

A More Efficient Representation of Obscuration for VRCC-3D+ Relations

Nathan Eloë, Chaman L. Sabharwal, and Jennifer L. Leopold

Abstract—VRCC-3D+ is an implementation of a region connection calculus that qualitatively determines the spatial relation between two 3D objects in terms of connectivity and obscuration. The eight connectivity relations are conceptually the same as RCC8, but calculated in 3D rather than 2D. The fifteen obscuration relations are calculated using the projection of the 3D objects on a particular 2D plane and the distance of the objects from the viewpoint. Herein we present a smaller, more precise set of VRCC-3D+ obscuration relations that retains the qualities of being jointly exhaustive and pairwise disjoint. However, this new set of relations overcomes two problems that existed in the previous set of fifteen relations: (1) lack of a precise mathematical definition for a key predicate, *InFront*, and (2) lack of an intuitive mapping of converse relations.

Index Terms—Computer vision, qualitative spatial reasoning, VRCC-3D, region connection calculus, spatial relations.

I. INTRODUCTION

QUALITATIVE spatial reasoning (QSR) in two dimensions is a well-studied field, and includes models such as the connectivity-based RCC systems [1], [2], [3], and obscuration-based systems such as LOS-14 [4], OCS-14 [5], and OCC [6]. These systems, while expressive, do not accurately portray the real world wherein objects exist and are perceived in three dimensions, not two. As computing power increases and the need to analyze three-dimensional data (e.g., stereoscopic video, robotic vision, etc.) increases, two-dimensional reasoning systems can be inefficient, or even inadequate, for sophisticated applications.

To ameliorate the shortcomings of two-dimensional QSR systems, Albath et al. developed RCC-3D [7], which eventually evolved into VRCC-3D+ [8]. VRCC-3D+ uses composite relations that express both connectivity and obscuration from a given perspective. The connectivity-based relations are the RCC8 relations (DC, EC, EQ, PO, TPP, TPPc, NTPP, NTPPc) defined in three dimensions; these relations have been an ongoing focus of optimization and refinement in the implementation as a QSR system [9]. The obscuration portion of the composite relations are refinements on the basic concepts of no obscuration (nObs), partial obscuration (pObs), equal obscuration (eObs), and complete obscuration (cObs).

Manuscript received on August 6, 2014, accepted for publication on September 22, 2014, published on November 15, 2014.

The authors are with Missouri University of Science and Technology, Computer Science Department, Rolla, MO-65409, USA (e-mail: {nwe5g8, chaman, leopoldj}@mst.edu). Corresponding author: Chaman L. Sabharwal (e-mail: chaman@mst.edu).

Over time these relations have been enhanced to improve their expressive power.

There are three criteria that the relations must meet to maintain the quality of the QSR system: the set of relations must be Jointly Exhaustive and Pairwise Disjoint (JEPD), every relation should map to exactly one converse relation, and the relations should have an intuitive mapping to natural language. If the relations are not jointly exhaustive, there will be physical configurations of objects that simply cannot be expressed by any relation. Relations that are not pairwise disjoint will result in ambiguous classification of object configurations. An intuitive mapping of the relations to natural language aids in human usefulness and usability of the system, and ensures that the expressive power of the system does not become needlessly complex; relations that cannot be differentiated in natural language typically do not add to the reasoning power of the system and overburden the computational complexity. Herein the authors focus on refining the obscuration terms of the composite VRCC-3D+ relations.

II. BACKGROUND AND RELATED WORK

A. Occlusion

Occlusion of one object by another object is contextually dependent on the observer's location (usually called the view point, the perspective reference point, or the center of perspective projection) relative to the objects. It follows that the occlusion decision can be made from the projection on a view plane. QSR applications are interested in deriving spatial obscuration relations and classification from projection of 3D objects on a 2D projection plane.

There are two types of projections as shown in Fig. 1: parallel and perspective. Both have their advantages and disadvantages. The parallel projection is easier to compute, but loses the concept of depth. With the perspective projection, the object is scaled by the distance from the view point then projected; depth information is preserved. Obscuration predicates are based on two parameters: the perspective projection in a plane and depth (distance of the object from the perspective reference point).

The terms *in front*, *occlusion*, and *closer* are closely related. In natural language, the term *in front* between two objects A and B is synonymously interpreted as "A is in front of B", "A occludes B", and "A is closer than B".

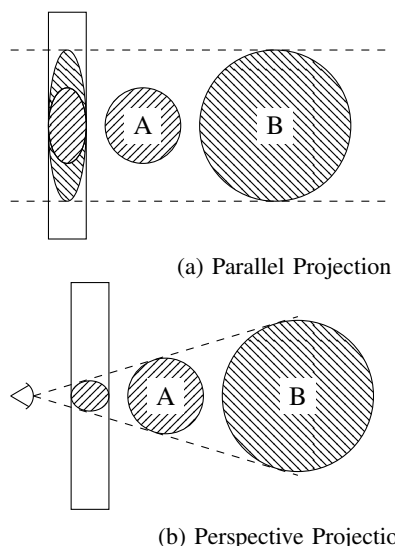


Fig. 1. The differences between parallel and perspective projections. In both cases, object A obscures object B.

B. QSR and Occlusion

One of the best known obscuration-based QSR systems is LOS-14 [4] which was introduced by A.P. Galton in 1994. It classifies regions based on what can be seen in the Lines Of Sight (LOS) from a given perspective. Fourteen relationships are defined based on obscuration (or the lack thereof) from a given viewpoint.

Another occlusion-based calculus is ROC-20 [10]. It is similar to LOS-14, but extends it to add support for concave objects, which allows for mutual obscuration. Every spatial relationship in ROC-20 is defined in terms of the occlusion that is present and an RCC8 relationship. This system is significantly more expressive than LOS-14 and can apply to a greater number of cases, as it correctly handles concave regions.

The Occlusion Calculus (OCC) was introduced by Kohler in 2002 [6] and characterizes relationships between objects by their respective projections into an image plane. The author states that the information obtained is from one perspective, and as such, this system should be paired with other QSR methods to get a fuller picture. The system sacrifices expressiveness for reduced computational and reasoning complexity.

Guha et al. introduced OCS-14 in 2011 [5]. This model was designed to correct for insufficiencies in earlier occlusion models that made them infeasible for use in computer vision. Earlier methods had not accounted for whether the occluder was a moving object or part of the static background, and whether or not the visible part of an object was a connected blob or a fragment. As OCS-14 is designed for computer vision, feasibility of computation is a concern, but not expressive power.

C. RCC-3D

RCC-3D [7] was designed by Albatth et al. to consider three dimensions, be computationally feasible, and give the most comprehensive spatial information about the system possible. Initially designed for use in analyzing the evolution of skeletal structures and other physical attributes, RCC-3D used the concepts of connectivity and obscuration to accomplish the design goals of completeness and computational feasibility. Because RCC-3D was to be used in visualizing physical changes over time, a GUI was deemed necessary. The resulting implementation was named VRCC-3D [11]. However, conceptual ambiguities that were uncovered in the implementation resulted in an evolution of the system, resulting in a revised model called VRCC-3D+ [8].

D. VRCC-3D+

Initially the obscuration portion of the VRCC-3D+ relationships simply were determined by overlapping boundaries and interiors of the projections of the objects in an image plane; the relations were limited to no obscuration (nObs), partial obscuration (pObs), complete obscuration (cObs), and equal obscuration (eObs). As the implementation of the system progressed, it became clear that a vital piece of information was missing; there was no concept of which object was obscuring the other. As such, an additional ternary predicate was added called *InFront*. For two objects A and B, possible values for *InFront*(A, B) were: YES (A is in front of B), NO (B is in front of A), and E (A and B are equidistant).

The ternary *InFront* predicate was used to refine the concepts of nObs, pObs, eObs, and cObs to express whether an object obscured the other, whether an object was obscured by the other (thereby adding *_c* to the relation name), or whether they obscured each other (thereby adding *_e* to the relation name). Some of these relations had an ambiguous combination of predicate values. As such, some of the relations were split, expanding the total number of relations to 15, as shown in Table I.

TABLE I
THE 15 CURRENT VRCC-3D+ OBSCURATION RELATIONS.

	IntInt	IntExt	ExtInt	InFront
nObs	F	T	T	YES
nObs_c	F	T	T	NO
nObs_e	F	T	T	EQUAL
pObs1	T	T	T	YES
pObs2	T	F	T	YES
pObs_c1	T	T	T	NO
pObs_c2	T	T	F	NO
pObs_e	T	T	T	EQUAL
cObs	T	T	F	YES
cObs_c	T	F	T	NO
cObs_e1	T	T	F	EQUAL
cObs_e2	T	F	T	EQUAL
eObs_e	T	F	F	EQUAL
eObs_c	T	F	F	NO
eObs	T	F	F	YES

III. IMPROVEMENTS TO THE RELATIONS

The first step in improving the obscuration relations is dealing with the mathematically imprecise predicate *InFront*. The ternary nature of this predicate and lack of rigorous mathematical definition led to different interpretations of the same scene by different entities. To replace this predicate, two new predicates are proposed: *Obscures* ($o(A, B)$) and *ObscuredBy* ($o_c(A, B)$). The *Obscures* predicate is defined as follows:

Let $f_O(x, y)$ be a function that maps the point (x, y) on the image plane back to the point (x', y', z') in object O that projects to the point (x, y) and is closest to the image plane. If no point in object O projects to point (x, y) , then $f_O(x, y) = (\infty, \infty, \infty)$. Also, let C be the location of the camera in world coordinates. The *Obscures* predicate for objects A and B is defined in Eq. (1):

$$o(A, B) = \begin{cases} T: & \exists x, y \text{ s.t. } |C - f_A(x, y)| < \\ & |C - f_B(x, y)| < \infty \\ F: & \text{otherwise} \end{cases} \quad (1)$$

In natural language, the meaning of this predicate is that it evaluates to true if there is a point at which the projections overlap and, within that projection, the first object hides some part of the second object. The definition of the converse relation $o_c(A, B)$ is simply $o_c(A, B) = o(B, A)$.

Note that the *Obscures* predicate only considers points at which the projection overlaps. This ameliorates cases such as that shown in Fig. 2. If we remove the condition that the distance between the camera and each of the two points be finite, object B would be reported to obscure object A at a point where the object A does not have a projection in the image plane.

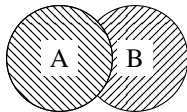


Fig. 2. Partial Obscuration: Object A obscures object B.

Table II shows the mapping of the original *InFront* predicate values to the values of the new *Obscures* and *ObscuredBy* predicates. Note that a value of EQUAL for the *InFront* predicate will map to either both o and o_c being true (T), or both being false (F).

The table of obscuration relations is rewritten as shown in Table III. The first simplification of the relation set follows directly from the predicate extension: when there is no obscuration between two objects, the projections do not overlap. As such o and o_c will always be false. It follows that *nObs* and *nObs_c* are impossible relations; only *nObs_e* is allowed by the predicate set, and only when the values of the new predicates are both false. This leads to the obscuration characterizations in Table IV.

TABLE II
MAPPING OF *InFront* TO o , o_c

<i>InFront</i>	$o(A, B)$	$o_c(A, B)$
YES	T	F
NO	F	T
EQUAL	F(or T)	F(or T)

TABLE III
THE 15 CURRENT VRCC-3D+ OBSCURATION RELATIONS WITH o AND o_c .

	IntInt	IntExt	ExtInt	o	o_c
nObs	F	T	T	T	F
nObs_c	F	T	T	F	T
nObs_e	F	T	T	F/T	F/T
pObs1	T	T	T	T	F
pObs2	T	F	T	T	F
pObs_c1	T	T	T	F	T
pObs_c2	T	T	F	F	T
pObs_e	T	T	T	F/T	F/T
cObs	T	T	F	T	F
cObs_c	T	F	T	F	T
cObs_e1	T	T	F	F/T	F/T
cObs_e2	T	F	T	F/T	F/T
eObs_e	T	F	F	F/T	F/T
eObs_c	T	F	F	F	T
eObs	T	F	F	T	F

TABLE IV
THE REDUCED SET OF VRCC-3D+ OBSCURATIONS (NObs AND NObs_c REMOVED).

	IntInt	IntExt	ExtInt	o	o_c
nObs_e	F	T	T	F	F
pObs1	T	T	T	T	F
pObs2	T	F	T	T	F
pObs_c1	T	T	T	F	T
pObs_c2	T	T	F	F	T
pObs_e	T	T	T	F/T	F/T
cObs	T	T	F	T	F
cObs_c	T	F	T	F	T
cObs_e1	T	T	F	F/T	F/T
cObs_e2	T	F	T	F/T	F/T
eObs_e	T	F	F	F/T	F/T
eObs_c	T	F	F	F	T
eObs	T	F	F	T	F

A. Handling Pathological Cases

Consider the projection shown in Fig. 3a. In this image, object B is partially obscuring object A, and A is partially obscuring object B. Under the previous set of 15 obscuration relations, the only way to express this would be *cObs_e*, when *pObs_e* is more intuitively correct. The reason for this is that the interior of the projection of object A does not intersect with the exterior of the projection of object B, resulting in an (intuitively incorrect) identification of *cObs* as the base obscuration type.

To address this issue, the structure of the relations themselves are examined. Currently, relations have a base of either *nObs*, *pObs*, *eObs*, or *cObs*. Appended to this base are the refinements of converse (*_c*) and equality (*_e*). To simplify this definition, a consistent structure is proposed: obscurations will have the form $xObs_y$, where x correlates to the *extent* of obscuration (Table V) and y corresponds to refinements on the obscuration (Table VI). To clarify the meaning of mutual

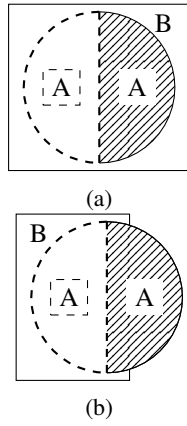


Fig. 3. Two examples of Mutual Obscuration. (a) A's projection completely contained in B's projection, (b) A's projection partially overlaps B's projection.

obscuration, a new suffix is introduced: *_m*.

A Cartesian product of the prefixes and suffixes show that there are 16 possible obscuration relations. However, it has already been shown that there can only be a single version of *nObs*; there is not a way to map suffixes directly to prefixes. As such, each relation must be individually handled.

TABLE V
PREFIX AND EXTENT OF OBSCURATION.

Prefix (<i>x</i>)	Meaning
n	No Obscuration
p	Partial Obscuration
e	Equal Obscuration
c	Complete Obscuration

TABLE VI
SUFFIX AND OBSCURATION REFINEMENT, WITH MAPPING TO *o* AND *o_c*.

Suffix (<i>y</i>)	Meaning	<i>o</i> (<i>A, B</i>)	<i>o_c</i> (<i>A, B</i>)
[none]	Obscures	T	F
_c	Is Obscured By	F	T
_e	Equally Obscure Each Other	F	F
_m (new)	Mutually Obscure Each Other	T	T

1) *Partial Obscuration (pObs)*: For partial obscuration, all cases where both objects are visible must be considered. The definitions of *pObs* and *pObs_c* remain unchanged. The characterization of *pObs_e* does not change, and has values of false for both *o* and *o_c*. The mutual refinement for *pObs* must handle the case shown in Fig. 3a as well as that shown in Fig. 3b. The case shown in Fig. 3b is straightforward. Figure 3a is more complicated: it must handle when object *A* is either object. As such, it maps to two characterizations of *pObs_m*. Table VIIa shows the new characterizations for all *pObs* relations.

2) *Equal Obscuration (eObs)*: Equal obscuration, by definition, occurs when the size and shape of the projections are identical; the values of the *IntInt*, *IntExt*, and *ExtInt* predicates will always be T, F, and F, respectively. The *eObs_e* obscuration should only occur if two objects are identical. Mutual equal obscuration can occur (Fig. 4), so that case must

be handled. Table VIIb shows the characterizations of the new *eObs* relations.

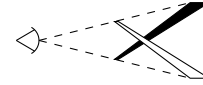


Fig. 4. Equal Mutual Obscuration

3) *Complete Obscuration (cObs)*: By definition, complete obscuration means that one object cannot be seen. As such, there is no *cObs_m* relation. Table VIIc shows the *cObs* characterizations.

B. Identification of Converse Obscurations

One of the problems with the old set of VRCC-3D+ obscuration relations was that there were cases where there was no consistent intuitive mapping from a relation to its converse. For example, *nObs_e*, *pObs_e*, and *eObs_e* map to themselves as converse, which is logical; if *A* and *B* obscure each other, then *B* and *A* should also obscure each other. However, there was not a single *cObs_e* relation but two. These two relations were the converse of each other. This inconsistency hindered both the implementation and reasoning with the system; it muddled the meaning of the *_e* suffix. Under this new relation set, every normal (no suffix) obscuration's converse relation is the converse obscuration (named with the *_c* suffix). The mutual (*_m*) and equal (*_e*) relations map to themselves as converse. Table VIII shows the full set of obscurations and their identified converse relation.

TABLE VIII
FULL OBSCURATION RELATION SET WITH IDENTIFIED CONVERSE RELATIONS.

	<i>IntInt</i>	<i>IntExt</i>	<i>ExtInt</i>	<i>o</i>	<i>o_c</i>	Converse
nObs_e	F	T	T	F	F	nObs _e
pObs	T	F	T	T	F	pObs _c
pObs_c	T	T	T	F	T	pObs
pObs_e	T	T	T	F	F	pObs _e
pObs_m	T	F	T	T	T	pObs _m
eObs	T	F	F	T	F	eObs _c
eObs_c	T	F	F	F	T	eObs
eObs_e	T	F	F	F	F	eObs _e
eObs_m	T	F	F	T	T	eObs _m
cObs	T	T	F	T	F	cObs _c
cObs_c	T	F	T	F	T	cObs
cObs_e	T	T	F	F	F	cObs _e

IV. EFFECT ON VRCC-3D+

The new set of obscurations directly benefits the implementation of VRCC-3D+ on two fronts: it is easier to verify that the implementation of the predicates is correct (leading to more correct results), and the reduced

TABLE VII
THE NEW VRCC-3D+ OBSCURATION CHARACTERIZATIONS.

(a) New pObs characterizations.

	IntInt	IntExt	ExtInt	<i>o</i>	<i>o_c</i>
pObs	T	T F	T	T	F
pObs_c	T	T	T F	F	T
pObs_e	T	T	T	F	F
pObs_m	T	T F T	T F T	T	T

(b) New eObs characterizations.

	IntInt	IntExt	ExtInt	<i>o</i>	<i>o_c</i>
eObs	T	F	F	T	F
eObs_c	T	F	F	F	T
eObs_e	T	F	F	F	F
eObs_m	T	F	F	T	T

(c) New cObs characterizations.

	IntInt	IntExt	ExtInt	<i>o</i>	<i>o_c</i>
cObs	T	T	F	T	F
cObs_c	T	F	T	F	T
cObs_e	T	T F	F T	F	F

set of obscurations and direct mapping to a converse relation improves the computational complexity of relation determination.

Migration from using the ternary *InFront* to two binary predicates immediately improved the unit test pass rate for obscuration relations from 29% to 77%. The expected output for the unit tests was determined by visual inspection; moving to the more mathematically precise predicates caused the implementation to more closely emulate human perception; several of the common errors reported stem from the perspective point being different for the person analyzing the file and the implementation, and more are due to floating point rounding errors. The majority of the errors that were fixed stemmed from an incorrect suffix on the base obscuration. Also, the use of two binary predicates instead of a single ternary predicate makes it trivial to implement a decision tree predicate picker similar to that presented in [12], which has been shown to improve the speed of computation.

The primary improvement in computational complexity is due to the unique mapping of a relation to a converse relation. In order to fully describe a scene containing objects A and B using VRCC-3D+, the system must compute the relationship between objects A and B, but also between B and A (the converse relation). To calculate an RCC8 relation in three dimensions, the computational complexity is in the worst case $O(f_a \times f_b)$, where f_a and f_b are the number of faces in objects A and B, respectively.

Generating the projections for objects A and B is $O(f_a)$ and $O(f_b)$, respectively; calculating the values of the predicate values is dependent on the (non-constant) complexity of performing intersection operations on the projections and the cost of ray casting to determine which object is closer to the view point. Regardless of the implementation of these operations, the cost of computing the values of the predicates is more than linear. In contrast to this, if the relationship between objects A and B is known, and both parts of the composite

relation have well-defined converses, the relationship between objects B and A can be determined to be the converse of each part of the composite relation: a lookup operation with complexity $O(1)$.

Every QSR system is designed as a balance of three criteria: ease of reasoning with the system, computational complexity, and expressive power of the system. An improvement to one aspect of the system comes at the cost of another. RCC-3D (the system that over time became VRCC-3D+) was initially designed to balance the three: by using compound relations, high expressive power and low computational complexity could be obtained without sacrificing too much in the way of ease of reasoning with the system. Herein, by reducing the 15 obscuration relations to 12 and introducing the concept of mutual obscuration as a refinement, both computational complexity and ease of reasoning have been improved.

Table IX shows the new set of 34 composite relations present in VRCC-3D+ (a reduction from the 46 relations stemming from the old set of 15 obscuration relations), and also serves to illustrate the importance of working to minimize the number of obscuration relations; if the four obscuration relations with multiple characterizations were expanded to 9 separate obscurations, the number of VRCC-3D+ relations would grow to 50 relations. Usage of a QSR system becomes increasingly more complex as the set of relations in the system increases.

An illustrative example of this is the composition table for the system which increases the speed of classifying relations by reducing the number of possible relations between two objects; in a scene with three objects A, B, and C, if the relationships between A and B, and B and C are known, the composition table reduces the set of relationships that are possible between objects A and C. This can be used in conjunction with a decision tree [12] to speed up computation. Calculation of this table has a non-constant polynomial complexity in the number of relationships.

TABLE IX
MAPPING OF RCC8 RELATIONS TO OBSCURATION RELATIONS

	$nObs_e$	$pObs$	$pObs_c$	$pObs_e$	$pObs_m$	$eObs$	$eObs_c$	$eObs_e$	$eObs_m$	$eObs$	$eObs_c$	$eObs_e$
DC	X	X	X		X	X	X			X	X	
EC	X	X	X		X	X	X			X	X	
PO		X	X	X	X	X	X		X	X	X	
EQ								X				
TPP							X				X	X
TPPc						X				X		X
NTPP											X	
NTPPc									X			

V. FUTURE WORK

Future work will focus on mapping the revised VRCC-3D+ relations to natural-language terms suitable for end-user applications involving spatial querying. Preliminary efforts in this direction have commenced for the VRCC-3D+ connectivity relations [13]. Given the ambiguity of natural-language terms such as *in front/behind*, *occludes*, and *closer/nearer*, it may prove difficult to find unambiguous mappings for the mathematically precise VRCC-3D+ obscuration relations. Extensive human experiment studies will need to be conducted, and likely domain-specific ontologies will have to be developed for the relation-to-term associations.

VI. CONCLUSIONS

In this paper, simplifications to the 15 obscuration relations present in VRCC-3D+ have been analyzed and presented. This change in the mathematical set of relations improved the computational correctness from 27% to 77%. The VRCC-3D+ obscuration relations are now easier to understand and computationally easier to implement because of the introduction of a new predicate for classification and a new class of obscuration.

REFERENCES

- [1] B. Bennett, "Spatial reasoning with propositional logics," in *Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning*, ser. KR'94, J. Doyle, E. Sandewall, and P. Torasso, Eds. San Francisco, CA.: Morgan Kaufmann, 1994.
- [2] J. Ouyang, Q. Fu, and D. Liu, "A model for representing topological relations between simple concave regions," in *Proceedings of the 7th International Conference on Computational Science*, ser. ICCS'07, vol. 1. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 160–167.
- [3] D. A. Randell, Z. Cui, and A. Cohn, "A spatial logic based on regions and connection," in *Proceedings of the 3rd Conference on Principles of Knowledge Representation and Reasoning*, ser. KR'92, B. Nebel, C.

Rich, and W. Swartout, Eds. San Mateo, California: Morgan Kaufmann, 1992, pp. 165–176. [Online]. Available: citeseer.ist.psu.edu/randell92spatial.html

- [4] A. Galton, "Lines of sight," in *AISB Workshop on Spatial and Spatio-Temporal Reasoning*, 1994.
- [5] P. Guha, A. Mukerjee, and K. S. Venkatesh, "OCS-14: You can get occluded in fourteen ways," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, ser. IJCAI'11, T. Walsh, Ed., 2011, pp. 1665–1670.
- [6] C. Kohler, "The occlusion calculus," in *Cognitive Vision Workshop*, ser. ICW'02, 2002.
- [7] J. Albath, J. L. Leopold, C. L. Sabharwal, and A. M. Maglia, "RCC-3D: Qualitative spatial reasoning in 3D," in *Proceedings of the 23rd International Conference on Computer Applications in Industry and Engineering*, ser. CAINE'10, 2010, pp. 74–79.
- [8] C. Sabharwal, J. Leopold, and N. Eloe, "A more expressive 3D region connection calculus," in *Proceedings of the 2011 International Workshop on Visual Languages and Computing (in conjunction with the 17th International Conference on Distributed Multimedia Systems)*, ser. DMS'11, 2011, pp. 307–311.
- [9] N. Eloe, J. L. Leopold, and C. L. Sabharwal, "Efficient computation of object boundary intersection and error tolerance in VRCC-3D+," in *Proceedings of the 18th International Conference on Distributed Multimedia Systems*, ser. DMS'12, 2012, pp. 67–70.
- [10] D. Randell, M. Witkowski, and M. Shanahan, "From images to bodies: Modelling and exploiting spatial occlusion and motion parallax," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, ser. IJCAI'01, vol. 1. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 57–63.
- [11] J. Albath, J. L. Leopold, and C. L. Sabharwal, "Visualization of spatio-temporal reasoning over 3D images," in *Proceedings of the 2010 International Workshop on Visual Languages and Computing (in conjunction with the 16th International Conference on Distributed Multimedia Systems)*, ser. DMS'10, 2010, pp. 277–282.
- [12] N. Eloe, J. L. Leopold, C. L. Sabharwal, and D. McGeehan, "Efficient determination of spatial relations using composition tables and decision trees," in *Proceedings of the IEEE Symposium on Computational Intelligence for Multimedia, Signal, and Vision Processing*, ser. CIMSIVP'13, 2013, pp. 1–7.
- [13] J. L. Leopold, C. L. Sabharwal, and K. Ward, "Spatial relations between 3D objects: The association between natural language, topology, and metrics," in *Proceedings of the 2014 International Workshop on Visual Languages and Computing*, ser. VLC'14, 2014, (To appear).