Thinking Inside the Box: A Comprehensive Spatial Representation for Video Analysis

Anthony G. Cohn
School of Computing
University of Leeds
Leeds, England
a.g.cohn@leeds.ac.uk

Jochen Renz
Research School of Computer Science
The Australian National University
Canberra, ACT, Australia
jochen.renz@anu.edu.au

Muralikrishna Sridhar
School of Computing
University of Leeds
Leeds, England
krishna@comp.leeds.ac.uk

Abstract

Successful analysis of video data requires an integration of techniques from KR, Computer Vision, and Machine Learning. Being able to detect and to track objects as well as extracting their changing spatial relations with other objects is one approach to describing and detecting events. Different kinds of spatial relations are important, including topology, direction, size, and distance between objects as well as changes of those relations over time. Typically these kinds of relations are treated separately, which makes it difficult to integrate all the extracted spatial information. We present a uniform and comprehensive spatial representation of moving objects that includes all the above spatial/temporal aspects, analyse different properties of this representation and demonstrate that it is suitable for video analysis.

Introduction

The field of Qualitative Spatial and Temporal Representation (QSTR) is now quite mature, with many calculi having been defined, and their computational properties having been well investigated(Cohn and Renz 2007). Increasingly, these calculi are being used for applications ranging from natural language semantics, robotics, GIS, and of particular concern to us here, high level interpretation of video data, e.g. (Sridhar, Cohn, and Hogg 2010; 2011b). The application of QSTR in video interpretation has been advocated for a variety of reasons; prime amongst these is that noise, and unimportant variations in occurrences of events can be abstracted away from at a qualitative level, rendering occurrences of events of the same kind identical, or at least much more similar.

Depending on the behaviour of the objects involved, it can be appropriate to model many different aspects of space. These include mereotopology, in which case one of the RCC calculi have been used (Sridhar, Cohn, and Hogg 2010; Dubba, Cohn, and Hogg 2010), relative trajectories or direction (e.g. (Fernyhough, Cohn, and Hogg 2000)). Other aspects of qualitative spatio-temporal information could be relevant, including relative sizes (Gerevini and Renz 2002) and relative speed (Delafontaine, Cohn, and de Weghe 2011). As has been remarked (Wölfl and Westphal 2009), the task of combining multiple representations is not entirely trivial, and a choice has to be made between an integrated calculus, and a loose combination of separate ones; however in the latter case there can be representational inefficiencies due to overlapping aspects of the calculi, which can also cause issues for inference, in particular detecting inconsistencies (Gerevini and Renz 2002).

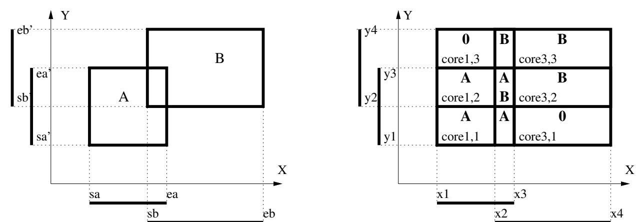

Figure 1: Two rectangles A and B and their projections (left). How the projections define the 9 cores (right).

Since a common representation of objects in video analysis is to use their minimum bounding rectangles (MBR)(de Campos et al. 2011; Thirde et al. 2007), rather than a more precise shape representation (although these are also used), we focus here on an integrated spatio-temporal representation for such rectangular regions. The fact that all regions are one-piece, rectangular, and aligned to the spatial axes, allows a number of representational efficiencies to be made.

A Comprehensive Rectangle Representation

In this section we formally define, motivate and explain our new representation. Our goal is to develop a comprehensive rectangle representation that allows us to represent all required spatial information between 2 rectangles. This includes the topology, direction, size, distance, and motion, which refers to the change of the other aspects over time.

Topology and direction between 2 rectangles is captured by the Rectangle Algebra (RA) (Balbiani, Condotta, and del Cerro 1999). The RA projects 2 axis-parallel rectangles A and B onto the x and y axes, which leads to 2 corresponding intervals on each axis. It then takes the Interval Algebra (IA) (Balbiani, Condotta, and del Cerro 1999) relation between each pair of intervals and defines an RA relation as a pair of IA relations, one for each axis (see Fig. 1). Since there are 13 IA relations, this leads to $13*13 = 169$ different RA relations. It is then straightforward to extract topology (e.g. their RCC8 relationship (Cohn and Renz 2007))[1] as

[1]This is possible since the rectangles are aligned to the axes – in

well as the external direction between A and B. To some degree we can also obtain the internal direction between A and B. Representation of motion is very limited in the RA as it is only possible to represent changes of RA relations over time, e.g. if $(<,<)$ changes to $(m,<)$ we know that A moved closer to B on the x axis. That is we only represent qualitative changes in topology or direction, but cannot represent intermediate changes such as "A is moving closer to B" that do not lead to a change in topology or direction. This requires additional size and distance information.

Consider again Fig. 1. The RA compares the interval relations between $[s_A, e_A]$ and $[s_B, e_B]$ and between $[s'_A, e'_A]$ and $[s'_B, e'_B]$. However, in order to represent relative movement of rectangles over time, we need to look at other intervals. E.g., assume that e_A is before s_B, then the relative change of the length of interval $[e_A, s_B]$ corresponds to whether A moves closer to B or away from B in the x-direction. Similarly, if the 2 intervals $[s_A, e_A]$ and $[s_B, e_B]$ overlap and if the interval $[s_B, e_A]$ becomes bigger, then A and B overlap more. If $[s_B, e_A]$ becomes smaller, then the area of B that is not overlapped by B becomes smaller. These different intervals are also useful in the static case as it allows us to express internal directions: e.g., that B is inside A and that it is in the bottom left corner of A.

In this way each possible interval between the points s_A, s_B, e_A, and e_B on each axis carries information and allows us to specify the relationship between 2 rectangles and how it changes over time in a much more expressive way than only considering RA relations.[2] For 4 endpoints s_A, s_B, e_A, and e_B we get up to 6 different intervals between these endpoints, on both axes up to 12.[3] It will always be the case that $s_A < e_A$ and $s_B < e_B$, but depending on the RA relation between A and B, it is possible that 2 of these endpoints are identical and we get only 3 different intervals. If 2 pairs are identical, we get only one interval on that axis.

Topology and Direction We propose a representation that captures all this information and allows a very detailed and comprehensive representation of the spatial and spatio-temporal relationships between rectangles (see Fig. 1).

Definition 1 (region of interest, core)**.** Given a 2D space $\mathcal{U}$ with 2 reference axes x and y and 2 rectangles $A, B \in \mathcal{U}$ that are parallel to x and y. Projecting A and B to x and y gives us 2 intervals $[s_A, e_A]$ and $[s_B, e_B]$ on x and 2 intervals $[s'_A, e'_A]$ and $[s'_B, e'_B]$ on y. For each axis, we rename and order the 4 endpoints of the 2 intervals such that $x_1 \leq x_2 \leq x_3 \leq x_4$ and $y_1 \leq y_2 \leq y_3 \leq y_4$.

The *region of interest* for rectangles A, B (written $roi(A, B)$) is the rectangle bounded by the intervals $[x_1, x_4]$ and $[y_1, y_4]$. The regions bounded by the intervals $[x_i, x_{i+1}]$ and $[y_j, y_{j+1}]$ are the *cores* of $roi(A, B)$, written as $core_{i,j}(A, B)$, for $1 \leq i, j \leq 3$. We call the intervals $[x_i, x_{i+1}]$ and $[y_j, y_{j+1}]$ for $1 \leq i, j \leq 2$ the *core intervals* on x or y, or the *core widths* and *core heights*, respectively.

It is clear that $roi(A, B)$ consists of exactly 9 cores. Since the underlying core intervals can have length 0, each core is either a rectangle, a line segment, or a point. The cores form a $3x3$ grid that divide the *roi* into 9 zones. A core is either part of A or not part of A and similarly for B. This leads to 4 different states. We introduce a 5th state in order to distinguish cores with a lower dimension.

Definition 2 (core state, region state)**.** Given 2 rectangles $A, B \in \mathcal{U}$, their $roi(A, B)$ and its 9 cores $core_{i,j}(A, B)$, for $1 \leq i, j \leq 3$, the *state of a core*, $state(core_{i,j}(A, B))$, in short $state_{i,j}(A, B)$, is defined as follows:

If $core_{i,j}(A, B)$ is a two dimensional region, then:
$state_{i,j}(A, B) = AB$, iff $core_{i,j}(A, B) \subseteq A \cap B$
$state_{i,j}(A, B) = A$, iff $core_{i,j}(A, B) \subseteq A - B$
$state_{i,j}(A, B) = B$, iff $core_{i,j}(A, B) \subseteq B - A$
$state_{i,j}(A, B) = \square$, iff $core_{i,j}(A, B) \not\subseteq A \wedge core_{i,j}(A, B) \not\subseteq$B

If $core_{i,j}(A, B)$ is a line segment or a point, we set its state to $\emptyset$. We write the state of $roi(A, B)$ as a 9-tuple $state(A, B) = [state_{1,1}(A, B), state_{1,2}(A, B), state_{1,3}(A, B), state_{2,1}(A, B), state_{2,2}(A, B), state_{2,3}(A, B), state_{3,1}(A, B), state_{3,2}(A, B), state_{3,3}(A, B)]$

Since A and B are rectangles, it is clear that not all states are possible. In fact the set of all cores that are part of a single rectangle A, i.e., that have state A or AB, must always form a rectangle too. Therefore, there are only 36 different possibilities of how A and B can be distributed over the cores.

However, not all combinations of the 36x36 assignments are possible states. E.g. if all 9 cores are part of A (and thus have either state A or AB), then only the centre core ($core_{2,2}$) can have state AB, since in all other cases some cores would have state $\emptyset$. We can show that only 169 different states are possible, corresponding exactly to the 169 RA relations (see www.comp.leeds.ac.uk/qsr/cores for a depiction and their correspondence with the RA relations).

Theorem 1. *Given 2 rectangles A, B and their $roi(A, B)$, there are 169 different states $state(A, B)$; these have a 1-to-1 correspondence to the 169 relations of the RA.*

Independent of how the 2 rectangles move, the location of each of the 9 cores with respect to its neighboring cores always stays the same, they always touch each other at the boundary. Since the cores can have zero width, it is possible that non-neighboring cores touch each other as well. This case is clearly determined by the state of the cores, and happens only if a core in the middle of a row or column (i.e., $i = 2$ or $j = 2$) has state $\emptyset$. Since location of cores and corresponding RA relation between cores are determined by $state(A, B)$, which covers the topology and direction between rectangles, the main additional information we can get is the size of cores as well as their width and height. It is clear that all 3 cores in the same row have the same height and that all 3 cores in the same column have the same width. Therefore, the width and height of all cores only depends on the length of the 6 corresponding core intervals $[x_i, x_{i+1}]$ and $[y_j, y_{j+1}]$, respectively, for $1 \leq i, j \leq 3$.

Size and Distance For the purpose of video analysis, the

general, for non aligned rectangles and non-convex regions, it is not possible to infer the RCC8 relationship from an RA relationship.

[2]This holds even if we use what could be called RA-INDU, the extension of RA that uses INDU (Pujari, Kumari, and Sattar 1999) for representing relations between the underlying intervals.

[3]These are called the implicit intervals of IA in (Renz 2012).

exact values of the width, height, and area of cores is generally not important. Instead, we are mainly interested in relative size measures. By comparing which core is larger than which other core, we can infer information such as relative closeness of rectangles. E.g., A is contained in B and is close to the bottom right corner of B. By comparing how the size of a core at one time point compares to its size at the next time point we can infer how objects move relative to each other. Given 6 intervals, we have to keep track of 15 relative size comparisons ($6 * 5/2 = 15$) between these intervals. However, rectangles can consist of multiple cores and in order to accurately compare their sizes, we may have to compare the sizes of combinations of cores together. This gives us 6 possible intervals on each axis, the 3 core intervals plus all possible unions of neighboring core intervals, a total of 12 intervals. We call these 12 intervals the *rectangle intervals* (RIs) $[X_i, X_j]$ and $[Y_i, Y_j]$ for any $1 \leq i < j \leq 4$, which give us $12 * 11/2 = 66$ different size comparisons.

A comparison of the relative size of the area of the 9 cores requires keeping track of $9 * 8/2 = 36$ different relative size relationships. A relative size comparison of all 36 possible rectangles would lead to $36 * 35/2 = 630$ relationships, even though many of them can be inferred from other relations.

Changes over Time When comparing changes over different time points, we can compare same with same, or we can compare everything with everything. In this paper we restrict ourselves to the same with same case and we compare each core with itself at different time points to see how the cores change. If we compare the relative height and width of cores, we need 6 comparisons altogether, for the area we need 9 comparisons, one for each core. We can also compare changes over time between sets of cores forming rectangles, in which case we have to compare 12 different intervals over different time points. For comparing changes over time for all possible rectangles formed by cores, we need 36 comparisons. The changes are recorded using a change function.

Definition 3 (change function, changes)**.** Given a set $\mathcal{V}$ of k variables $v_1, \ldots, v_k$ over a domain $\mathcal{D}$ and an order relation R on $\mathcal{D}$. V_t is the assignment of values from $\mathcal{D}$ to each variable of $\mathcal{V}$ at time point t. We define a *change function* $ch(\mathcal{V}) : \mathcal{V} \mapsto \{<, =, >\}$ where $ch(v_t) =$ '=' if $v_t - v_{t-1} = 0$, $ch(v_t) =$ '<' if $v_{t-1} - v_t < 0$, and $ch(v_t) =$ '>' if $v_t - v_{t-1} < 0$. The *changes from* V_{t-1} *to* V_t for each of the k variables of $\mathcal{V}$, written as $ch(\mathcal{V})$, is the k-tuple $[ch(v_1), \ldots, ch(v_k)]$.

Building an Integrated Representation In order to obtain qualitative information from video, we have to detect and track the relevant objects and their MBRs. For every pair of MBRs A, B and in every frame, we then record the 4 x and the 4 y coordinates of the 4 lines bounding A and B in both directions. These coordinates define the 9 cores as described above. We then determine the status of each core with respect to A and B. This is all the information we need to extract. All qualitative relations can be inferred from this information: the RA relations between A and B which give us topology and direction are derived from $state(A, B)$, all relative size information between the cores can be derived from the x and y coordinates of the bounding lines.

Instead of keeping track of 66 different relative size comparisons between 2 rectangles A and B, we choose a more compact representation of the relative size of their cores. We take the 12 RIs and determine their total order with respect to their length. Some of these intervals might have the same length and for some, the order is predetermined due to one being contained in the other. We then assign each of the 12 RIs its rank in the total order, with rank 1 being assigned to the smallest interval. Intervals of same size will get the same rank. If m intervals have the same rank k, then the next largest interval will get rank $k + m$. The highest rank is less than 12 if the largest interval has the same length as another interval. It is clear that we can obtain each of the 66 different relative size relations between the 12 RIs by comparing their rank: if they have equal rank, they have equal size, the one with lower rank is smaller.

Definition 4 (rank, ranking)**.** Given a set $\mathcal{V}$ of k variables $v_1, \ldots, v_k$ over a domain $\mathcal{D}$ and an order relation R on $\mathcal{D}$. For each assignment of values to variables in $\mathcal{V}$, we can sort $\mathcal{V}$ with respect to R. $rank(v) : \mathcal{V} \mapsto \{1, \ldots, k\}$ is the *rank* of $v \in \mathcal{V}$ wit respect to R, where same value implies same rank. The *ranking* of $\mathcal{V}$, written as $ranking(\mathcal{V})$, is the k-tuple $[rank(v_1), \ldots, rank(v_k)]$.

Integrating all the concepts we defined so far gives us a compact representation of the relevant qualitative information about 2 rectangles.

Definition 5 (CORE-9, CORE-9$^+$, CORE-9^{++})**.** Given 2 rectangles $A, B \in \mathcal{U}$, their $roi(A, B)$, the set $\mathcal{C}_{(A,B)}$ of 9 cores $core_{i,j}(A, B)$, for $1 \leq i, j \leq 3$, the set $\mathcal{C}^+_{(A,B)}$ that consists of all 36 unions of cores that form rectangles, the set of 6 core intervals $\mathcal{CI}_{(A,B)}$, and the set of 12 rectangle intervals $\mathcal{CI}^+_{(A,B)}$. In order to refer to a specific element in $\mathcal{CI}_{(A,B)}$ we use as superscript the interval/core we are referring to, e.g. $\mathcal{CI}^{[x_1,x_2]}_{(A,B)}$, or $\mathcal{C}^{[2,3]}_{(A,B)}$.

CORE-9(A, B, t) is a qualitative representation of rectangles A and B at time t with the following components:
$state(A, B)$, the state of the 9 different cores;
$ranking(\mathcal{C}_{(A,B)})$, the ranking of the core areas;
$ranking(\mathcal{CI}_{(A,B)})$, the ranking of the core intervals;
$ch(\mathcal{C}_{(A,B)})$, the changes of $\mathcal{C}_{(A,B)}$ compared to time $t - 1$;
$ch(\mathcal{CI}_{(A,B)})$, the changes of $\mathcal{CI}_{(A,B)}$ compared to time $t - 1$

CORE-9$^+$ uses the 12 RIs $\mathcal{CI}^+_{(A,B)}$ instead of $\mathcal{CI}_{(A,B)}$. In addition, CORE-9^{++} also uses $\mathcal{C}^+_{(A,B)}$ instead of $\mathcal{C}_{(A,B)}$

A *CORE-9 relation* is any valid assignment of $state(A, B)$, $ranking(\mathcal{CI}_{(A,B)})$, $ranking(\mathcal{C}_{(A,B)})$, $change(\mathcal{C}_{(A,B)})$, and $change(\mathcal{CI}_{(A,B)})$.

Note that $ch(\mathcal{CI}_{(A,B)})$ and $ch(\mathcal{C}_{(A,B)})$ are not completely independent. E.g. if $ch(\mathcal{CI}^{[x_2,x_3]}_{(A,B)})$ ='<' and $ch(\mathcal{CI}^{[y_1,x_2]}_{(A,B)})$ ='<', then $ch(\mathcal{C}^{[2,1]}_{(A,B)})$ must be '<' too.

Expressiveness of CORE-9

CORE-9 integrates a number of different aspects of space. We have already shown that it covers the RA relations, that is topology and direction between rectangles; indeed, one can straightforwardly define RCC-8 relations from CORE-9 and the relations of a qualitative direction calculus such as the Cardinal Direction Calculus (CDC) (Ligozat 1998). It is fairly straightforward to show that it also covers relative size and relative distance and, what is more, that it integrates these 4 aspects. With this integration, it is possible to further refine the different aspects. E.g., we can refine the topological overlap relation by direction, i.e., from which direction one rectangle overlaps the other. We can further refine them by relative size, e.g. how much of a rectangle overlaps the other one, and by relative distance. Relative distance in combination with the other types of relations is very powerful. It allows us, for example, to specify if a rectangle overlaps another one closer to the left or right, the top or bottom. Other relative distance measures can further refine the overlap relationship. We can make similar refinements for the other topological relations. In addition, CORE-9 allows us to represent changes of these relations over time. Because of the relative distance relations, we can also express that rectangles approach each other or move away from each other, overlap more or overlap less, and in which directions, in effect defining a form of the QTC calculus ((Delafontaine, Cohn, and de Weghe 2011) for rectangles, which could be used, e.g. to model action sequences such as overtaking cars in a qualitative natural way.

CORE-9 encodes much more than just RCC, CDC and QTC. It is clear that relative size information is also explicitly encoded, via the $ranking$ function of Definition 5. This allows us, for example, to represent that A is in the bottom left of B, or that A overlaps B in the x direction more than in the y direction. CORE-9 can also encode much more directional information than is possible in CDC: it can encode *internal directions*, e.g. to represent that A is in the NE of B. Also for the case of QTC, it can handle cases where A and B are not disjoint, so that it is possible to represent, e.g., that A is part of B, and that it is moving west. CORE-9 can also represent that a region is growing or contracting.

There are some things that CORE-9 cannot represent. Relative speed is one such aspect (the third component of most QTC family calculi) – it would require comparisons between arbitrary RIs at different times, not just the same ones (as is the case in the present $ranking$ function). This would be a straightforward extension.

Smoothness of CORE-9

A very important property that any useful qualitative representation should have is *smoothness*, i.e., if there are small changes to what is being represented, there should only be small changes in the representation. Ideally, only parts of the representation should change, that correspond to parts that are changing, i.e., changes should be *local*. Having smooth and local changes is particularly important for representing moving objects in video, where there are often small changes between every frame. It allows us to define a metric of similarity between different core representations that captures the degree of change – if only local changes exist, then two representations will be broadly similar.

In the following, we sketch how small continuous changes affect our representation. We begin with changes to the state of the 9 cores and analyse which changes to the 2 represented rectangles can cause these changes. There are only two kinds of small continuous changes that have this effect: (1) if the two rectangles share a common edge segment at one time point and the change affects the common edge, or (2) if the two rectangles share a common edge segment which they previously did not share. Since the change is continuous, this means that one of the core intervals changes from non-zero length to zero length, or from zero length to non-zero length. It is clear that for cores corresponding to these core intervals the status changes between $\emptyset$ and either A, B, or AB. We can show in a simple case analysis that all other cores keep their previous states.

Apart from these changes to the state of cores, continuous changes to the rectangles also lead to continuous changes in the sizes of core and RIs. It is clear that these changes lead to smooth and local changes of the sizes relations of the intervals. However, CORE-9 does not represent these relations directly, but only the rank of intervals with respect to their relative size. By analysing how the rank of intervals can change and how this depends on continuous change, we can show that intervals that are not involved in a change do not change their rank, and that intervals that are involved change rank at most to the next higher or lower rank.

Continuous changes of the represented rectangles are also recorded in the $ch(...)$ entries of CORE-9. Clearly these entries only change locally, and they change as smoothly as the rectangles change. As a consequence of our analysis, we obtain the following theorem.

Theorem 2. *CORE-9 is a smooth and local representation, that is, continuous changes to the represented rectangles lead to a smooth and local change in the representation.*

An HMM Framework For Smoothed Relations

In this section we briefly report on 2 experiments comparing the efficacy of the proposed uniform representation against the combination of separate representations on actual video data. First we show that an approach that models topology and direction jointly performs better in obtaining smoothed relations, and secondly that this jointly obtained representation is better at event detection.

We can exploit a Hidden Markov Model (HMM) to obtain smoothed relations from noisy video data, extending the approach of (Sridhar, Cohn, and Hogg 2011a), which showed that jitter of RCC8 relations could be thus reduced. The temporal model used the CN graph (CNG) of RCC8: each state of the HMM is labelled with an RCC8 relation. The transition probabilities are defined using the CNG of RCC8 in such a way that transitions to the same state have a relatively higher probability compared to the transition between states allowable in the CNG and transitions between states not possible according to the CNG have a zero transition probability. A novel distance measure between regions provided the observations for the HMM. The probability distribution between the states and the observations was modelled by an observation model for each state. It is possible to extend this formalism and design a HMM for CORE-9. Our current implementation does not extend to the full CORE-9 representation yet, but focuses on comparing single Multi-

Observation HMM (MOHMM) whose input is defined in terms of a vector intervals on the x and y axes against the relations produced by a pair of non-integrated HMMs for each of the topological and directional aspects separately (a Parallel HMM architecture – PaHMM(Chen et al. 2009)), so that at any time it is possible to infer a pair of topological and directional respectively. Manually annotated spatial relationships are used both for training the parameters of the HMMs and for evaluating them. The evaluation dataset consists of 36 videos each of 150200 frames and containing one or more of 6 verbs: approach, bounce, catch, jump, kick and lift (a subset of the videos and verbs at www.visint.org). The evaluation of the HMMs involved determining the extent to which the episodes output by the system temporally align with those of the ground truth. Accuracy was measured in terms of the mean and variance of the percentage of temporal overlap, between the outcome of each of the HMMs and the ground truth in a 10-fold cross validation: the MOHMM had an accuracy of 72.5%, while the PaHMM with the separate representations only had 62%.

In a second experiment, the relationships between pairs of object tracks obtained by each of the HMMs were *re-represented* in terms of a 3 layered graph structure called interaction graphs(Sridhar, Cohn, and Hogg 2010; 2011b), between which a similarity measure can be defined and used to perform learning tasks such as event detection. We used the event detection framework of the latter paper to learn 2 sets of event models arising from the interaction graphs obtained using PaHMM and the MOHMM respectively. An event was regarded as being detected if the detected interval overlapped the ground-truth interval by more than 50%. On a leave-one out cross validation the event models arising out of the PaHMM outputs yielded a mean F1 score of 38.6%, while the MoHMM yielded 44.5%. Thus in both experiments the integrated representation outperformed the representations computed separately, giving supporting evidence for the benefits of the integrated CORE-9 representation. A full set of experiments for CORE-9 remains to be conducted.

Final Comments

Video analysis presents a challenge to the field of KR. It requires us not only to detect and track objects in video, but to infer what the objects are doing. QSTR provides an effective representation for this task as the exact location and extent of the objects we track is typically much less important than their qualitative relationships. Many different QSTRs have been proposed. The hypothesis we have started to explore in this paper is that an integrated representation may prove to be more effective, as well as being arguably more elegant. Our representation is suited for representing rectangles, which is appropriate when objects are represented by their MBR. Our representation is very compact and can be easily and smoothly extracted from video. This integrated approach appears to give better performance on event detection as demonstrated in a sample dataset.

A variety of future work is possible; some of this has already been alluded to above. A more thorough experimental evaluation is also required. A further avenue is to extend the HMM smoothing to all aspects of the CORE-9 representation, specifically, the relative size information.

Acknowledgements The first and third authors acknowledge financial support of the DARPA Mind's Eye program (project VIGIL, W911NF-10-C-0083). The second author is supported by an ARC Future Fellowship (FT0991917).

References

Balbiani, P.; Condotta, J.-F.; and del Cerro, L. F. 1999. A new tractable subclass of the rectangle algebra. In *Proceedings of IJCAI'99*, 442–447.

Chen, C.; Liang, J.; Zhao, H.; Hu, H.; and Tian, J. 2009. Factorial HMM and parallel HMM for gait recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C* 39(1):114–123.

Cohn, A. G., and Renz, J. 2007. Qualitative spatial reasoning. In van Harmelen, F.; Lifschitz, V.; and Porter, B., eds., *Handbook of Knowledge Representation*. Elsevier.

de Campos, T.; Barnard, M.; Mikolajczyk, K.; Kittler, J.; Yan, F.; Christmas, W. J.; and Windridge, D. 2011. An evaluation of bags-of-words and spatio-temporal shapes for action recognition. In *WACV*.

Delafontaine, M.; Cohn, A. G.; and de Weghe, N. V. 2011. Implementing a qualitative calculus to analyse moving point objects. *Expert Syst. Appl.* 38(5):5187–5196.

Dubba, K. S. R.; Cohn, A. G.; and Hogg, D. C. 2010. Event model learning from complex videos using ILP. In *Proc. ECAI*, 93–98.

Fernyhough, J.; Cohn, A.; and Hogg, D. 2000. Constructing qualitative event models automatically from video input. *Image and Vision Computing* 18:81–103.

Gerevini, A., and Renz, J. 2002. Combining topological and size information for spatial reasoning. *Artif. Intell.* 137(1-2):1–42.

Ligozat, G. 1998. Reasoning about cardinal directions. *J. Vis. Lang. Comput.* 9(1):23–44.

Pujari, A. K.; Kumari, G. V.; and Sattar, A. 1999. Indu: An interval and duration network. In *Proc. of Australian Joint Conference on AI*, 291–303.

Renz, J. 2012. Implicit constraints for qualitative spatial and temporal reasoning. In *Proc. KR*.

Sridhar, M.; Cohn, A. G.; and Hogg, D. C. 2010. Unsupervised learning of event classes from video. In *Proc. AAAI*, 1631–1638. AAAI Press.

Sridhar, M.; Cohn, A. G.; and Hogg, D. C. 2011a. From video to RCC8: Exploiting a distance based semantics to stabilise the interpretation of mereotopological relations. In *COSIT*, volume 6899 of *LNCS*, 110–125. Springer.

Sridhar, M.; Cohn, A. G.; and Hogg, D. C. 2011b. Benchmarking qualitative spatial calculi for video activity analysis. In *Proc. IJCAI Workshop Benchmarks and Applications of Spatial Reasoning*, 15–20.

Thirde, D.; Borg, M.; Aguilera, J.; Wildenauer, H.; Ferryman, J. M.; and Kampel, M. 2007. Robust real-time tracking for visual surveillance. *EURASIP J. Adv. Sig. Proc.* 2007.

Wölfl, S., and Westphal, M. 2009. On combinations of binary qualitative constraint calculi. In Boutilier, C., ed., *IJCAI*, 967–973.