Reasoning about Dynamic Depth Profiles

Mikhail Soutchanski[1] and **Paulo Santos**[2]

Abstract. Reasoning about perception of depth and about spatial relations between moving physical objects is a challenging problem. We investigate the representation of depth and motion by means of depth profiles whereby each object in the world is represented as a single peak. We propose a logical theory, formulated in the situation calculus (SC), that is used for reasoning about object motion (including motion of the observer). The theory proposed here is comprehensive enough to accommodate reasoning about both sensor data and actions in the world. We show that reasoning about depth profiles is sound and complete with respect to actual motion in the world. This shows that in the conceptual neighbourhood diagram (CND) of all possible depth perceptions, the transitions between perceptions are logical consequences of the proposed theory of depth and motion.

1 Introduction

The present paper describes a logical theory for representing knowledge about objects in space and their movements as noted from the viewpoint of a mobile robot. One of the main purposes of this theory is to equip the robot with the basic machinery for deriving and manipulating *symbolic* information about the motion of objects (and the robot itself). Methodologically speaking, the work described here belongs to the *cognitive robotics* area [5, 11]. The aim of cognitive robotics is to endow robots with high-level cognitive skills by importing techniques from the field of knowledge representation, especially those that use formal logic as their theoretical foundation; [5] provides a detailed review of recent theoretical and practical results.

The goal of this paper is to provide a formal account both for perception of motion and for actions by integrating a novel formalism about perception of depth [12] with a principled formalism for handling actions and change in a dynamic world, the situation calculus (SC) [11]. The central idea in the work reported in [12] is the construction of a logical representation for the data obtained from a mobile robot's range finding sensor assuming a simplification of the depth maps[3] in the form of a *horizontal slice* across the centre of the visual field. Each horizontal slice is represented as a two-dimensional profile of a one-and-a-half planar view of a scene, which preserves the horizontal shape and depth of objects in the scene. This planar view constitutes the basic entity of the reasoning system called *depth profile calculus* (DPC). Occlusion and parallax are two main perceptual events that this theory accounts for (inspired by [10] and other qualitative spatial reasoning approaches [1]). In the sequel, we talk about a range finding sensor assuming that it is either a stereo vision system, a laser range finder, or both.

In the present paper we extend the language of DPC with the definitions provided by SC. This new theory also includes concepts of Euclidean geometry to describe visibility and motion of objects (including the observer). Thus, the formalism presented here facilitates not only sensor data assimilation but also reasoning about actions in the world. We expect that this research will result in an automated reasoning system capable of recognizing the plans of other vehicles based on knowledge about its own motion and data from a range finding sensor. The proposed framework can find applications similar to those of cognitive vision systems that summarize conceptually a sequence of images of dynamic scenes in terms of motion verbs or as natural language sentences [4, 7, 8]. However, in contrast to systems reviewed in [8], that see a scene from a birds-eye view, we are interested in scenes from an *ego-centric* viewpoint that is more natural from a cognitive standpoint.

This paper makes several important contributions. To the best of our knowledge, the proposed logical theory (called $\mathcal{TDM}$) is the first theory for reasoning about actions that goes well beyond simple examples discussed in [11] by developing an elaborated theory for reasoning about motion and perception of depth. Also, we show that in the conceptual neighbourhood diagram (CND) of all possible depth perceptions, the transitions between perceptions are logical consequences of the proposed theory of depth and motion. This theory can be used for solving the projection problem (if a logical property holds after doing a sequence of actions), for plan recognition [3], for reasoning about incomplete information and for solving other reasoning tasks that cannot be solved using CNDs alone.

2 Depth Profiles

In this work, space is represented by depth profiles that are graphical descriptions of horizontal slices of depth maps as perceived from the observer's viewpoint. Each depth profile corresponds to one scene and is the result of one horizontal slice taken at mid-height of the scene objects. The robot has a limited field of view defined by the direction that it is facing, the maximum depth perceived and the angle of its camera aperture. The area determined by this angle is called the *visibility cone*.

The depth and the edges of depth map regions intersected by horizontal slices define depth profiles (Figure 1, (b) and (d)). Peaks in these profiles are named *depth peaks* and are the primitive spatial entities that can be sensed by the robot. The edges of a peak in a profile (represented in its horizontal axis, cf. Figure 1(b)) are related to the boundaries of images of visible objects in a horizontal slice, while the vertical dimensions of a peak (the axis *depth*) provide the depth of the perceived objects from the observer's viewpoint. For instance, Figure 1(a) and (c) show an observer perceiving two objects (b_1 and b_2) from two distinct viewpoints, ν_1 and ν_2, respectively. Figures 1(b) and (d) depict the depth profiles relative to ν_1 and ν_2, where b_1 and b_2 are represented by the peaks p and q respectively. The depth L in Fig. 1 is the outermost boundary still visible to the observer.

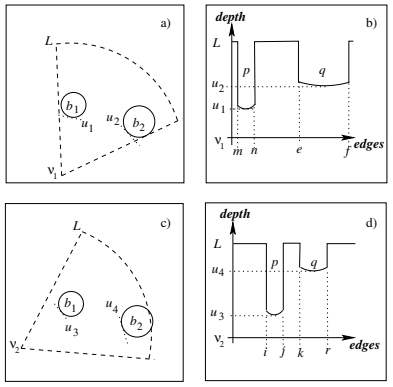

Figure 1. (a) and (c) are birds-eye views of two objects b_1, b_2 from two distinct points ν_i; (b) and (d) depth profiles respective to (a) and (c).

Besides depth and edges, depth profiles encode information about the apparent sizes of visual objects from the robot's perspective. In this work we use the width of peaks as a measure of the apparent sizes (e.g., in Figure 1(b) the size of the peak q is defined by $(f - e)$).

[1] Department of Computer Science, Ryerson University, Toronto, Canada, email: mes@cs.ryerson.ca

[2] IAAA – Elec. Eng. Dep., Centro Universitário da FEI, São Paolo, Brasil, email: psantos@fei.edu.br

[3] A depth map indicates the range from observer to each point on an observed scene.

The observer-relative apparent displacement between pairs of objects can also be retrieved from depth profiles by measuring the distance between the nearest points of distinct depth peaks. In this work, angular distances in a depth profile are measured from the leftmost boundary of the profile to the rightmost point of a peak. Thus, for instance, in Figure 1(b) the observer-relative distance separating p and q (i.e., an angle between adjacent borders of p and q) is given by $f-(f-e)-n=e-n$.

In order to maintain the correspondence between objects and peaks through time, three domain constraints are assumed: *object persistence*, *smoothness of motion*, and *non interpenetration* [6]. Object persistence stands for the assumption that the domain objects cannot disappear or appear instantaneously (without crossing the boundaries). Smoothness of motion is the assumption that objects in the world cannot instantly jump from one place to another. Finally, in an environment where non interpenetration holds, objects do not pass through each other. Depth peaks, however, can *approach* each other, *recede* from each other, *coalesce*, or *split*. Also, an individual peak can *extend* (when an object or the observer approach each other), *shrink* (when they move apart), *appear* (when an object moves closer to an observer through the boundary *L*), *vanish* (when an object moves away beyond the boundary *L*). These changes in peaks are relations in the depth profile calculus (DPC) that represent transitions in the sensor data. These transitions are related to changes in the observed objects in the world and compose the conceptual neighbourhood diagram[4] (CND) shown in Fig. 2 (see also [12]). A CND is a graph whereby in its vertices are relations on some specific objects, and in its edges, the transitions between these relations.

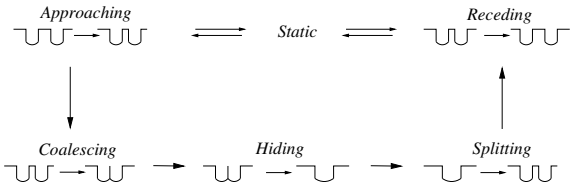

Figure 2. Conceptual neighbourhood diagram

However, DPC is only capable of describing the observed changes in the world, thus overlooking reasoning about the actions that caused these transitions. The present paper deals with this issue by proposing a logical theory of depth and motion such that the transitions in the CND become logical consequences from the proposed theory.

In order to reduce the inherent complexity of dealing with object's shapes, and to facilitate the treatment of motion in the world, this work assumes that the environment is only populated by cylinders. In fact, approximating object's shapes to cylinders is a traditional assumption in Computer Vision, recall David Marr's cylindrical representation of the human body for instance. The assumption that each profile contains exactly one peak per body is an initial approximation to the problem. Future research on noisy sensors will force us to relax this assumption, as (for instance) reflections of light on an observed object may cause it to be perceived as various distinct peaks.

3 Situation Calculus

The situation calculus (SC) is a predicate logic language for axiomatising dynamic worlds. In recent years, it has been considerably extended beyond the original language to include stochastic actions, concurrency, continuous time, etc, but in all cases, its basic ingredients consist of *actions*, *situations* and *fluents* [11]. It can use also additional sorts for objects in the domain.

[4] The CND for single peak relations was omitted here for brevity.

Actions are first-order terms consisting of an action function symbol and its arguments. In the approach to representing time in SC of [11], one of the arguments to such an action function symbol—typically, its last argument—is the time of the action's occurrence. E.g., $endMove(R, loc(3,8), loc(5,10), 105.7)$ denotes the action of the robot ending its motion from location $loc(3,8)$ to location $loc(5,10)$ at time 105.7 (e.g., measured in seconds). There is a corresponding action $startMove(R, loc(3,8), loc(5,10), 44.9)$ that started the process of moving between these two locations at time 44.9. All actions are instantaneous (i.e, with zero duration). Durations of actions extended in time can be captured using processes as in [11]. In addition to physical actions, that change some properties of the world, there are also sensing actions that change what the robot knows about the world (only the robot can do a sensing action). E.g., the action $sense(p, loc(x_r, y_r), t)$ is a sensing action executed at time t from the location $loc(x_r, y_r)$ that gets from sensors a depth profile p which can include one or several peaks.

A *situation* is a first-order term denoting a sequence of actions. Such sequences are represented using a binary function symbol do: $do(\alpha, s)$ denotes the sequence resulting from adding an action term α to the sequence s. The special constant S_0 denotes the *initial situation*, namely the empty action sequence. Every situation refers uniquely to a sequence of actions.

Relations or functions whose values vary from situation to situation are called *fluents*, and are denoted by predicate or function symbols whose last argument is a situation term. To simplify the presentation we do not consider functional fluents. In addition to fluents representing properties of the world, one can reason in the SC not only about effects of physical actions on these properties, but also about sensing actions and their effects on knowledge using an epistemic fluent $K(s', s)$ [5, 11]. In this paper, we consider only a simple form of literal-based knowledge about fluents that can be represented by the fluents themselves (understood as subjectively perceived properties), as proposed in [13]. This approach (knowledge about literals only) is more practical and sufficient for our purposes because we make a full observability assumption: a value of any fluent at each moment of time in any situation is known. The correctness of this approach to reasoning about sensing actions is discussed in [9].

The SC includes the distinguished predicate $Poss(a, s)$ to characterize actions a that are possible to execute in s; see other details and definitions in [11]. A *basic action theory* (BAT) $\mathcal{D}$ is a set of axioms for a domain theory written in the SC with the following five classes of axioms to model actions and their effects.

Action precondition axioms $\mathcal{D}_{ap}$: There is one for each action term $A(\vec{x})$, with syntactic form $Poss(A(\vec{x}), s) \equiv \Pi_A(\vec{x}, s)$. Here, $\Pi_A(\vec{x}, s)$ is an uniform formula with free variables among $\vec{x}, s$. These are the preconditions of action A: A is possible if and only if the condition $\Pi_A(\vec{x}, s)$ is true.

Successor state axioms (SSA) $\mathcal{D}_{ss}$: There is one for each relational fluent $F(\vec{x}, s)$, with syntactic form $F(\vec{x}, do(a, s)) \equiv \Phi_F(\vec{x}, a, s)$, where $\Phi_F(\vec{x}, a, s)$ is a uniform formula with free variables among $a, s, \vec{x}$ having the syntactic form $a = PositiveAction \wedge \gamma^+(\vec{x}, s) \vee \cdots$
$\vee\ F(\vec{x}, s) \wedge \neg(a = NegativeAction \wedge \gamma^-(\vec{x}, s)\ \vee\ \cdots)$,
where *PositiveAction* is an action that has positive effect on the fluent F, $\gamma^+(\vec{x}, s)$ is the formula expressing a context in which this positive effect can occur. Similarly, *NegativeAction* is an action that can make the fluent F false if the uniform formula $\gamma^-(\vec{x}, s)$ holds in the situation s. These characterize the truth values of the fluent F in the next situation $do(a, s)$ in terms of the situation s, and they embody a solution to the frame problem for deterministic actions [11].

Unique names axioms for actions $\mathcal{D}_{una}$: These state that the ac-

tions of the domain are pairwise unequal.

Initial database $\mathcal{D}_{S_0}$: This is a set of sentences whose only situation term is S_0; it specifies the initial problem state.

Foundational axioms Σ for situations with time are given in [11]. These axioms use a new function symbol $start(s)$, denoting the start time of situation s. However, below we use a predicate $start(s,t)$ for similar purposes. (We use predicates rather than functions to make connection with Prolog implementation more transparent.) Similar to [11], we do not write axioms for time t, but assume that it can vary over rationals or reals with the standard interpretation.

4 Depth Profiles in the Situation Calculus

This section describes the logical formalism to represent transitions between depth profiles within the situation calculus. The most important aspect of this formalism is expressed in its successor states axioms (SSA) whereby, as we shall see, the relations representing transitions in the perceptions of depth are combined with the actions that caused such transitions. Therefore, an agent using these axioms is capable of describing (by means of DPC relations) perceived changes in the world as well as reasoning about the effects of its own actions and those of other agents in the domain.

Formally, we introduce a many-sorted first-order language that uses quantifiers over depth profiles (p), time points (t), depth (u), size (z), angular distance (d) between peaks or between a peak and the left border, rotation angles (ω), directions (θ) that the robot is facing (we measure directions in terms of angular distance from North), physical bodies (b), coordinates (x and y) and viewpoints (v) which are locations $loc(x,y)$. This theory also includes the term $pk(b,u,z,d)$ (read as "peak of a body b located at depth u from the current viewpoint has size z and angular distance d from the left border").

Let the motion of the robot or any other body be described by the term $startMove(b,l_1,l_2,time)$ – b starts moving between locations l_1 and l_2 at the moment $time$ – and the term $endMove(b,l_1,l_2,time)$ – b ends the process of moving between l_1 and l_2. The robot R can pan its range finding sensor in any direction: this is represented by the pair of actions $startPan(\omega,time)$ and $endPan(\omega,time)$, where the rotation angle ω is positive if it is clockwise and negative if it is counter-clockwise. The robot R located in $loc(x_r,y_r)$ can also get a profile p from sensors by doing the action $sense(p,loc(x_r,y_r),time)$. These actions are characterized by the following precondition axioms $\mathcal{D}_{ap}$ (where $fieldView(\beta)$ is the visibility cone, $facing$ and $location$ are fluents introduced below).

$poss(startMove(b,l_1,l_2,t),s) \equiv location(b,l_1,s) \wedge$
$\neg\exists l,l' moving(b,l,l',s) \wedge l_1 \neq l_2 \wedge start(s,t') \wedge t \geq t'.$

$poss(endMove(b,l_1,l_2,t),s) \equiv moving(b,l_1,l_2,s) \wedge$
$start(s,t') \wedge t \geq t' \wedge l_1 \neq l_2.$

$poss(startPan(\omega,t),s) \equiv \neg\exists w' rotating(w',s) \wedge start(s,t') \wedge t \geq t'.$

$poss(endPan(\omega,t),s) \equiv rotating(\omega,s) \wedge start(s,t') \wedge t \geq t'.$

$poss(sense(p,loc(x_r,y_r),t_2),s) \equiv$
$start(s,t_1) \wedge t_2 \geq t_1 \wedge location(R,loc(x_r,y_r),s) \wedge$
$(\exists b,u,z,d,\theta,\beta)\, pk(b,u,z,d) \in p \wedge u > 0 \wedge z > 0 \wedge d > 0 \wedge$
$facing(\theta,loc(x_r,y_r),s) \wedge fieldView(\beta) \wedge$
$visible(loc(x_r,y_r),b,\beta,\theta,s) \wedge$
/* there are no invisible peaks in p */
$(\neg\exists b_I,u_I,z_I,d_I)\,(pk(b_I,u_I,z_I,d_I) \in p \wedge$
$\neg visible(loc(x_r,y_r),b_I,\beta,\theta,s)\,),$

or in English, sensing a profile p is a possible action, if p includes a peak (with positive attributes) from a visible object and has no peaks from objects that are currently not visible (given robot's orientation and aperture). The predicate $visible(v,b,\beta,\theta,s)$ means that a body b is visible from the current viewpoint v if the field of view is β and the robot is facing a direction θ in the situation s. This predicate can be easily maintained for any s, following the movements of the observer or other bodies, using a variation of the rotational sweep-line algorithm proposed for straight-line segments by D.T. Lee in 1978 (see [2] for details). For any snapshot of the world s, and a geometric configuration of circles (representing 2D projections of cylinders), this is an efficient algorithm that takes $O(n \cdot \log n)$ time, where n is the number of circles. Each circle is reduced to a chord between tangent points computed by shooting a tangent ray to the circle from the viewpoint. The algorithm answers boolean queries whether a chord is visible (i.e., if at least one point on the chord is visible) or is completely occluded in a current situation. To take into account that the robot's field of view has an angular aperture of $\beta \leq 180^\circ$, this predicate takes an extra argument β. Also, $visible$ takes an argument θ representing the direction that the robot is facing. In the sequel, we will not provide any axioms for this predicate, but rely on its intended interpretation provided by an external computational geometry module. Similarly, we do not provide any axioms for dense linear orders, but assume that time, coordinates and others vary over rationals or reals, with the standard interpretation.

The fluent $facing(\theta,loc(x_r,y_r),s)$ reads as "the robot located in $loc(x_r,y_r)$ is facing the direction that makes an angle θ with North in the situation s"; the fluent $location(b,l,s)$ means that b is located in the point l in s and the fluent $moving(b,l_1,l_2,s)$ represents the process of moving between two locations. The axioms for motion are simple, but they can be easily elaborated by taking into account equations of continuous motion. The predicate $rotating(s)$ holds if the robot pans its range finding sensor in s. These fluents are characterized by the following self-explanatory axioms.

$facing(\theta_2,loc(x_r,y_r),do(a,s)) \equiv\ (\exists t,\omega,\theta_1)$
$(a = endPan(\omega,t) \wedge facing(\theta_1,loc(x_r,y_r),s) \wedge \theta_2 = \theta_1 + \omega) \vee$
$facing(\theta_2,loc(x_r,y_r),s) \wedge (\neg\exists t,\omega)(a = endPan(\omega,t)).$

$location(b,loc(x,y),do(a,s)) \equiv\ (\exists t,x',y')$
$a = endMove(b,loc(x',y'),loc(x,y),t) \vee location(b,loc(x,y),s) \wedge$
$(\neg\exists t,x_2,y_2)(a = endMove(b,loc(x,y),loc(x_2,y_2),t)\,).$

$moving(b,l_1,l_2,do(a,s)) \equiv\ (\exists t) a = startMove(b,l_1,l_2,t) \vee$
$moving(b,l_1,l_2,s) \wedge (\neg\exists t)(a = endMove(b,l_1,l_2,t)).$

$rotating(\omega,do(a,s)) \equiv\ (\exists t) startPan(\omega,t) \vee$
$rotating(\omega,s) \wedge (\neg\exists t)(a = endPan(\omega,t)).$

In the sequel, we use a convenient abbreviation for the Euclidean distance between two points:

$$euD(loc(x_1,y_1),loc(x_2,y_2),dist) \stackrel{def}{=} dist = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2}.$$

Three further predicates refer to peak attributes: the peak's depth, size, and the angular distance of a peak from the left border. The predicate $depth(pk(b,u,z,d),u,loc(x_r,y_r),s)$ holds if the peak's depth is u in the situation s when the viewing point is in $loc(x_r,y_r)$. Similarly, $size(pk(b,u,z,d),z,$-$loc(x_r,y_r),s)$ holds in s if the peak's angular size is z. Finally, $dist(pk(b_1,u_1,z_1,d_1),pk(b_2,u_2,z_2,d_2),d,loc(x_r,y_r),s)$ holds if the angular distance between two peaks is d in the situation s. Below, only the SSA for $depth$ is shown, $size$ and $dist$ are analogous.

The predicate $depth(pk(b,u,z,d),u,loc(x_r,y_r),do(a,s))$ holds after the execution of an action a at a situation s if and only if a was a sensing action that picked out the peak of b with depth u or the robot R (or an object b) moved to a location such that the Euclidean distance from the object to the observer (the depth of the object b) becomes u in the resulting situation. This SSA is formally expressed in the following formula, that also includes a frame axiom stating that the value of the fluent $depth$ remains the same in the absence of any action that explicitly changes its value.

$depth(pk(b,u,z,d),u,loc(x_r,y_r),do(a,s)) \equiv$
$(\exists t,p) a = sense(p,loc(x_r,y_r),t) \wedge pk(b,u,z,d) \in p \ \vee$
$(\exists t,x,y,x_1,y_1,r,e)(a = endMove(R,loc(x_1,y_1),loc(x_r,y_r),t) \wedge$
$location(b,loc(x,y),s) \wedge location(R,loc(x_1,y_1),s) \wedge$
$radius(b,r) \wedge euD(loc(x,y),loc(x_r,y_r),e) \wedge (u = e - r)) \ \vee$
$(\exists t,x_1,y_1,x_2,y_2,r,e)(a = endMove(b,loc(x_1,y_1),loc(x_2,y_2),t) \wedge$
$location(R,loc(x_r,y_r),s) \wedge location(b,loc(x_1,y_1),s) \wedge$
$radius(b,r) \wedge euD(loc(x_r,y_r),loc(x_2,y_2),e) \wedge (u = e - r)) \vee$
$depth(pk(b,u,z,d),u,loc(x_r,y_r),s) \wedge$
$location(R,loc(x_r,y_r),s) \wedge (\exists x,y).location(b,loc(x,y),s) \wedge$
$(\neg\exists t,l,p',u',z',d',x_1,y_1)\ (a = endMove(R,loc(x_r,y_r),l,t) \vee$
$a = endMove(b,loc(x,y),loc(x_1,y_1),t) \ \vee$
$a = sense(p',loc(x_r,y_r),t) \wedge pk(b,u',z',d') \in p' \wedge u \neq u').$

In addition to the predicates on peak attributes we can define a set of relations representing *transitions* between attributes of single peaks. These transitions account for the perception of moving bodies and can be divided into two kinds: predicates referring to *transitions in single peaks* and *transitions between pairs of peaks*.

Transitions on single peaks are: $extending(pk(b,u,z,d),loc(x_r,y_r),s)$, which states that a peak $pk(b,u,z,d)$, representing an object b, is perceived from $loc(x_r,y_r)$ as extending (or expanding in size) in situation s; $shrinking(pk(b,u,z,d),loc(x_r,y_r),s)$, states that $pk(b,u,z,d)$, representing a visible object b, is shrinking (contracting in size) in s; $appearing(pk(b,u,z,d),loc(x_r,y_r),s)$ means that $pk(b,u,z,d)$, unseen in a previous situation, is perceived in a situation s; and, $vanishing(pk(b,u,z,d),loc(x_r,y_r),s)$ that represents the opposite of $appearing$. Finally, $peak\_static$ represents that the peak attributes do not change in the resulting situation $do(a,s)$ wrt s. For instance, SSA for $extending$ (below) states that a peak is perceived as extending in a situation $do(a,s)$ iff there was a sensing action that perceived that its angular size is greater in $do(a,s)$ than in s, or the robot (or the object) moved to a position such that the computed angular size of the object in $do(a,s)$ is greater than its size in situation s. In either case, the depth in both situations, depth u' in $do(a,s)$ and depth u in s, has to be smaller than an L (the furthermost point that can be noted by the robot sensors), representing in this case a threshold on depth that allow the distinction between $extending$ and $appearing$. Thus, if the peak depth u in situation s was such that $u \geq L$, i.e., the peak was too far, but the depth $u' < L$ in $do(a,s)$, i.e., the peak is closer to the viewpoint in the resulting situation, then the peak is perceived as $appearing$, rather than extending ($shrinking$ and $vanishing$ are analogous). Examples of situations in which these fluents hold are given in Figure 1: if the observer moves from viewpoint ν_2 to ν_1 (Figure 1(c) and (a)), the peak from b_2 is perceived as $extending$ (the peak q from b_2 is greater in Figure 1(b) than in (d)). If the change is from ν_1 to ν_2, instead, q would be $shrinking$, whereas if only one of the distances was smaller than L, then q would be $appearing$ or $vanishing$, according to the differences noted in s and in $do(a,s)$. For simplicity, we present a high-level description of the SSA only.

$extending(peak,viewpoint,do(a,s))$ **iff**
- a is a sensing action which measured that the angular size of $peak$ is currently larger than it was at s **or**
- a is an $endMove$ action terminating the process of robot's motion resulting in the viewpoint such that a computed size of $peak$ from the viewpoint is larger than it was at s **or**
- a is an $endMove$ action terminating the motion of an object to a new position such that from robot's viewpoint a computed size of $peak$ became larger than it was at s **or**
- $extending(peak,viewpoint,s)$ **and** % frame axiom % a is none of those actions which have effect of decreasing the perceived angular size of $peak$

One of the predicates referring to the transition between pairs of peaks is $approaching(pk(b_1,u_1,z_1,d_1),pk(b_2,u_2,z_2,d_2),loc(x_r,y_r),s)$, which represents that peaks $pk(b_1,u_1,z_1,d_1)$ and $pk(b_2,u_2,z_2,d_2)$ (related, respectively, to objects b_1 and b_2) are approaching each other in situation s as perceived from the viewpoint $loc(x_r,y_r)$. (The following relations have analogous arguments to those of $approaching$, they were omitted here for brevity.) Similarly, $receding$, states that two peaks are receding from each other. The predicate $coalescing$, states that two peaks are coalescing. Analogously to coalescing, the relation $hiding$ represents the case of a peak coalescing completely with another peak (corresponding to total occlusion of one body by another). The predicate $splitting$, states the case of one peak splitting into two distinct peaks; finally, two_peak_static, states that the two peaks are static.

Axioms constraining the transitions between pairs of peaks are straightforward, but long and tedious (due to involved geometric calculations). Therefore, for simplicity, we discuss only a high-level description of the SSA for $approaching$ (the axioms for $receding$, $coalescing$, $shrinking$ and $hiding$ are analogous). The axiom for $approaching$ expresses that two depth peaks are approaching iff an apparent angle between them obtained by a sensing action is smaller at the situation $do(a,s)$ than at s *or*, the observer (or an object) moved to a position such that a *calculated* apparent angle is smaller at $do(a,s)$ than at s. In the latter case, the apparent angle between peaks from b_1, b_2 is calculated by the predicate $angle(loc(x_{b_1},y_{b_1}),loc(x_{b_2},y_{b_2}),loc(x_\nu,y_\nu),r_{b_1},r_{b_2},\gamma)$ that has as arguments, respectively, the location of the centroids of objects b_1 and b_2, the location of viewpoint ν, the radii of b_1 and b_2 and γ is an angle that we want to compute. The computations accomplished by $angle$ include the straightforward solution (in time $O(1)$) of a system of equations (including quadratic equations for the circles representing the perimeter of the objects and linear equations for the tangent rays going from the viewpoint to the circles). Similarly to the threshold L used in the SSA for $extending$ above, the SSA for $approaching$ uses a pre-defined (hardware dependent) threshold Δ (roughly, the number of pixels between peaks) that differentiates $approaching$ ($receding$) from $coalescing$ ($splitting$). Another threshold is used in an analogous way to differentiate $coalescing$ from $hiding$. Figure 1 also exemplifies a case where approaching can be entailed. Consider for instance a robot going from viewpoint ν_1 to ν_2, in this case, the angular distance $(k - j)$ between peaks p and q in Fig. 1(d) is less than $(e - n)$ in Fig. 1(b). Moving from viewpoint ν_2 to ν_1 would result in the entailment of $receding$. If it was the case that the apparent distance between the objects was less than Δ, $coalescing$ or $splitting$ could be entailed.

$approaching(peak1,peak2,viewpoint,do(a,s))$ **iff**
- a is a sensing action that measured the angle between $peak1$ and $peak2$ and this angle is smaller than it was at s **or**
- a is an $endMove$ action terminating the process of robot's motion resulting in the viewpoint such that a computed angle between $peak1$ and $peak2$ is currently smaller than it was at s **or**
- a is an $endMove$ action terminating the motion of an object to a new position such that from robot's viewpoint a computed angle between peaks decreased in comparison to what it was at s **or**
- $approaching(peak1,peak2,viewpoint,s)$ **and** % frame axiom% a is none of those actions which have an effect of increasing the perceived angle between $peak1$ and $peak2$.

We name Theory of Depth and Motion ($\mathcal{TDM}$) a theory consisting of the precondition axioms $\mathcal{D}_{ap}$ for actions introduced in this section, SSA $\mathcal{D}_{ss}$ for all fluents in this section, an initial theory $\mathcal{D}_{S_0}$ (with at least two objects and the robot), together with $\mathcal{D}_{una}$ and Σ.

5 Perception and Motion in $\mathcal{TDM}$

The previous section introduced SSA for depth profiles constraining the fluents on depth peaks to hold when either a particular transition in the attributes of a depth peak was sensed, or the robot (or an object) moved to a position such that a particular transition happens. It is easy to see that the axioms presented above define the conceptual neighbourhood diagram (CND) for depth profiles (Fig. 2).

It is worth noting also that the vertices in the conceptual neighbourhood diagram (and the edges connecting them) in Figure 2 represent all the percepts that can be sensed given the depth profile calculus in a domain where objects and the observer can move. Therefore, we can say that perception in $\mathcal{TDM}$ is *sound* and *complete* wrt motion, in the sense that the vertices and edges of the CND in Fig. 2 result from object's motion (i.e. perception is sound) and that every motion in the world is accounted by a fluent or by an edge between fluents in this CND (i.e. it is complete).

Our first result is a schema applying to each fluent in $\mathcal{TDM}$ that represents perception of relations between peaks.

Theorem 1 *(Perception is sound wrt motion). For any fluent F in the CND the following holds:*

$$\mathcal{TDM} \models a \neq sense(p, loc(x_r, y_r), t') \supset$$
$$(\neg F(\vec{x}, s) \wedge F(\vec{x}, do(a, s)) \supset (\exists b, l_1, l_2, t) a = endMove(b, l_1, l_2, t))$$
$$\mathcal{TDM} \models a \neq sense(p, loc(x_r, y_r), t') \supset$$
$$(F(\vec{x}, s) \wedge \neg F(\vec{x}, do(a, s)) \supset (\exists b, l_1, l_2, t) a = endMove(b, l_1, l_2, t)).$$

For any fluents F and F' in $\mathcal{TDM}$ if there is an edge between F and F' in the CND then the following holds:

$$\mathcal{TDM} \models a \neq sense(p, loc(x_r, y_r), t') \supset$$
$$(F(\vec{x}, s) \wedge \neg F'(\vec{x}, s) \wedge \neg F(\vec{x}, do(a, s)) \wedge F'(\vec{x}, do(a, s)) \supset$$
$$(\exists b, l_1, l_2, t)\, a = endMove(b, l_1, l_2, t)).$$

Proof sketch: The proof of this theorem rephrases the explanation closure axiom that follows from the corresponding SSA (see [11] for details). For every vertex in the CND (i.e., for every perception-related fluent F of $\mathcal{TDM}$), if the last action that the robot did is not a sense action, then the change in the value of this fluent can happen only due to an action $endMove$. In addition, we show that for every edge linking two distinct fluents F and F' of the CND in Fig. 2, the transition is due to a move action such that in the resulting situation, the fluent F ceases to hold, but F' becomes true. □

The next theorem states that every motion in the domain is accounted by a vertex or by an edge of the CND in Fig. 2. We denote by F_i, F_j all perception-related fluents (F_i and F_j can be different vertices or can be the same).

Theorem 2 *(Perception is complete wrt motion). For any moving action a in $\mathcal{TDM}$ there is a fluent F_i or an edge between two fluents F_i and F_j in the CND:* $\mathcal{TDM} \models$

$$(\exists b, l_1, l_2, t) a = endMove(b, l_1, l_2, t) \supset \Big[\bigvee_i F_i(\vec{x}, do(a, s)) \vee$$
$$\bigvee_{i,j} \big(F_i(\vec{x}, s) \wedge \neg F_j(\vec{x}, s) \wedge \neg F_i(\vec{x}, do(a, s)) \wedge F_j(\vec{x}, do(a, s))\big)\Big]$$

Proof sketch: The proof follows from the geometric fact that the twelve numbered regions defined by the bi-tangents between two objects (Figure 3) define all possible qualitatively distinct viewpoints to observe these objects. It is easy to see that for every motion of the observer within each region or across adjacent regions in Figure 3 there is an action A mentioned in the SSAs that corresponds to this motion. Therefore, it follows from SSAs that, either a vertex of the CND (a fluent F) describes the perception resulting from the motion, or there are two fluents F and F' such that F ceases to hold after doing a, but F' becomes true. For instance, take a robot in Region 5 (Fig. 3) facing the two objects a and b, but moving backward from them. The SSAs would allow the conclusion that the peaks referring to a and b would be $approaching$ and $shrinking$. On the other hand, a robot (still facing a and b) crossing from Region 5 to 6 would be able to entail the transition from $approaching$ to $coalescing$ by using SSAs.

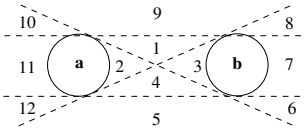

Figure 3. Bi-tangents between two visible objects.

6 Discussion and conclusion

We propose a logical theory built within the situation calculus for reasoning about depth perception and motion of a mobile robot amidst moving objects. The resulting formalism, called Theory of Depth and Motion ($\mathcal{TDM}$), is a rich language that allows both sensor data assimilation and reasoning about motion in the world, where their effects are calculated with Euclidean geometry. We show that reasoning about perception of depth in $\mathcal{TDM}$ is sound and complete with respect to actual motion in the world. This result proves the conjecture made in [12] which hypothesises that the transitions in the conceptual neighbourhood diagrams of the depth profile calculus are logical consequences of a theory about actions and change. Note that $\mathcal{TDM}$ relies on standard models of dense orders, computational geometry and other quantitative abstractions, but this pays off at the end: we can obtain logical consequences about purely qualitative phenomena (e.g., objects approaching each other) from $\mathcal{TDM}$. This theory is an important contribution of our paper.

Future research includes the implementation of the proposed formalism in a simulator of a dynamic traffic scenario. We expect that the theory presented in this paper will allow the reasoning system to recognize and summarize (in simple sentences) plans of other vehicles based on knowledge about its own motion, and its perceptions.

Acknowledgements: Thanks to Joshua Gross, Frédo Durand, Sherif Ghali for comments about computing visibility efficiently in dynamic 2D scenes. This research has been partially supported by the Canadian Natural Sciences and Engineering Research Council (NSERC) and FAPESP, São Paulo, Brazil.

REFERENCES

[1] A. G. Cohn and J. Renz, 'Qualitative spatial representation and reasoning', in *Handbook of Knowledge Representation*, 551–596, (2008).
[2] M. de Berg et al, *Computational Geometry, Algorithms and Applications (Chapter 15), 2nd Edition*, Springer, 2000.
[3] A. Goultiaeva and Y. Lespérance, 'Incremental plan recognition in an agent programming framework', in *Cognitive Robotics, Papers from the 2006 AAAI Workshop*, pp. 83–90, Boston, MA, USA, (2006).
[4] Gerd Herzog, *VITRA: Connecting Vision and Natural Language Systems*, http://www.dfki.de/vitra/, Saarbrücken, Germany, 1986-1996.
[5] H. Levesque and G. Lakemeyer, 'Cognitive robotics', in *Handbook of Knowledge Representation*, 869–886, Elsevier, (2008).
[6] R. Mann, A. Jepson, and J. M. Siskind, 'The computational perception of scene dynamics', *CVIU*, **65**(2), 113–128, (1997).
[7] A. Miene, A. Lattner, U. Visser, and O. Herzog, 'Dynamic-preserving qualitative motion description for intelligent vehicles', in *IEEE Intelligent Vehicles Symposium (IV-04)*, pp. 642–646, Parma, Italy, (2004).
[8] Hans-Hellmut Nagel, 'Steps toward a cognitive vision system', *AI Magazine*, **25**(2), 31–50, (2004).
[9] R. P. A. Petrick, *A Knowledge-level approach for effective acting, sensing, and planning*, Ph.D. dissertation, University of Toronto, 2006.
[10] D. Randell, M. Witkowski, and M. Shanahan, 'From images to bodies: Modeling and exploiting spatial occlusion and motion parallax', in *Proc. of IJCAI*, pp. 57–63, Seattle, U.S., (2001).
[11] Raymond Reiter, *Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems*, MIT, 2001.
[12] Paulo Santos, 'Reasoning about depth and motion from an observer's viewpoint', *Spatial Cognition and Computation*, **7**(2), 133–178, (2007).
[13] M Soutchanski, 'A correspondence between two different solutions to the projection task with sensing', in *Proc. of the 5th Symposium on Logical Formalizations of Commonsense Reasoning*, pp. 235–242, New York, USA, May 20-22, (2001).