

Title	全方位カメラを用いた距離制限制約付き美術館問題に関する研究
Author(s)	小林, 公樹
Citation	
Issue Date	2009-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/8140
Rights	
Description	Supervisor:浅野哲夫, 情報科学研究科, 修士

修 士 論 文

全方位カメラを用いた距離制限制約付き
美術館問題に関する研究

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

小林公樹

2009 年 3 月

修 士 論 文

全方位カメラを用いた距離制限制約付き 美術館問題に関する研究

指導教官 浅野哲夫 教授

審査委員主査 浅野哲夫教授
審査委員 上原隆平准教授
審査委員 平石邦彦教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

710031 小林公樹

提出年月: 2009 年 2 月

概 要

建物を監視する際，現在では，防犯カメラが利用される．防犯カメラとして全方位カメラという通常のカメラよりも撮影範囲の広い特殊なカメラを使用する事で建物内でのカメラの設置台数を減らす事が期待されるが，通常のカメラと特性や仕組みが異なるため設置場所を決める際には，注意が必要である．そのため，本研究では防犯カメラとして全方位カメラを用いて建物全体を監視する際の最適配置を求める事，防犯カメラ設置台数を削減することを目的とする．

目次

第1章	はじめに	1
1.1	研究の背景と目的	1
第2章	全方位カメラの性質	3
2.1	全方位カメラとは	3
2.2	従来のカメラとの違い	4
第3章	美術館問題と自由視点合成	6
3.1	美術館問題とは	6
3.2	自由視点の合成	6
3.2.1	モーフィング	6
第4章	問題の定義と分類	8
4.1	問題の定義	8
4.2	準備	8
4.3	美術館問題との関係	8
第5章	最適配置を求めるアルゴリズム	10
5.1	基本設定パラメータ	11
5.2	辺での最適配置	11
5.3	角での最適配置	11
5.4	各辺での余り	16
5.5	スタート位置の調整	18
5.6	二分探索	21
5.7	長方形での最適配置	22
5.8	直交多角形での最適配置	23
5.8.1	270°の角での最適配置	23
5.8.2	最適配置アルゴリズム	26
第6章	まとめ	27

第1章 はじめに

1.1 研究の背景と目的

近年、建物内のセキュリティーでは防犯カメラが必要不可欠である。高層ビルや巨大な駅などが立ち並ぶ都市では、建物内部の間取りが複雑になっており防犯カメラの数も膨大になってきている。そのため、建物内を必要最低限のカメラの設置により監視することができればコスト、メンテナンス等の手間を削減する事ができる。そのため設置するカメラの台数を減らす事や、効率よくそれらを設置する位置を考える事はとても重要である。

本研究では、設置する防犯カメラとして全方位カメラを用いる。このとき、建物内に設置すべき防犯カメラの最小値を求めたい。全方位カメラは、1回の撮影で周囲 360° の風景を撮影可能することができる。全方位カメラは設置位置よりある一定の距離離れると極端に解像度が低下するという特性を持っている。これは、撮影できる範囲に距離制限が生じるという事である。全方位カメラを用いることで環境全体を少数で監視することができるが、カメラの特性や仕組みが通常のカメラとは異なるため通常の防犯カメラと同じように利用するのは難しい。そのため、全方位カメラを用い必要最低限の設置により建物を監視する問題を考える事はとても重要である。

また、全方位カメラを用いることにより、美術館問題をより現実的な問題として捉えることができる。美術館問題とは、 n 個の辺で構成された多角形 P の内部領域をギャラリーに見立てて、警備員を配置し監視する際に警備員の最小人数を求める問題である。美術館問題では警備員は 360° の視野を持ち、その視野に含むものを常に監視できると仮定している、これは警備員や通常の防犯カメラを想定したときには、無理のある仮定である。しかし、全方位カメラを用いることにより、美術館問題での警備員の視野を実際に確保する事ができ、警備員の持つ 360° の視野を現実的に考える事が出来る。また、美術館問題では一般の多角形について問題を考えているが実際の美術館や建物の間取りは直交多角形である場合が多い。そのため本研究では、全方位カメラの特性を生かし、より現実的な問題を考えた。

本研究では上記の問題を計算幾何学における最適化問題として捉え全方位カメラを用いて撮影することにより生じる画像の距離の変化による解像度低下を距離制限の制約とみなし、美術館問題を直交多角形という一般的な制約のもとで考察することにより設置カメラの台数が最小となる最適配置を考案した。

全方位カメラを用いることは建物内のウォークスルーを構成する際にも役立つ。ここで言うウォークスルーとは、撮影した全方位画像より構成された仮想空間の中を実際に歩き

進んでゆく人物の視点を表現することである．通常のカメラにより，ウォークスルーを構成しようと思うと人物の視点方向情報が必要になる．また，構成するウォークスルーも道のりが決定されていなければならない．後から変更することはできない．建物内の全てを歩き回れるようなウォークスルーを構成したいと思うと撮影しなければならない場所も膨大になる．例えば，ある地点から 360° 全て見渡せるような画像が欲しいとしよう．ここで，通常のカメラの1回で撮影出来る角度を 120° だとすると， 360° の画像を得ようと思うとその地点で少なくとも3回は撮影しなければならない．また撮影の際にどの写真がどの方向を撮ったものなのかという情報も管理しなければならない．そして，これらの操作を沢山行なわなければいけない．

しかし，全方位カメラを用いることで1回の撮影で 360° 全ての情報を得られるため，方向情報が不要になる．そのため，ウォークスルーの道のりも自由に変更可能となる．また，撮影済みの複数の全方位画像から撮影していない場所での全方位画像を合成することも出来るのでカメラの設置台数を大幅に削減することができる．

第2章 全方位カメラの性質

2.1 全方位カメラとは

全方位カメラとは，双曲面ミラーと従来のカメラを組み合わせたものである．図 2.1 に示すように，三次元上での点を $P(X, Y, Z)$ ，全方位カメラの画像上の点を $p(x, y)$ とし，画像上の任意の点 p と双曲面の焦点 F' とを結ぶ直線が，双曲面と交差する点を p' とすると，双曲面のもう一つの焦点 F と p' とを結ぶ直線上に点 p の三次元上での点 P が存在する．そのため，水平方向 360° の画像情報を一枚の画像に取り込むことができる．また，入力画像を基に焦点 F を中心としたパノラマ画像，透視投影変換画像を容易に作り出すことができる．全方位カメラの仕組みを図 2.1 に示す．全方位カメラは双曲面の性質から二つの

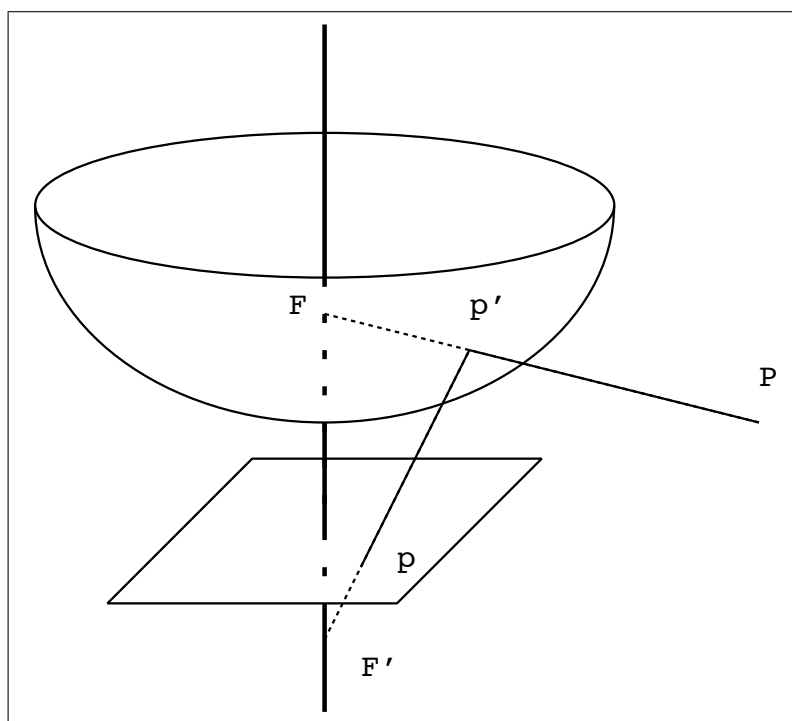


図 2.1: 全方位カメラの仕組み

焦点を持ち，一つの焦点に向かう光線は双曲面ミラーで反射されてもう一つの焦点へ集まる．三次元上の点 $P(X, Y, Z)$ と全方位カメラ画像上の点 $p(x, y)$ には以下のような関係が

成り立つ．

$$\frac{X^2+Y^2}{a^2} - \frac{Z^2}{b^2} = -1 \quad (2.1)$$

$$\text{焦点} : F : (0, 0, c), F' : (0, 0, -c) \quad (2.2)$$

$$c = \sqrt{a^2 + b^2} \quad (2.3)$$

$$x = \frac{Xf(b^2-c^2)}{(b^2+c^2)(Z-c)-2bc\sqrt{(Z-c)^2+X^2+Y^2}} \quad (2.4)$$

$$y = \frac{Yf(b^2-c^2)}{(b^2+c^2)(Z-c)-2bc\sqrt{(Z-c)^2+X^2+Y^2}} \quad (2.5)$$

全方位カメラで撮影された画像の例を図 2.2 に示す．

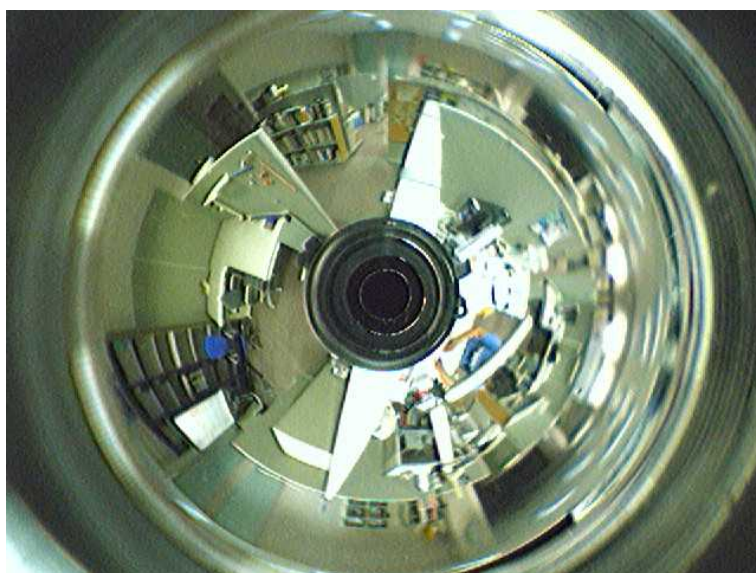


図 2.2: 全方位カメラ出力画像

2.2 従来のカメラとの違い

全方位カメラと通常のカメラを比較すると、全方位カメラは周囲 360° の画像が得られるため、画像間での対応点の消失が起こりにくいという違いが上げられる。また、通常のカメラで全周囲の画像を取得するには複数回の撮影を行ない、それらの画像を一つの画像として編集する必要があるが、全方位カメラではこれらの手間を省くことが出来る。

しかし、360° の範囲を一枚の画像に収めるため通常のカメラと比較して解像度が低下する。また、画像は双曲面ミラーに映ったものなので画像の中心からの距離によって解像度が異なる。美術館問題では警備員は 360° の視野を持ち、その視野に含むものを常に監視している事となっているが、これは警備員や通常の監視カメラでは無理のある仮定であ

る．しかし，全方位カメラを用いることにより，美術館問題での警備員の視野を実際に確保する事ができ，警備員の持つ 360° の視野を現実的に考えることが出来る．

第3章 美術館問題と自由視点合成

3.1 美術館問題とは

美術館問題とは、 n 個の辺で構成された多角形 P の内部領域をギャラリーに見立てて、警備員を配置し監視する際に警備員の最小人数を求める問題である。ギャラリーが単純 n 直交多角形の場合には、必要十分な警備員数は $\lfloor \frac{n}{3} \rfloor$ 人であることが知られている [1],[2]。

ただし、 $\lfloor x \rfloor$ は実数 x の整数部分である。また、ギャラリーに h 個の穴がある場合でも、ギャラリー全体の辺数および穴を構成する辺数の総数が n である時 $\lfloor \frac{n}{3} \rfloor$ 人の警備員で必要十分である事も知られている [3]。

しかし、一般の多角形 P に対しては P が単純である場合でさえ NP 困難であることが知られている [2]。

3.2 自由視点の合成

全方位画像の自由視点の合成とは、撮影済みの全方位画像を用いて撮影していない場所での全方位画像を合成することである。図 3.1 のように自由視点を合成したい場所 (図中では対応点) の全方位画像を周囲の全方位画像から必要な部分を切り出し合成することで得る。全方位画像の自由視点合成により取得済みの画像から実際に画像を取得していない地点での画像を合成することが出来るため建物内に設置するカメラを減らすことができる。以下では [4] の手法を使って複数の全方位画像から自由視点画像を生成する方法について述べる。

3.2.1 モーフィング

得られた全方位画像にモーフィングを適用することにより高速に自由視点全方位画像を生成する手法を用いる。この手法は環境中の複数の地点で全方位画像を取得し、画像間の対応を与えることにより自由視点における全方位画像を生成する。全方位カメラの位置、姿勢および入力画像間の対応点は既知とする。

1. 入力画像上の対応点の座標を入力とする全方位ステレオによって、その対応点の 3 次元位置を計算する。

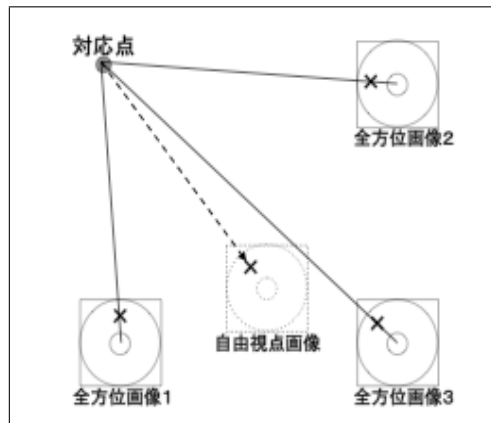


図 3.1: 対応点の自由視点画像への投影

2. 1 で得た 3 次元点を自由視点上に投影する.
3. 投影された点群に対して デローネイ三角形分割を行ない, 三角形パッチを生成する .
4. 3 で得た三角形パッチに対応する三角形の画像を各入力画像から切り出し, 自由視点位置に基づいて算出される重みを用いてブレンドし, 自由視点における全方位画像とする .

操作の詳細は [4] を参考にする .

第4章 問題の定義と分類

4.1 問題の定義

n 辺から成る多角形 P として表現されるギャラリーの壁面を全方位カメラにより監視したい．ギャラリーの壁面をどの点も少なくとも 1 台の全方位カメラにより指定範囲の解像度で監視できるようにするには，最小何台のカメラをどのように配置すればよいか．

4.2 準備

全方位カメラにより撮影された画像は，撮影位置より遠ざかると解像度が低くなるという性質を持っている．そのため，一定の解像度で監視を行うためにはある範囲内で撮影する必要がある．これは，つまり撮影に距離制限が生じる事になる．カメラの設置位置からの距離制限の事を R と呼ぶことにする．これは設置位置を中心とした半径 R の円となる．解像度を確保したい場所に関しては近くの位置から撮影が必要となる．今回の場合は，絵画が飾ってある壁面を R 以内の距離で撮影する必要がある．

また，全方位カメラは双曲面ミラーを 2 つ合わせた構造をしておりミラーを真下から見るとミラーの中心には穴があいている．そのため，撮影された画像には図 4.1 に示すように撮影不可能な範囲が出来る．この撮影不可能範囲は，平面で見ると円になる．この円の半径を r とする．この r と R を用いると図 4.2 に示すように，本研究の問題は半径 R と半径 r の円で定まる環形で多角形の辺をくまなくカバーする問題となる．

以降この r 以上 R 以下の領域を撮影可能領域と呼ぶことにする．

4.3 美術館問題との関係

一般の多角形 P に対して $r = 0$, $R = \infty$ とすると美術館問題と同じ問題となるため，この問題は NP 困難である．そこで，ここでは問題を簡単にするために多角形が直交多角形であるという制約を付けて問題を考える．

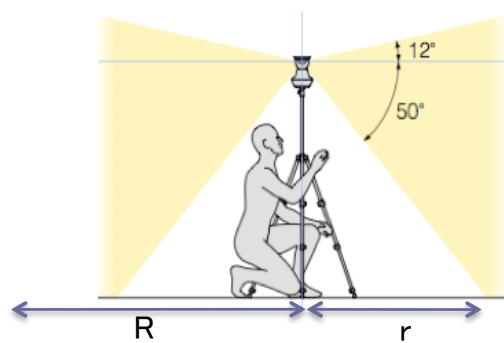


図 4.1: 全方位カメラの撮影原理と撮影不可能領域の定義

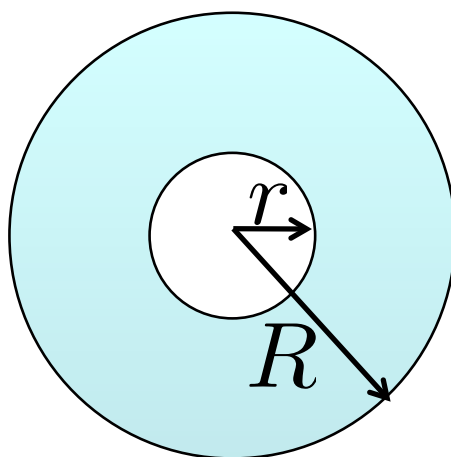


図 4.2: 撮影可能領域を表わす環形 2 つの同心円に囲まれた領域

第5章 最適配置を求めるアルゴリズム

ギャラリーを表す多角形 P が長方形である場合について問題を考え次に直交多角形の場合に一般化する．基本的な設置の方針としては， P の辺上で最初のカメラの設置場所を決めそこから時計回りで隙間なく敷き詰めていく．隙間のある設置方法も考えられるが隙間のある設置方法はカメラの個数を増やすことなく隙間のない設置方法に移行できる．そのためここでは，隙間のない設置方法のなかで設置台数が最小のものを発見することにより最適配置を求める．

また，最初のカメラの設置位置を変化させることにより個数が変化する場合があります．これについても調べなければならない．スタート位置の変化により個数が変化する例について具体的に見てみよう．ここでは， $r = 1.0, R = 6.0$ と設定し，ギャラリーを表わす多角形 P として，最も単純な図形である一辺の長さが 22.0 の長さの正方形を考えよう．このような簡単な場合ですら最適解を求めるのは自明ではない．最初のスタート位置を左上の頂点より右に 2.0 離れた地点にしてみよう．このとき，図 5.10 の (a) に示すように全ての壁をカバーするには 7 台のカメラ設置が必要となる．次にスタート位置を左上の角より右に 5.8 離れた所にすると，コーナー位置に置いたカメラにより 2 つの辺に撮影可能領域を拡大できるためカメラの設置台数は 6 台になる図 5.10 の (b) ．

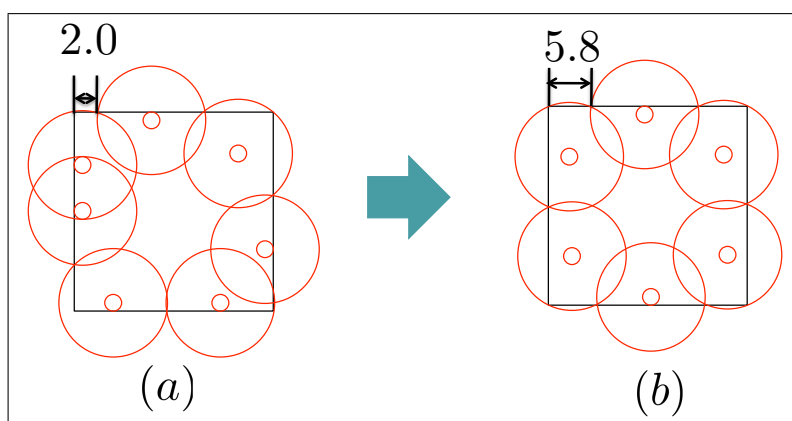


図 5.1: スタート位置の違いによる個数の変化.

(a) では 7 箇所で撮影しなければならないが，(b) のようにコーナーをうまく利用すると 6 箇所で済む

5.1 基本設定パラメータ

カメラの設置台数を最小にしたいので設置したカメラの撮影距離を最大にする必要がある．そのため，撮影距離が最大になるカメラの設置位置，つまり最適配置を求める必要がある．カメラを設置した時の壁の最大撮影距離の事を以下では最大撮影可能長と呼ぶ事にする．長方形の場合，撮影の種類を撮影する状況によって分類すると辺での撮影と角での撮影の2つに分類する事ができる．それぞれについて最適配置を求めなければならない．

5.2 辺での最適配置

1つの辺をカバーするだけなら， (r, R) で定義される環形で表わされる撮影可能領域と対象の壁面との共通部分の長さ（撮影される壁の部分の長さ）が最大になるようにカメラを置くのがよい．そのような位置は図5.2に示すように小さい方の円が壁面に接するように置くことで達成される．また，この時に撮影出来る距離を d_{max} と呼ぶことにする．簡単な計算で $d_{max} = 2\sqrt{R^2 - r^2}$ であることがわかる．また，図5.3のように撮影距離 d_{max} を保ったままギャラリーを表す多角形の頂点が半径 R の円の円周上にくるように設置すると， d_{max} を定める壁面と直角の壁面にも少し撮影出来る範囲がある．その状態を示したのが図5.3である．同図のように直角に交わる2つの辺を同時にカバー出来るようにカメラを置いたとき，2つの辺にわたって撮影出来るので得策である．隣接壁面での撮影部分の長さが最大になるのは，大きい方の円がちょうど角の頂点を通るときである．簡単な観察により，その長さ δ_{max} は， $2r$ に等しいことがわかる．

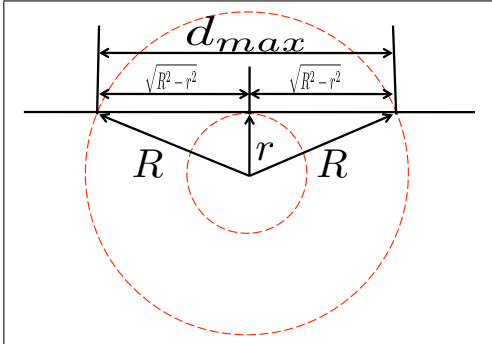


図 5.2: 最大撮影可能長 d_{max} の定義

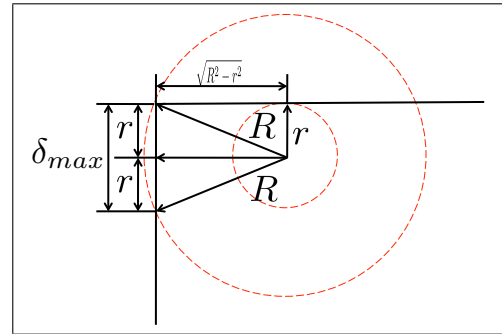


図 5.3: 隣接壁面最大撮影可能長 δ_{max} の定義

5.3 角での最適配置

1つの頂点だけをカバーするだけならば，図5.4の(a)に示すように (r, R) で定義される環形で表わされる撮影可能領域と対象壁面との共通部分の長さが最大になるようにカメ

ラを置くのがよい．そのような位置は簡単な観察により撮影可能領域の中心が角の二等分線上にあり，かつ大きい方の円がちょうど角の頂点を通る時である事がわかる．しかし，今回は隙間無く敷き詰めていくため図 5.4 の (a) に示すように角だけで局所的な最大撮影可能長を考えるのではなく図 5.4 の (b) に示すように敷き詰める際に出る任意の余りに対しての最大撮影可能長を考える必要がある．

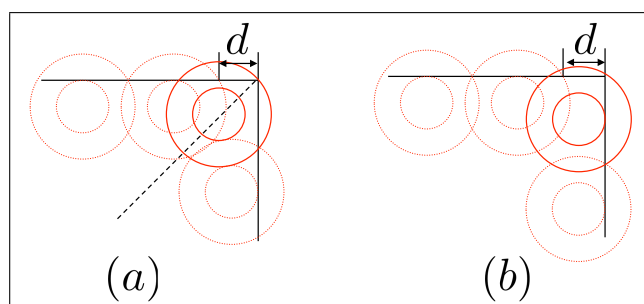


図 5.4: 角でのカメラ設置方法

(a) のように局所的な最大撮影可能長を求めるのではなく (b) のように任意の余りに対しての最大撮影可能長を求める．

以下では角でのカメラ設置時の関係式より敷き詰めの際の任意の余りに対する隣接壁面での最大撮影可能長を求める．角を原点とし x 軸を任意の余りの出る辺, y 軸を敷き詰めの際の次の辺とする．撮影可能領域の円の中心を (C_x, C_y) とする図 5.5 . 円の中心 (C_x, C_y) は，撮影不可能領域の半径 r があるため， $C_x \geq r, C_y \geq r$ を満たさなければならない．また，頂点（ここでは原点）は撮影可能領域の半径 R 以内に入っていないといけないため $C_x^2 + C_y^2 \leq R^2$ を満たさなければならない． x を余り y を次の辺での最大値とするため， $(C_x - x)^2 + C_y^2 = R^2$, $C_x^2 + (C_y - y)^2 = R^2$ となる．

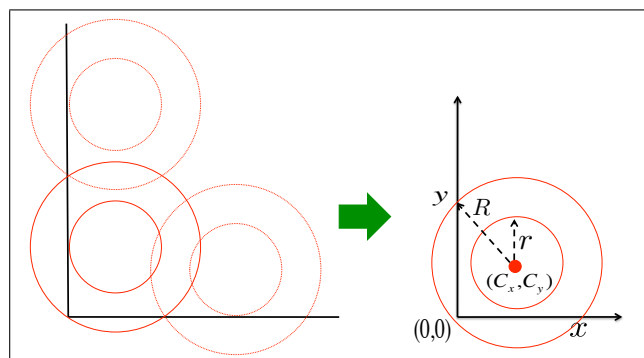


図 5.5: 任意の余りに対する最大撮影可能長を求める

以上の関係をまとめると以下のような式になる .

$$C_x \geq r, C_y \geq r, \quad (5.1)$$

$$C_x^2 + C_y^2 \leq R^2, \quad (5.2)$$

$$(C_x - x)^2 + C_y^2 = R^2, \quad (5.3)$$

$$C_x^2 + (C_y - y)^2 = R^2. \quad (5.4)$$

次に以上の式より y の最大値を求める . 式 5.4 より

$$\begin{aligned} (C_y - y)^2 &= R^2 - C_x^2, \\ C_y - y &= \pm \sqrt{R^2 - C_x^2}, \\ y &= C_y \pm \sqrt{R^2 - C_x^2}. \end{aligned} \quad (5.5)$$

y を最大にしたいので

$$y = C_y + \sqrt{R^2 - C_x^2}. \quad (5.6)$$

式 5.3 より次式を得る .

$$\begin{aligned} C_y^2 &= R^2 - (C_x - x)^2, \\ C_y &= \pm \sqrt{R^2 - (C_x - x)^2}. \end{aligned} \quad (5.7)$$

式 5.6 より y を最大にしようとする C_y を最大にしなければならないので

$$C_y = \sqrt{R^2 - (C_x - x)^2}. \quad (5.8)$$

式 5.6 と式 5.8 より次式を得る .

$$y = \sqrt{R^2 - (C_x - x)^2} + \sqrt{R^2 - C_x^2}. \quad (5.9)$$

式 5.9 を C_x について微分すると

$$y' = \frac{1}{2} \frac{-2(C_x - x)}{\sqrt{R^2 - (C_x - x)^2}} + \frac{1}{2} \frac{-2C_x}{\sqrt{R^2 - C_x^2}}. \quad (5.10)$$

$y' = 0$ とし変極点を求めると次のようになる .

$$\begin{aligned}
& \frac{(-C_x+x)\sqrt{R^2-C_x^2}-C_x\sqrt{R^2-(C_x-x)^2}}{\sqrt{R^2-(C_x-x)^2}\sqrt{R^2C_x^2}} = 0, \\
& (-C_x+x)\sqrt{R^2-C_x^2}-C_x\sqrt{R^2-(C_x-x)^2} = 0, \\
& (-C_x+x)\sqrt{R^2-C_x^2} = C_x\sqrt{R^2-(C_x-x)^2}, \\
& (-C_x+x)^2(R^2-C_x^2) = C_x^2(R^2-(C_x-x)^2), \\
& (x-C_x)^2(R^2-C_x^2) = C_x^2R^2-C_x(x-C_x)^2, \\
& (x-C_x)^2R^2-C_x^2(x-C_x)^2 = C_x^2R^2-C_x^2(x-C_x), \\
& (x-C_x)^2 = C_x^2, \\
& x-C_x = C_x, \\
& x = 2C_x, \\
& C_x = \frac{x}{2}.
\end{aligned} \tag{5.11}$$

変極点は $C_x = \frac{x}{2}$ となる .

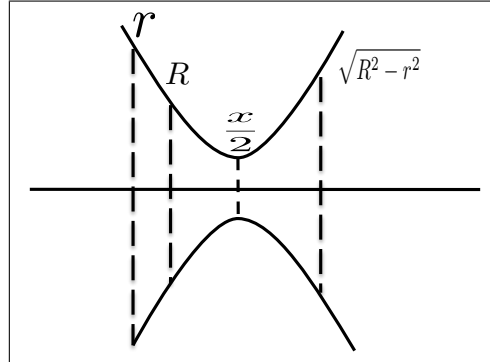


図 5.6: 関数 y

しかし , この関数 y は増加関数なのか減少関数なのかわからないので以下の 3 つの値の最大値を最大撮影可能距離とする . まず初めに C_x の取りうる値の下限である r を代入すると

$$y_r = \sqrt{R^2 - (r-x)^2} + \sqrt{R^2 - r^2}$$

となるが , もし x が R 以上の値の時は $C_x = R$ とする .

$$y_R = \sqrt{R^2 - (R-x)^2}$$

次に変極点の値を代入すると次式を得る .

$$\begin{aligned} y_{\frac{x}{2}} &= \sqrt{R^2 - \left(\frac{x}{2} - x\right)^2} + \sqrt{R^2 - \frac{x^2}{4}} \\ &= 2\sqrt{R^2 - \frac{x^2}{4}} \end{aligned} \quad (5.12)$$

次に , C_x の取りうる上限の値である $\sqrt{R^2 - r^2}$ を代入することにより次式を得る .

$$\begin{aligned} y_{\sqrt{R^2 - r^2}} &= \sqrt{R^2 - (\sqrt{R^2 - r^2} - x)^2} + \sqrt{R^2 - R^2 + r^2} \\ &= \sqrt{r^2 + 2\sqrt{R^2 - r^2}x - x^2} + \sqrt{r^2} \end{aligned} \quad (5.13)$$

以上の 3 値の最大値を取り最大撮影距離を求める .

$$y_{max} = \max\{y_{r,R}, y_{\frac{x}{2}}, y_{\sqrt{R^2 - r^2}}\} \quad (5.14)$$

また , 余りが δ_{max} 以下である時は最大値として d_{max} を返す .
最大値を求める関数の擬似プログラムをアルゴリズム 1 に示す .

Algorithm 1 Returnmax(x) 任意の余りに対して隣接する壁面での最大撮影可能長を求める関数

```

1:  $R, r$  : カメラの半径
2:  $\delta_{max}, d_{max}$  最大撮影可能長
3:  $maxarray[]$  : 配列
4:
5: input x 余り
6: if  $x \leq \delta_{max}$  then
7:   return  $d_{max}$ 
8: end if
9: if  $x \geq R$  then
10:  配列  $maxarray$  に  $\sqrt{R^2 - (x - R)^2}$  の値を追加
11: else
12:  配列  $maxarray$  に  $(\sqrt{R^2 - (r - x)^2} + \sqrt{R^2 - r^2})$  の値を追加
13: end if
14: 配列  $maxarray$  に  $2\sqrt{R^2 - \frac{x^2}{4}}$  の値を追加
15: 配列  $maxarray$  に  $\sqrt{r^2 + 2\sqrt{R^2 - r^2}x - x^2} + \sqrt{r^2}$  の値を追加
16: return  $maxarray$  の最大値

```

5.4 各辺での余り

設置台数を求めるためには敷き詰めた時に出る各辺での余りにより次の辺での最適配置を求めるため余りを配列などで保持する必要がある．そのため，各辺での余りを求め配列に追加していく．各辺での余りはスタート位置を設定した最初の辺では辺長から初期値である δ_{max} を引いたものを d_{max} で割った余りを配列に追加する．次の辺では，先程求めた余りを使い関数 $Returnmax(x)$ により次の辺での最大撮影可能長を求める．辺長から先程求めた最大撮影可能長を引き d_{max} で割った余りを配列に追加．次の辺でも同様の操作をする．最後の辺では，辺長からスタート位置により求めた最大撮影可能長と一つ前の辺で求めた最大撮影可能長を引いたものを d_{max} で割った余りを配列に追加する．各辺での余りの入った配列を返す関数の擬似プログラムをアルゴリズム 2 に示す．

Algorithm 2 Returnarray(x) 各辺での余りを格納した配列を返す関数

```
1:  $R, r$  : カメラの半径
2:  $\delta_{max}, dmin$  最大撮影可能長
3:  $la, lb$  長方形の辺の長さ
4: array[] 余りを入れる配列
5:
6: input x スタート位置
7: array[0]  $\leftarrow (la - x) \% d_{max}$ 
8: for  $i = 0$  to 1 do
9:   if  $i == 0$  then
10:     $l \leftarrow lb$ 
11:   else
12:     $l \leftarrow la$ 
13:   end if
14:   array[i + 1]  $\leftarrow (l - returnmax(array[i])) \% d_{max}$ 
15: end for
16: array[3]  $\leftarrow (lb - returnmax(array[2]) - returnmax(x)) \% d_{max}$ 
17: return array 余り配列
```

そして，求めた余りをもとにカメラの設置台数を計算する．台数を求める部分はアルゴリズム 2 とほとんど同様である．各辺での余りを求め配列に追加すると同時にスタート位置の辺と最後の辺以外では， $\lceil (\text{辺長} - \text{前の辺の余りにより求めた最大撮影可能長}) \div d_{max} \rceil$ によりその辺での設置台数を求め合計台数加算する．スタート位置の辺では， $\lceil (\text{辺長} - \delta_{max}) \div d_{max} \rceil$ により計算し合計台数に加算する．

最後の辺では， $\lceil (\text{辺長} - \text{スタート位置の値により求めた最大撮影可能長} - \text{前の辺の余りにより求めた最大撮影可能長}) \div d_{max} \rceil$ により台数を求め合計台数に加算する．ここで，スタート位置のカメラの個数を勘定に入れていなかったのが最後に合計台数に加える．以

上の操作により合計台数を求める．設置台数を求める関数の擬似プログラムをアルゴリズム 3 に示す．以上の操作によりスタート位置を決定すれば設置する台数を求める事が出

Algorithm 3 Returncnt(x) カメラの合計設置台数を求める関数

```

1:  $R, r$  : カメラの半径
2:  $\delta_{max}, dmin$  最大撮影可能長
3:  $la, lb$  長方形の辺の長さ
4:  $array[]$  余りを入れる配列
5:
6: input  $x$  スタート位置
7:  $array[0] \leftarrow (la - x) \% dmax$ 
8:  $array$  に  $(la - x) \% dmax$  を追加 # 最初の辺でのあまり
9:  $sumcnt = ((la - x) / dmax).ceil$  # 最初の辺での個数
10: for  $i = 0$  to 1 do
11:   if  $i == 0$  then
12:      $l \leftarrow lb$ 
13:   else
14:      $l \leftarrow la$ 
15:   end if
16:    $array[i + 1] \leftarrow (l - returnmax(array[i])) \% dmax$ 
17:    $sumcnt+ = ((l - returnmax(array[i])) / dmax).ceil$ 
18: end for
19:  $array[3] \leftarrow (lb - returnmax(array[2]) - returnmax(x)) \% dmax$ 
20:  $sumcnt+ = ((lb - returnmax(array[2]) - returnmax(x)) / dmax).ceil$ 
21:  $sumcnt+ = 1$  スタート位置に設置したカメラの数
22: return  $sumcnt$  個数

```

来る．

5.5 スタート位置の調整

敷き詰めのスタート位置を変化させることによりカメラの個数が変化する場合がある．そのため，スタート位置を変化させながらカメラの台数を計算しそのうちの最小値を見なければならない．

しかし，スタート位置候補は指数通りあり全てについて設置台数を求めるわけにはいかない．そこで以下では有限回の操作で最適配置を求めるアルゴリズムについて紹介する．

最初に設置するカメラ位置を決定しそのカメラを時計回りにずらしていくことによりスタート位置を変化させ全体の設置台数の変化を見る．この時のずらす幅は高々カメラ1台分で良い．なぜなら1台分以上移動させるとそれは次のカメラに移ったことになるためである．最初に設置するカメラとして長方形の左上の頂点とする図 5.7．ここで，図 5.7 のカメラの撮りうるこの辺での最小値から最大値までスタート位置を変化させ個数の変化をみる．これによりカメラを1台分時計回りにずらしたことになる．

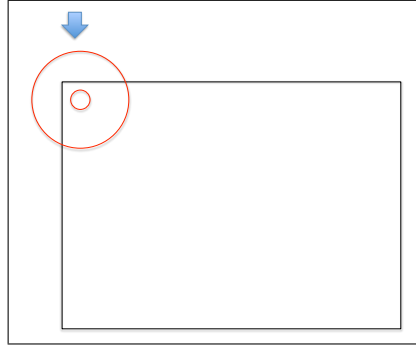


図 5.7: スタート位置.最初にカメラを設置する場所をここにする

最初に設置するカメラの撮影距離の最小値は本来ならば0になるはずだが，図 5.3 のように辺での最大撮影可能長 d_{max} を撮りながら隣接壁面でも δ_{max} 撮影可能であるため，最初の辺での最小値を δ_{max} とする．また，最大値は図 5.2 に示すように d_{max} となる．そのため，図 5.8 に示すようにスタート位置 d を $\delta_{max} \leq d \leq d_{max}$ ($2r \leq d \leq 2\sqrt{R^2 - r^2}$) の範囲で変化させる事になる．次に個数を調べなければならないスタート位置候補を見つける．そこで各頂点に設置されたカメラに注目し構造変化のあり，なしを見る．スタート位置 d を適当な値に決定し各頂点で出る余りを $d_1, \dots, d_n$ とする．スタート位置を変化させていくと $d_1, \dots, d_n$ の最小値から順に次の辺に移る．ここで， $\alpha = \min\{d_1, \dots, d_n\}$ とし，スタート位置を $d + \alpha$ に変化させたときに出る余りを $d'_1, \dots, d'_n$ とする．ここで d_i と d'_i を比べた時にもし $d_i \leq d'_i$ となっていれば図 5.10 に示すように構造変化が起っており，そうでないなら図 5.9 に示すように構造変化は起っていない．

そのため，全ての $d_i \leq d'_i$ となっているような頂点を配列などに蓄えておく．このようなスタート位置を探索する理由は，スタート位置はカメラ1台分しか変化しないため，構造変化が起きた時にのみ個数が変化する可能性があるためである．この操作をカメラが丁

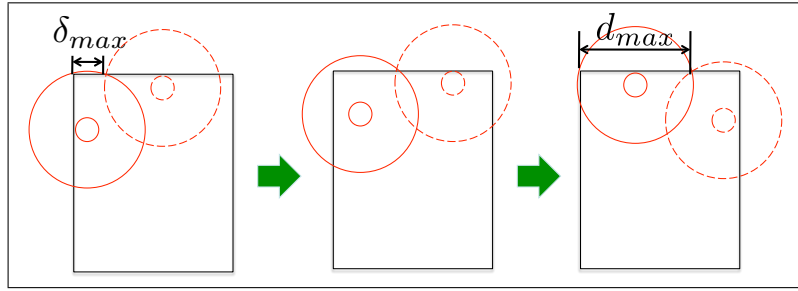


図 5.8: スタート位置の変化

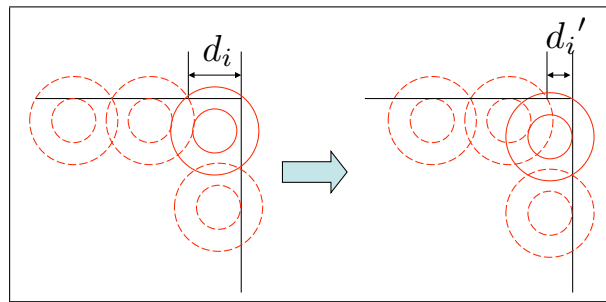


図 5.9: $d_i > d_i'$ 構造変化が起こらない例.角を担当するカメラ (実線円) が変化していない

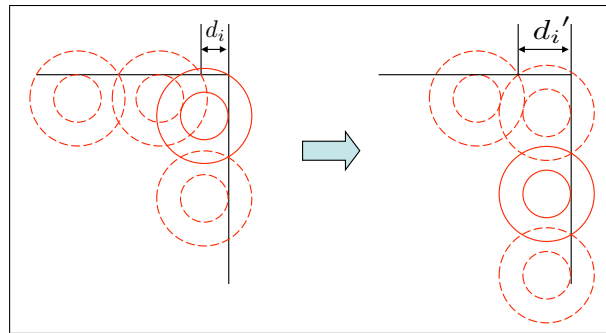


図 5.10: $d_i \leq d_i'$ 構造変化が起きる例 角を担当するカメラ (実線円) が変化している

度 1 台分右にずれるまで繰り返す．つまり，最初に述べた通りスタート位置 d の初期値を δ_{min} とし， $d + \alpha \leq d_{max}$ になるまで繰り返す．アルゴリズム 4 にスタート位置の調整のアルゴリズムを示す．

Algorithm 4 Start(a) スタート位置の調整

```
1: r,R カメラの半径
2:  $\delta_{max} = 2r, d_{max} = 2\sqrt{R^2 - r^2}$ 
3: e=0.99 誤差
4: la,lb 四角形の各辺
5: result[] グローバル配列 (ちょうど構造変化が起きているスタート位置を格納する配列)
6: array1[]  $\leftarrow \text{Returnarray}(d_{min})$  グローバル配列 (余りを格納する配列)
7:  $min \leftarrow \text{array1.min}$  グローバル
8:
9: input a カメラの設置位置の初期値
10: while  $a \leq d_{max}$  do
11:   array2  $\leftarrow \text{Returnarray}(a)$ 
12:   list = [] 構造変化が起きる余りの番号を入れる配列
13:   for  $i = 0$  to  $\text{array1.length} - 1$  do
14:     if  $\text{array1}[i] \leq \text{array2}[i]$  then
15:       list に  $i$  を追加
16:     end if
17:   end for
18:   for  $i = 0$  to  $\text{list.length} - 1$  do
19:     if  $i == 0$  then
20:       result に  $bsearch1(\text{list}[i], \delta_{max}, \delta_{max} + min), bserch2(\text{list}[i], \delta_{max}, \delta_{max} + min)$  を追加
21:       //二分探索により構造変化が起きる直前と余りが0になるスタート位置を求める,
22:     else
23:       result に  $bsearch1(\text{list}[i], \delta_{max}, a + min), bserch2(\text{list}[i], \delta_{max}, a + min)$  を追加
24:     end if
25:      $a += \text{array2.min}$ 
26:   end for
27: end while
```

5.6 二分探索

構造変化が起きる直前と余りが0になるスタート位置を二分探索により発見する．スタート位置の初期値(ここでは δ_{max})を探索の左端とし，探索範囲の右端として二分探索が実行された時のスタート位置とする．何故この2つだけを求めるかというと，余りが $\delta_{max}, 0$ のときに隣接壁面の撮影で最も得をするため，この2つを求める．また，この2つを用いないで最小個数を構成できないため，二分探索により求めたスタート位置のいずれかが最小個数を構成する．構造変化が起きるスタート位置を正確に出すのは容易ではないため，ある誤差範囲の中に求める値が入れば結果としてその時のスタート位置を返す．そのため e を誤差の範囲とし，求めた余りが $\delta_{min} \pm e$ の範囲に入っていれば構造変化が起きる直前の場所としてその時のスタート位置を返す．同様の方法で，余りが0になる地点も求める．この時も e を誤差の範囲とし，スタート位置を求める．

つまり， $left$ を探索の左端とし， $left$ の値を δ_{max} ，探索範囲の右端を $right$ とし， $right$ の値を二分探索が実行されたときのスタート位置とする， mid を中央値とし mid の値を $(left + right) \div 2$ とする．この mid の値をスタート位置とし，アルゴリズム2の関数を使い余りの入った配列を得る．この余りの入った配列の要素の該当する値が $\delta_{min} - e$ 以上 $\delta_{min} + e$ 以下であるならば構造変化が起きる直前の位置として mid を返す．また，この範囲に入っていないならば引き続き二分探索を続行する．求めた余りがスタート位置を変化させる前の余り以上の値ならば左側を探索し，そうでないならば右側を探索する．この探索を余りが誤差範囲に入るまで繰り返す．同様に0になる地点も求める．

以下に構造変化が起きる直前のスタート位置を求めるのための二分探索の擬似プログラムをアルゴリズム5に示す．余りが0になるスタート位置を求めるための二分探索も殆ど同様である．

Algorithm 5 bsearch1(a,b,c) 2分探索

```
1: array1[] グローバル配列 (余りを格納)
2:
3: input a, b, c 構造変化が起きているスタート位置のどれか, スタート位置の初期値,
   変化させたスタート位置
4: array[] ローカル配列
5: left  $\leftarrow b$ 
6: right  $\leftarrow c$ 
7: mid  $\leftarrow (left + right)/2$ 
8: array  $\leftarrow rearray(mid)$ 
9: if ( $dmin - e \leq array[a] || array[a] \leq dmin + e$ ) then
10:  return mid
11: else
12:  if array1[a]  $\leq array[a]$  then
13:    bsearch(a, left, mid) 左を探索
14:  else
15:    bsearch(a, mid, right) 右を探索
16:  end if
17: end if
```

5.7 長方形での最適配置

最後に今まで述べた関数を使い長方形での最適配置を求める．二分探索により，構造変化が起きる直前と余りが0になるスタート位置を求めた．今度はそのスタート位置よりアルゴリズム3を用いて設置する台数を求める．求めた設置台数の最小値を結果として返す．これにより，スタート位置候補のうち最小個数を構成する可能性のあるスタート位置全てについて調べたため，この結果が最適解となる．実行部分の擬似プログラムをアルゴリズム6に示す．

Algorithm 6 main(a) 実行部

```
1:  $dmin = 2r, dmax = 2\sqrt{R^2 - r^2}$ 
2: result[] グローバル
3: array1[]  $\leftarrow Returnarray(dmin)$  グローバル
4:  $min \leftarrow array1.min$  グローバル
5: array[]
6:
7: start(dmin) 初期値として dmin を代入
8: for  $i$  to result.length - 1 do
9:   array に recnt(result[i]) を追加
10: end for
11: print array.min
```

5.8 直交多角形での最適配置

次にギャラリーを表わす多角形 P が直交多角形である場合について考える．直交多角形の場合，角が $90^\circ, 270^\circ$ の2種類ある． 90° での任意の余りに対する最大値は長方形の時の方法で求めることが出来るが， 270° の場合の任意の余りに対する最大値を求める必要がある．

5.8.1 270° の角での最適配置

図 5.11 の b の領域のように 270° の角では，撮影領域には入っているが撮影出来ない場所が存在する．そのため， 270° の角では両方の壁を撮影しようとするカメラを配置できる場所が限られる．

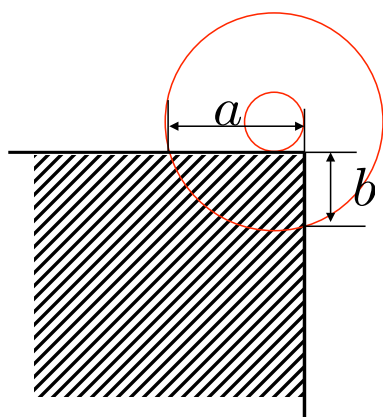


図 5.11: 撮影領域

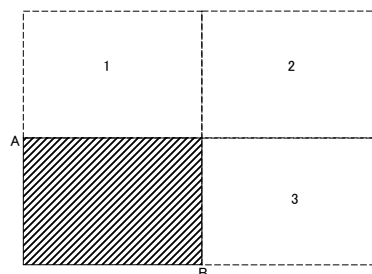


図 5.12: 撮影可能領域

図 5.12 の 1 の領域にカメラを設置してしまうと壁 B が撮影出来ない．任意の余りに対して次の壁での撮影距離を最大にしたいためどちらか一方しか撮影できない設置方法ではいけない．そのためには図 5.12 の 1 の領域には設置できない．同様に図 5.12 の 3 の領域では壁 A が撮影できないため設置できない．両方の壁を撮影するためには図 5.12 の 2 の領域内に設置しなければならない．また，撮影不可能な領域の半径 r があるため領域は図 5.13 の 2 の領域になる．

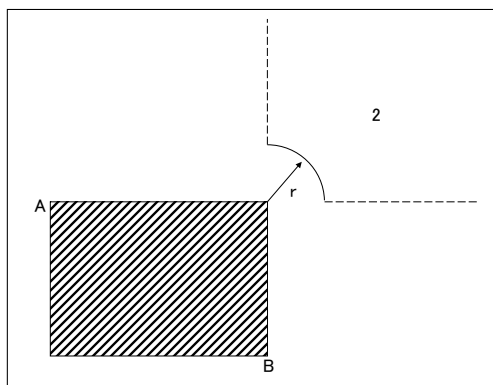


図 5.13: 設置候補地

ここで壁に対して設置位置の角度があまりに小さいと有効な画像が得られない．そのためここではある程度壁に対して角度を付けて設置する．この角度の事を以下では許容角 (θ) と呼ぶ事にする．許容角を導入すると設置可能位置は以下の図 5.14 のようになる．

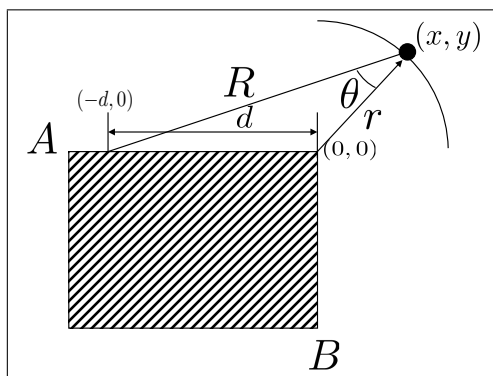


図 5.14: 許容角

許容角を設定すると撮影可能な余りの値の上限が出てくる．多角形の頂点を原点とし，カメラを設置する場所を (x, y) とする図 5.14．余弦定理より，次式を得る．

$$\begin{aligned} d^2 &= r^2 + R^2 - 2rR \cos \theta, \\ d &= \sqrt{r^2 + R^2 - 2rR \cos \theta}. \end{aligned} \quad (5.15)$$

撮影可能な余りは式 5.15 のようになる．次に， d の値を用いて半径 r, R の二つの円の方程式より，次式を得る．

$$x^2 + y^2 = r^2, \quad (5.16)$$

$$(x + d)^2 + y^2 = R^2. \quad (5.17)$$

式 5.16 より次式を得る．

$$y^2 = r^2 - x^2. \quad (5.18)$$

式 5.17 を展開する．

$$x^2 + 2dx + d^2 + y^2 = R^2. \quad (5.19)$$

式 5.19 に式 5.18 を代入すると次式を得る．

$$\begin{aligned} x^2 + 2dx + d^2 + r^2 - x^2 &= R^2, \\ 2dx + d^2 + r^2 &= R^2, \\ 2dx &= R^2 - r^2 - d^2, \\ x &= \frac{R^2 - (r^2 + d^2)}{2d}. \end{aligned} \quad (5.20)$$

式 5.18 より

$$\begin{aligned} y^2 &= r^2 - x^2, \\ y &= \pm \sqrt{r^2 - x^2}, \\ y &\geq 0 \\ y &= \sqrt{r^2 - x^2}. \end{aligned} \quad (5.21)$$

式 5.21 に式 5.20 の値を代入すると次式を得る．

$$\begin{aligned} y &= \sqrt{r^2 - \left(\frac{R^2 - (r^2 + d^2)}{2d}\right)^2} \\ (x, y) &= \left(\frac{R^2 - (r^2 + d^2)}{2d}, \sqrt{r^2 - \left(\frac{R^2 - (r^2 + d^2)}{2d}\right)^2}\right) \end{aligned} \quad (5.22)$$

ここで，もし余りが式 5.15 の値以下の時は式 5.15 と式 5.22 より (x, y) を求め次に最大撮影距離を計算する．この時の最大値は，隣接壁面でも許容角を満たさなければならないため $\sqrt{r^2 + R^2 - 2rR \cos \theta}$ となる．もし，余りが式 5.15 の値以下でないなら余りに対してカメラ 1 台設置し，次の壁では頂点を新たなスタート位置として計算する．

5.8.2 最適配置アルゴリズム

多角形の各辺の長さや頂点の角度は与えられているものとする。直交多角形の場合、本質的な変更点は角での最適配置の部分だけである。角度の判定をし 90° ならばアルゴリズム 1 の *Returnmax* を使い最適配置を求め、そうでないなら 270° の時の求め方で求める。アルゴリズム 7 に直交多角形での任意の余りに対する最大撮影可能長を求める擬似プログラムを示す。

Algorithm 7 直交多角形

```
1: input x 余り
2: if 角は  $90^\circ$ ? then
3:   return Returnmax(x)
4: else
5:   if  $x \leq \sqrt{r^2 + R^2 - 2rR \cos \theta}$  then
6:     return  $\sqrt{r^2 + R^2 - 2rR \cos \theta}$ 
7:   else
8:     return 0 頂点を新たなスタート地点とする。
9:   end if
10: end if
```

第6章 まとめ

本論文では，距離制限制約付きの美術館問題を全方位カメラを用いてギャラリーを表す多角形が直交多角形である場合について長方形の場合を基礎とし最適配置を示した．今後の課題としては，ギャラリーを表す多角形が穴のある直交多角形の場合やより一般的な応用として一般の多角形への拡張が考えられる．また，実際の建物での防犯カメラとしての利用やウォークスルーの構築など実装実験等への応用も考えられる．また，全方位カメラを防犯カメラとして使用した場合と通常の防犯カメラの場合の設置費用の比較などを行い実際の使用を考慮した問題として考えていく事なども今後の課題としてあげられる．

謝辞

研究にあたり，数々のご助言を頂いた浅野哲夫教授に深く感謝致します．また，学生生活を含め様々な面でサポートして頂いた上原隆平准教授，清見礼助教授に厚く御礼申し上げます．

参考文献

- [1] Joseph O'Rourke, *Art Gallery Theorems and Algorithms*, OXFORD UNIVERSITY PRESS, 1987.
- [2] Joseph O'Rourke, "*An alternative Proof of the Rectilinear Art Gallery Theorem*" *Journal of Geometry*, Vol.21,pp.118-130, 1983.
- [3] F.Hoffmann "*On the Rectilinear Art Gallery problem*", LNCS, 433, Springer Verlag, pp.716-728, 1995.
- [4] 富手要, 山澤一誠, 横矢直和, 複数の全方位画像を用いた広範囲なウォークスルーの実現, 画像の認識・理解シンポジウム (MIRU2002) 講演論文集, Vol. II, pp. 353-358, (2002.8).