# Using Occlusion Calculi to Interpret Digital Images

**David Randell**[1] and **Mark Witkowski**[1]

**Abstract**. This paper reports on an investigation using occlusion calculi to interpret digital images. Using a minimal set of digital, region-relation detectors, and assuming a continuous interpretation of physical space, we show how existing calculi can be augmented and embedded in the *Event Calculus* to interpolate and recover a larger set of occlusion relations than are otherwise available at the basic detector level.

## 1 INTRODUCTION

Abstracting and reasoning about high-level symbolic spatial information about extended bodies (or regions) has long been a motivating ontological assumption used in *Qualitative Spatial Reasoning (QSR)* research. That several suitably expressive *QSR* logics have subsequently turned out to have interesting and useful computational properties has added to this general degree of confidence in their use [2,3]. *QSR* applications cover such diverse areas as querying spatial databases linked to *GIS*, the encoding of vague concepts, and in processing visual information using real-world images [3,10]. In terms of practical applications however, few have applied themselves to the task of deploying these logics on simulated or real-world machine-vision systems.

Given the relative absence of published work along these latter lines, plus the fact that *QSR* has now been an active research topic in AI for at least 17 years, it is perhaps time to enquire why, and despite the large number of spatial logics QSR has spawned, QSR has not made more headway here than it has. To this end, we have honed in on a subset of QSR logics known as *occlusion* calculi [5, 8, 10, 11] and in particular have singled out Galton's *Lines Of Sight* (*LOS-14*) calculus [5] with which to interpret 3D convex bodies in terms of their projected images. We wish to examine to what extent occlusion calculi can be used to interpret idealised, real-world two-dimensional digital images returned by a camera.

If we assume (as we do) that our regions are derived directly as the segmented images of physical bodies, then spatial occlusion events becomes a significant issue as several ontological assumptions of existing QSR logics (e.g. that the extent of the regions in question is known *a priori*, and that the embedding space is continuous) no longer hold. As we show, these and other assumptions have significant ramifications for digital geometry [7] and QSR logics in general.

In order to strengthen and simplify our example, we (i) modify the ontological base on which the original *LOS-14* calculus was built to handle a *discrete* (digitised) rather than a continuous model of space, (ii) embed the logic in an abductive inferential framework using the *Event Calculus* and exploit Conceptual Neighbourhood Diagrams (CNDs) in order to interpolate (by abduction) spatial relations not directly encoded in our set of region-relation detectors, (iii) simplify the segmentation process by individuating regions as aggregates of pixels falling within a specific colour-hue range, and (iv) assume a computational model of region identity, to track regions over time (e.g. [12]).

[1] Dept. of Computing, Imperial College London, UK.
email:{d.randell,m.witkowski}@imperial.ac.uk

The rest of this paper is structured as follows. In section 2 we introduce occlusion calculi. Three important limiting assumptions are challenged, (i) that the extent of regions is known *a priori*, (ii) that embedding space is continuous, and (iii) that region boundary information can be reliably extracted from real images. In section 3 we introduce a discrete mereotopological theory. This is used to define the logical properties of regions and relations extracted from segmented digitised images, and specify the logical properties of a small set of region, and region-relation detectors. In section 4 the detectors are mapped to the relation set of *LOS-14*. Section 5 presents an overview of a practical implementation of the detectors. In section 6, the *Event Calculus* is used to abductively infer occlusion relations from an assumed temporal succession of detector states extracted from a sequence of images. Concluding remarks are made in section 7.

## 2 OCCLUSION CALCULI

Region-based Occlusion Calculi for vision applications originated with Galton's *Lines of Sight* paper (*LOS-14*) [5]; this was followed by *The Region Occlusion Calculus* (*ROC-20*) [10], [11], and Kohler's *Occlusion Calculus* [8]. These calculi spring from the general development of *QSR* formalisms, calculi and algebras that have been developed over the years for modelling sub-theories of space. In contrast to other *QSR* spatial formalisms, occlusion calculi adopt a distinctive *viewpoint-centred* model of space where visual spatial relations on bodies (or regions) are mapped to corresponding spatial relations on their images.

At least two of these logics (*LOS-14 and ROC-20*) assume a continuous embedding model of space, and each factors out a *Jointly Exhaustive and Pairwise Disjoint* (*JEPD)* set of relations. Constraints on the interpretation of the regions, primitive functions and relations used, mean the degree of theory overlap between them varies. For example, Galton's *LOS-14*, defined on discrete convex bodies, identifies 14 *JEPD* relations (see Figure 1), whereas *ROC-20* relaxing this convexity constraint identifies 20. All the relations defined in *LOS-14* have analogues in *ROC-20*, and both are built on the set of relations identified in the spatial logic *RCC-8* [9]. In contrast, Kohler's *Occlusion Calculus* builds on the weaker set of relations defined in the spatial logic *RCC-5* [2].

As is common with many *QSR* theories, each *JEPD* relation set is re-worked into a conceptual neighbourhood diagram (*CND*). These diagrams encode the continuous transitions allowed between pairs of spatial relations [4]. For example, in *LOS-14*, a direct transition exists between *C/2* (clears), and *JC/2* (just clears) but not between *C/2* and *PH/2* (partially hides); where here and elsewhere, the *R/n* notation used, identifies an n-ary relation. The same *CND* can also be used to justify interpolated relations, as in the case where *JC/2* is inferred given a detected transition between *C/2* and *PH/2* – see Figure 1.

## 2.1 Occlusion Calculi: some assumptions

Given that all the occlusion calculi described here are to a large extent, theoretically motivated, their ontological assumptions need to be carefully examined given we seek to ground the logical primitives in real image data.

The first assumption noted is that the extent of the bodies is assumed known *a priori*, or is at least directly inferable from other information present. However, in the presence of opacity, the mode and extent of visual overlap between bodies is more often than not indeterminate. For example, consider the *CND* of *LOS-14* depicted in Figure 1. Of the 14 possible occlusion relations, 12 rely on part of one body being hidden by another, while only 8 have both bodies directly visible. On the assumption that the physical bodies giving rise to image regions are opaque, and therefore that their hidden extent cannot be determined from any given viewpoint, only three distinct relations of this type can be reliably distinguished. This is discussed in detail later.

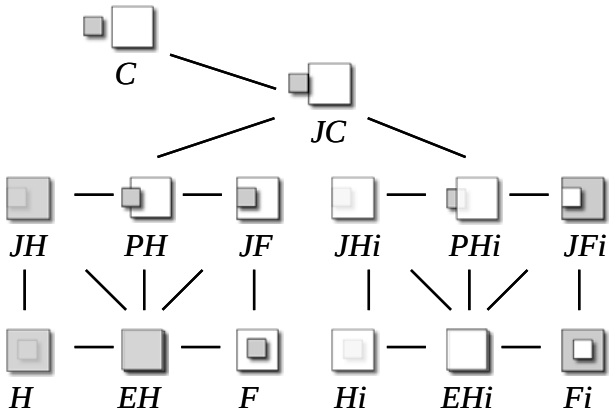


**Figure 1:** The conceptual neighbourhood diagram LOS-14

The second assumption is that all the existing occlusion calculi assume a *continuous* model of space. As is now well known, the spatial logic *RCC8* used in [5,7,9] disallows *atomic* regions; otherwise the whole calculus is rendered inconsistent [9]. Yet clearly, the space modelled by pixel-arrays is discrete, and if individual pixels here cannot be treated as atomic regions, how then can digital images provide a model for these calculi? The continuous vs. discrete issue for *QSR* formalisms is also discussed in [6], [13] where discrete variants of the *RCC8* calculus are motivated and developed. Also, and related to this point, is how we interpret the primitives of these logics. Take for example the relation *C/2* used in *LOS-14* (see Figure 1), or its analogue relation *NonOccludesDC/2* used in *ROC-20*. Whether we choose to interpret these relations in terms of lines of sight (as in *LOS-14*) or as a pencil of half-lines extending from some viewpoint point *v*, to infinity (as in *ROC-20*), a fundamental difference arises when using the projective plane defined on the real numbers compared with a viewer-centred discrete interpretation of space. In the first case, the spatial relation *C(A,B)* is always true whatever the image scale assumed. But in the latter case, whether or not *A* and *B* are visually separated depends upon other parameters of the physical system being modelled, such as the image scale or the resolution of the imaging system. This is, of course, an example of the *granularity* issue that is well known in AI research.

The third and last assumption highlighted here is that region-boundary information can be reliably extracted from digitised images. The emphasis on boundaries is a direct consequence of adopting mereotopological logics, for it is precisely the mode of *contact* between the visual boundaries of regions that determines what spatial relation is true. However, the reliable interpretation of boundary information from real-world digital images is problematic and detectors based on this information tend to be brittle and often unreliable in use.

These assumptions are of course by no means exhaustive. For example, sensor noise, fast sampling rates, the effects of variable lighting conditions, of shadows and specular reflections all affect the segmentation of the raw digital data returned by any current physical imaging system. However, the main assumptions stemming from the ontology and models used in occlusion calculi have been identified. It is specifically these assumptions that we now address in a *dynamic QSR* framework.

# 3 DISCRETE MEREOTOPOLOGY

We use Galton's *two-sorted mereotopological logic* [6] as a *specification language* to define the logical properties of our region and region-relation detectors defined on aggregates of pixels as regions. Two sorts are declared: *pixels* (*cells* in Galton's case) and *regions* as aggregates of pixels.

In [6], definitions of the classical mereological relations and the Boolean composition of regions are given and interpreted set-theoretically. A null region $\varnothing$ is declared, that contains no pixels, and the universal region *U*, that contains every pixel in the embedding space. Containment of a pixel *x* in a region *X* is modelled as set membership, i.e. as $x \in X$; and adjacency of two pixels is captured by the reflexive, symmetric relation *A(x,y)*, meaning pixel *x* is adjacent to or equal to pixel *y*. We use a square-based pixel array, and adjacency here is interpreted as 8-adjacency, meaning every non-boundary pixel of the array is surround by 8 neighbours forming a combined 3x3 pixel matrix.

The basic relation-based definitions proceed as follows. Inclusion between regions is defined as: $X \subseteq Y \equiv \forall x \in X \rightarrow x \in Y$, and the (non-null) part/whole relation as: $P(X,Y) \equiv X \subseteq Y \ \& \ X \neq \varnothing$. Overlap is defined as: $O(X,Y) \equiv X \cap Y \neq \varnothing$. For the weaker relation of connection (or contact) between regions, the adjacency relation now appears: $C(X,Y) \equiv \exists x \exists y \ [x \in X \ \& \ y \in Y \ \& \ A(x,y)]$. This enables the external connection relation: *EC(X,Y)* and the remaining discrete analogue counterparts of the *RCC8* relations and other related concepts (For example, Galton defines neighbourhood, boundary, interior, closure and covers metric spaces, change and continuity) to be defined. To avoid confusion with the RCC8 relations these will be referred to as *RCC8D* [6]. We will use only three of these *RCC8D* relations: DisConnected, *DC(X,Y)*; Externally Connected, *EC(X,Y)*; and Non Tangential Proper Part, *NTPP(X,Y)*, mirroring the detectors. Taking *U-X* as the region difference between *U* and *X*, these are defined as follows: $DC(X,Y) \equiv \neg C(X,Y)$, $EC(X,Y) \equiv C(X,Y) \ \& \ \neg O(X,Y)$, and $NTPP(X,Y) \equiv DC(X,U\text{-}Y)$.

While *CNDs* exist for discrete mereotopological logics (e.g. *RCC8D*), the effect of the digital sampling serves to significantly increase the number of possible distinct relation-relation transitions when compared with their continuous counterparts. Here, we assume an underlying continuous model of physical space while sampling the relations into a discrete representation. This has the advantage of (i) allowing physically realisable interpretations of 'anomalous' transitions close to the resolution limit of the detectors, and (ii) exploiting much stronger constraints on possible transitions than directly supported by the discrete model of space.

### 3.1 Extensions

A finite disjoint set of individual region-membership predicates: $R_1,...,R_n$, is defined on aggregates of pixels. The exact nature of the membership property $R_i$ need not concern us here; but will typically range over some physical property predicated on bodies, such as colour as assumed in our worked example. As uniform patches of colour are used as the basis of both object (or body) and region-identity, and is used by our region segmentation algorithm to label individual regions, we can state the equivalence: $R(X) \equiv \forall x[x \in X \rightarrow R(\{x\})]$. Another property, that of a maximally-connected region, *Max(X)*, is also adopted; in which aggregates of adjacent pixels satisfying property $R_i$ are clustered into connected subsets. This ensures a minimal number of explicitly labelled regions at the detector level: $Max(X) \equiv \bigvee_{i=1}^{n} [R_i(X) \,\&\, \forall x \forall y[x \in X \,\&\, y \notin X \,\&\, A(x,y)] \rightarrow \neg R_i(\{y\})]$. While not developed here, individual regions can then be classified into connected and disconnected (i.e. scattered) regions, see [6] for details. Finally, an additional property of *well-formed region* (wfr) is defined on aggregated sets of pixels as satisfying the definition: $Region(X) \equiv \bigvee_{i=1}^{n} [R_i(X) \,\&\, \forall x[x \in X \rightarrow \exists y \exists z \exists u[\& \; y \in X \,\&\, z \in X \,\&\, u \in X \,\&\, x \neq y \,\&\, x \neq z \,\&\, x \neq u \,\&\, y \neq z \,\&\, u \neq u \,\&\, z \neq u \,\&\, A(x,y) \,\&\, A(x,z) \,\&\, A(x,u)]]]$. A well-formed region defines a set of pixels *X*, satisfying some property *R*, where every pixel element of *X* has at least 3 distinct immediate neighbours of that set. Regions not satisfying this condition are passed to our algorithm for pre-processing prior to segmentation and labelling of the regions.

## 4 REGION CONTACT DETECTORS

The simplest detectors report whether or not a labelled region X (from viewpoint *v*) is detected: *visible(X,v) = true*, *iff* from *v*, pixel-region *X* is detected in the pixel-array, otherwise false. This allows us to infer when, for example, *visible(a,v)=false*, that *a* exists but is hidden from *v*. These relations (with their reified regions) re-appear as the primitive relation: *Visible(x,v)* in the object language.

The pixel-aggregate detector definitions for *DC**, *EC** and *NTPP** are now as follows (see Figure 2 for examples):

$$DC^*(X,Y) \equiv \bigvee_{i=1,j=1}^{n} [R_i(X) \,\&\, R_j(Y)] \,\&\, \neg\exists x \exists y \,[x \in X \,\&\, y \in Y \,\&\, A(x,y)]]$$

$$EC^*(X,Y) \equiv \bigvee_{i=1,j=1}^{n} [R_i(X) \,\&\, R_j(Y)] \,\&\, \exists x \exists y \,[x \in X \,\&\, y \in Y \,\&\, A(x,y)] \,\&\, \neg\, NTPP^*(X,Y)$$

$$NTPP^*(X,Y) \equiv \bigvee_{i=1,j=1}^{n} [R_i(X) \,\&\, R_j(Y)] \,\&\, [NTPP(X,Y') \,\&\, Y'=X+Y]$$

Each of these relations are mapped to three implemented detectors, dc*, ec* and ntpp*, with ntppi* as the inverse of ntpp*. Note that the ntpp* *detector* does not of itself embody the transitive property of the (*RCC8 and RCC8D*) relation *NTPP/2*, as the detector operates locally. Thus, if ntpp*(A,B) and ntpp*(B,C), then dc*(A,C) will be initially reported. However, as the domain model restricts bodies to convex regions, we infer ntpp*(A,C) via its mapping to *NTPP/2*. The justification for labelling embedded nested regions as *NTPP** (as shown in Figure 2) follows from the (RCC8 and RCC8D) theorem: $\forall xyz[NTPP(sum(x,y),z) \rightarrow [NTPP(x,z) \,\&\, NTPP(y,z)]]$, i.e. every part of a non-tangential part of region forms a non-tangential relationship to the whole.

In terms of the *LOS-14* occlusion relations mapping to the relation detectors, only one is clearly unambiguous, namely *C*, which maps to *DC**; while the others: *JC,PH,PHi,JF,JFi* map to *EC**; and *F*, which maps to *NTPP**, are either ambiguous or reintroduce some other assumption into the interpretation. In the case of the latter, for example, a one-one mapping is only possible if we make (as we do) the additional convexity assumption. Having now reduced the 14 occlusion relations down to a detector that can only distinguish three classes of relations, how then can we recover the missing ones?

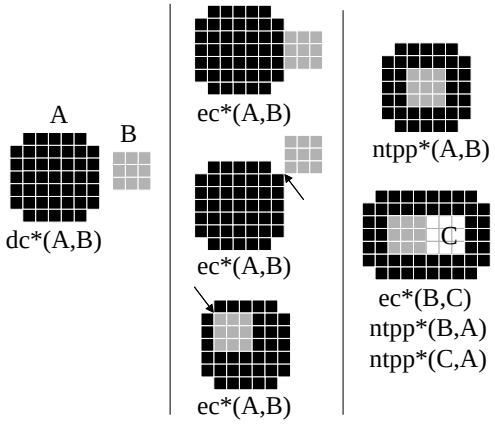

Figure 2: Discrete digitised regions

### 4.1 Mapping between Bodies and Regions

To illustrate the interplay between bodies and digitised regions we now embed the relations: *DC**, *EC** and *NTPP** (and the inverse relation *NTPPI**) into *LOS-14*. An *explicit* viewpoint variable, *v*, and an image function *image(x,v)*, that maps body *x*, and viewpoint *v* to its projected image is added, and each dyadic *LOS-14* relation, is re-worked into a ternary relation – see [10] for further details.

The first set of axioms map the occlusion relations to their *RCC8D* spatial relations on images; then following the group of definitions, another group of axioms map between selected *LOS-14* relations and the primitive region-relation detectors. For brevity we omit from the antecedents of these axioms conjuncts that encode the visibility or otherwise of bodies given particular occlusion relationships. However, these can be inferred from the model used to illustrate the CND in Figure 1.

$\forall x \forall y \forall v[C(x,y,v) \rightarrow DC(image(x,v),image(y,v))]$
$\forall x \forall y \forall v[JC(x,y,v) \rightarrow EC(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [PH(x,y,v) \rightarrow PO(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [JF(x,y,v) \rightarrow TPP(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [F(x,y,v) \rightarrow NTPP(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [JH(x,y,v) \rightarrow TPPi(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [H(x,y,v) \rightarrow NTPPi(image(x,v),image(y,v))]$
$\forall x \forall y \forall v\ [EH(x,y,v) \rightarrow EQ(image(x,v),image(y,v))]$

$PHi(x,y,v) \equiv def.\ PH(y,x,v)$, $JHi(x,y,v) \equiv def.\ JH(y,x,v)$
$JFi(x,y,v) \equiv def.\ JF(y,x,v)$, $Hi(x,y,v) \equiv def.\ H(y,x,v)$
$Fi(x,y,v) \equiv def.\ F(y,x,v)$, $EHi(x,y,v) \equiv def.\ EH(y,x,v)$

$\forall x \forall y \forall v[C(x,y,v) \rightarrow DC^*(image(x,v),image(y,v))]$
$\forall x \forall y \forall v[\Phi(x,y,v) \rightarrow EC^*(image(x,v),image(y,v))]$
*where:* $\Phi \in \{JC,PH,JF,JFi,PHi\}$
$\forall x \forall y \forall v\ [F(x,y,v) \rightarrow NTPP^*(image(x,v),image(y,v))]$

## 5 THE DETECTOR ALGORITHM

This section summarises the main points of the algorithm (and program) used to implement the detectors and transform input

bitmap images to regions and extract the QSR relations between them. The (C++) program accepts as input bitmap images (of size `x` by `y`) and, in step 1, assigns a region value in the range 0 to `maxr`-1 (max regions) for each unique colour value encountered (the array `r`). Test images may be prepared using the block fill tool in any standard bitmap manipulation program. The program also allows camera images to be grabbed directly using a Videre Mega-D firewire camera or imported via bitmap images. Prior to use, real images are subjected to a hue separation process to extract distinct regions. The resulting region based image is subjected to a region-conditioning process that implements the notion of well-formed region (wfr) introduced earlier in section 3.1. Every pixel with fewer that three like neighbours is converted to the identity of the majority of adjacent pixels (or arbitrarily where the count is equal). The process is applied iteratively until the segmentation results in a set of labelled regions $X_1,\ldots,X_n$ that satisfy *Region*($X_1$) &…& *Region*($X_n$). This sequence of activities is indicated below by `get_regions()`. The process has the additional effect of overcoming the effects of region boundary pixel aliasing inherent in digital imaging, and of removing "speckle" noise pixel groups. The image regions are considered to be embedded in a background area, which is not taken as a region.

In step 2 the `relations` array, which records the detector relations between all combinations of regions, is initialised to dc* (`DCSTAR`). The image sized array `mask` is used to track those pixels and regions that have been processed. It is initialised to discount all background pixels.

In step 3 each region value is compared to its neighbour (here, `neighbour()` returns the coordinates of an adjacent pixel) and processes each region as it encounters each new region edge. `TraceOutline()` recursively traces around the outside edge of the new region building a mask (`omask`) of those connected edge pixels. Note that here `all_neighbours()` successively returns the coordinates of all eight connected pixels. Next, `FillRegion()` applies a recursive flood-fill to the current region, noting the edge pixels encountered (`IsEdge()`). Any edge pixel (that also belongs to a region) that has previously been tagged in omask as an outside region indicates that the current and touching regions form a symmetric ec* relationship (`SetECRegion()`), and that any other connected region must therefore form an (asymmetric) internal ntpp*/ntppi* relationship (`SetNTPPRegion()`). Note that under this arrangement evidence of an ntpp* relation effectively takes precedence over that for an ec* one. This process is repeated until all regions have been processed. Finally the `relations` array is displayed and reformatted for further processing (Figure 4).

```
process(raw_image)
{ // Step 1: convert image to regions
  proc_image = get_regions(raw_image);
  forevery(x, y)
    r(x,y) := ToRegionValue(proc_image[x,y]);
  // Step 2: initialise relations[] to dc*
  forevery(i < maxr,j < maxr)
    relations[i,j] := DCSTAR;
  initialise(mask);
  // Step 3: process all regions
  forevery(x, y)
    { p = r[x,y]
    if(p != r[neighbour(p)] && ¬mask[neighbour(p)]
      { clear(omask);
      cval := r[neighbour(p)];
      TraceOutline(cval,x,y);
      FillRegion(cval,x,y); } }
  Display_regions(r); }

TraceOutline(cval,x,y)
{ if(r[x,y] == cval) return;
  if(omask[x,y]) return;
  if(IsOuterEdge(cval,x,y)) return;
  omask(x,y) := 1;
  forevery({x',y'} := all_neighbours(x,y))
    TraceOutline(cval,x',y'); }

FillRegion(cval,x,y)
{ if(mask[x,y]) return;
  if(r[x,y] != cval) return;
  IsEdge(cval,x,y);
  mask[x,y] := 1;
  forevery({x',y'} := all_neighbours(x,y))
    FillRegion(cval,x',y'); }

bool IsOuterEdge(cval,x,y)
{ result := false;
  if(r[x,y] == cval) return(result);
  forevery({x',y'} := all_neighbours(x,y))
    { if(r[x',y'] == cval)
      { omask[x',y'] := 1; result := true; } }
  return(result); }

IsEdge(cval,x,y)
{ forevery({x',y'} := all_neighbours(x,y))
    { if(cval != r(x',y'))
      if(omask[x',y'])
        SetECRegion(x,y,x',y');
      else
        SetNTPPRegion(x,y,x',y'); } }

SetECRegion(x1,y1,x2,y2)
{ if(relations(r[x1,y1],r[x2,y2]) == DCSTAR)
    { relations[r[x1,y1],r[x2,y2]] = ECSTAR;
    relations[r[x2,y2],r[x1,y1]] = ECSTAR; } }

SetNTPPRegion(x1, y1, x2, y2)
{ relations[r[x1,y1],r[x2,y2]] = NTPPISTAR;
  relations[r[x2,y2],r[x1,y1]] = NTPPSTAR; }
```

Figure 3 shows one part-processed image in a sequence captured from the viewpoint of our upper torso humanoid robot *Ludwig*. The objects are each of a distinctive colour and the raw image was captured using a Canon A80 digital camera, then suitably cropped and re-sized, but otherwise unmodified, prior to presentation to the program. The hue bands were adjusted by hand to achieve a clear region segmentation. Figure 4 shows the result of applying the algorithm to the image; note the diagonal symmetry.

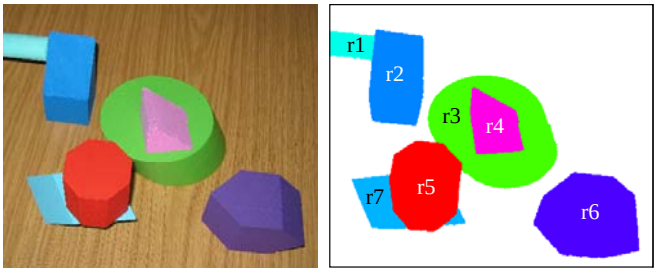


**Figure 3:** Camera image before and after region extraction

| | r1 | r2 | r3 | r4 | r5 | r6 | r7 |
|---|---|---|---|---|---|---|---|
| r1 | - | ec* | dc* | dc* | dc* | dc* | dc* |
| r2 | ec* | - | dc* | dc* | dc* | dc* | dc* |
| r3 | dc* | dc* | - | ntppi* | ec* | dc* | dc* |
| r4 | dc* | dc* | ntpp* | - | dc* | dc* | dc* |
| r5 | dc* | dc* | ec* | dc* | - | dc* | ec* |
| r6 | dc* | dc* | dc* | dc* | dc* | - | dc* |
| r7 | dc* | dc* | dc* | dc* | ec* | dc* | - |

**Figure 4:** The resulting detector relations matrix

## 6 RECOVERING OCCLUSION RELATIONS

We use the *Event Calculus* [14,15], with its primitive ontology of actions (and events), fluents, and time points, for representing and reasoning about occlusion events over time. For the purposes of this paper we will restrict our discussion to the *Happens(α,t)* and the *HoldsAt(β,t)* predicates. Visual occlusion events are modelled as time indexed *fluents* arising from the result of actions, e.g. moving and changing the viewpoint. Thus, for example, where from viewpoint *v*, body *A* partially hides body *B* at time *t*: this becomes: *HoldsAt(PH(A,B,v),t).*

As a simple example, consider the sequence where two bodies A and B pass through the *LOS-14* transition path: *C(A,B)→ JC(A,B)→ PH(A,B)→ JF(A,B)→ F(A,B)* respectively from times *t0* to *t4* – see Figure 1. In the interest of clarity only, we will assume a set of totally ordered time points: *t0* through to *t4*. We also will assume that throughout this period the bodies are visible. Initially, we infer *C(A,B)* at *t0* (from dc*) and *F(A,B)* (from ntpp*) at *t4*, while all the intermediate states between the time points *t1* to *t3* currently remain indistinguishable. This is resolved as follows.

From *Happens(dc*(image(A,v0),image(B,v0),t0)* we can abductively infer *HoldsAt(C(A,B,v0),t0),* and from the change of the detector at *t1* (with only one direct *CND* transition) that *HoldsAt(JC(A,B,v1),t1)* occurred. Given *Happens(ntpp*(image(A, v4),image(B,v4),t4)* we know *HoldsAt(F(A,B,v4),t4).* From the *CND* only *EH(A,B)* and *JF(A,B)* neighbour *F(A,B).* However, given that during period from *t0* to *t4*, the detection of both *A* and *B* took place, then *EH(A,B)* at *t3* is ruled out – leaving *Holds(JF(A,B,v3),t3)*. And given it is not possible to pass from *JF* directly to *JC* without passing through *PH* (from the *CND*), then another event is interpolated, namely: *HoldsAt(PH(A,B,v2),t2*) thus completing the sequence and satisfying the constraints imposed by the *CND*.

As for the remaining paths, clearly, parity of reasoning accounts for the path: *C(A,B)→ JC(A,B)→ PHi(A,B)→ JFi(A,B)→ Fi(A,B)*. By registering when a body is or is not visible, other sequences can be similarly determined: for example, PH→ *EH→ F* (and its inverse sequence). It turns out that within this dynamic setting, 10 of the 14 relations (interpreted as occlusion events) can now be accounted for, leaving each of the pairs: *JH* and *H*; and *JHi* and *Hi* as necessarily indeterminate. Hence, despite the paucity of the visual detectors as to occlusion events, and with minimal modelling assumptions, it becomes possible to infer the occlusion of bodies in three-dimensional space. A similar approach appears in [16] where *Aspect Graphs* replace the role of CNDs

## 7 CONCLUSIONS

This paper demonstrates how, within an abductive framework, a rich set of occlusion spatial relations can be reliably inferred using a sparse and simple set of region-relation detectors. The decision to use simple detectors is justified on several grounds. Firstly, the difficult task of providing a clear interpretation for the detection of fine-tuned spatial relationships is now transferred to the inference mechanism. Constraints within the domain theory and associated *CND* are used in the recovery and justification of the interpolated relation made. Secondly, the sparse set of detectors is robust in use and hence less likely to return hypotheses that will later need to be revised. Thirdly, that the theoretically motivated minimal 3-adjacency neighbour property of pixels in segmented regions (and assumed by our algorithm) has additional practical consequence of providing a despeckle and general clean-up operation on raw images. This not only reduces the need to label more potential regions, it also reduces the need to explain away data not directly supported by the intended model.

We have concentrated on region-contact spatial relations. However, other spatial logics can be similarly grounded in image data to capture more detail in the model. For example, variants on Allen's Interval logic [1] can be used to encode regions being to the left/right or above/below each other - see [10] for an example of this. While *LOS-14* assumes a domain of discrete convex bodies, the segmentation algorithm used does not capture this geometric property. However, by implementing this geometrical property, 7 of the 14 *LOS-14* relations can be directly detected. In the case where convexity condition no longer applies, *ROC-20* can used to replace *LOS-14* and similarly mapped to the set of detectors – see [10] for the mapping between the relation sets: *ROC-20*, *LOS-14* and *RCC-8.*

## ACKNOWLEDGEMENTS

This work was supported by EPRSC under grant: EP/C530683/1(P) "Abductive Robot Perception: Modelling Granularity and Attention in Euclidean Representational Space", which is gratefully acknowledged. We also wish to thank Murray Shanahan, and to the anonymous referees for detailed comments on an earlier draft of this paper.

## REFERENCES

[1] Allen, J. F. (1983) Maintaining Knowledge about Temporal Intervals, Communications of the ACM, 26(11), pp. 832-843

[2] Bennett, B. (1994) Spatial Reasoning with Propositional Logic, KR-94, pp. 51-62

[3] Cohn, A. G. (1997) Qualitative Spatial Representation and Reasoning Techniques, KI-97, Springer LNAI 1303, pp. 1-30

[4] Freksa, C. (1992) Conceptual neighborhood and its role in temporal and spatial reasoning, in: Singh, M., Trave-Massuyes, L. (eds.) Decision Support Systems and Qualitative Reasoning, North-Holland, Amsterdam, pp. 181-187

[5] Galton, A. P. (1994) Lines of Sight, in AISB Workshop on Spatial and Spatio-Temporal Reasoning

[6] Galton, A. P. (1999) The Mereotopology of Discrete Space, in Proc. COSIT 1999, pp. 251-266

[7] Klette, R. and Rosenfeld, A. (2004) Digital Geometry: Geometric Methods for Digital Picture Analysis, Morgan Kaufmann, Elesevier Inc., San Francisco.

[8] Kohler, C. (2002), The Occlusion Calculus, in Proc. Workshop on Cognitive Vision, Zurich, Switzerland

[9] Randell, D. A., Cui, Z. and Cohn A. G. (1992) A Spatial Logic Based on Regions and Connection, KR-92, pp. 165-176

[10] Randell, D. A., Witkowski, M. and Shanahan, M. (2001) From Images to Bodies: Modelling Spatial Occlusion and Motion Parallax, IJCAI-01, pp. 57-63

[11] Randell, D. A. and Witkowski, M. (2002) Building Large Composition Tables via Axiomatic Theories, KR-02, pp. 26-36

[12] Randell, D. A. and Witkowski, M. (2004) Tracking Regions Using Conceptual Neighbourhoods, ECAI-2004 Workshop on Spatial and Temporal Reasoning, pp. 63-71

[13] Roy, A. J. and Stell, J. G. (2002) A Qualitative Account of Discrete Space, GIScience 2002, pp. 276-290

[14] Shanahan, M. P. (1997), Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia, MIT Press

[15] Shanahan, M. P. (1999) The Event Calculus Explained, in Artificial Intelligence Today, Springer LNAI 1600, pp. 409-430

[16] Shanahan, M. P. and Randell, D. A. (2004) A Logic-Based Formulation of Active Visual Perception, KR-04, pp. 64-72