Aberystwyth University

Reasoning about shadows in a mobile robot environment

Fenelon, Valquiria; Santos, Paulo E.; Dee, Hannah Mary; Cozman, Fabio G.

Reasoning about shadows in a mobile robot environment

Valquiria Fenelon, Paulo E. Santos, Hannah M. Dee & Fabio G. Cozman

Volume 22, Number 2, March/Ap
ISSN: 0924–669X   CODEN API

ONLINE FIRST

APPLIED INTELLIGENCE

*The International Journal of
Artificial Intelligence,
Neural Networks, and
Complex Problem-Solving Technologies*

Editor-in-Chief:

Moonis Ali

Springer

Available
online
www.springerlink.com

Springer

Springer

Reasoning about shadows in a mobile robot environment

**Valquiria Fenelon · Paulo E. Santos · Hannah M. Dee ·
Fabio G. Cozman**

Abstract This paper describes a logic-based formalism for qualitative spatial reasoning with cast shadows (Perceptual Qualitative Relations on Shadows, or PQRS) and presents results of a mobile robot qualitative self-localisation experiment using this formalism. Shadow detection was accomplished by mapping the images from the robot's monocular colour camera into a HSV colour space and then thresholding on the V dimension. We present results of self-localisation using two methods for obtaining the threshold automatically: in one method the images are segmented according to their grey-scale histograms, in the other, the threshold is set according to a prediction about the robot's location, based upon a qualitative spatial reasoning theory about shadows. This theory-driven threshold search and the qualitative self-localisation procedure are the main contributions of the present research. To the best of our knowledge this is the first work that uses qualitative spatial representations both to perform robot self-localisation and to calibrate a robot's interpretation of its perceptual input.

Keywords Qualitative spatial reasoning · Cognitive robotics · Knowledge representation

V. Fenelon · F.G. Cozman
Escola Politécnica, Universidade de S. Paulo, São Paulo, Brazil

P.E. Santos (✉)
Centro Universitário da FEI, S. Paulo, Brazil
e-mail: psantos@fei.edu.br

H.M. Dee
Department of Computer Science, Aberystwyth University, Aberystwyth, UK

1 Introduction

Cast shadows as cues for depth perception have been used to enhance depictions of natural scenes since the Renaissance [12]. Recent research within psychology suggests that the human perceptual system gives preferential treatment to information from shadows when inferring motion in depth and perceiving 3D scene layout. These studies suggest that information coming from shadows can override such basic notions as conservation of object size [8, 25, 27]. Casati [5] points out that cast shadows also contain information that is not used during passive perception, for instance, information about the presence and location of the light source and the caster; the intensity of the source; the caster's shape; the screen texture; and the distance between the caster and the screen.

Whilst psychologists have demonstrated the centrality of shadows to our own perception of depth, size and motion, much work in computer vision and robotics starts from the premise that shadows are sources of noise rather than information. The present work falls within the small but growing area of research which aims to treat shadows not as sources of noise, but as sources of information. This requires not only a model of the kinds of information that shadows can purvey, but also a robust and accurate shadow detection system. Researchers within both computer vision and robotics have been working in this area—many engaged in shadow suppression in videos from fixed cameras, but some engaged in the more challenging task of shadow identification, localisation and use [14].

The contribution of this paper is the investigation of a qualitative self-localisation method using information from cast shadows and the development of theory-driven threshold search for shadow-caster segmentation. We discuss the experimental evaluation of these methods using two techniques for automatically obtaining the threshold to segment

images from a robot's camera. In the first technique, the images are segmented according to their grey-scale histogram. In the second, the threshold is searched according to a prediction about the robot's location, given a shadow-based qualitative map.

This paper is organised as follows. Section 2 outlines related research from within both computer vision and robotics. Section 3 describes the theory upon which this work is based: Perceptual Qualitative Relations about Shadows (PQRS), which formalises the problem of reasoning about shadows within a qualitative spatial reasoning context. The adaptive thresholding methods considered in this work are presented in Sect. 4, and experimental results are described in Sect. 5. A discussion of open issues is presented in Sect. 6, and Sect. 7 concludes this paper.

Throughout this paper, constants are written in upper-case letters and variables in lower case, unless explicitly stated otherwise.

2 Related research

When considering the task of segmenting moving objects from a static background, shadows are a frequent source of false positives [11, 28] and therefore shadow *suppression* is a major research area. In this context, shadow detection in computer vision almost always involves some model of the colour of the *screen* (or, in computer vision terminology: *background*) and detection is performed using a model of shadows characterising them as *'roughly the same colour as the background, but darker'*. Perhaps the simplest shadow detection method proposed is that of [42], in which a grey-scale image is thresholded and the darker pixels are labelled *shadow*; however this approach fails on complex images and in situations where lighting changes due to either environmental effects or egomotion. Prati in [32] provides an overview and a taxonomy of early shadow-detection techniques, dividing them into *model-based* and *non-model-based*; however, this categorisation does not apply well to more recent works, many of which can be thought of as ensemble methods [28, 31].

Cucchiara et al. in [11] use detected moving objects and a background model as their starting point. The pixel values of moving objects are converted to the HSV (Hue, Saturation and Value) colour space, and then these objects are investigated to determine whether they are real moving objects or merely shadow pixels. This is accomplished by considering observed and background values of all three HSV components, considering the difference between foreground and background values for H and S, and the ratio of the two V values. This captures the intuitive observations that shadows are about the same hue as the same part of the scene unshadowed, slightly more saturated, and darker. A similar approach based upon the observation of colour changes

in cast shadows is presented in [36]. Stauder et al. in [40] use assumptions about the background (*it will dominate the scene*), the nature of shadows and luminance (*shadows are darker and tend to have uniform shading*) and the presence of moving and static edges. Other methods for shadow filtering use a model of the shadow's caster, either assuming it is rectangular (like a car) as in [45], or upright (from a moving person for instance) as in [20, 35]. These assumptions improve filtering considerably, but break down when shadows from arbitrary objects are considered.

There are a few systems within computer vision that use cast shadows as sources of information rather than noise. The work reported in [4] uses known 3D locations and their cast shadows to perform camera calibration and light location (using known casters and screen to tell about the light source); Caspi and Werman in [6] use the moving shadows cast by known vertical objects (e.g., flagpoles, or the side of buildings) to determine the 3D shape of objects on the ground (using the shadow to tell about the shape of the screen). Balan et al. [1] use shadows as a source of information for detailed human pose recognition: they show that using a single shadow from a fixed light source can provide a similar disambiguation effect as using additional cameras.

In robotics, the story is similar. Fitzpatrick and Torres-Jara in [16], inspired by work suggesting that humans use the shadows of their own limbs when judging limb location [7], track the position of a robotic arm and its shadow cast on a table to derive an estimate of the time of contact between the arm and the table. Shadows are detected in this work using a combination of two methods: in the first method, a background model of the workspace is built without the arm and then used to determine light changes when the arm is within the camera view. The second method compares subsequent frames in order to detect moving regions of light change. The authors motivate their work pointing out that depth from shadows and stereopsis may work as complementary cues for robot perception, while the latter is limited to surfaces rich in textures, the former works well for smooth (or even reflective) surfaces. Cheah et al. [9] present a novel controller for a robot manipulator, providing a solution to the problem of trajectory control in the presence of kinematic and dynamic uncertainty. In order to evaluate their results, an industrial robot arm was controlled using the visual observation of the trajectory of its own shadow. Lee et al. [23] use cast shadows inside pipes to detect landmarks: by fitting bright lights to the front of their pipe inspection robot, they can determine when a pipe bends by detecting cast shadows.

Information from shadows is also considered in unmanned autonomous planetary exploration. Tompkins et al. [41] describe an autonomous path planning system that takes into account various conditions of the robot's state, including particularities of the terrain and lighting. In this context,

the information about shadows cast by terrain irregularities allows the rover to plan a trajectory that maximises the trade-off between the exposure of the solar cells to sun light and the limited resources in planetary missions. Kunii and Gotoh [22] propose a *Shadow Range Finder* system that uses the shadow cast by a robot arm on the surface of a terrain in order to obtain depth information around target objects, thus providing low-cost, energy-saving sensors for the analysis of the terrain surrounding rock samples of interest.

More recently, we developed an initial representation of cast shadows in terms of a spatial formalism based on occlusion relations (presented in [37]). This representation, called Perceptual Qualitative Relations about Shadows (PQRS), is used in a qualitative self-localisation procedure for a mobile robot in an office-like environment. The present paper builds upon this idea and, therefore, a more complete and accurate description of the PQRS formalism is presented in the next section.

The idea of qualitative self-localisation first appeared in [24] whereby a tessellation of a mobile robot's environment is obtained from the set of lines connecting pairs of point-wise landmarks. The space bounded by these lines define regions, which can then be treated as vertices of a topological map. The spatial representation behind this idea was further developed in [17, 39, 44]. In particular, [17] considers extended convex (instead of point-wise) objects as landmarks and the decomposition of space is based on the notions of occlusion and visibility, which has much in common to the PQRS formalism investigated in this paper. However, in contrast to [17], the present work shows empirical results from the application of these ideas.

A more complete survey of research on shadown perception (in psychology, artificial intelligence and computer vision) is presented in [14].

3 Perceptual qualitative relations about shadows (PQRS)

Perceptual Qualitative Relations about Shadows (PQRS) [37] is a theory inspired by the idea that shadows provide the observer with the viewpoint of the light source, as shadows are projections of casters from the light source's location. Equivalently, we can say that every point in the shadow region is totally occluded by the caster from the viewpoint of the light source.[1] This idea is developed by representing relations of occlusion and shadows within the scope of

the Qualitative Spatial Reasoning (QSR) field of research, which is part of the Artificial Intelligence sub-area known as Knowledge Representation and Reasoning [13]. The goal of QSR is to provide appropriate formalisms for representing and reasoning about spatial entities, such as part-whole relations, connectivity, orientation, line segments, size and distance, and so on [2, 3, 10, 19, 46]. In practice, QSR formalisms are based on a number of constraints that reflect the structure of space, which are represented as a set of qualitative (i.e., non-numerical) relations. With these relations, QSR methods also facilitate the representation of high-level domain knowledge, adding a more abstract (conceptual) level to the systems in which they are applied, including robotics and vision systems [15]. Therefore, research on qualitative spatial reasoning for robotics does not preclude the use of more traditional quantitative methods, but complements them.

PQRS assumes the existence of a major, static, light source denoted by L, situated above the observer (in agreement with recent research on the psychophysics of perception [26]). It is also assumed that the scenes are observed from a viewpoint v, and that shadows are cast on a single screen Scr, assumed to be much larger than the shadow and not necessarily planar.

The foundation of PQRS is the QSR theory named the Region Occlusion Calculus (ROC) [34], which is itself built upon one of the best known QSR approaches: the Region Connection Calculus (RCC) [33]. RCC is a first-order axiomatisation of spatial relations based on a reflexive, symmetric and non-transitive dyadic primitive relation of *connectivity* ($C/2$) between two regions. Informally, assuming two regions x and y, the relation $C(x, y)$, read as "*x is connected with y*", is true if and only if the closures of x and y have at least one point in common.

Assuming the $C/2$ relation, and two spatial regions x and y, some mereotopological relations between two spatial regions can be defined, such as:

– *disconnected from* (*DC*): $DC(x, y) \equiv \neg C(x, y)$;
– *part of* (*P*): $P(x, y) \equiv \forall z(C(z, x) \rightarrow C(z, y))$;
– *equal to* (*EQ*): $EQ(x, y) \equiv P(x, y) \wedge P(y, x)$;
– *overlaps* (*O*): $O(x, y) \equiv \exists z(P(z, x) \wedge P(z, y))$;
– *partially overlaps* (*PO*): $PO(x, y) \equiv O(x, y) \wedge \neg P(x, y) \wedge \neg P(y, x)$;
– *proper part of* (*PP*): $PP(x, y) \equiv P(x, y) \wedge \neg P(y, x)$;
– *externally connected* (*EC*): $EC(x, y) \equiv C(x, y) \wedge \neg O(x, y)$;
– *tangential proper part* (*TPP*): $TPP(x, y) \equiv PP(x, y) \wedge \exists z(EC(z, x) \wedge EC(z, y))$;
– *non-tangential proper part* (*NTPP*): $NTPP(x, y) \equiv PP(x, y) \wedge \neg \exists z(EC(z, x) \wedge EC(z, y))$.

RCC also includes the inverse relations of P, PP, TPP and $NTPP$, which are represented by a capital 'I' appended to the relative relation: PI, PPI, $TPPI$ and $NTPPI$.

---

[1] This holds if we assume a point light source; in the real world with shadows from larger sources, we can make a distinction between the shadow body (or *Umbra*) which is totally occluded, and the *Penumbra*, which is partially occluded by the caster from the viewpoint of the light source. For the current work, with robots, small light sources, and noisy sensors, we can assume a point light source without losing generality.

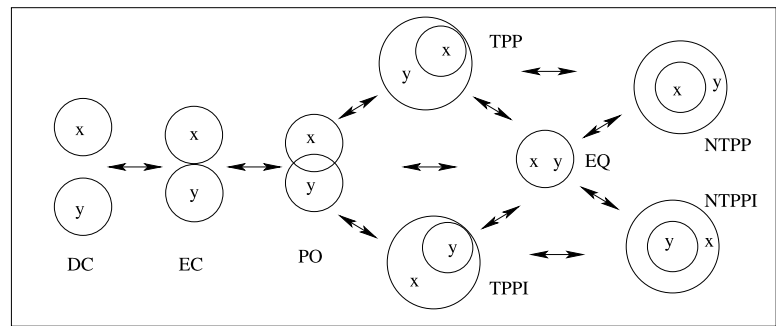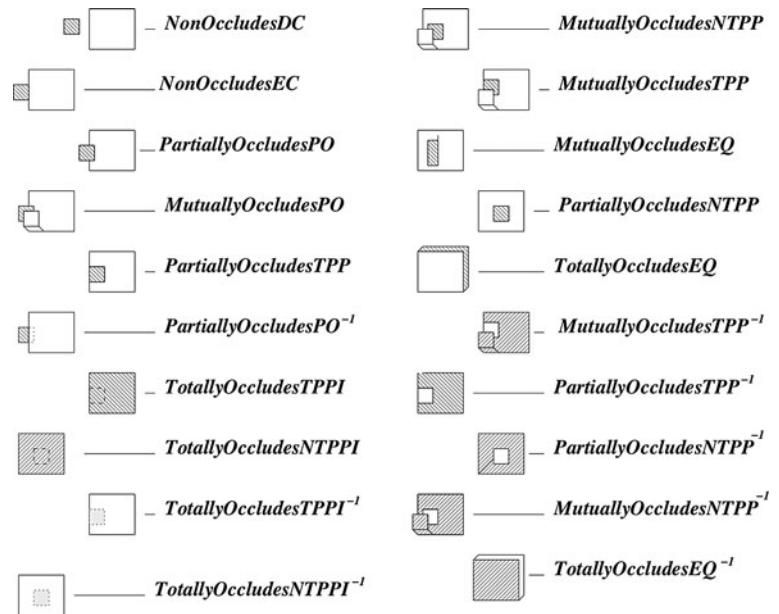**Fig. 1** The RCC8 relations and their conceptual neighbourhood diagram

Fig. 2 The 20 ROC relations

The set constituted by the relations *DC*, *EQ*, *PO*, *EC*, *TPP*, *NTPP*, *TPPI*, and *NTPPI* is the jointly exhaustive and pairwise disjoint set usually referred to as RCC8, these relations are depicted in Fig. 1 [33].

The continuous transitions between the RCC8 relations are shown as a *conceptual neighbourhood diagram* (CND) in Fig. 1. Conceptual neighbourhood diagrams are standard techniques in spatial reasoning [18]. Briefly, a CND is a graph representing in its vertices relations on some specific objects, and in its edges, the continuous transitions between these relations. The concept of continuous transitions, in this context, means that in between adjacent vertices of the graph (i.e. two edge-connected relations), there is no other possible relation which these regions could assume.

Using RCC8 relations, along with the primitive relation *TotallyOccludes*(x, y, v) (which stands for "*x* totally occludes *y* with respect to the viewpoint *v*"), the Region Occlusion Calculus (ROC) represents the various possibilities of interposition relations between two arbitrarily-shaped objects. In particular, it is possible to define occlusion relations for *non occlusion* (*NonOccludes*/3), *partial occlusion* (*PartiallyOccludes*/3) and *mutual occlusion*

(*MutuallyOccludes*/3). In fact, Randell et al. [34] define 20 such relations (shown in Fig. 2).

Informally, in ROC, occlusion relations represent the relative 3D position between pairs of objects with respect to an observer, whereas RCC represents the relative relations between the 2D projections of these objects from an observer's viewpoint. In order to make it clearer, Fig. 2 shows a graphical representation of the ROC relations between two objects: a white box and a dashed box. For instance, the relation *NonOccludesDC* is represented by the two boxes completely separated (i.e. non-occluding and disconnected); the relation *PartiallOccludesTPP* is depicted with the small dashed box occluding the white box, while the 2D projection of the dashed box is a tangential proper part of the 2D projection of the white box.

More formally, Region Occlusion Calculus makes a distinction between the occupancy regions of bodies and their images (or projections) from the viewpoint of an observer. This distinction is made by assuming two functions: the function *region*(x), which maps a body *x* to its 3D occupancy region, and the function *image*(x, v) that maps a body *x* to the body's 2D projection, as seen from

viewpoint v. It is worth pointing out that the viewpoint in ROC is modelled as a pinhole camera whose parameters, however, are not important for the qualitative theory. For instance, given two bodies X and Y and a viewpoint V, the statement $PartiallyOccludesTPP(X, Y, V)$ is defined as $PartiallyOccludes(X, Y, V)$ and $TPP(image(X, V), image(Y, V))$.

It is worth pointing out also that the "I" in the relations $TotallyOccludesTPPI(o, s, v)$ and $TotallyOccludesNTPPI(o, s, v)$ represents the inverse of TPP and PP, respectively; so, for instance, $TotallyOccludesTPPI(o, s, v)$, means that the caster o totally occludes its shadow s, but $image(s)$ is the tangential proper part of $image(o)$ (i.e., $TPPI(image(o, v), image(s, v))$).

Using these definitions, ROC is constrained by a number of axioms, of which we only cite two (($A1$) and ($A2$)), since only these are relevant for proving the PQRS theorems proposed below.

Formula ($A1$) is the ROC axiom that states that "if x totally occludes y from a viewpoint v, x totally occludes any part of y from the viewpoint v":

$(A1)$ $\forall x\, y\, z\, v$ $\big[\big[TotallyOccludes(x, y, v) \wedge$
$$P\big(region(z), region(y)\big)\big] \rightarrow$$
$$TotallyOccludes(x, z, v)\big].$$

Formula ($A2$) below states that "if x totally occludes y from v, no part of y totally occludes any part of x from the viewpoint v":

$(A2)$ $\forall x\, y\, z\, v$ $\big[TotallyOccludes(x, y, v) \rightarrow$
$$\forall z\, u\big[\big[P\big(region(z), region(x)\big) \wedge$$
$$P\big(region(u), region(y)\big)\big] \rightarrow$$
$$\neg TotallyOccludes(u, z, v)\big]\big]$$

In order to simplify notation we use the following abbreviation [34]:

$Occludes(x, y, v) \equiv$
$$\exists z\, \exists u\, P\big(region(z), region(x)\big) \wedge$$
$$P\big(region(u), region(y)\big) \wedge$$
$$TotallyOccludes(z, u, v)$$

Considering the ROC relations between a caster o and its shadow s, from a viewpoint v (and the fact that a shadow never occludes its caster, as proved in theorem (T1) below) only the following ROC relations have models in PQRS:

– $NonOccludesDC(o, s, v)$;
– $NonOccludesEC(o, s, v)$;
– $PartiallyOccludesPO(o, s, v)$;

– $PartiallyOccludesTPP(o, s, v)$;
– $PartiallyOccludesNTPP(o, s, v)$;
– $TotallyOccludesTPPI(o, s, v)$;
– $TotallyOccludesEQ(o, s, v)$, and
– $TotallyOccludesNTPPI(o, s, v)$.

Apart from the ROC relations inherited by PQRS, this theory also assumes the primitive $Shadow(s, o, Scr, L)$ that represents that a shadow s is cast by a caster o, from the light source L, on the screen Scr. The axiom constraining the $Shadow/4$ relation is represented by Formula ($A3$) below.

$(A3)$ $Shadow(s, o, Scr, L) \leftrightarrow \neg\exists o'\big(o \neq o'\big) \wedge$
$$Occludes\big(o', o, L\big) \wedge$$
$$PO\big(region(s), region(Scr)\big) \wedge$$
$$TotallyOccludes(o, s, L).$$

Axiom($A3$) states that the shadow of a caster o is the region in a screen Scr that is totally occluded by o from the light source viewpoint L (if there is no other object o' occluding o from L).

With this formalism it is possible to prove a number of theorems about commonsense facts. For instance, it follows directly from Axioms ($A2$) and ($A3$) that no shadow occludes its own caster,[2] as denoted by Theorem ($T1$) below.

$(T1)$ $Shadow(s, o, Scr, L) \rightarrow \neg TotallyOccludes(s, o, L)$.

It is also a consequence of Axiom ($A3$) and the ROC axioms that no shadow casts a shadow itself (cf. Theorem ($T2$)):

$(T2)$ $Shadow(s, o, Scr, L) \rightarrow \neg Shadow\big(s', s, Scr, L\big)$.

Proof Let's assume, reasoning by contraposition, that both (a) $Shadow(s, o, Scr, L)$ and (b) $Shadow(s', s, Scr, L)$ are true, for any object o and shadows s' and s. From (b) and axiom ($A3$) we have that $TotallyOccludes(s, s', L)$, similarly from (a) and ($A3$), we have that $TotallyOccludes(o, s, L)$. From the transitivity of $TotallyOccludes/3$, we have: $TotallyOccludes(o, s', L)$. Therefore, from ($A3$) we conclude that $Shadow(s', s, Scr, L)$ is false, since the condition

$$\neg\exists o'\big(o \neq o'\big) \wedge Occludes\big(o', o, L\big)$$

does not hold. Therefore, the initial hypothesis leads to a contradiction, thus its negation holds, proving the thesis. □

We can also prove that if two shadows of distinct objects partially overlap, then the objects will be in a relation of

---

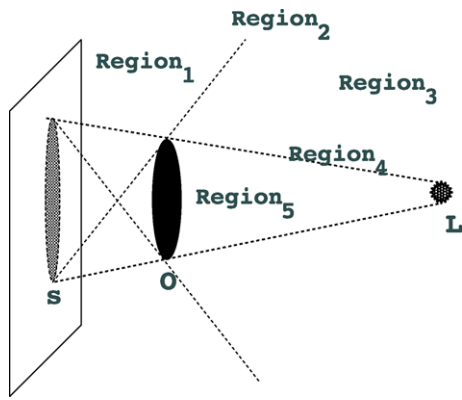[2]Note that we are only dealing with cast shadows, and not self-shadows.

Fig. 3 Distinct regions implied by the observation of a shadow and its caster, where **O** is the caster, **S** its shadow, **L** is the light source, and the constants $Region_1$, $Region_2$, $Region_3$, $Region_4$ and $Region_5$ represent the regions for the relative localisation wrt shadow and object (for brevity, only the top half of the figure is labelled, the bottom half is symmetrical). It is worth noting that, in this figure, $Region_2$ and $Region_4$ have zero-width boundaries

occlusion with respect to the light source, as expressed in Theorem $(T3)$.

$(T3)$ $Shadow(s, o, Scr, L) \wedge Shadow(s', o', Scr, L)$

$$\wedge\, O\big(region(s'), region(s)\big)$$

$$\rightarrow Occludes(o, o', L) \vee$$

$$Occludes(o', o, L).$$

Proof From $Shadow(s, o, Scr, L)$ and axiom $(A3)$ we have (a) $TotallyOccludes(o, s, L)$, and similarly from $Shadow(s', o', Scr, L)$ we have (b) $TotallyOccludes(o', s', L)$. From $O\big(region(s'), region(s)\big)$ and the definition of $O/2$ we have (c) $\exists z\, P(z, s) \wedge P(z, s')$.

From $(A1)$, (a) and (b) it is true that o totally occludes any part of s (and analogously to o' and s'), thus from (c) it is true that: $\exists z\, P(z, s) \wedge P(z, s') \wedge TotallyOccludes(o, z, L) \wedge TotallyOccludes(o', z, L)$.

By construction, o and o' are disconnected, therefore the previous formula implies that either $occludes(o, o', L)$ or $occludes(o', o, L)$. $\square$

Theorem $(T3)$ is an example of an inference that presupposes a relational theory, i.e., it is a result that cannot be achieved with the traditional numerical methods used in robotics.

3.1 Relative location within PQRS

The formalism given above can be used to reason about shadows from arbitrary viewpoints: by relating shadows to occlusion we can determine five distinct regions defined by the lines of sight between the light source, the caster and

its shadow as represented in Fig. 3. We use these distinct regions for robot self-localisation, requiring that we make two practical adjustments. $Region_2$ and $Region_4$ in Fig. 3 are in fact *boundaries* separating $Region_1$ and $Region_3$, and between $Region_3$ and $Region_5$ respectively. Therefore, it is virtually impossible for a robot to locate itself on them. In a real robot environment with noisy sensors, $Region_2$ and $Region_4$ are extended, assuming an interval of uncertainty around these boundaries. Also, in a mobile robot environment, where shadows are usually connected to their casters, only the part of the shadow farthest from the caster (which we shall call *top* for convenience) is considered when constructing the diagram. In order to formally state the distinction between whole shadows and their top parts, we introduce a function $top(s)$ that maps a shadow (s) to its top part (as represented in Fig. 4). This is analogous to the *region* and *image* functions used in ROC, as mentioned above:

$$top : Shadow \rightarrow Shadow$$

In practice, however, obtaining the top part of the shadow is very hard (particularly considering situations where the shadow can be occluded by its caster from a viewpoint). In our implementation we solve this problem by analysing the ROC relations between the top part of the object with respect to the object's shadow, since the top of the shadow is generated by the top part of its caster.

Using the top part of a cast shadow and the extended version of $Region_2$ and $Region_4$, the diagram depicted in Fig. 3 becomes that represented in Fig. 5. As it is more suitable to robot self-localisation, only the latter will be used throughout this paper.

Considering the diagram in Fig. 5, any viewpoint v located on $Region_1$ will observe the (top of) shadow s and the object o as $NonOccludesDC(o, top(s), v)$; similarly, if v observes o and $top(s)$ from $Region_3$ it should see that $PartiallyOccludesPO(o, top(s), v)$ and from $Region_5$ that $TotallyOccludesNTPPI(o, top(s), v)$. In $Region_4$, v would observe $TotallyOccludesTPPI(o, top(s), v)$. In $Region_2$, v perceives object and shadow as $NonOccludesEC(o, top(s), v)$. These facts are included in PQRS by Axioms $(A4)$ to $(A8)$ below, where the predicate $located(r, v, o, s)$ represents that an observer v is located at a region r according to the object o and its shadow s.

$(A4)$ $located(Region_1, v, o, s) \leftarrow$

$\quad Shadow(s, o, Scr, L) \wedge$

$\quad NonOccludesDC\big(o, top(s), v\big) \wedge v \neq o;$

$(A5)$ $located(Region_2, v, o, s) \leftarrow$

$\quad Shadow(s, o, Scr, L) \wedge$

$\quad NonOccludesEC\big(o, top(s), v\big) \wedge v \neq o;$

Fig. 4 (**a**) Diagram representing the top part of a shadow; (**b**) snapshot of the top part of a shadow, as seen from the robot's camera
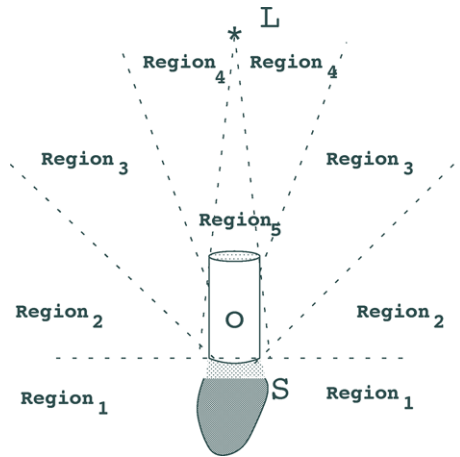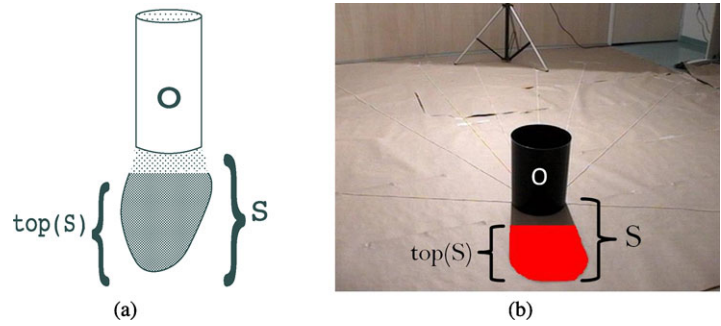

Fig. 5 Regions implied by the observation of a shadow and its caster (represented by the constants $Region_1$, $Region_2$, $Region_3$, $Region_4$ and $Region_5$), where L is the light source, O is the object (caster), S is its shadow. Only the top part of the shadow is used in the definition of the regions in this diagram

(*A6*) $located(Region_3, v, o, s) \leftarrow$

$Shadow(s, o, Scr, L) \wedge$

$PartiallyOccludesPO\big(o, top(s), v\big) \wedge v \neq o;$

(*A7*) $located(Region_4, v, o, s) \leftarrow$

$Shadow(s, o, Scr, L) \wedge$

$TotallyOccludesTPPI\big(o, top(s), v\big) \wedge v \neq o;$

(*A8*) $located(Region_5, v, o, s) \leftarrow$

$Shadow(s, o, Scr, L) \wedge$

$TotallyOccludesNTPPI\big(o, top(s), v\big) \wedge v \neq o.$

In the algorithms for qualitative self-localisation presented in Sect. 4.1 below, it is convenient to introduce the relation $neighbour(x, y)$; $neighbour/2$ represents the neighbouring regions with respect to the robot's location, as represented in Formula (*A9*) below, for an object o, a viewpoint v and a shadow s.

(*A9*) $neighbour(x, y) \leftrightarrow \big(located(x, v, o, s) \wedge EC(x, y)\big)$

In Formula (*A10*) we define a predicate *predict_future_loc*(y, v, o, s) that represents a prediction for the robot's future location y, to a neighbouring region of the robot's current location x, if a moving action occurs. The moving action *move*/1 is assumed as primitive in the present formalism. In practice, it is directly related to the robot's actuators.

(*A10*) $located(x, v, o, s) \wedge move(v) \wedge neighbour(x, y) \leftrightarrow$

$predict\_future\_loc(x, v, o, s) \vee$

$predict\_future\_loc(y, v, o, s)$

Using ROC it is also possible to relate shadows with the relative distance of objects from a viewpoint. Region Occlusion Calculus relates occlusion with relative distance using the relation *nearness* ($N(x, y, z)$), read as "x is nearer to y than x is to z", that was first defined in [43]. *Nearness* is incorporated into ROC by the following axiom [34]:

(*A11*) $\forall x\, y\, v\, \big[PartiallyOccludes(x, y, v) \rightarrow N(v, x, y)\big]$

representing that "if a body x partially occludes a body y with respect to some viewpoint v then x is nearer to v than y is to v".

Within PQRS this implies the commonsense fact that $N(L, o, s)$ (for a light source L, an object o and its shadow s) and consequently that $N(L, o, Scr)$: "the light source is nearer the caster than it is to its shadow". This fact, allied with the qualitative self-localisation (according to Fig. 5), facilitates inferences about relative depth from monocular views in some cases.

It is worth pointing out that Axiom (*A11*) does not hold in cases of *mutual occlusion* (such as those presented in Fig. 2), but only under *partial occlusion*. In fact, if mutual occlusion occurs, the premise of Axiom (*A11*) is false, since the set of ROC relations is joint exhaustive and pairwise disjoint.

4 Robot environment, self-localisation and adaptive thresholding methods

The idea for qualitative robot self-localisation using cast shadows, presented in the previous section, was implemented on our Pioneer PeopleBot mobile robot using its

Fig. 6 Images depicting environment, sample input (with ambiguous shadow-base relations) and segmented object/shadow

(a) robot environment   (b) Two shadows   (c) Example segmented image

---
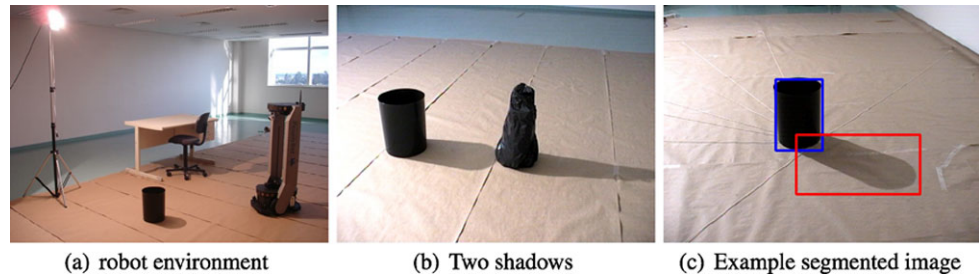
Algorithm 1 PERCEPTION-ACTION(th, Scene, v)

1: segment *Scene* using the threshold *th* to obtain a caster O and the top of its shadow $top(S)$
2: **if** $NonOccludesDC(O, top(S), v)$ **then**
3:    robot is on $Region_1$
4: **else if** $NonOccludesEC(O, top(S), v)$ **then**
5:    robot is on $Region_2$
6: **else if** $PartiallyOccludesPO(O, top(S), v)$ **then**
7:    robot is on $Region_3$
8: **else if** $TotallyOccludesTPPI(O, top(S), v)$ **then**
9:    robot is on $Region_4$
10: **else if** $TotallyOccludesNTPPI(O, top(S), v)$ **then**
11:    robot is on $Region_5$
12: **else**
13:    *FAIL*
14: **end if**

---

monocular colour camera to obtain snapshots of objects and their shadows in an office-like environment, containing a major (but not necessarily single) light source, as shown in Fig. 6(a). Shadow detection was accomplished by first mapping the images captured by the camera into a HSV colour space. These images were then segmented by thresholding on the V dimension, whereby high values (light objects) were filtered out and low values (dark objects) are casters. Shadows were located within a value range in between light and dark objects. Morphological operators and the saturation value were used to filter noise (such as reflections of the light source on the object or background shadows). The robot was set to navigate through the room and to analyse its position with respect to object-shadow locations according to the diagram shown in Fig. 5. An example of the snapshots used in this work is shown in Fig. 6(c). Shadow correspondence, which is the problem of matching each shadow to its caster [25, 27], is solved in this work by assuming a simple heuristic: the shadow that is connected to an object's base is the shadow of this object. When there are various shadows connected to the object's base, the caster is associated with the shadow that is further away from the light source (Fig. 6(b) shows an example of such situation).

Given a threshold *th*, a *Scene* and a viewpoint v, Algorithm 1 summarises the basic method for self-localisation described and built upon in this section.

In Algorithm 1 the ROC relations between a caster O and its shadow S are evaluated according to a threshold on the distance between the (top part of) the shadow's bounding box when *Non Occlusion* holds. If the shadow is in some degree occluded by its caster, from the observer's viewpoint, the ROC relation is evaluated according to a percentage of the shadow's bounding box that can be observed from behind the caster: $PartiallyOccludesPO(O, top(S), v)$ is interpreted when more than (or equal to) 10 % of the bounding box is observed, but not all of it; $TotallyOccludesTPPI(O, top(S), v)$ is assumed when less than 10 % is still observed; and, when no part of the shadow is seen from behind the caster, $TotallyOccludesNTPPI(O, top(S), v)$ is concluded.

4.1 Adaptive thresholds for foreground/background segmentation

In this work we compare the use of three distinct methods for thresholding images to find shadows. The first method (forming the baseline) is a fixed threshold across all images, selected by hand. The second is Otsu's method [30], and the third is a threshold search related to the robot's predicted location according to PQRS (which we call the *knowledge-based threshold*).

Otsu's method [30] is an adaptive thresholding method, often used in computer vision, that finds the threshold (*th*) which maximises the inter-class variance σ between two groups of pixels. Formula (1) expresses σ in terms of the threshold-dependent class probabilities ($\omega_1(th)$ and $\omega_2(th)$) and class means ($\mu_1(th)$ and $\mu_2(th)$) of groups 1 and 2.

$$\sigma^2(th) = \omega_1(th)\omega_2(th)\big[\mu_1(th) - \mu_2(th)\big]^2 \qquad (1)$$

The *knowledge-based threshold* method uses belief about the robot's previous location and information about the robot's motion in order to make a prediction about its current location. This procedure is represented in Algorithm 2 (THRESHOLD-AND-POSITION) and it works as follows. The robot starts in any of the object-shadow regions, as depicted in Fig. 5. From this position the robot executes a motion. Given the speed and direction of this motion, the robot makes a prediction about where it is going to be with respect to the map in Fig. 5. This prediction is given by odometry

---

Algorithm 2 THRESHOLD-AND-POSITION(s_i, th_0, $scene$, v)

1: $th_{min} = th_0 - 5$, $th_{max} = th_0 + 5$
2: $s_j = $ PERCEPTION-ACTION(th_0, $Scene$, v)
3: **if** ($s_i == s_j$) **then**
4:     return (th_0, s_j)
5: **else**
6:     (th_{aux}, s_j) = VARY-THR(s_i, th_0, th_{min}, th_{max}, $scene$)
7:     **if** ($th_{aux} < th_{max}$) **then**
8:         return (th_{aux}, s_j)
9:     **else**
10:         (th_{aux}, s_j) = VARY-THR(s_{i+1}, th_0, th_{min}, th_{max}, $scene$)
11:         **if** ($th_{aux} < th_{max}$) **then**
12:             return (th_{aux}, s_j)
13:         **else**
14:             (th_{aux}, s_j) = VARY-THR(s_{i-1}, th_0, th_{min}, th_{max}, $scene$)
15:             **if** ($th_{aux} < th_{max}$) **then**
16:                 return (th_{aux}, s_j)
17:             **else**
18:                 return *FAIL*
19:             **end if**
20:         **end if**
21:     **end if**
22: **end if**

Algorithm 3 VARY-THR(s, th_0, th_{min}, th_{max}, $scene$)

1: $step = 1$;
2: $th_{aux} = th_{min} - step$
3: **while** (($s \neq s_j$) and ($th_{aux} < th_{max}$)) **do**
4:     $th_{aux} = th_{aux} + step$
5:     $s_j = $ PERCEPTION-ACTION(th_{aux}, $scene$, v)
6: **end while**
7: return (th_{aux}, s_j)

---

and is represented by Axiom (A10) in the formalism above. We summarise the knowledge-based threshold search as follows:

- The robot starts in a known position, and performs a moving action. This puts it in a new position, which we can estimate (but not know precisely, due to actuator noise and odometry drift).
- In its new position, the robot captures a snapshot of the target object and uses it to decide on its location. This is accomplished by calling the function PERCEPTION-ACTION (Algorithm 1).
- If the location interpreted from the image matches that predicted, then the robot moves on.

- If not, the function VARY-THR (Algorithm 3) is called to vary the threshold until a match is found between its predicted and interpreted positions.
- If there is no such match, VARY-THR is called twice again to verify whether there is a threshold that allows the robot to match its position with respect to one of the neighbouring regions of the predicted location.
- The algorithm terminates with failure if it does not find a suitable threshold that would place it on the predicted region or on one region either side of the prediction. Through this consideration of neighbouring regions, we can better handle errors introduced by noisy odometry.

The pseudocode THRESHOLD-AND-POSITION (Algorithm 2) has as input the prediction of the robot's position s_i after its motion, a threshold *th* and a *scene* snapshot observed from the viewpoint v. This algorithm uses the variables th_0, th_{min}, th_{max} and th_{aux} for thresholds, where th_0 is a starting threshold, and th_{min} and th_{max} are the minimum and maximum thresholds, respectively. The variables s_0, s_i, s_j represent the robot's position, and the variables s_{i-1}, s_{i+1} represent regions that are neighbours of s_i. These regions are possible outcomes of a single (non-deterministic) motion whose goal was s_i. As well as these symbols, the pseudocode VARY-THR (Algorithm 3) uses the constant *step* that represents the step of threshold variation. This work assumes the following values: $step = 1$, $th_{min} = th_0 - 5$ and $th_{max} = th_0 + 5$.

The main function in the set of algorithms presented here is Algorithm 2, THRESHOLD-AND-POSITION. This calls Algorithm 1: PERCEPTION-ACTION and Algorithm 3: VARY-THR, outputting the robot's location (s_j) and the current threshold. Thus the termination, complexity and correctness of Algorithm 2 depends on these attributes with respect to the latter two functions.

Algorithm 1 is a branching *if-then-else* statement representing the various shadow-object regions with respect to the diagram in Fig. 5, given a snapshot of the world. The algorithm outputs a default value (*FAIL*) when no region can be assigned. Thus, it always terminates and runs in constant time. This algorithm is correct, since given an appropriate threshold *th*, it will output the correct location of the robot within the five qualitatively distinct regions shown in Fig. 5. Algorithm 3 always terminates, since it only applies a linear search (with fixed step) on a finite set of thresholds. In this work the set of thresholds is the interval of natural numbers between th_{min} and th_{max} and it is traversed with step 1. Thus, the maximum time complexity of Algorithm 3 is $O(n)$, where n is the number of steps from th_{min} to th_{max}, i.e., $n = |th_{min}, th_{max}|/step$. Algorithm 3 returns *FAIL* if Algorithm 1 fails (i.e. in the cases when the target object is not present in the scene, or a threshold cannot be found to segment a shadow from its caster).

Given the correctness of Algorithms 1 and 3, the correctness of Algorithm 2 is obtained by considering two cases: when the robot is located in the region it predicts and the case when the robot's prediction is wrong. Assuming perfect sensors this proof is straightforward since, in the absence of sensor noise, if the robot is in its predicted position, the algorithm will find an appropriate threshold. This is accomplished by calling Algorithm 3 (line 6 of Algorithm 2) that searches for all thresholds available with a small step with respect to threshold variation. On the other hand, if the robot's prediction is wrong, the robot will either be in one of the neighbouring regions to the predicted region, i.e. either in s_{i+1} or s_{i-1} (as a result of its non-deterministic actuators), or it will be lost. The algorithm covers these cases, since it makes two further calls to Algorithm 3 (lines 10 and 14 of Algorithm 2), or fails if it is the case that the robot is lost. In the presence of sensor noise, however, the algorithm may encounter random errors, so its correctness can only be verified experimentally, as presented in the next section.

The running time complexity of Algorithm 2 is directly proportional to the complexity of Algorithm 3. Therefore, its running time is $O(n)$, where $n = |th_{min}, th_{max}|/step$ as mentioned above.

To the best of our knowledge this is the first work where a low-level visual parameter such as a segmentation threshold has been obtained as a result of the robot's prediction of its location according to a qualitative theory. In this way, we use the PQRS theory not only for robot self-localisation based upon shadow perception, but also for the refinement of the shadow perception itself. The next section presents an empirical evaluation of this technique.

5 Experiments

This section describes the results of the experiments on robot localisation with respect to the map in Fig. 5. In these experiments, the robot collected 587 snapshots around the target object, which provides the frame of reference (e.g. the black bucket in Fig. 6(c)). The target was not always within view of the camera, which represents the main source of noise in the experiments, amounting to 20 % of the entire dataset. Therefore, in the best case scenario, i.e. in the unlikely case that the algorithm never fails in its estimation of robot location, we could not exceed 80 % correctness.

As we are interested in qualitative localisation, during the experiments, the position of the object, the light source, the sizes of the objects, shadows, the sizes of the qualitative regions induced by shadow, and objects (according to Fig. 5) were unknown. However, the velocity and direction of the robot's motion were given by the robot's moving action.

In this section we present results for the following experiments of robot's self-localisation:

Table 1 Percentage of correct answers from using fixed thresholds, Otsu's method and the knowledge-based adaptive thresholding, where "# images" is the number of snapshots taken at each region

Region	# images	fixed threshold (%)	Otsu (%)	knowledge based (%)
$Region_1$	225	28	21	66
$Region_2$	171	26	4	34
$Region_3$	138	64	21	80
$Region_4$	36	47	25	44
$Region_5$	17	47	76	59
Global	587	38	18	58

– a baseline experiment that uses fixed thresholds for image analysis;
– experiments using Otsu's method for adaptive thresholding;
– experiments using the knowledge-based threshold method.

The results of the baseline experiment are represented in the third column of Table 1, showing a global performance of 38 % on localising the robot in every region. In this experiment, the threshold was set manually during a calibration phase with the robot in $Region_3$. A high accuracy was obtained in this region (around 60 % with respect to $Region_3$). Within other regions the results were lower than 50 %. The poor performance outside of $Region_3$ is because the foreground/background segmentation is not optimal for images obtained under other light conditions (i.e., the threshold selected is sensitive to the relative position configurations between robot, caster and light produced by the agent's motion). Calibrating the fixed threshold with the robot on other regions has a similar effect.

The obvious approach for improving the poor results obtained by fixed-thresholding is to move to a dynamic method and *adjust the thresholds for each snapshot taken*. The technique we have used to perform this adjustment is the Otsu method [30] (cf. Sect. 4.1). This should be able to automatically find the threshold for segmenting objects of interest (i.e. casters and their shadows) from background. The results obtained are represented on the fourth column of Table 1.

The results obtained with a variable threshold method (Otsu's method), surprisingly, were much worse than those obtained with a fixed threshold. For global localisation, the method answered correctly on 18 % of the total 587 snapshots. The best performance of Otsu's method was obtained on region 5 (76 %). However, due to the small size of this region with respect to the others, only 17 snapshots where obtained from it. Thus, this result cannot be generalised. The localisation accuracy on the other regions using Otsu's method was below 50 %. Investigation of the pixel value

distributions indicated that the problem is that these distributions are not in general bi-modal, which increases the difficulty of searching for an appropriate threshold from the image histogram.

In our third set of results, the robot was set to vary the threshold until the interpretation of the target object and its shadow matches a robot's prediction of its location (using Algorithm 2, as explained in Sect. 4.1). The results obtained are represented in the fifth column of Table 1 ("knowledge based"), which shows that the system achieved an accuracy of around 58 % overall. Thus the incorporation of top-down knowledge about shadow appearance, and reasoning based upon past location and agent motion can greatly assist in the refinement of a simple shadow-detection algorithm, outperforming a traditional algorithm for adaptive thresholding.

6 Discussion and open issues

In this work we investigated robot self-localisation using qualitatively distinct regions defined from a visual observation of cast shadows. These regions were defined on a formal framework within a qualitative spatial reasoning standpoint, which provides a principled approach to reasoning about space.

Central to the problem of qualitative self-localisation using shadows is the segmentation of casters and shadows from the background, which was accomplished here by thresholding on value (V) on the HSV colour space. In the present paper we proposed a new strategy for calibrating this threshold, where the prediction about the robot's location is used to search for a match between the interpreted position (as given from visual observation) and its predicted location. In order to evaluate this method, we have presented three sets of experiments in which different ways of defining the threshold were tried. In the first set of experiments a hand-coded fixed threshold was used, in the second set of experiments we used Otsu's method [30] in order to find the threshold values from the image histogram. The third set of experiments presents the results of applying our proposed method for matching the prediction with the observation.

The intuition behind the experiments with fixed thresholds was to provide a lower-bound for the evaluation of our idea, since (as we hypothesised) nothing could perform worse than a hand-coded threshold. Experiments with Otsu's method were then to set the standard, as this is a frequently used method for adaptive thresholding. However, it turned out that Otsu's performance was in fact worse than using the fixed threshold. This is due to the fact that the first set of experiments used the best threshold that could be found, after a number of trials where the value was changed

by hand. Otsu's method, however, had to deal with arbitrary images, where it had to maximise a value that is dependent on an *a priori* hypothesis of bi-modal pixel distribution. This was not the case in some of the snapshots taken by the robot: a great number of them suffered from the effect of reflections of the light source on the caster. Moreover, from some angles, there was a negative gradient of luminosity just behind the object.

The method for calibrating the threshold using the prediction about the robot's location performed better than the other two, obtaining an accuracy of around 58 % with respect to our dataset containing 587 snapshots of the robot's environment. Considering that the dataset used was composed of 20 % of images where self-localisation was impossible (since the target object was not present in these data points), our algorithm had a performance of around 70 % over the images where self-localisation was possible. In the remaining 30 % of cases, the errors were due to the proximity of the robot to the borderlines separating the regions and due to wrong predictions caused by errors in the robot actuators that led the robot beyond the predicted region or its neighbours. In the cases of borderline proximity, the threshold variation was affecting the observed regions between object and shadow. Therefore, no appropriate threshold was found that matches the predicted region (or one of its neighbours). We are currently working on more robust shadow detection algorithms that should improve on this results. The cases of wrong predictions shall be tackled in future research by including multiple object-based landmarks for qualitative self-localisation.

The framework presented in this paper has not been tested in a dynamic environment. In such cases, multiple shadows from moving objects may cause a very negative effect on self-localisation methods as presented in this paper. We believe that this issue could be solved by extending the ideas presented in this work by not only allowing the system to predict the motion of the robot, but also the actions of other agents in the robot environment. Therefore, the algorithm would be able to single out the target object that it is using to locate itself. This is a matter for future research.

Another open issue of this work is the qualitative localisation in situations where shadows from multiple objects can generate (potentially inconsistent) hypotheses. We are currently investigating ways of doing this by incorporating the ideas proposed in [17, 38] into our framework. Besides this, we believe that the work reported in the present paper could be enhanced by incorporating the idea of cognitive maps [29], as well as modern methods of SLAM [21].

Although this work explores a qualitative theory about space, the choice of qualitative methods does not preclude the use of quantitative or statistical methods. Rather,

we believe that qualitative methods in robotics should complement more traditional numerical algorithms, providing an additional processing level in situations where it is possible to extract and model qualitative information.

7 Conclusion

This paper has demonstrated how the incorporation of qualitative spatial representation and *a priori* knowledge about shadow regions can be combined to enhance a simple shadow-detection algorithm based upon thresholding. Future work will consider the incorporation of more sophisticated shadow detection algorithms and the extension of the current snapshot-based system to one which incorporates continuous video from a dynamic environment.

We have raised a number of questions in this work, and we consider these questions in themselves to be a useful contribution. For example, how can shadows improve object localisation when contrasted with object-based methods? Under what conditions can shadows be effectively exploited? How can we combine predictive shadow-based localisation with predictive localisation based upon object's pose? These are all questions which we hope to consider in more depth in future work.

References

1. Balan AO, Black MJ, Haussecker H, Sigal L (2007) Shining a light on human pose: on shadows, shading and the estimation of pose and shape. In: Proc international conference on computer vision (ICCV), pp 1–8
2. Balbiani P, Condotta JF (2002) Spatial reasoning about points in a multidimensional setting. Appl Intell 17(3):221–238
3. Bennett B, Cohn AG, Wolter F, Zakharyaschev M (2002) Multidimensional modal logic as a framework for spatio-temporal reasoning. Appl Intell 17(3):239–251
4. Cao X, Foroosh H (2007) Camera calibration and light source orientation from solar shadows. Proc Comput Vis Pattern Recogn 105(1):60–72
5. Casati R (2004) The shadow knows: a primer on the informational structure of cast shadows. Perception 33(11):1385–1396
6. Caspi Y, Werman M (2006) Vertical parallax from moving shadows. In: Proc computer vision and pattern recognition (CVPR), New York, USA
7. Castiello U, Lusher D, Burton C, Disler P (2003) Shadows in the brain. J Cogn Neurosci 15(6):862–872
8. Cavanagh P (2005) The artist as neuroscientist. Nature 434:301–307
9. Cheah CC, Liu C, Slotine JJE (2006) Adaptive tracking control for robots with unknown kinematic and dynamic properties. Int J Robot Res 25(3):283–296
10. Cohn AG, Renz J (2008) Qualitative spatial representation and reasoning. In: Handbook of knowledge representation. Elsevier, Amsterdam, pp 551–596
11. Cucchiara R, Grana C, Neri G, Piccardi M, Prati A (2001) The sakbot system for moving object detection and tracking. In: Video-based surveillance systems: computer vision and distributed processing (Part II—detection and tracking). Kluwer Academic, Dordrecht, pp 145–158
12. da Costa Kauffmann T (1979) The perspective of shadows: the history of the theory of shadow projection. J Warburg Courtauld Inst 38:258–287
13. Davis R, Shrobe H, Szolovits P (1993) What is a knowledge representation? AI Mag 14(1):17–33
14. Dee H, Santos P (2011) The perception and content of cast shadows: an interdisciplinary review. Spat Cogn Comput 11(3):226–253
15. dos Santos MV, de Brito RC, Park HH, Santos P (2009) Logic-based interpretation of geometrically observable changes occurring in dynamic scenes. Appl Intell 31(2):161–179
16. Fitzpatrick P, Torres-Jara E (2004) The power of the dark side: using cast shadows for visually-guided touching. In: Proc of the 4th IEEE/RAS international conference on humanoid robots, pp 437–449
17. Fogliaroni P, Wallgrün JO, Clementini E, Tarquini F, Wolter D (2009) A qualitative approach to localization and navigation based on visibility information. In: COSIT'09: proceedings of the 9th international conference on spatial information theory. Springer, Berlin, pp 312–329
18. Freksa C (1991) Conceptual neighbourhood and its role in temporal and spatial reasoning. In: Decision support systems and qualitative reasoning. Elsevier, Amsterdam, pp 181–193
19. Guesgen HW (2002) Reasoning about distance based on fuzzy sets. Appl Intell 17(3):265–270
20. Hsieh JW, Hu WF, Chang CJ, Chen JS (2003) Shadow elimination for effective moving object detection by Gaussian shadow modeling. Image Vis Comput 21(6):505–516
21. Kang JG, Kim S, An, SY, Oh SY (2012) A new approach to simultaneous localization and map building with implicit model learning using neuro evolutionary optimization. Appl Intell 36(1):242–269
22. Kunii Y, Gotoh T (2003) Evaluation of shadow range finder: SRF for planetary surface exploration. In: Proc of the IEEE international conference on robotics and automation (ICRA), pp 2573–2578
23. Lee JS, Roh SG, Kim DW, Moon H, Choi HR (2009) In-pipe robot navigation based upon the landmark recognition system using shadow images. In: Proc of the IEEE international conference on robotics and automation (ICRA), pp 1857–1862
24. Levitt TS, Lawton DT (1990) Qualitative navigation for mobile robots. Artif Intell 44:305–360
25. Mamassian P (2004) Impossible shadows and the shadow correspondence problem. Perception 33:1279–1290
26. Mamassian P, Goutcher R (2001) Prior knowledge on the illumination position. Cognition 81(1):B1–B9
27. Mamassian P, Knill DC, Kersten D (1998) The perception of cast shadows. Trends Cogn Sci 2(8):288–295
28. Martel-Brisson N, Zaccarin A (2007) Learning and removing cast shadows through a multidistribution approach. IEEE Trans Pattern Anal Mach Intell 29(7):1134–1146

29. Mendonça M, de Arruda LVR, Neves F Jr. (2012) Autonomous navigation system using event driven-fuzzy cognitive maps. Appl Intell 37(2):175–188

30. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9:62–66

31. Porikli F, Thornton J (2005) Shadow flow: a recursive method to learn moving cast shadows. In: Proc international conference on computer vision (ICCV)

32. Prati A, Mikic I, Trivedi M, Cucchiara R (2003) Detecting moving shadows: algorithms and evaluation. IEEE Trans Pattern Anal Mach Intell 25(7):918–923

33. Randell D, Cui Z, Cohn A (1992) A spatial logic based on regions and connection. In: International conference on knowledge representation and reasoning, Cambridge, US, pp 165–176

34. Randell D, Witkowski M, Shanahan M (2001) From images to bodies: modeling and exploiting spatial occlusion and motion parallax. In: International joint conference on artificial intelligence, Seattle, US, pp 57–63

35. Renno JRR, Orwell J, Thirde DJ, Jones GA (2004) Shadow classification and evaluation for soccer player detection. In: Proc British machine vision conference, Kingston upon Thames, UK, pp 839–848

36. Salvador E, Cavallaro A, Ebrahimi T (2004) Cast shadow segmentation using invariant color features. Comput Vis Image Underst 95(2):238–259

37. Santos P, Dee H, Fenelon V (2009) Qualitative robot localisation using information from cast shadows. In: Proc of the IEEE international conference on robotics and automation (ICRA), pp 220–225

38. Sapena O, Onaindäa E (2008) Planning in highly dynamic environments: an anytime approach for planning under time constraints. Appl Intell 29(1):90–109

39. Schlieder C (1993) Representing visible locations for qualitative navigation. In: Qualitative reasoning and decision technologies, pp 523–532. CIMNE

40. Stauder J, Mech R, Ostermann J (1999) Detection of moving cast shadows for object segmentation. IEEE Trans Multimed 1(1):65–76

41. Tompkins P, Stentz A, Whittaker WL (2001) Automated surface mission planning considering terrain, shadows, resources and time. In: Proceedings of the 6th international symposium on artificial intelligence, robotics and automation in space (i-SAIRAS'01), Montreal, Canada

42. Troccoli A, Allen PK (2004) A shadow-based method for model registration. In: IEEE computer vision and pattern recognition workshop (CVPRM)

43. van Benthem J (1991) The logic of time: a model theoretic investigation into the varieties of temporal ontology and temporal discourse. Kluwer Academic, Dordrecht

44. Wagner T, Huebner K (2004) Egocentric qualitative spatial knowledge representation for physical robots. In: 8th international workshop on RoboCup 2004 (Robot World Cup Soccer Games and Conferences). Lecture notes in artificial intelligence. Springer, Berlin, pp 134–149

45. Wang Y, Loe KF, Tan T, Wu JK (2005) A dynamic hidden Markov random field model for foreground and shadow segmentation. In: Proc IEEE workshop on applications of computer vision

46. Zimmermann K, Freksa C (1996) Qualitative spatial reasoning using orientation, distance, and path knowledge. Appl Intell 6(1):49–58

Valquiria Fenelon has got her MSc from Centro Universitário da FEI, S. Paulo, Brazil in 2009 and is currently a PhD candidate at University of S. Paulo, Brazil.

Paulo E. Santos received his PhD degree in electrical engineering from Imperial College, London, UK in 2003, working on the development of spatial reasoning systems for mobile robots. He was a research assistant at the School of Computing, University of Leeds, UK from mid-2003 to 2005. Currently Paulo is an associate professor at FEI, S. Paulo, Brazil in the field of Artificial Intelligence and Robotics.

Hannah M. Dee is a lecturer in computer science at Aberystwyth University working on the areas of computer vision for the analysis of human behaviour; the detection of shadows and reasoning about shadows. Before joining Aberystwyth, Hannah was a research fellow in Grenoble and at the University of Leeds, from where she was awarded a PhD degree in 2005.

Fabio G. Cozman received the PhD degree in robotics in 1997 from the School of Computer Science, Carnegie Mellon University. He is a professor at the University of São Paulo, Brazil. He has worked on the theory and applications of sets of probability measures, Bayesian networks, and semisupervised learning.