spatial metaphors of web use. *Spatial Cognition & Computation*, 14(4), 306–320.

Mrva, J., Neupauer, S., Hudec, L., Sevcech, J., Kapec, P., 2019. Decision support in medical data using 3D decision tree visualisation. *Proceedings of EHB 2019*, IEEE, Iasi, Romania, 1–4.

Nagel, T., Klerkx, J., Vande Moere, A., Duval, E., 2013. Unfolding – A library for interactive maps. A. Holzinger, M. Ziefle, M. Hitz, M. Debevc (eds), *Proceedings of SouthCHI 2013*, Maribor, Slovenia, 497–513.

Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., Seifert, C., 2023. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *ACM Computing Surveys*, 3583558.

Nunes, I., Jannach, D., 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction*, 27(3-5), 393–444.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85), 2825–2830.

Poorthuis, A., Van der Zee, L., Guo, G., Keong, J. H., Dy, B., 2020. Florence: a Web-based Grammar of Graphics for Making Maps and Learning Cartography. *Cartographic Perspectives*.

Robinson, A. C., 2011. Highlighting in geovisualization. *Cartography and Geographic Information Science*, 38(4), 373–383.

Roth, R. E., 2017. Visual variables. D. Richardson, N. Castree, M. F. Goodchild, A. Kobayashi, W. Liu, R. A. Marston (eds), *International Encyclopedia of Geography: People, the Earth, Environment and Technology*, John Wiley & Sons, Ltd, 1–11.

Satyanarayan, A., Moritz, D., Wongsuphasawat, K., Heer, J., 2017. Vega-Lite: A grammar of interactive graphics. *IEEE TVCG*, 23(1), 341–350.

Schulz, H.-J., 2011. Treevis.net: A tree visualization reference. *IEEE Computer Graphics and Applications*, 31(6), 11–15.

Stack Overflow, 2018. What is the difference between YAML and JSON? accessed: November 26, 2022.

Stasch, C., Scheider, S., Pebesma, E., Kuhn, W., 2014. Meaningful spatial prediction and aggregation. *Environmental Modelling & Software*, 51, 149–165.

Stusak, S., Schwarz, J., Butz, A., 2015. Evaluating the memorability of physical visualizations. B. Begole, J. Kim, K. Inkpen, W. Woo (eds), *Proceedings of CHI 2015*, 3247–3250.

Van Ho, Q., Lundblad, P., Astrom, T., Jern, M., 2012. A web-enabled visualization toolkit for geovisual analytics. *Information Visualization*, 11(1), 22–42.

White, T., 2017. Symbolization and the visual variables. J. P. Wilson (ed.), *Geographic Information Science & Technology Body of Knowledge*. Issue: Q2.