

仮想世界データベースのための位置問合せ機能の提案

佐藤 こず恵[†] 渡辺 知恵美[†] 増永 良文^{††}

[†] お茶の水女子大学大学院人間文化研究科 〒112-8610 東京都文京区大塚 2-1-1

^{††} お茶の水女子大学理学部 〒112-8610 東京都文京区大塚 2-1-1

E-mail: [†]{kozue,chiemi}@dmlab.is.ocha.ac.jp, ^{††}masunaga@is.ocha.ac.jp

あらまし 我々は、バーチャルリアリティ技術とデータベースを統合した仮想世界データベースシステム (VWDB) の研究を進めている。VWDBにおいて、一度仮想空間から出たユーザが再度入り、前に行ったことのある場所に行きたいが何処だか分からなくなってしまった場合や、人からこんな風にオブジェクトが配置されている場所があったと聞き、そこに行ってみたいという場合に、その場所を問い合わせる方法として、オブジェクトを配置することでその風景を再現することによる問合せ (これを Query-by-Scene と名付ける) を提案する。また、探索方法として、風景を再現する際のユーザの記憶違いを許容しない素朴法とアバタの向き、オブジェクトの方向の記憶違いをある程度許容する 9-交差法を提案する。

キーワード バーチャルリアリティ, 仮想世界, 位置問合せ, Query-by-Scene

A Proposal of Spatial Position Identification Function in VWDB: A Virtual World Database System

Kozue SATO[†], Chiemi WATANABE[†], and Yoshifumi MASUNAGA^{††}

[†] Graduate School of Humanities and Sciences, Ochanomizu University Otsuka 2-1-1, Bunkyo-ku, Tokyo, 112-8610 Japan

^{††} Faculty of Sciences, Ochanomizu University Otsuka 2-1-1, Bunkyo-ku, Tokyo, 112-8610 Japan
E-mail: [†]{kozue,chiemi}@dmlab.is.ocha.ac.jp, ^{††}masunaga@is.ocha.ac.jp

Abstract The virtual world database system named VWDB is currently under development at Ochanomizu University. This system is constructed by integrating a set of virtual reality systems and an object-oriented database system so that it can support database functions for cyberspace applications. In this paper, we propose a novel query function for VWDB users to find a place in the VWDB virtual world when a user wants to go back the place where he/she has once visited, or the place where his/her friend recommended to go. In order to issue such queries, we introduced a Query-By-Example-like approach in that the user arranges a set of special objects in a query space to draw the scene from memory where he/she wants to go back. Two methods named the "naive method" and the "9-intersection method" are introduced to process the query. It is shown that unlike the naive method, the 9-intersection method is capable of forgiving the users' faulty memory when drawing the scene from memory.

Key words virtual reality, virtual world, spatial position identification, Query-by-Scene

1. はじめに

バーチャルリアリティ (Virtual Reality:以下 VR) 技術は医療, 建築, エンターテイメントなどさまざまな分野で利用されている。我々は, この VR 技術とデータベースを統合した仮想世界データベースシステム (Virtual World DataBase system: 以下 VWDB) の研究を進めている [1] [2] .

VWDB ではユーザが自由にウォークスルーしているため,

例えば, VWDB から出てもう一度 VWDB に入った時, 前に行ったことのある場所に行きたいが何処だか分からなくなってしまった, とか, VWDB に入ったことのある人からこんな風にオブジェクトが配置されている場所があったと聞き, そこに行ってみたい, といったことが考えられる。この場合, 以前に行ったことがあれば「確か自分の前のあの辺にオブジェクト A が, 左のあっちの方に B が, 右斜め後ろに C があるような場所だった。」と目的地を思い浮かべることがあるだろうし, 人に聞

いて得た情報から目的地を想像することもあるだろう。これは、現実世界で例えば茗荷谷駅はどこか、と聞かれた時に、正確に住所などは思い出せなくても、駅を出て目の前にはマクドナルドが、左斜め前には交番があって、すぐ左にドトールがあった、という風景が思い浮かべることが出来るのと同じである。

そこで、我々は VWDB で目的地を問い合わせるために、この思い浮かべた風景 (scene) をそのまま問合せ可能とする機能を実現することを考えた。この問合せ機能を、Query-by-Scene (QBS) と名付ける。

これまで行われてきた空間的問合せに関する研究は、その多くが問合せ言語の形式に基づいてデータベースにオブジェクト間の関係を問い合わせるものだった(「オブジェクト A と B の関係は」とか「オブジェクト A の北にあるオブジェクトは」といったもの)[3]。しかし、地図や絵などの問合せを行う際に、人間の持っている感覚を問合せ言語で表すには限界があると考えられ、問合せ言語を記述する替わりに、Query-by-Example の考えから問合せをペンでスケッチして表現する、Query-by-Sketch などの方法が提案されている [4]。

しかし、VWDB においては「前のあの辺」や「あっちの方」といった自分を中心とする距離的・方向的感覚を表現する時、実際にユーザが仮想空間内で目的地周辺のオブジェクト群を、自分がそこに立っていると仮定して配置する方法、つまり QBS が有効であると考えた。ただ、この方法はユーザが思い浮かべたシーンを再現するので、記憶違いでオブジェクトを実際の距離や方向とずれて配置させてしまうことが考えられる。したがって、QBS の探索方法はこの記憶違いをある程度許容する必要がある。

本稿では、QBS による位置問合せとその探索方法を 2 つ提案し、比較検討する。

2. QBS による問合せ

2.1 QBS とは

本研究では VWDB がサポートする 3 次元仮想空間を対象とし、過去に一度は仮想空間に入ったことのあるユーザが再度仮想空間に入り、以前行った場所に行きたいが、どこかわからなくなってしまった場合を想定して位置問合せ方法を提案する。この場合、ユーザは実際に仮想空間をウォークスルーしたことがあるため、

- 目的地付近のオブジェクトの大まかな方向、距離、大きさなどを自分から見た視覚的イメージとして持っている。
- 方向関係は自分から見てのものなので、ワールド座標における北や東などではなく、自分から見て前、右、後ろ、左といった方向で記憶している。

と考えられる。

我々は、仮想空間で見たものであるならば、スケッチするなど自分の持つイメージを 2 次元にマッピングして表現するよりも、体験したのと同じ仮想空間で、自分からの距離がこれぐらい、あちらの方向にあった、というように表現してもらう方が、より自然であると考えた。そこで、本研究では思い浮かべ

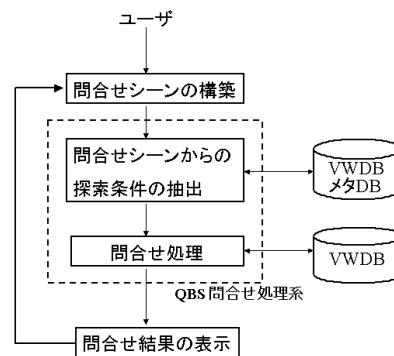


図 1 QBS の処理の流れ

たシーンを再現してもらおう Query-by-Scene (QBS) による位置問合せを提案する。

また、仮想空間でユーザが意識している方向関係は自分から見た前後左右の方向なので、QBS ではワールド座標に依存しない探索方法が必要となる。その探索方法も併せて提案する。

2.2 QBS の処理

QBS は図 1 に示す流れで処理を行う。

問合せシーンの構築

ユーザは以前行ったことのある場所のシーンを思い浮かべながら、問合せ空間でそのシーンを再現し、問合せシーンとする。

問合せシーンからの探索条件の抽出

ユーザが構築した問合せシーンを構成しているオブジェクト群の非空間的属性や空間的属性から、VWDB メタデータベースを参照しながら探索条件を抽出する。

問合せ処理

探索条件を VWDB に発行する。本来、探索条件の抽出とこの問合せ処理は、それぞれ独立させた処理として行う必要があるが、現時点ではまとめて QBS 問合せ処理系として扱っている。

結果表示

問合せの結果をユーザに提示する。問合せ結果として得られるのは、ユーザが構築したシーンに配置されたオブジェクト群の候補と問合せシーンを構築した際にユーザが自分があると仮定した位置である。よって、結果として得られた位置にユーザを移動させたり、オブジェクトを仮想空間の中で強調表示させたりする方法が考えられる。

この結果に満足できない場合には、再び問合せシーンの構築を行い問い合わせる。

2.3 問合せシーンの構築

今回対象とする仮想空間に存在するオブジェクトは、例えば box1, box2, cone, pyramid などのタイプを非空間的属性として持ち、位置、大きさ、形状などを空間的属性として持っている (今回は非空間的属性が同じならば、大きさ、形状も同じであるとしている)。また、一意に識別できるような識別子などは持たないものとする。

我々は、問合せとしてオブジェクトを配置させるための問合せ空間を提供し、ユーザはその空間で目的地周辺の風景を思い

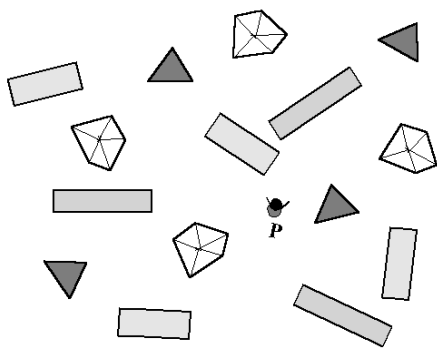


図 2 例として利用する仮想空間を鳥瞰的に見た図

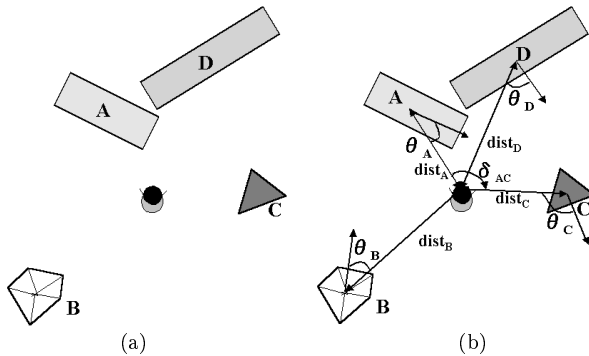


図 3 問合せシーンの構築例 (鳥瞰図)

浮かべながら、次のような手順でオブジェクトを配置させ、問合せシーンを構築する。ここで、この配置させるオブジェクトを問合せ用オブジェクトと呼ぶことにする。

- (1) 仮想空間に存在するオブジェクトのタイプ一覧リストから、思い浮かべたオブジェクトと同じタイプの問合せ用オブジェクトを選択する。
- (2) 問合せ用オブジェクトを思い浮かべたシーンと同じ位置に配置する。
- (3) (1), (2) の操作を繰り返して、思い浮かべたシーンを再現し、問合せシーンとして構築する。

3. 探索方法のモデル化

ユーザが構築した問合せシーンから、それが仮想空間内のどの位置であるかを特定するための探索方法を二つ述べる。

3.1 素朴法

ユーザが問合せシーンを構築した際のアバタとオブジェクト群の距離的・方向的関係やオブジェクト同士の関係を忠実に利用する方法として、素朴法 (naive method) を提案する。

素朴法の説明は、図 2 のようにオブジェクトが配置されている仮想空間を対象とし、ユーザが問合せ空間で P の地点を思い浮かべて、鳥瞰的に見ると図 3 (a) のように問合せシーンを構築した場合を例にして行う。

ここでは説明のために、各オブジェクトを A, B, C, D という名前で識別することにし、距離や角度を以下のとおりで定義した (図 3 (b))。なお、ここで用いる距離や角度はすべて平面上のものとし、高さは含めないものとする。

$dist_A$: アバタと A の距離 ($dist_B, dist_C, dist_D$ も同様)

オブジェクトは形状を持っているので、形状を考慮すると、アバタに一番近いオブジェクトの点との距離 (最短距離) やアバタから一番遠い点との距離 (最長距離) も考えられるが、ここではオブジェクトの重心の座標との距離としている。

$DIST(x, y)$: 地点 x と地点 y の距離

$dist_\epsilon$: 距離に関する許容誤差

θ_A : アバタに対する A の角度 ($\theta_B, \theta_C, \theta_D$ も同様)

アバタと A の重心を結んだ線を、A を中心に反時計回りに回転させ、底面の多角形の辺と初めて垂直に交わる角度とする。

δ_{AC} : A と C の相対角度

A の重心とアバタを結んだ直線をアバタを中心に回転させ、C の重心とアバタを結んだ直線と一致した最小角度。よって反時計回りを正として $-\pi$ から π の範囲で扱う。

δ_ϵ : 角度 δ に関する許容誤差

以下に素朴法で図 3 の問合せの処理を行う手順を示す。ここでは $dist_A \leq dist_C \leq dist_D \leq dist_B$ であるとする。

仮想空間内にある A, B, C, D と同じタイプのオブジェクト群を $\{a_i\}, \{b_j\}, \{c_k\}, \{d_l\}$ とする。

1. 各 a_i に対し、距離が $dist_A$ かつ θ_A の地点 ($x_{im}, m=1,2$ とする) を求める (図 4(a))。A は直方体であるため、 $\theta_A + 180^\circ$ の地点からも A と同様に見ることが出来るので、一つの a_i に対して 2 つの地点 (x_{i1} と x_{i2}) が求まる。
2. 各 c_k に対し、距離が $dist_C$ かつ θ_C の地点 y を求め、以下の条件を満たすか計算する。

$$DIST(x_{im}, y) \leq dist_\epsilon$$

$$x_{im} \text{ において, } \delta_{AC} - \delta_\epsilon \leq \delta_{a_i c_k} \leq \delta_{AC} + \delta_\epsilon$$
 もし条件を満たせば、組合せ (x_{im}, a_i, c_k) を作る。
3. 各 d_l に対し、距離が $dist_D$ かつ θ_D 又は $\theta_D + 180^\circ$ の地点 z_m ($m=1,2$) を求め、2 で得られた組合せ (x_{im}, a_i, c_k) に対して、以下の条件を満たすか計算する。

$$DIST(x_{im}, z_m) \leq dist_\epsilon$$

$$x_{im} \text{ において, } \delta_{AD} - \delta_\epsilon \leq \delta_{a_i d_l} \leq \delta_{AD} + \delta_\epsilon$$
 もし条件を満たせば、組合せ (x_{im}, a_i, c_k, d_l) を作る。
4. 各 b_j で、距離が $dist_B$ かつ b_j の向きに対して θ_B の地点 w を求め、3 で得られた組合せ (x_{im}, a_i, c_k, d_l) に対して、以下の条件を満たすか計算する。

$$DIST(x_{im}, w) \leq dist_\epsilon$$

$$\delta_{AB} - \delta_\epsilon \leq \delta_{a_i b_j} \leq \delta_{AB} + \delta_\epsilon$$
 もし条件を満たせば、組合せ ($x_{im}, a_i, c_k, d_l, b_j$) を作る。
5. 4 で得られた組合せ ($x_{im}, a_i, c_k, d_l, b_j$) を問合せ結果とする。

図 4 は図 2 の一部に素朴法を適用させているところを示している。

この方法は、アバタから一番近い距離にある問合せ用オブジェクトとタイプが同じオブジェクト群に対して、ユーザがい

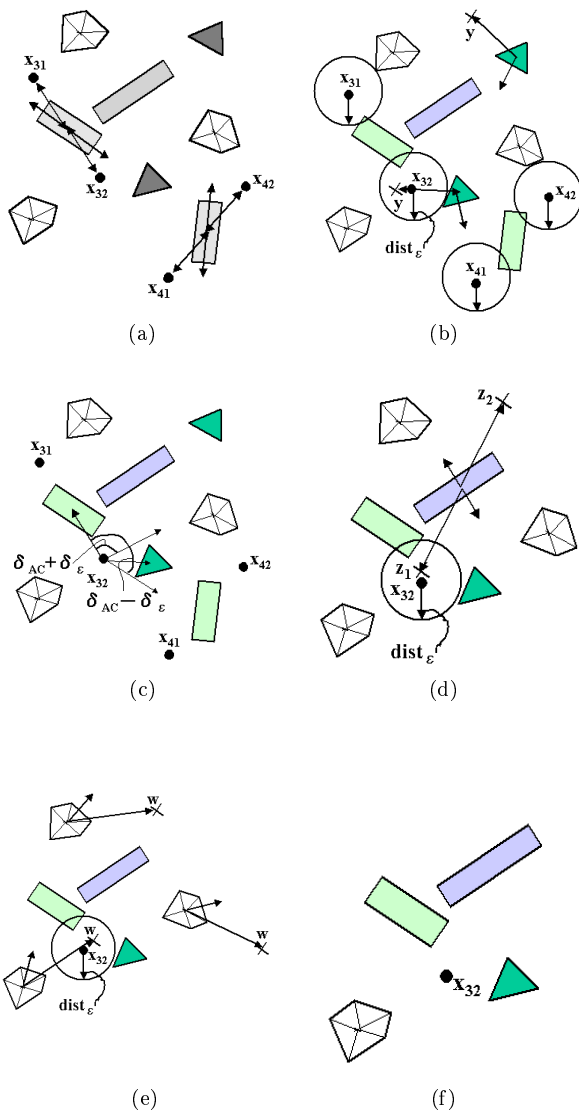


図4 素朴法の探索例

ると仮定した地点を候補として挙げ、その候補地点が妥当であるかどうかを他のタイプのオブジェクト群と計算していく方法である。計算は、アバタからの距離が小さいオブジェクトと同じタイプのオブジェクト群から順に行っていく。よって、 $dist_A \leq dist_C \leq dist_D \leq dist_B$ の関係になっている図3の問合せの処理では、1で a_i 、2で c_k 、3で d_l 、4で b_j の計算を行う。

手順1～4で求めた x_{im} y_{zn} w は、対称性のないオブジェクトでは1点に決まるが、対称性のあるオブジェクトでは複数求まる。例えば、AやDのような直方体だったら2点、もし立方体だったら4点、正n角柱や正n角錐だったらn点求まる。また、円柱や円錐などは正多角形や正多角錐で近似して扱うこととしている。

[特徴]

素朴法は、オブジェクトとアバタの位置関係を θ と $dist$ で、アバタを中心としてオブジェクト間の位置関係を δ で計算している。よって、ユーザが問合せシーンを構築した時に、問合せ用オブジェクトの距離や角度に対して記憶違いをした場合、求め

たい問合せ結果が得られないことも考えられる。特に、問合せ用オブジェクトを配置させた時にアバタからの距離が大きい場合には、 θ に誤差があると、候補地点を求める時に大きく影響してしまい、 $dist_\epsilon$ 内に候補地点が入らず問合せ結果から外れてしまうことが多いと考えられる。

3.2 9 - 交差法

素朴法は、問合せシーンを構築した時の記憶違いをあまり許容することができないと考えられた。しかし、実際にシーンを構築する時には位置や角度に関して記憶違いをしていることが考えられる。そこで、アバタを中心とした方向を近似して扱う方法として、9 - 交差法 (9-intersection method) を提案する。

9 - 交差法の説明も、図2のようにオブジェクトが配置されている仮想空間を対象とし、ユーザが問合せ空間で鳥瞰的に見ると図3(a)のように問合せシーンを構築した場合を例にして行う。

この方法における距離や方向を以下のとおりに定義した。

$dist_A$: アバタとAの重心座標との距離 ($dist_B$ $dist_C$ $dist_D$ も同様)

$DIST(a_i, b_j)$: オブジェクト a_i とオブジェクト b_j の距離

$dist_\epsilon$: 距離に関する許容誤差

$DIR(avt, O_1)$: アバタ (avt) から見たオブジェクト (O_1) の方向

この方向は [4] を参考にして、図5(a)のようにアバタの位置と向きを元に格子を作り、オブジェクトがどの格子の位置にあるかを \emptyset と $-\emptyset$ で表すことにした。

$$DIR(avt, O_1) = \begin{pmatrix} LF_{avt} \cap O_1 & F_{avt} \cap O_1 & RF_{avt} \cap O_1 \\ L_{avt} \cap O_1 & 0_{avt} \cap O_1 & R_{avt} \cap O_1 \\ LB_{avt} \cap O_1 & B_{avt} \cap O_1 & RB_{avt} \cap O_1 \end{pmatrix}$$

LF_{avt} はアバタから見て left-front の方向を、 F_{avt} は front、 RF_{avt} は right-front、 R_{avt} は right、 RB_{avt} は right-behind、 B_{avt} は behind、 LB_{avt} は left-behind、 L_{avt} は left を表している。例えば、A、B、C、Dの方向は次のように表せる。

$$DIR(avt, A) = \begin{pmatrix} -\emptyset & -\emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset \end{pmatrix}$$

$$DIR(avt, B) = \begin{pmatrix} \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset \\ -\emptyset & \emptyset & \emptyset \end{pmatrix}$$

$$DIR(avt, C) = \begin{pmatrix} \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset \\ -\emptyset & \emptyset & \emptyset \end{pmatrix}$$

$$DIR(avt, D) = \begin{pmatrix} \emptyset & \emptyset & -\emptyset \\ \emptyset & \emptyset & -\emptyset \\ \emptyset & \emptyset & \emptyset \end{pmatrix}$$

EF: LF_{avt} F_{avt} RF_{avt} のいずれか又は複数が $-\emptyset$ で、それ以外が \emptyset 。アバタから見て「前の方にある」ことを表す。図5ではAとDがEFである。

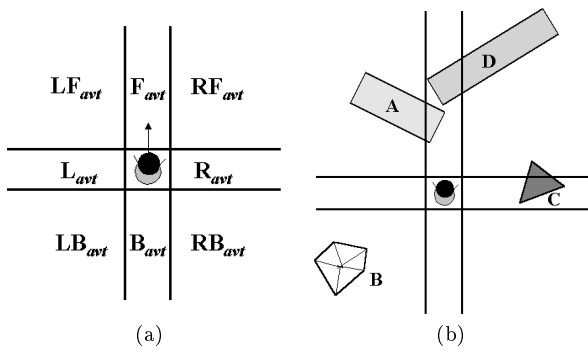


図5 アバタのローカル座標を基にした 9-intersection

- EB: LB_{avt} B_{avt} RB_{avt} のいずれか又は複数が $\neq \emptyset$ で、それ以外が \emptyset . アバタから見て「後ろの方にある」ことを表す. 図5では B が EB である .
- EL: LF_{avt} L_{avt} LB_{avt} のいずれか又は複数が $\neq \emptyset$ で、それ以外が \emptyset . アバタから見て「左の方にある」ことを表す. 図5では B が EL である .
- ER: RF_{avt} R_{avt} RB_{avt} のいずれか又は複数が $\neq \emptyset$ で、それ以外が \emptyset . アバタから見て「右の方にある」ことを表す. 図5では C が ER である .

以下に 9 - 交差法で図5 (b) の問合せの処理を行う手順を示す .

仮想空間内にある A , B , C , D と同じタイプのオブジェクト群を $\{ a_i \} \{ b_j \} \{ c_k \} \{ d_l \}$ とする .

- (a_i b_j c_k d_l) に対して , 距離条件を満たすか計算する . 条件を満たすならば , ステップ 2 に進む .

[距離条件]

$$DIST(a_i, b_j) \leq dist_A + dist_B + 2dist_\epsilon$$

$$DIST(a_i, c_k) \leq dist_A + dist_C + 2dist_\epsilon$$

$$DIST(a_i, d_l) \leq dist_A + dist_D + 2dist_\epsilon$$

$$DIST(b_j, c_k) \leq dist_B + dist_C + 2dist_\epsilon$$

$$DIST(b_j, d_l) \leq dist_B + dist_D + 2dist_\epsilon$$

$$DIST(c_k, d_l) \leq dist_C + dist_D + 2dist_\epsilon$$

これらをすべて満たすならば , 距離条件を満たすという .

- (a_i b_j c_k d_l) に対して , 方向条件を満たすか計算する . 条件を満たしていれば , 次のステップに進む .

[方向条件]

アバタを挟んで反対の位置に見えるオブジェクト (EF と EB のオブジェクトや EF と ER のオブジェクト) は , その二つが前後 , 又は左右に見ることの出来るアバタの向きがワールド座標系に対して角度 θ で求めることが出来る . その θ が存在するかどうかを判定する . この例では EF と EB の関係で A と B , D と B が , EL と ER の関係で B と D が考えられ , 3 つの角度 θ_1 θ_2 θ_3 が求められる .

$$\theta_1 \cap \theta_2 \cap \theta_3 = \theta_{dir}$$

$\theta_{dir} \neq \emptyset$ ならば方向条件を満たすという .

- (a_i b_j c_k d_l) を問合せ結果とする .
- 全ての組合せ (a_i b_j c_k d_l) に対して 1 ~ 3 を行う .

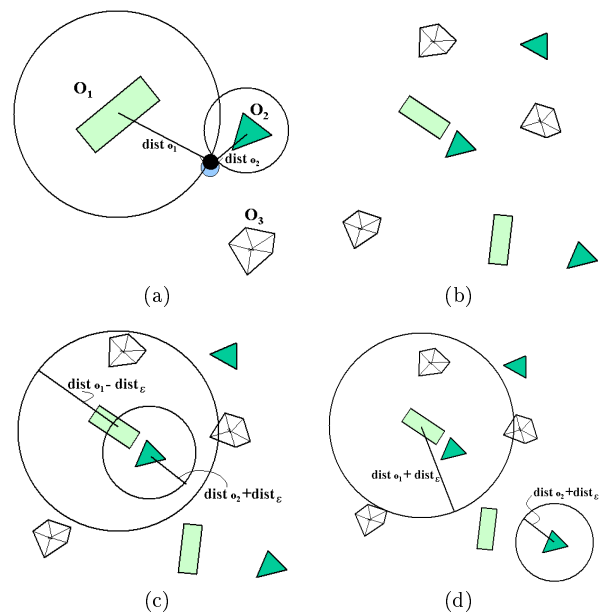


図6 距離条件

この方法は , ユーザが配置した問合せ用オブジェクトの候補オブジェクト群に対して , そのように配置させることができるユーザの位置 , 向きが存在するかどうかを判定していく方法である .

[距離条件について]

距離条件は , 条件を満たしているか判定しているオブジェクトに対して , アバタが存在することのできる地点があるかどうかを見ている . 例えば , 図6 (a) のように問合せ用オブジェクトを配置させ , 図6 (b) のオブジェクトがそのオブジェクトに該当するかをみる . 図6 (a) では , o_1 を中心に半径 $dist_{o_1}$ の円と o_2 を中心に半径 $dist_{o_2}$ の円を描くと , 2 つの円は少なくとも1点 , 多くても2点で交わる . 条件ではこれに誤差 $dist_\epsilon$ を考慮し , $|dist_{o_1} - dist_{o_2}| - 2dist_\epsilon \leq DIST(o_1, o_2)$ は図6 (c) のようなオブジェクトの組合せを , $DIST(o_1, o_2) \leq dist_{o_1} + dist_{o_2} + 2dist_\epsilon$ は図6 (d) のようなオブジェクトの組合せを除外する .

[方向条件について]

方向条件は , EF のオブジェクトがアバタの前に , EB のオブジェクトがアバタの後ろに , EL のオブジェクトがアバタの左に , ER のオブジェクトがアバタの右にあるようなローカル座標を作ることができるかを判定している .

ここで , 条件として用いる θ の具体的な求め方について説明する . ユーザが図6 (a) のように配置したとする . a_3 と c_4 に対して , o_1 が EL , o_2 が ER の関係にあることから a_3 が左 , c_4 が右に見ることのできるユーザの視線の角度 θ を求める . 考え方は , 図7 (a) (b) に示すとおり , 問合せ空間でアバタを囲っている格子の幅を考慮して θ を求めるのだが , 計算上の煩雑さを少なくするために , アバタを囲っている格子の幅は限りなく0に近く (c) (d) のようにアバタが線上にいるものとして計算を行う . そして , この2つの線は a_3 と c_4 の各頂点を結んだ平面上の線のワールド座標系に対する傾きが最大のものと

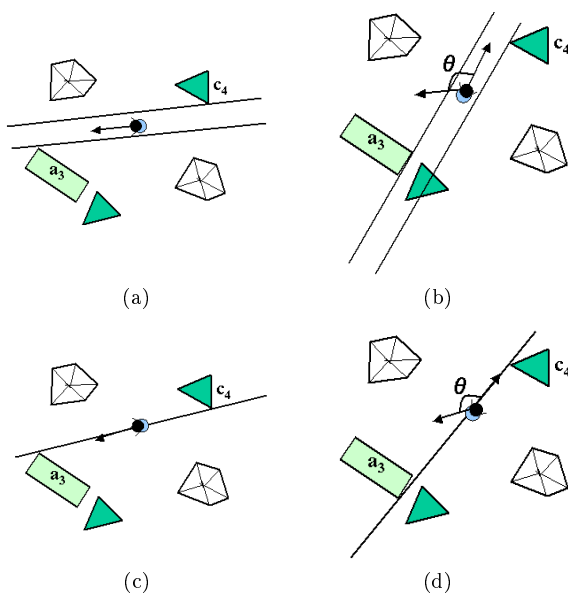


図 7 方向条件の θ

最小のものになっている。

[特徴]

一般に、アバタの位置は点で表されるが、アバタの前後左右に距離を設けて格子を作ることにより、ユーザが問合せシーンを構築した時の方向の記憶違いを許容することができる。もし、アバタを点として考えて前後左右だけで方向を捉えると、境界線近くのオブジェクトの方向は、少しのずれで前と後ろ、右と左が変わってしまうことになる。

一方、そのために図 8 のような配置上の違いは識別しないので、ユーザが意図しないオブジェクト群も条件を満たしてしまう場合も増えると考えられる。

また、方向条件として EF と EB, EL と ER のようにアバタを挟んで反対に位置するオブジェクトを利用しているので、どこか一方方向に集中して配置した場合には、方向条件を作ることが難しくなる。よってユーザが問合せシーンを構築する時に、最低二組は EF と EB や EL と ER の関係になるようなオブジェクトを配置するというような制約が必要となる。

3.3 素朴法と 9-交差法の比較

素朴法は、問合せ処理の課程でアバタとオブジェクトの角度 θ やアバタに対するオブジェクト同士の角度 δ を利用しているので、ユーザが目的地付近の風景を再現する際のユーザの向きや、風景に存在するオブジェクトの方向、距離などに関する記憶違いをあまり許容することが出来ない。しかし、9-交差法はアバタから見たオブジェクトの方向を 8 方向で近似しているので、記憶違いをかなり許容することが出来ると考えられる。

例えば、実際には図 6 (a) のような配置になっているオブジェクトを、(b) のように D のオブジェクトをずれて配置した場合、素朴法ではこのオブジェクト群は検索結果として出ないが ((c)), 9-交差法では距離条件は満たしており、かつ例えば (d) のようなローカル座標が作れるので、問合せ結果として出る。

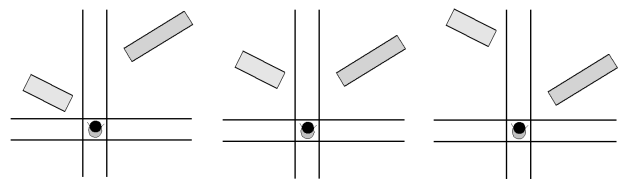


図 8 9-intersection による方向の近似

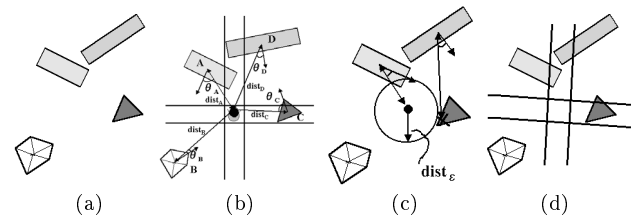


図 9 素朴法と 9-交差法

4. まとめと今後の課題

Query-by-Scene による位置問合せとその探索方法、素朴法と 9-交差法を提案した。今回はすべてタイプの異なる 4 つの問合せオブジェクトを配置した場合を例にして説明したが、同じタイプのオブジェクトを複数配置する場合、3 つや 5 つのオブジェクトを配置する場合も同様に考えることが出来る。

今回示した方法は、ユーザが問合せオブジェクトの配置を行った時の正確性が重要となってくる。例えば、距離に関する誤差 $dist_e$ に関して、正確性が高ければ誤差は小さくとることができるし、低ければ誤差は大きくとる必要がある。また、それに伴って意図している結果以外の問合せ結果がどれだけ含まれるかの適合率は変化するだろうし、意図している結果がもれてしまうことも出てくると考えられる。実際にどれくらい正確に再現できるかの実験を行って適合率の変化などのデータを集め、検討していくのと同時に、時間計算量についても考慮していく必要がある。

今回は仮想空間内にランドマークなどの一意に決まるようなオブジェクトは無いものとして考えたが、仮想空間における位置問合せにはランドマークは有効である [5]。ランドマークになるようなオブジェクトがある場合の問合せ方法も併せて考えていきたい。

文 献

- [1] 渡辺知恵美, 大杉あゆみ, 佐藤こず恵, 増永良文: VWDB: 仮想世界データベースシステム-基本的概念とシステム構成-, 情報処理学会第 62 回全国大会講演論文集 (2001).
- [2] 渡辺知恵美, 大杉あゆみ, 佐藤こず恵, 増永良文: 仮想世界データベースシステム: VWDB における共有型作業環境のためのトランザクション概念の導入, 情報処理学会論文誌, Vol.43, SIG9(TOD15), pp.55-67 (2002).
- [3] X. Liu, S. Shekhar, S. Chawla.: Object-based Directional Query Processing in Spatial Databases, *IEEE Transactions of Knowledge and Data Engineering*, (2001).
- [4] M. Egenhofer.: Query Processing in Spatial-Query-by-Sketch, *Journal of Visual Languages and Computing*, Vol. 8, No. 4, pp. 403-424 (1997).
- [5] ナイジェル・フォアマン, ラファエル・ジレット編, 竹内謙彰, 旦直子監修, “空間認知研究ハンドブック,” 二瓶社, 2001.