Applying the Qualitative Model for Reasoning about 3D perspectives (Q3D) to solve a Spatial Test about Rotations*

Marti Ejarque , Zoe Falomir
Umeå University, Sweden
{marti.ejarque,zoe.falomir}@umu.se

Abstract

This paper explores how qualitative spatial reasoning can be used to solve Shepard–Metzler-style rotation problems. We combine two qualitative models: a qualitative 3D perspective model (Q3D), which represents objects through symbolic depth descriptions from multiple orthographic views, and a qualitative rotation model (QOR), which captures how these descriptions transform under rotations in three-dimensional space. This makes it possible to compare two block-based objects by determining whether one can be transformed into the other through a valid sequence of qualitative rotations. Together, the two models provide a symbolic framework for representing spatial structure, reasoning across perspectives, and identifying object equivalence under rotation, while also making explicit the qualitative transformations involved in mental rotation.

1 Introduction

Qualitative Spatial and Temporal Representations and Reasoning (QSTR) [Cohn and Renz, 2007; Ligozat, 2011; Sioutis and Wolter, 2021] provides symbolic models for reasoning about spatial and temporal information when precise metric data are unavailable, unnecessary, or cognitively less appropriate. Instead of representing space through numerical coordinates, qualitative models describe relations such as topology, direction, orientation, proximity, intersection and relative position. These representations have been successfully applied in several areas, including robotics [Mast *et al.*, 2016; Falomir *et al.*, 2011, 2013], computer vision [Suchan, 2022], ambient intelligence [Falomir and Olteţeanu, 2015], architecture and design [Li and Schultz, 2024; Bhatt and Freksa, 2015], geographic information systems [Wei *et al.*, 2024; Clementini and Cohn, 2024; Moratz *et al.*, 2025], safety analysis [Grigoleit *et al.*, 2017], autonomous driving [Monsen *et al.*, 2025] and education [Kragten *et al.*, 2024; Santamarta *et al.*, 2023]. The most relevant calculi in QSR was reported by Dylla *et al.* [2017].

A relevant feature of qualitative spatial representations is that they are close to the way humans often describe and reason about space. They make it possible to abstract away from exact measurements, while preserving the spatial distinctions that are important for the task. This connection with human spatial cognition has motivated the use of qualitative models in sketch recognition [Lovett *et al.*, 2006], visual problem solving [Lovett and Forbus, 2011], paper folding reasoning tests [Falomir, 2016; Falomir *et al.*, 2021], analogy in Raven's Progressive Matrices [Lovett and Forbus, 2017], logic composition of qualitative shapes when solving cognitive tests [Falomir *et al.*, 2020; Pich and Falomir, 2018], modeling 2D spatial abstraction during mental rotation tasks [Lovett and Schultheis, 2014], and reasoning about support and stability in educative games [Jaiswal and Falomir, 2026].

This paper integrates the Qualitative 3D depth model (Q3D) [Ejarque and Falomir, 2026; Falomir, 2015; Falomir and Oliver, 2016] with the Qualitative Object Rotation (QOR) model Falomir and Costa [2025]; Falomir [2026]; Falomir *et al.* [2026] in order to infer rotations in 3D objects similar to those presented by Shepard-Metzler metal rotation tasks [Shepard and Metzler, 1971]. The Q3D model represents an object through symbolic depth matrices associated with orthographic views such as `Front`, `Right`, and `Up`. In this paper, we use complete six-face Q3D states, including `Front`, `Back`, `Left`, `Right`, `Up`, and `Down`, so that rotations can be applied consistently even when some faces are not visible in the test item.

The contribution of this paper is threefold. First, it shows how Q3D depth descriptions can be transformed under qualitative rotation while remaining in the same symbolic representation. Second, it formulates rotation identification as a finite graph-search problem over complete Q3D states, while allowing object matching through partial visible views. Third, it provides an interpretable output in the form of a shortest qualitative rotation path, which can be understood as a symbolic explanation of the mental rotation.

The rest of the paper is organized as follows. Section 2 introduces the Shepard-Metzler mental rotation task and reformulates it as a qualitative transformation problem. Section 3.1 presents the Q3D depth model used to describe three-dimensional objects through symbolic depth matrices. Section 4 describes the qualitative rotation operators and the graph-based method for identifying rotated objects from depth matrices. Section 5 reports the results of the rotation generation and sequence recovery experiments. Finally, Section 6 discusses limitations and future work.

2 The problem: the Shepard and Metzler test

The Shepard-Metzler mental rotation test [Shepard and Metzler, 1971] is a classical spatial reasoning task in which participants compare two three-dimensional objects and decide whether they depict the same object under rotation or correspond to a different one. The objects are usually composed of connected cube-like blocks, which allow for simplicity but also non-trivial configurations. The task requires reasoning about the transformations that preserve the structure of the object.

Mental rotation is especially relevant because it requires identifying whether two perceived objects correspond to the same

*Correspondence to: Marti Ejarque. E-mail:marti.ejarque@umu.se

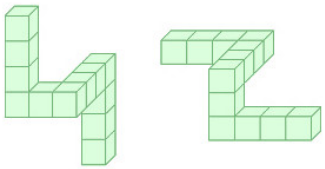


Figure 1: Shepard–Metzler test item: original object on the left and transformed candidate on the right. (Image by Jennifer Oneske, Creative Commons CC0 License).

three-dimensional structure under a change of orientation.

From the perspective of qualitative spatial reasoning, the task can be reformulated as: Given two qualitative descriptions of three-dimensional objects, we ask whether there exists a sequence of valid rotations that transforms the first description into the second.

$$O_1 \equiv O_2 \quad \text{iff} \quad \exists \rho_1, \ldots, \rho_n \text{ such that } \rho_n(\cdots \rho_1(O_1)) = O_2.$$

Here, each ρ_i denotes a qualitative rotation operator. The central problem is therefore to represent the object in a form that supports rotation, comparison, and explanation.

This paper focuses on Shepard-Metzler-style objects represented through Q3D depth descriptions. Instead of using metric coordinates or rendered images, each object is encoded by qualitative depth matrices associated with its canonical views. This allows the mental rotation problem to be treated as a symbolic transformation problem: the model generates rotated Q3D states and checks whether one of them matches the candidate object. This formulation makes explicit which spatial transformations are involved in solving a rotation item. Rather than returning only a binary answer, the model can also return the sequence of qualitative rotations that connects the reference object to the candidate.

In the following sections, we use this task as a test case for combining Q3D object descriptions with qualitative rotation operators. The goal is not to simulate continuous mental imagery, but to show that a symbolic qualitative model can represent the relevant spatial structure, transform it under rotation, and identify object equivalence through an interpretable rotation path.

3 Preliminaries

3.1 The Qualitative 3D depth model (Q3D)

When reasoning about real objects in space, humans naturally think in terms of three-dimensional volumes and multiple perspectives. The Q3D model [Falomir, 2015; Falomir and Oliver, 2016] formalises this intuition by encoding orthographic projections through symbolic depth values rather than numeric measurements. The Q3D depth Descriptor of an object is defined by three views:

$$Q3D_{RS} = \{F, R, U \in View \mid Q3Dmatrix \subseteq N_{depths}\}$$

$$N_{depths} = \{a, b, c, d, \cdots, *\}$$

such that,

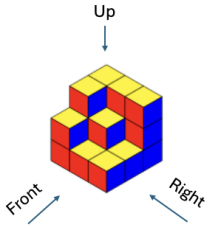
$$view(F, Object, H, W, Q3Dmatrix(H, W)),$$
$$view(R, Object, H, L, Q3Dmatrix(H, L)),$$
$$view(U, Object, L, W, Q3Dmatrix(L, W)).$$

where F, R, and U denote the `Front`, `Right`, and `Up` perspectives, while H, W, and L represent the height (H), width (W), and length (L) of the object's bounding box. Each matrix entry belongs to the symbolic set:

$$Q3Dmatrix(D_1, D_2) \subseteq \{a, b, c, d, \ldots, *\}.$$

where D_1 and D_2 are the dimensions of the matrix (i.e. pairs of 3D dimensions: H, W and H, L and L, W) and the elements of each matrix describe ordinal depth layers composed of equally sized volumetric units: a denotes the surface of the object, b the first level of depth (one unit removed), c the second level of depth (two units of volume removed), d the third level of depth (three units of volume removed) and so on. The symbol $*$ indicates that all volumes along a depth column have been removed, so there is a hole. These symbols encode relative ordering rather than metric distance.

Figure 2 presents an object described by the Q3D where volumes of a specific size produced a $3 \times 3 \times 3$ grid with volumetric features for the `Front`, `Right`, and `Up` sides. Note that another volumetric size could have been used producing other dimensions for the object descriptor (e.g. $6 \times 6 \times 6$).



Q3D descriptor for this 3x3x3 object:

Front	Right	Up
$[b,c,c]$	$[*,c,a]$	$[a,a,a]$
$[a,b,c]$	$[c,b,a]$	$[a,b,c]$
$[a,a,a]$	$[a,a,a]$	$[b,c,c]$

Logically as Prolog facts:

$view(front, obj, 3, 3, [[b,c,c],[a,b,c],[a,a,a]]).$
$view(right, obj, 3, 3, [[*,c,a],[c,b,a],[a,a,a]]).$
$view(up, obj, 3, 3, [[a,a,a],[a,b,c],[b,c,c]]).$

Figure 2: Example of (a) an object observed from `Front`, `Right`, and `Up` sides; (b) an intuitive representation of the Q3D descriptor; (c) the logic facts produced by the Q3D descriptor for this specific object.

Consistency Checking across Views Beyond representation, Q3D imposes geometric compatibility constraints between views. A Q3D description is *consistent* [Falomir, 2015] if all pairwise projections can be jointly realised as orthographic views of a single three-dimensional object:

$$\begin{aligned} consistent\_Q3D(F,R,U) \leftrightarrow\ & consistent\_perspective(F,R) \wedge \\ & consistent\_perspective(F,U) \wedge \\ & consistent\_perspective(R,U). \end{aligned}$$

Intuitively, two views are consistent if there exists (at least) a 3D arrangement of volumetric blocks whose projections simultaneously satisfy the qualitative depth relations encoded in both views. Thus, `Front` –`Right` consistency requires that the volume values in last column of the `Front` view are consistent with the depth columns of the `Right` view, since they meet in an object edge and therefore these volumes are the same seen from both views. `Front` – `Right` consistency implies `Right` –`Front` consistency, which is defined as:

$$\begin{aligned} F_{i,3}(a) &\leftrightarrow R_{k,1}(a) \\ F_{i,3}(b) &\leftrightarrow R_{k,2}(a) \wedge (R_{k,1}(b) \vee R_{k,1}(c) \vee R_{k,1}(*)) \\ F_{i,3}(c) &\leftrightarrow R_{k,3}(a) \wedge (R_{k,1}(b) \vee R_{k,1}(c) \vee R_{k,1}(*)) \wedge \\ &\quad (R_{k,2}(b) \vee R_{k,2}(c) \vee R_{k,2}(*)) \\ F_{i,3}(*) &\leftrightarrow \neg R_{i,k}(a) \end{aligned}$$

where $\forall k, i \in \{1,2,3\}$. Similarly, `Front` – `Up` consistency and `Right` – `Up` consistency are defined.

Inference of Depth Values in Occluded Object Sides The Q3D model also provided deterministic rules for inferring depth values in the occluded sides of an object: `Back`, `Left`, and `Down`. For that, these rules exploit space continuity and depth interrelations. For example, `Front` and `Back` are opposite sites, if the Q3D include an item indicating the last level of depth in a view (e.g., level c in `Front` in an object of $3 \times 3 \times 3$ dimension), that involves the existence of the first level of depth (a) in the opposite view, that is `Back`. The same interrelations appear in opposite `Right` and `Left` sides and in `Up` and `Down`. Regarding space continuity, the case of holes involves that symbols ($*$) propagate from-to opposite sides of an object.

In summary, the Q3D depth model introduced: (i) a symbolic volumetric representation, (ii) cross-view consistency constraints, and (iii) inference mechanisms based on space continuity (e.g. hole continuity) and interrelations (e.g. depth relativity). It successfully showed that spatial structure and depth relations can be inferred from complementing perspectives. The Q3D reasoning model was implemented and tested in a SWI-Prolog application using 3×3×3 objects Falomir [2015] and then it was implemented in a videogame [Falomir and Oliver, 2016] with the goal of training players' spatial reasoning skills.

3.2 The Qualitative Object Rotations model (QOR)

The Qualitative Model of Object Rotation (QOR) [Falomir *et al.*, 2026] formalizes the relationship between an object perspectives and how they evolve during manipulation. To achieve this, the model relies on two main components: the Qualitative Object Descriptor (QOD) [Falomir and Costa, 2025], and the Rotation Reference System (QOR_{RS}) which defines the rotation movements that can be applied to a QOD, determining how the object features change their location and orientation from the observer's point of view.

$$QOR = \langle QOD_{RS}, QOR_{RS} \rangle,$$

where the QOD_{RS} describes an object O (or its associated bounding box) which has 6 canonical sides parallel in pairs, and $view_{O_{xyz}}$ is the view of that object O defined by 3 sides (x,y,z) perpendicular to each other. Each side is characterized by a *feature* (content), a *location* and an *orientation*. The type of features can vary from a simple symbol to an image/texture or a range of depths depending on the pattern recognition techniques used (e.g. pixels in a digital image).

The Qualitative Object Descriptor (QOD) provides a description of any three-dimensional object by embedding it within a conceptual bounding box. A view of an object is derived from a set of three single-side descriptors $\{QOD_x, QOD_y, QOD_z\}$, as follows:

$$view(Scene_{ID}, Object_{ID}, Perspective_{xyz}) \leftarrow \{QOD_x, QOD_y, QOD_z\}.$$

Each side of each object is characterized by features (pixels, voxels, symbols, etc.) which are spatially described by its location and its orientation reference systems (RS).

$$side(Scn, Obj, Location_{RS}, Feature_{RS}, Orientation_{RS}).$$

where

$$Location_{RS} = \langle (x,y,z), Location_N, Location_G \rangle$$

$Location_N \in \{F, R, U, D, B, L\}$
$Location_G = \{x,y,z \mid x \perp y, y \perp z, x \perp z\}$
where F is front, R is right, U is up, D is down, B is back, L is left, and (x,y,z) indicates coordinates in space which represents the planes defining the object bounding box.

Table 1: Location_{RS} definition for a bounding box of volume 3.

	$Location_N$	(x,y,z)	$Location_G$
F	*front*	$(x,y,3)$	plane $z=3$
B	*back*	$(x,y,0)$	plane $z=0$
R	*right*	$(3,y,z)$	plane $x=3$
L	*left*	$(0,y,z)$	plane $x=0$
U	*up*	$(x,3,z)$	plane $y=3$
D	*up*	$(x,0,z)$	plane $y=0$

$$Orientation_{RS} = \langle O_A, O_G \rangle$$

defined as follows:

$O_A \in \{0q, 1q, 2q, 3q, \textit{non-oriented}\}$
$O_{G_2} \in \{0°, 90°, 180°, 270°, \textit{any}\}$
$O_R \in \{+q, +2q, -2q, -q, \textit{same}\}$
$O_{G_2} \in \{+90°, +180°, -180°, -90°, 0\}$

where Orientation_A refers to the set of concepts that define an absolute orientation, e.g. "turned a quarter clockwise (1q)"; and Orientation_{G_1} refers to the geometric counterpart, that is, the turning angle clockwise in degrees (°) which is incrementing in steps of 90°, Orientation_R refers to the concepts that define relative orientation, e.g. "orientation increased a quarter (+q)" and Orientation_{G_1} define the corresponding increasing turning angles.

$$Feature_{RS} = \langle Feature_D, Feature_N \rangle$$

where $Feature_D$ refers to raw data from sensory input, such as a set of pixels for a symbol or voxels for depth, and $Feature_N$ is the symbolic concept assigned to that raw data. The mapping from $Feature_D$ to $Feature_N$ is achieved through an external abstraction process, either via manual annotation or automated perception mechanisms (e.g., computer vision classifiers), which explicitly grounds the qualitative symbols to the physical world.

In this paper, the features describing our objects are volumes described by the $Q3D_{RS}$.

$$Feature_{RS} = \langle Q3D_{RS} \rangle$$

Next, the Qualitative Object Rotation (QOR) completes the QOD descriptor by defining rotation actions.

The Qualitative Object Rotation (QOR) model relates rotation labels ($Rotation_N$) and their geometric definitions ($Rotation_G, Axis$). It is formalized as the relation:

$$rotation(Rotation_N, Rotation_G, Axis)$$

where the **Axes** are defined by the lines passing through the centroids of opposite object sides (see Figure 3), following the standard Euler definition relative to the object's centroid. $Rotation_G$ specifies the rotation angle in discrete 90° increments, and $Rotation_N$ assigns a qualitative label to the action. Table 2 describes the minimal QOR rotations.

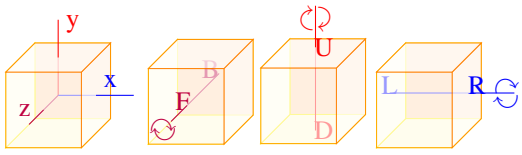


Figure 3: Operators for QOR depending on 3 axes (x,y,z) in two possible directions: axis front-back (fl,z); axis up-down (ud,y), and axis right-left (rl,x).

Table 2: QOR_{RS} specification. An icon representation using arrows has been used for increasing the readability of this paper.

Icon	*rotation*(Rotation$_N$, Rotation$_G$, Axis).
↑	*rotation*(towards-up, 270, x).
↓	*rotation*(towards-down, 90, x).
←	*rotation*(towards-left, 270, y).
→	*rotation*(towards-right, 90, y).
↷	*rotation*(towards-up-right, 270, z).
↶	*rotation*(towards-up-left, 90, z).

Table 3: Inference table which relates the initial location and the final location of a feature with the corresponding rotation movement and the final relative orientation.

from/to	**front**	**back**	**up**	**down**	**right**	**left**
front	$\curvearrowright^{+q}$ ∨ $\curvearrowleft^{-q}$		$\uparrow$	$\downarrow$	$\rightarrow$	$\leftarrow$
up	$\downarrow$	$\uparrow^{2q}$	$\rightarrow^{-q}$ ∨ $\leftarrow^{+q}$		$\curvearrowright^{+q}$	$\curvearrowleft^{-q}$
down	$\uparrow$	$\downarrow^{2q}$		$\rightarrow^{+q}$ ∨ $\leftarrow^{-q}$	$\curvearrowleft^{-q}$	$\curvearrowright^{+q}$
right	$\leftarrow$	$\rightarrow$	$\curvearrowleft^{-q}$	$\curvearrowright^{+q}$	$\uparrow^{+q}$ ∨ $\downarrow^{-q}$	
left	$\rightarrow$	$\leftarrow$	$\curvearrowright^{+q}$	$\curvearrowleft^{+q}$		$\uparrow^{-q}$ ∨ $\downarrow^{+q}$
back		$\curvearrowleft^{+q}$ ∨ $\curvearrowright^{-q}$	$\downarrow^{2q}$	$\uparrow^{2q}$	$\leftarrow$	$\rightarrow$

Location and Orientation change. By observing the relation between a rotation action and: (i) the changing location and (ii) the relative orientation changes, a new inference table can be built and it is shown by Table 3) where the leftmost column presents the initial feature location, while the top row presents the final location. The intersection cell specifies the rotation (Rotation$_N$) required to perform this transformation according to QOR_{RS} specification (Table 2). The superscript ΔO added to Rotation$_N^{\Delta O}$ in Table 3 specifies the relative orientation change applied to the feature during movement. The absence of a superscript implies that the orientation of the feature remains the same. Empty cells indicate that QOR rotation cannot directly move a feature from the starting location to the ending location (e.g. between opposite sides).

4 Method: Mixing Q3D+QOR for Rotation Inference and Face Transformation

4.1 Problem Representation

We represent each object as a discrete 3D structure with six oriented faces: Front, Back, Left, Right, Up, and Down. Each face is encoded as a depth matrix, so the object preserves both its global spatial structure and the local depth pattern on each face.

This representation is appropriate because a rotation acts as two levels. First, it permutes which face occupies each spatial position in the object. Second, it may reorient the depth matrix associated with a face, even when the face remains in the same relative location. As a result, the state of the object after rotation is determined by both face relocation and in-plane matrix transformation. This separation is the key abstraction used throughout the method: the object is treated as a structured set of depth matrices, and every transformation is defined in terms of how those matrices move and how they are rotated internally.

4.2 Applying Rotations to Q3D Model

A rotations is represented as a transformation of a complete six-face state. In our implementation, each rotation is specified by a declarative rule indicating three pieces of information: the source face, the transformation that must be applied, and the target face. Then, a rotation is not encoded as a numerical operation over 3D coordinates, but as a symbolic transformation over Q3D depth maps. The possible transformations of a face matrix are:

$$none, \quad rot_{90}^{cw}, \quad rot_{90}^{ccw}, \quad rot_{180}.$$

These transformations only change the position of the symbols inside the matrix. They do not change the symbols themselves. Therefore, a cell labeled a, b, c, or $*$ preserves its qualitative depth meaning after the rotation.

The rotation model Falomir [2026] distinguishes between rotations around different object axes. Pitch rotations, denoted here as rotations `towards_up` and `towards_down`, affect the vertical arrangement of the object. For instance, in a rotation `towards_up`, the faces move according to the following qualitative correspondence:

$$F \rightarrow U, \qquad U \rightarrow B, \qquad B \rightarrow D, \qquad D \rightarrow F.$$

Yaw rotations, denoted as `towards_left` and `towards_right`, affect the horizontal arrangement of the object. For a rotation `towards_left`, the visible side faces are reassigned as:

$$R \rightarrow F, \qquad F \rightarrow L, \qquad L \rightarrow B, \qquad B \rightarrow R.$$

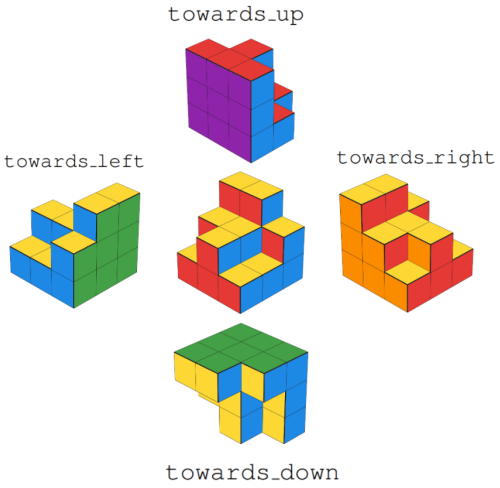


Figure 4: Schematic effect of viewpoint-changing rotation on a Q3D object.

The implementation also considers in-plane rotations (Roll), corresponding to quarter turns around the observer's viewing axis. These are denoted as `towards-up-right` and `towards-up-left`. In such rotations, the `Front` and `Back` faces remain in their positions and their matrices rotate in place, while the surrounding faces cycle around them (towards-up-right):

$$U \rightarrow R \rightarrow D \rightarrow L \rightarrow U$$

for one direction, and the inverse cycle for the other. This allows the model to capture rotations that do not change which face is frontal, but do change the orientation of the object within the image plane.

This preserves the qualitative character of the representation. The method does not reconstruct a metric 3D model and rotate it geometrically. Instead, it reasons directly over Q3D descriptions: rotations are computed by relocating symbolic depth maps

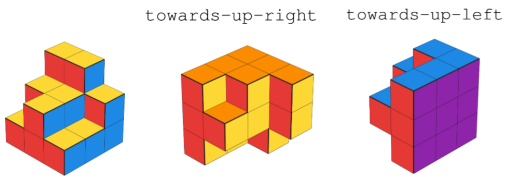


Figure 5: Schematic effect of in-plane rotation on a Q3D object.

and reindexing their matrix positions. As a result, the transformed object remains expressed in the same qualitative language as the original object, which allows rotated states to be compared directly with other Q3D descriptions.

4.3 Rotation Identification from Depth Matrices

Once rotations can be applied to Q3D states, the next problem is to identify whether two qualitative descriptions correspond to the same object under rotation. We formulate this as a search over a finite rotation graph. Each node in this graph is a complete six-face Q3D state, and each directed edge corresponds to one elementary rotation. In the current implementation, the considered rotations are `towards_up`, `towards_down`, `towards_left`, `towards_right`, `towards-up-right`, and `towards-up-left`.

The graph is generated from the initial object by repeatedly applying all available rotations to every newly discovered state. Whenever a new six-face configuration is produced, it is added as a node. If the same configuration has already been reached through another sequence, it is not expanded again. This produces the finite set of reachable orientations for the object, together with the rotation labels that connect them.

The order and size of this graph are bounded by the number of rigid orientations of a cube. A complete orientation can be specified by choosing which of the six faces is in the front position and, once this is fixed, which of the four remaining side faces is in the up position. This means that an asymmetric object has at most $6 \cdot 4 = 24$ distinct orientation states, so the graph order is $|V| \leqslant 24$. Since the implementation applies the six elementary rotation operators from every discovered node, the graph has six directed labelled outgoing transitions per node. Therefore, its size is $|E| = 6|V|$, and in the generic asymmetric case $|E| = 6 \cdot 24 = 144$.

The comparison between objects can then be performed at two levels. The rotation graph is constructed using the complete six-face state, because the hidden faces are needed to apply rotations consistently. However, the matching step can be restricted to the visible perspectives used in the test item. In our case, we use the `Front`, `Right`, and `Up` views, abbreviated as FRU. So, two states are considered observationally equivalent when their FRU signatures coincide ($FRU(S_i) = FRU(S_j)$).

This distinction is important for modelling mental rotation tests. A participant is normally shown only a limited view of an object, not its complete six-face Q3D description. Therefore, the method uses the full state internally to guarantee coherent rotations, but identifies matches through the visible depth matrices. Given a fixed object and a transformed candidate, the model retrieves all graph nodes with the same FRU description as the fixed object and all graph nodes with the same FRU description as the candidate.

The inferred transformation is then obtained by searching for a shortest path between these two sets of nodes. Since every edge corresponds to one rotation, the shortest path is the sequence with the minimum number of qualitative rotations connecting the two observed descriptions:

$$S_i \xrightarrow{\rho_1,\ldots,\rho_n} S_j, \qquad FRU(S_i) = FRU(O_1), \quad FRU(S_j) = FRU(O_2).$$

If such a path exists, the candidate is classified as a rotated version of the fixed object. The sequence $\rho_1, \ldots, \rho_n$ is therefore not only a computational result, but also an interpretable description of how the first object can be mentally rotated into the second one.

4.4 Use Case

To illustrate the method, consider a mental rotation item composed of two qualitative objects: a fixed reference object and a transformed candidate. Both objects are represented as six-face Q3D states, although the comparison is performed through the FRU description, as mentioned before. The task is to determine whether the transformed candidate can be obtained from the fixed object by a sequence of valid qualitative rotations.

The process starts from the Q3D description of the reference object:

$$S(O) = \langle F, B, L, R, U, D \rangle.$$

A copy of this object is then transformed by applying one or more rotation operators. For example, a candidate object may be generated by the sequence:

$$S(O) \xrightarrow{\texttt{towards\_left}} S_1 \xrightarrow{\texttt{towards-up-right}} S_2.$$

The resulting state S_2 represents the transformed object. Its full six-face description is available internally, but the inference procedure only uses the visible subset:

$$FRU(S_2) = \langle F_2, R_2, U_2 \rangle.$$

The inference procedure searches the graph described before for a state whose `Front`, `Right`, and `Up` matrices match those of the transformed candidate. If such a state is found, the candidate is interpreted as a rotated version of the reference object. The output of the method is the shortest qualitative rotation sequence that explains the transformation. This sequence provides an explicit symbolic explanation of the match. In this sense, the method does not only answer whether two objects are rotationally equivalent, but it also describes how one object can be transformed into the other.

This use case captures the role of the proposed method in Shepard-Metzler style tests. The reference and candidate objects are not compared through metric coordinates or rendered images. Instead, both are converted into qualitative depth descriptions, and the model reasons over the transformations of those descriptions. A correct answer is therefore grounded in a sequence of qualitative rotations over Q3D states.

5 Results

We evaluate the method on Q3D object descriptions by applying qualitative rotations to an initial object and then testing whether the system can recover the transformation from the visible `Front`, `Right`, and `Up` views. The evaluation focuses on three aspects: whether rotations generate coherent Q3D states, whether the applied rotation sequence can be recovered from partial views, and how a model behaves when several orientations share the same visible description.

5.1 Forward Rotation Generation

The first result concerns the generation of rotated Q3D descriptions. Starting from an initial six-face state, the implemented

rotation operators produce a new six-face state in the same qualitative format. That is, after each rotation, the object is still represented by the six canonical perspectives `Front`, `Back`, `Left`, `Right`, `Up`, and `Down`, and each perspective is again encoded as a Q3D depth matrix.

This shows that the rotation model is closed over the Q3D representation. This is important because it allows rotations to be composed, meaning that a qualitative rotation produces another valid Q3D state. A transformed state can immediately be used as the input to another rotation, making it possible to generate sequences such as:

$$S_0 \xrightarrow{\texttt{towards\_left}} S_1 \xrightarrow{\texttt{towards-up-right}} S_2 \xrightarrow{\texttt{towards\_up}} S_3.$$

Each intermediate state S_i remains directly comparable with other Q3D descriptions.

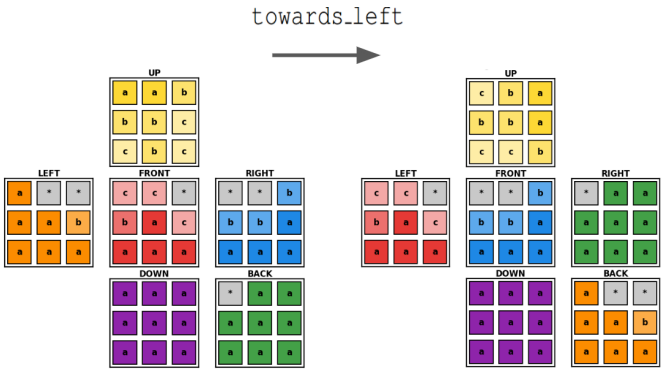


Figure 6: Example of a Q3D object before and after applying a qualitative rotation. Each state is represented by the six canonical depth matrices, showing that the rotation operators generate new Q3D states in the same symbolic format.

Figure 6 illustrates this process for one example object. The left side shows the original Q3D state, while the right side shows the state obtained after applying one qualitative rotation. The symbols inside the matrices are preserved, but their positions and face assignments change according to the selected rotation. For example, rotations around the vertical axis reassign the lateral faces while rotating the `Up` and `Down` matrices in place. Also, in-plane rotations keep the `Front` and `Back` faces in their positions while rotating them and cycling the surrounding faces.

The generated states also provide a direct way to inspect the effect of each operator. Since every rotated object is represented as a cube-net of depth matrices, the result of a rotation can be checked face by face. This makes the transformation transparent: one can observe both which faces have moved to new positions and how the internal orientation of their matrices has changed. This visual and symbolic inspection is useful for verifying that the qualitative rotation operators behave as expected before using them fo sequence recovery.

5.2 Rotation Sequence Recovery

The recovery experiment shows that the model can use partial qualitative views to reconstruct an explicit rotation path between two object descriptions. Rather than only deciding whether a transformed candidate is compatible with a reference object, the system returns a sequence of qualitative rotations that explains the transformation.

To illustrate this, a candidate is generated by first rotating the reference object to the left and then applying an in-plane quarter turn. Given only the `Front`, `Right`, and `Up` views of the reference and transformed states, the model recovers the following paths:

$$\rho_1 = \texttt{towards\_left} \rightarrow \texttt{towards-up-right},$$
$$\rho_2 = \texttt{towards\_up} \rightarrow \texttt{towards\_left},$$
$$\rho_3 = \texttt{towards-up-left} \rightarrow \texttt{towards\_up}.$$

Each step corresponds to a qualitative update of the Q3D state. The first rotation changes the viewpoint by cycling the lateral faces, while the second rotation performs an in-plane quarter turn. As a result, the visible matrices of the final state match the transformed candidate. Moreover, the recovered path is not necessarily identical to the sequence originally applied to generate the transformed candidate. A candidate may be produced through a long sequence of rotations, even when an equivalent orientation can be reached through fewer steps. For this reason, the inference procedure searches the rotation graph for the shortest path between the reference and the transformed FRU descriptions. The returned sequence should therefore not be understood as a reconstruction of the exact sequence that was used to generate it.

For example, consider a candidate generated by the following sequence:

$$\rho = \texttt{towards\_left x4} \rightarrow \texttt{towards-up-right}.$$

The first four rotations return the object to the same horizontal orientation, so the observed transformation is equivalent to a single in-plane quarter turn. Therefore, the model does not return the full applied sequence. Instead, it recovers the shorter explanation:

$$\rho_{inf} = \texttt{towards-up-right}.$$

This behavior is intentional. The inference procedure searches the rotation graph for a shortest path between the reference and transformed FRU descriptions. The returned sequence should therefore be understood as a minimal qualitative explanation of the observed transformation, rather than as a reconstruction of the exact sequence that was used to generate the candidate.

5.3 Dataset Evaluation

Base objects. The dataset is build from six base $3 \times 3 \times 3$ objects, each defined as a complete six-sides Q3D state (*Front*, *Back*, *Left*, *Right*, *Up*, *Down*) extracted from Ejarque and Falomir [2026]; Falomir [2015]; Falomir and Olier [2016]. Figure 7 shows all six templates as reconstructed from these depth views. They were chosen to span a range of structural complexity, measured as the number of unit cubes occupied out of the 27 available in the bounding box: from sparse configurations (*object5*, *object6*: 11/27) to dense ones (*object3*, *object4*: 22/27), with *object1* and *object2* in between.

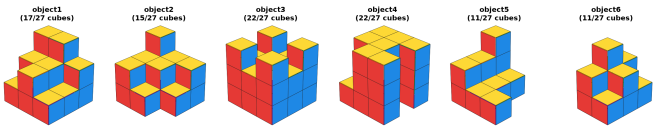


Figure 7: The six base template objects used to build the dataset, with the number of occupied unit cubes (out of 27) shown under each name.

For each object, we generate a sample by applying short predefined sequences of rotations and recording the resulting target

object. In addition to exact targets, we generate distractors to increase ambiguity. Distractors are constructed in two ways: by removing cubes only when the removal does not cause any other cube to become unsupported, and by creating symmetric variants of the original object through mirroring. This yields targets that are visually plausible but may not correspond to any valid rotation sequence from the initial object.

The dataset contains 444 samples in total, with 297 solvable cases and 147 unsolved cases under the shortest-path solver. This allows to test whether our solver can recover the shortest sequence of rotations from a given initial object to a target configuration, even when the target is a distractor. The solvable samples verify that the graph-based solver can recover valid minimal paths, while the unsolved samples highlight cases where the target cannot be reached from the initial object under the allowed moves.

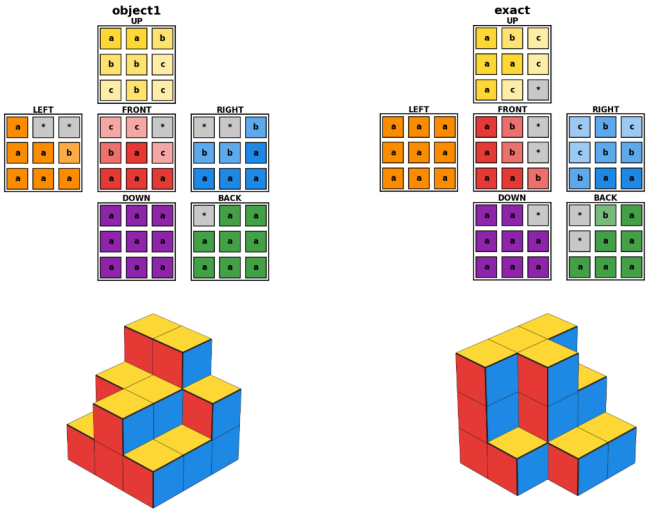


Figure 8: Representative solvable example from the dataset. The original sequence is `towards_up` → `towards-left`.

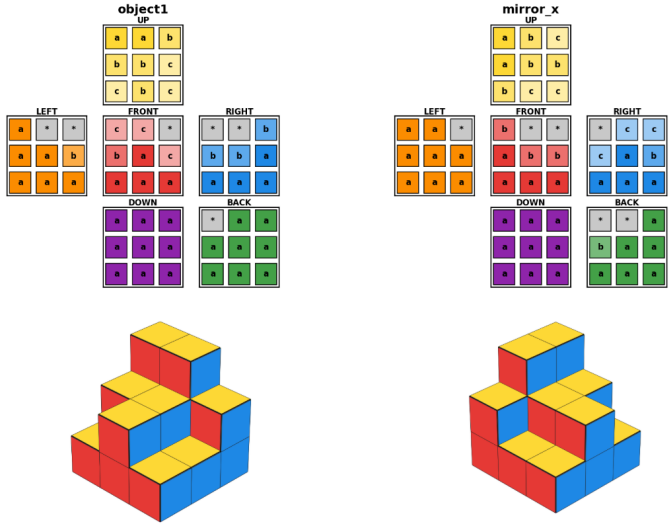


Figure 9: Representative distractor example from the dataset. Although the target is structurally plausible, no valid path exists from the initial object, since it's a mirror.

Figure 8 and 9 show representative examples of a solvable case and an unsolved distractor case, respectively. For solvable example, solver outputs three sequences:

`towards_up` → `towards-left`,
`towards_left` → `towards-up-right`
`towards-up-right` → `towards-up`

Since this is a qualitative shortest-path evaluation, there is no single incorrect answer in the usual sense: the target is either reachable by one or more valid shortest sequences or it is not. In solvable cases, multiple shortest paths may exist, so the key criterion is whether the solver recovers the complete set of minimal solutions. We also observe that some distractor targets are deliberately visually ambiguous, which makes it difficult to determine whether they represent the same object by inspection alone. This suggests a useful direction for future work on stronger disambiguation and perceptual comparison mechanisms.

6 Discussion about Future Work

This paper presented a qualitative approach for reasoning about object rotation in Shepard-Metzler style tasks. By combining Q3D depth descriptions with qualitative rotation operators, the model represents objects as symbolic six-face states, generates rotated descriptions, and recovers short transformation paths from visible `Front`, `Right`, and `Up` views.

Future work will focus on extending the approach beyond controlled block objects. One direction is to connect Q3D descriptions with perceptual input, such as RGB-D images voxel grids, or point clouds, so that the model can reason about real objects under partial observation. This would also support affordance checking, where qualitative features such as holes, flat regions, or accessible surfaces determine whether an object can be grasped, stacked, inserted, or used as support.

A second direction is to use the model as a component for solving real 3D puzzles, where objects must be rotated, aligned, or matched with target shapes. The same representation could also support Best-Next-View Planning in robotics: since Q3D makes visible and hidden structure explicit, an agent could choose the next viewpoint expected to reduce ambiguity the most.

As a final direction, the implementation should be generalized to arbitrary $H \times W \times L$, just like Q3D, and evaluated on larger object sets, symetric objects, and partially observed cases.

Code Availability

The code and dataset used in this work are available here.

Acknowledgments

We acknowledge the funding by the Wallenberg AI, Autonomous Systems and Software Program (WASP) awarded by the Knut and Alice Wallenberg Foundation, Sweden.

References

M. Bhatt and C. Freksa. Spatial computing for design: an artificial intelligence perspective. In John S. Gero, editor, *Studying Visual and Spatial Reasoning for Design Creativity*, pages 109–127. 2015.

Eliseo Clementini and Anthony G. Cohn. Extension of rcc*-9 to complex and three-dimensional features and its reasoning system. *ISPRS Int. J. Geo Inf.*, 13(1):25, 2024.

A. G. Cohn and J. Renz. *Qualitative Spatial Reasoning, Handbook of Knowledge Representation*. Elsevier, Wiley-ISTE, London, 2007.

Frank Dylla, Jae Hee Lee, Till Mossakowski, Thomas Schneider, André van Delden, Jasper van de Ven, and Diedrich Wolter. A survey of qualitative spatial and temporal calculi: Algebraic and computational properties. *ACM Comput. Surv.*, 50(1):7:1–7:39, 2017.

Marti Ejarque and Zoe Falomir. An Extended Qualitative Model for Reasoning about 3D Objects using Depth and Different Perspectives. In Ed Manley et al., editor, *The 17th Conference on Spatial Information Theory (COSIT) 22-25 September, York, UK*, LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2026.

Zoe Falomir and Vicent Costa. Comparing qualitative object descriptors using a visual similarity measure. In Vicenç Torra, Yasuo Narukawa, and Josep Domingo-Ferrer, editors, *Modeling Decisions for Artificial Intelligence - 22nd International Conference, MDAI 2025, València, Spain, September 15-18, 2025, Proceedings*, volume 15957 of *Lecture Notes in Computer Science*, pages 303–314. Springer, 2025.

Zoe Falomir and Eric Oliver. Q3D-Game: A Tool for Training Users' 3D Spatial Skills. In Paulo Novais and Shin'ichi Konomi, editors, *Intelligent Environments 2016 - Workshop Proc. of 12th Int. Conference on Intelligent Environments, IE 2016, London, UK, September 14-16, 2016*, volume 21 of *Ambient Intelligence and Smart Environments*, pages 187–196. IOS Press, 2016.

Z. Falomir and A.-M. Olteţeanu. Logics based on qualitative descriptors for scene understanding. *Neurocomputing*, 161:3–16, 2015.

Z. Falomir, E. Jiménez-Ruiz, M. T. Escrig, and L. Museros. Describing images using qualitative models and description logics. *Spatial Cognition and Computation*, 11(1):45–74, 2011.

Z. Falomir, L. Museros, V. Castelló, and L. Gonzalez-Abril. Qualitative distances and qualitative image descriptions for representing indoor scenes in robotics. *Pattern Recognition Letters*, 38:731–743, 2013.

Zoe Falomir, Albert Pich, and Vicent Costa. Spatial reasoning about qualitative shape compositions. *Ann. Math. Artif. Intell.*, 88(5-6):589–621, 2020.

Zoe Falomir, Ruben Tarin, Aurelio Puerta, and Pablo Garcia-Segarra. An interactive game for training reasoning about paper folding. *Multim. Tools Appl.*, 80(5):6535–6566, 2021.

Zoe Falomir, Ramon Garcia-Pou, and Vicent Costa. A qualitative model to reason about object rotations – applied to solve the Cube Comparison Test. 2026. Under Review.

Zoe Falomir. A qualitative model for reasoning about 3d objects using depth and different perspectives. In T. Lechowski, P. Wallega, and M. Zawidzki, editors, *LQMR 2015 Workshop*, volume 7 of *Annals of Computer Science and Information Systems*, pages 3–11. PTI, 2015.

Z. Falomir. Towards a qualitative descriptor for paper folding reasoning. In *Proc. of the 29th International Workshop on Qualitative Reasoning*, 2016. Co-located with IJCAI'2016 in New York, USA.

Zoe Falomir. A Qualitative Model to Reason about Object Rotations (QOR) applied to solve the Cube Comparison Test (CCT), 2026.

Florian Grigoleit, Sebastian Holei, Andreas Pleuss, Robert Reiser, Julian Rhein, Peter Struss, and Jana von Wedel. The qsafe project - developing a tool for current practice in functional safety analysis. In Marina Zanella, Ingo Pill, and Alessandro Cimatti, editors, *28th International Workshop on Principles of Diagnosis (DX'17), Brescia, Italy, September 26-29, 2017*, Kalpa Publications in Computing, pages 297–312. EasyChair, 2017.

Abhishek Jaiswal and Zoe Falomir. A qualitative model for reasoning about path and support. In *International Workshop on Qualitative Reasoning (QR'26), co-located at the 35nd Joint International Conference on Artificial Intelligence (IJCAI)*, Bremen, Germany, August 2026.

Marco Kragten, Tessa Hoogma, and Bert Bredeweg. Integration of a teacher dashboard in a hybrid support approach for constructing qualitative representations. In Rafael Ferreira Mello, Nikol Rummel, Ioana Jivet, Gerti Pishtari, and José A. Ruipérez-Valiente, editors, *Technology Enhanced Learning for Inclusive and Equitable Quality Education - 19th European Conference on Technology Enhanced Learning, EC-TEL 2024, Krems, Austria, September 16-20, 2024, Proceedings, Part I*, Lecture Notes in Computer Science, pages 208–221. Springer, 2024.

Beidi Li and Carl Schultz. Clingo2dsr - a clingo-based software system for declarative spatial reasoning. *Spatial Cognition & Computation*, 0(0):1–51, 2024.

G. Ligozat. *Qualitative Spatial and Temporal Reasoning*. MIT Press, Wiley-ISTE, London, 2011.

A. Lovett and K. Forbus. Cultural commonalities and differences in spatial problem-solving: A computational analysis. *Cognition*, 121(2):281 – 287, 2011.

A. Lovett and K. Forbus. Modeling visual problem solving as analogical reasoning. *Psychological Review*, 124(1):60 – 90, 2017.

Andrew M. Lovett and Holger Schultheis. Modeling spatial abstraction during mental rotation. In Paul Bello, Marcello Guarini, Marjorie McShane, and Brian Scassellati, editors, *Proceedings of the 36th annual conference of the Cognitive Science Society*, pages 886 – 891. Cognitive Science Society, 2014.

A. Lovett, M. Dehghani, and K. Forbus. Learning of qualitative descriptions for sketch recognition. In *Proc. 20th Int. Workshop on Qualitative Reasoning (QR), Hanover, USA, July*, 2006.

Vivien Mast, Zoe Falomir, and Diedrich Wolter. Probabilistic reference and grounding with PRAGR for dialogues with robots. *J. Exp. Theor. Artif. Intell.*, 28(5):889–911, 2016.

Julius Monsen, Jakob Suchan, and Mehul Bhatt. Probabilistic answer set programming driven ranking of dynamic spacetime belief models. In *Rules and Reasoning: 9th International Joint Conference, RuleML+RR 2025, Istanbul, Turkey, September 22–24, 2025, Proceedings*, page 156–175, Berlin, Heidelberg, 2025. Springer-Verlag.

Reinhard Moratz, Niklas Daute, James Ondieki, Markus Kattenbeck, Mario Krajina, and Ioannis Giannopoulos. Bilateral spatial reasoning about street networks: Graph-based RAG with qualitative spatial representations. *CoRR*, abs/2512.15388, 2025.

Albert Pich and Zoe Falomir. Logical composition of qualitative shapes applied to solve spatial reasoning tests. *Cogn. Syst. Res.*, 52:82–102, 2018.

Vicent Santamarta, Pablo Garcia-Segarra, Noelia Ventura-Campos, Aida Moreno-Rus, and Zoe Falomir. Reversal error in mathematics: Training against it using ε8404 coach.

In Ismael Sanz, Raquel Ros, and Jordi Nin, editors, *Artificial Intelligence Research and Development - Proceedings of the 25th International Conference of the Catalan Association for Artificial Intelligence, CCIA 2023, Món Sant Benet, Spain, 25-27 October 2023*, Frontiers in Artificial Intelligence and Applications, pages 241–249. IOS Press, 2023.

Roger N. Shepard and Jacqueline Metzler. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703, 1971.

Michael Sioutis and Diedrich Wolter. Qualitative spatial and temporal reasoning: Current status and future challenges. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4594–4601. ijcai.org, 2021.

Jakob Suchan. *Declarative reasoning about space and motion in visual imagery - theoretical foundations and applications*. PhD thesis, Bremen University, Germany, 2022.

Lijun Wei, Valérie Gouet-Brunet, and Anthony G. Cohn. Location retrieval using qualitative place signatures of visible landmarks. *Int. J. Geogr. Inf. Sci.*, 38(8):1633–1677, 2024.