

ロープウェイ式害獣検知システムにおけるロープの設置手法

小倉 広大¹ 勝間 亮¹

概要: 近年, シカ, イノシシ, サルなどの害獣による農村部への被害が深刻化している. 害獣被害を抑える対策として, WSN(Wireless Sensor Networks) による害獣モニタリングシステムにより害獣を検知し, 音や光等で追い払うことで, 被害を軽減することが期待されている. しかし, 一般的なモニタリングシステムでは監視カメラを固定しており, 植物の生育などにより撮影領域が減少するなどの問題に対処することが難しい. そこで, 本稿では, ロープ上を監視カメラが移動できるロープウェイ式害獣検知システムを提案し, ロープウェイ式害獣検知システム導入時におけるロープの設置コストをできるだけ少なくする手法について提案する. 提案手法では, できるだけ少ない監視カメラで効率的にモニタリングを行うために, 複数に分割されている監視対象領域をまとめて監視できる領域をクラスタリング方式で求め, ロープを設置する位置を決定する. シミュレーション実験により, 提案手法では, 一般的なクラスタリング手法である k-means 法と比べると, より少ない監視カメラで全監視領域をモニタリングできることが確認できた.

A Method to Deploy Ropes for Ropeway-type Vermin Detection Systems

KODAI OGURA¹ RYO KATSUMA¹

1. はじめに

近年, 中山間地域などにおいて, シカ, イノシシ, サルなどの野生獣, いわゆる害獣による農村部への被害が深刻化・広域化している. 特に畑を荒らされる被害が多く発生しており, 日本では, 2008 年度~2014 年度まで, 毎年約 200 億円にも上るほどの被害額がでているほどである [1] [2] [3]. 害獣被害は農作物への被害だけでなく営農意欲の減退や, 耕作放棄地の増加をもたらすなど被害額として数字に現れる以上に農村に深刻な影響を与えており, 対策を講じる必要がある.

害獣対策として一般的なものには, 害獣を捕獲する罠を山中に設置する方法や, 山と畑との境界に防止柵や電気柵を設置し, 畑への侵入を阻止する方法がある. しかし, 罠を設置する方法は害獣を捕獲したかを確認するために管理者が見回りをしないといけなく, 防止柵や電気柵を設置する方法は, 安全面の不安や, 破損や漏電の確認をする必要が出てくるなど人手がかかる.

そこで, WSN(Wireless Sensor Network) を利用して, 罠に出入りする頭数をセンサと人工知能で解析し, 罠に入る個体数が最大となるタイミングでゲートを閉じるように自動操作できるシステムや, 捕獲網をワイヤーでつり下げ, 網の下に害獣が大量に入ったことを監視カメラで確認した上で, 網を落として捕獲するシステム等が利用されている [1]. アイニク株式会社では, 加速度センサ付 RF 監視タグ, 赤外線センサ付 RF 監視タグにより罠に掛かった害獣を検知し, その情報を M2M でメールサーバに転送し, 管理者に捕獲メールを送るシステムを紹介している [4]. また, 小川は, web 上で双方向通信・制御できるロボット監視カメラを作成し, 監視と追い払い機能を同時に実現できる機器を開発した [5]. このロボット監視カメラは日中はカメラの動体検知機能を使い, 夜間は赤外線感知の外部センサー機能を使い害獣を検知する. このように, 害獣をモニタリングして対策を講じる方法が注目を集めている. しかし, カメラを固定する既存のモニタリングシステムでは, 一度設置したらその設置場所から動かすことが困難なため, 植物の生育などでモニタリング不能となった場合に対処が容易ではない.

¹ 大阪府立大学
Osaka Prefecture University, Sakai, Osaka 599-8531, Japan

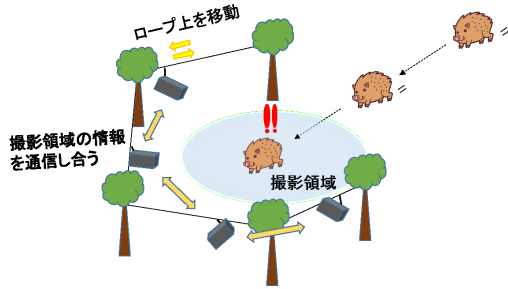


図 1 ロープウェイ式害獣検知システム概略図

そこで、移動式監視カメラを使った害獣検知手法が必要となってくる。移動式監視カメラを利用することで、障害物を避けて映すことが可能であり、撮影領域を適切に保つことができる。例えば、地上走行型や空中飛行型のノードに監視カメラを搭載すると、山中では木々が生い茂っているため、木の根や枝にひっかり転倒や故障の恐れがある。そこで、本研究では木々の枝と枝との間にロープを張り巡らせ、ロープ上を移動可能な複数のカメラが協調移動しながら対象領域を監視するロープウェイ式害獣検知システムを提案する。ロープを設置したい位置に枝がない場合は、ポールを立てるなどをして、ロープを張る。

ロープウェイ式害獣検知システムの概略図を図 1 に示す。張り巡らしたロープ 1 本につき、1 個の監視カメラを設置し、害獣を検知する。監視カメラはロープ上を自由に移動でき、適切な位置に移動して対象領域を監視することができる。また、監視カメラに通信機を搭載することで、他の監視カメラと撮影領域についての情報を通信し合い、協調移動することで、自身が映していない領域を他の監視カメラが映すことで、システム全体の撮影領域を大きくできる。以下、ロープ上に監視カメラ、通信機等を搭載したノードを設置するものとし、監視ノードと呼ぶ。ロープウェイ式害獣検知システムを導入する利点は、上に挙げたことの他にも、設置コストが低いので、害獣がよく現れる場所に監視ノードを多く配置することができることや、故障を想定して予備の監視ノードを配置することが可能なのがある。ロープウェイ式害獣検知システムでは、監視ノードはロープ上を移動するため、撮影領域はロープの設置場所に影響を受ける。そのため、どこにロープを設置するかは重要な問題になる。

本稿では、ロープウェイ式害獣検知システムにおいて、導入する監視ノードの数をできるだけ少なくしつつ、満足な監視領域を得るための設置手法を提案する。あらかじめ複数の監視したい領域が与えられているとき、同時に複数の領域を監視できる位置にカメラを設置すると効率的である。そのようなカメラ設置すると効率的である。そのようなカメラ位置はクラスタリング手法である k-means 法で求めることができる [6]。しかし、本問題はモニタリングの質を保つためにロープの距離や角度の制限が設けられるた

め、単純な k-means 法では適切なロープの設置位置を求めることができない可能性が高い。そこで提案手法では、集約領域を決定する処理と、ロープの設置位置を決定する処理を交互に行い、毎処理後にロープの長さや角度を考慮することで適切なロープの設置位置を求める。また、監視領域を集約する際は、監視カメラの撮影可能距離よりも小さい範囲を想定して集約することにより、監視カメラが所属するロープ上のある一点に存在するときのみモニタリング可能といった領域を作らないように工夫する。シミュレーション実験により、k-means 法を用いて全集約領域を決定した後にロープの設置位置を決定する方法に比べ、提案手法では必要な監視ノードの数を少なくすることが出来た。また、監視カメラの撮影可能距離内で監視領域を集約した場合と、撮影可能距離の $\frac{3}{4}$ の距離で集約した場合を比較すると、必要な監視ノードの数はあまり変わらなかった。

2. 問題設定

本稿のねらいは、ロープウェイ式害獣検知システム導入時の効率の良いロープの設置手法の提案である。本章では、害獣検知を考える上で想定する環境、条件について述べる。

2.1 監視領域

害獣検知では、監視ノードの撮影領域内で、害獣の姿を撮影するため、監視ノードの設置場所が大事となる。山中を移動する動物はやみくもに行き来するのではなく、コースを決めて移動する。例えば、人が作った林道などがそうである。動物が通るコースのことを獣道といい、獣道では、足元の下草は食われて短くなっていたり、低木の小枝は折れたりしているため、肉眼で発見しやすい。害獣検知ではそのような獣道などの害獣が現れやすい場所を監視対象とする。

本稿では、害獣検知を考える上で、あらかじめ、獣道などの監視対象において特に監視したい場所が点として定まっているものとする。つまり、監視領域は点として与えられる。以下、監視領域を監視点という名称で呼ぶ。監視点の集合を $S = \{s_0, s_1, \dots, s_n\}$ で定義する。監視点には固有の ID が割り振られている。

2.2 ロープ上に配置する監視ノード

ロープ上には各ロープにつき 1 台の監視ノードを設置する。監視ノードは通信機、監視カメラ、小型 PC、センサを搭載している。監視ノードは自身が設置されているロープ上でのみ自由に移動でき、他のロープに移ることはできない。各監視ノードは半径 R_c の通信範囲に存在する他のノードと通信して情報を交換することができる。通信機は Zigbee を用い、その通信可能距離は機器の電波出力により数メートルから数キロメートルまで様々に変更可能である。監視カメラは 360°カメラを想定する。監視カメラの撮影可能距離を r_d とする。

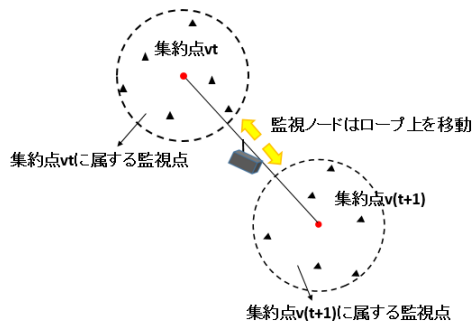


図 2 ロープ上を移動する監視ノードの撮影領域

2.3 監視点の集約

提案手法では、複数の監視点を同時に監視することができる場所にロープを張ることで、用いる監視ノードの数を減らす。複数の監視点を同時に監視できる場所を点として求め、その点を集約点と呼ぶ、集約点の集合を $V = \{v_0, v_1, \dots, v_m\}$ で定義する。ロープをある集約点と別の集約点とを結ぶように張ることで、そのロープ上を移動できる監視ノードは2つの集約点に移動できる。そのため、1つの監視ノードで2つの集約点に含まれる監視点を監視できる。具体例を図2に示す。ロープが集約点 v_t と v_{t+1} を結ぶように張られているとすると、監視ノードはそのロープ上を移動することができるため、集約点 v_t , v_{t+1} に属する監視点を監視することが可能である。

ある集約点 $v \in V$ からロープの結び先となる集約点の集合を $E(v)$ とし、 E を任意の v に対する $E(v)$ の集合とすると、集約点と設置されるロープの位置関係は、集約点を頂点とし、ロープの結び先となる集約点の関係を辺とした無向グラフ $G = (V, E)$ とみなせる。

2.4 集約距離係数

監視ノードは集約点から張られるロープ上を移動するが、その際に、監視ノードがある集約点に属する監視点をすべて監視できるようなロープ上の領域は、その集約点から一番遠い監視点に依存する。例えば、図3のように、一番遠い監視点までの距離が r_d であり、監視点と集約点を結んだ先にロープが張られているとき、そこを移動する監視ノードはその集約点上でのみ、全ての監視点を監視できることになり、移動することができず、撮影領域の減少に柔軟に対応できない問題が発生する。この問題を解決するために、監視点集約フェーズで監視点を集約する際には、監視ノードの撮影可能距離 r_d の範囲にある監視点を集約するのではなく、 r_d より小さい範囲内で集約することにする。ここで、集約距離係数 $q (0 \leq q \leq 1)$ を導入する。集約距離係数 q を導入した際の一番遠い監視点と集約点との位置関係を図4に示す。集約する際に、 qr_d の範囲内の監視点を集約することにより、ロープ上を $(1-q)r_d$ の距離まで移動できることが保証される。この集約距離係数はシステム

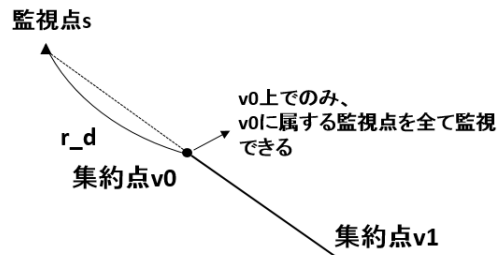


図 3 集約距離係数を設定しない場合の一番遠い監視点と集約点

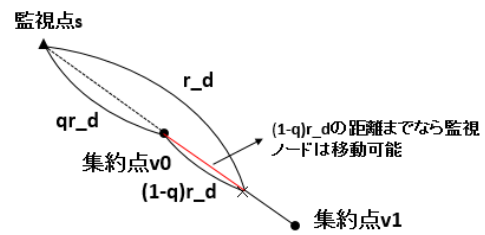


図 4 集約距離係数を設定する場合の一番遠い監視点と集約点の関係

導入時に決めるものとする。

2.5 問題の入出力, 条件, および目的

本問題の入力として監視点の集合 S を設定する。出力は無向グラフ $G = (V, E)$ である。ロープを枝と枝との間に設置する場合、長すぎると監視ノードの重さによりロープがたるんでしまい、撮影領域を一定に保つことが困難であるため、枝と枝との間に設置できるロープの長さには上限があるものとする。そのため、ロープの両端となる2つの集約点の距離はロープの長さの上限より小さくなくてはならない制約がある。ロープの長さの上限を l_{max} で表現する。

ある集約点 $v \in V$ の位置から監視できる監視点の集合を $A(v)$ とし、 A を任意の v に対する $A(v)$ の集合とすると、各監視点はいずれかの集約点から監視可能な状態でなければならぬので、すべての監視点 $s \in S$ に対して $S = \cup A(v)$ にならなければならない。また、ある集約点において複数本のロープが設置される場合、それらのロープは同じ方向ではなく、異なる方向に設置されることが望ましい。異なる方向にロープを設置することで、その集約点に属する監視点を異なる方向から撮影できることが可能になるため、障害物があり撮影領域が減少しても、もう一方のロープ上の監視ノードから撮影することで、満足な撮影領域を保持できる。そこで、同じ集約点に結ばれるロープの角度は $\frac{\pi}{2}$ より大きくするという制約条件を設ける。

1つ以上の集約点に属する監視点を集約済監視点、そうでない監視点を未集約監視点と呼ぶ。集約済監視点は A 、未集約監視点は $S - A$ で表すことができる。本問題の目

的は設定した監視点を全て集約し、その上で設置するロープの本数を減らし、必要な監視ノードを少なくすることである。

3. 全監視領域をカバーするロープの設置手法

本章では、ロープウェイ式害獣検知システムを導入する際の効率的なロープの設置手法について述べる。提案手法では、クラスタリング方式で複数の監視点を集約点にまとめることにより、設置するロープの本数を減らし、必要な監視ノードの数を最小にする。集約点は、その集約点に属する監視点の代表点であり、集約点の位置は属する監視点の座標の平均座標とする。集約点を決定する際に、全監視点を集約できるように集約点を全て決定してから、その後集約点を結ぶようにロープを設置すると、ロープの長さの上限 l_{max} を超えてしまう可能性がある。そこで、提案手法では、初期に一つの集約点を作成し、作成済の集約点を始端とするロープの終端の集約点(次集約点と呼ぶ)を次々と作成していく方式をとる。最もシンプルな集約点を求めてから、順次更新していくため、集約点の集合 V は最初は $V = \{\emptyset\}$ であり、その後1つずつ追加されていく。追加する際に固有の ID を割り振る。一番最初に追加される集約点を初期集約点と呼ぶ。

監視点を集約点にまとめる処理を監視点集約フェーズ、次集約点を求めることを次集約点決定フェーズと呼び、これらのフェーズを交互に行う。次集約点決定フェーズで新たに集約点が追加されない場合に未集約監視点除去フェーズに移行する。未集約監視点除去フェーズでは、監視点集約フェーズにおいて、どの集約点にも属さない監視点を集約する。このフェーズでは、未集約監視点の各監視点の座標を持つ集約点を集約点集団 V に追加し、それぞれに対して監視点集約フェーズを実行する。追加された集約点のロープの結び先となる集約点を決定する際は、次集約点決定フェーズを実行するのではなく、集約点集団 V の中から、一番近い集約点を選択する。もし、選択された集約点までの距離がロープの上限を超えてしまう場合は、中間に集約点を設定する。未集約監視点除去フェーズを終了すると、提案手法を終了する。

提案手法では、最初の集約点、つまり初期集約点はあらかじめ設定された監視点の中から適当に1つ選択し、その座標で初期化しないとイケない。初期集約点からロープを伸ばしていくので、システム全体の集約点の数、ロープの設置位置は最初に選択監視点によって影響を受ける。そのため、最適なロープの設置位置を求めるには、最初に選択する監視点を1つずつ変更し、全探索する必要が出てくる。初期集約点を変更しながら、監視点集約フェーズ、次集約点決定フェーズ、未監視点除去フェーズを行い、そのうち、設置されるロープの数、つまり必要な監視ノードの数が少ないときの無向グラフ $G = (V, E)$ を提案手法の解とする。

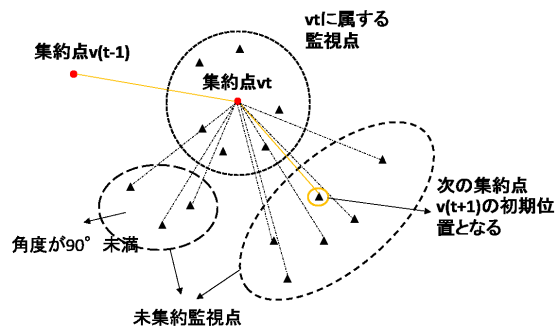


図5 ロープの角度を考慮した次集約点決定フェーズ

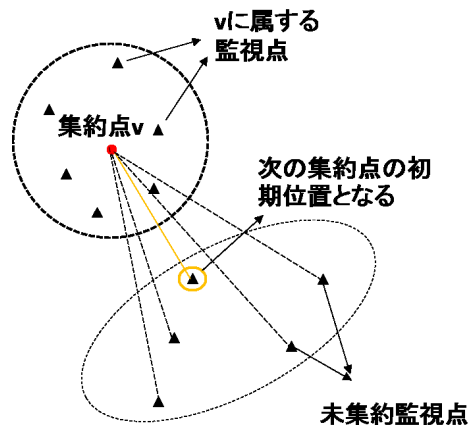


図6 距離のみを考慮した次集約点決定フェーズ

3.1 監視点集約アルゴリズム

ここでは、監視点集約フェーズにおいて、監視点を集約する際のアルゴリズムについて述べる。監視点集約フェーズでは、集約点と監視点との距離が qr_d 以下の監視点をその集約点に属する監視点とし、集約点の位置を属する監視点の平均座標に設定する。集約点に属する監視点が収束するまで集約点の位置を更新し、収束した後、次集約点決定フェーズに移行する。

集約点 v_t が周辺の監視点を集約するアルゴリズムは以下の通りになる。

- (1) 集約点 v_t から距離 qr_d 以内になる監視点をその集約点に属する監視点とし、 $A(v_t)$ を更新する
- (2) $A(v_t)$ に属する監視点の平均位置を計算する
- (3) 集約点 v_t の位置 ($v_t.pos$) をステップ2で求めた平均位置に更新する
- (4) $A(v_t)$ に属する監視点が収束するまでステップ2,3を繰り返す
- (5) 次集約点設定フェーズに移る

3.2 次集約点設定アルゴリズム

ここでは、次集約点設定フェーズにおいて、次集約点を決定する際のアルゴリズムについて述べる。監視点集約フェーズで集約点の座標を決定した後は、ロープを結ぶために結び先となる次の集約点を適切に決定しないとイケない

い。次集約点決定フェーズでは、ロープの設置位置となる次集約点を決定する。次集約点は、未集約監視点から選ばれ、その未集約監視点の座標を次集約点の初期座標とする。ロープの設置角度の制約条件を満たすため、次集約点を決定するさいは、集約点に結ばれるロープと、未集約監視点との角度が $\frac{\pi}{2}$ 以上となる未集約監視点のうち、集約点との距離が一番小さい監視点の座標を次集約点の初期座標とし、集約点集団 V に追加する。もし、角度の制約条件を満たす未集約監視点と、集約点との距離がロープの上限を超えてしまう場合は、ロープの上限以下の内、一番近い未集約監視点の座標を次集約点の初期座標とし、集約点集団 V に追加する。また、ロープの上限以下の距離を持つ未集約監視点が存在しない場合は次集約点を決定しないまま次集約点決定フェーズを終了する。

次集約点を決定する様子を図5に示す。集約点 v_t の次集約点 v_{t+1} を決定する例を図5に示す。集約点 v_t はロープの結び先として、集約点 v_{t-1} が存在しているとする。対象とする未集約監視点を s_{t+1} とすると、次集約点 v_{t+1} を決定する際は、 $\frac{\pi}{2} \leq \angle v_{t-1}v_t s_{t+1}$ を満たし、かつ集約点 v_t との距離 $v_t s_{t+1}$ がロープの上限以内で一番小さい監視点 s_{t+1} の位置を次集約点 v_{t+1} の初期位置として設定し、集約点集合 V に追加する。

集約点の数が1つしかない場合、まだロープが設置されていない状態なので、図6のように、次集約点は、未集約監視点の中から一番近い監視点の座標を初期位置とする集約点を設定し、集約点集団 V に追加する。もし、一番近い監視点までの距離がロープの長さの上限を越す場合は、中間地点に集約点を設定する。その後、図5で示した通り、角度を考慮して別の集約点を決定する。つまり、集約点が1つしか存在しない場合は、次集約点決定フェーズで、2つの集約点が追加される。

集約点 v_t の次集約点 v_{t+1} を設定するアルゴリズムは以下の通りとなる。集約点 v_t には、集約点 v_{t-1} からロープが張られていることとし、ロープの上限は l_{max} である。また、未集約監視点 $\{S - \{A\}\}$ のうち対象とする監視点を s_{t+1} で表現する。

- (1) 集約点 v_t が初期集約点ならステップ2へ、そうでないならステップ4へ
- (2) 集約点 v_t の位置 $(v_t.pos)$ と未集約監視点 $\{S - \{A\}\}$ に属する全ての監視点との距離 $v_t s_{t+1}$ を計算し、 $v_t s_{t+1} \leq ropeMax$ を満たす s_{t+1} のうち、 $v_t s_{t+1}$ の値が一番小さい監視点 s_{t+1} を求める
- (3) 集約点 v_{t+1} の位置 $(v_{t+1}.pos)$ を $(s_{t+1}.pos)$ で初期化し、集約点集合 V に追加するまた、 $E(v_t), E(v_{t+1})$ にそれぞれ、 v_{t+1}, v_t を追加する
- (4) 集約点 v_t の位置 $(v_t.pos)$ と未集約監視点 $\{S - \{A\}\}$ の全ての監視点との距離 $v_t s_{t+1}$ を計算し、 $\frac{\pi}{2} \leq \angle v_{t-1}v_t s_{t+1}$ を満たし、かつ $v_t s_{t+1} \leq rope_{max}$ を満た

す監視点 s_{t+1} のうち、 $v_t s_{t+1}$ の値が一番小さい監視点 s_{t+1} を決定する

- (5) 集約点 v_{t+1} の位置 $(v_{t+1}.pos)$ をステップ4で求めた監視点の座標 $(s_{t+1}.pos)$ で初期化し、集約点集合 V に追加する また、 $E(v_t), E(v_{t+1})$ にそれぞれ、 v_{t+1}, v_t を追加する
- (6) ステップ5で該当する監視点 s_{t+1} が得られない場合は、ステップ2に行き、距離のみで次集約点 v_{t+1} を決定する

3.3 未集約監視点除去フェーズ

未集約監視点除去フェーズでは、監視点集約フェーズにおいて、どの集約点にも属さなかった監視点を集約していく。アルゴリズムは以下の通りとなる。ロープの上限は l_{max} である。

- (1) 未集約監視点 $\{S - \{A\}\}$ から1つ取り出し、監視点 s_t とする。もし、 $\{S - \{A\}\} = \{\emptyset\}$ なら未集約監視点除去フェーズを終了する
- (2) 集約点 v_t の位置 $(v_t.pos)$ を $(s_t.pos)$ で初期化し、集約点集合 V に追加する
- (3) 集約点 v_t に対して、監視点集約フェーズを行う
- (4) 集約点 v_t から一番近い集約点 v_{t+1} を計算する
- (5) 距離 $v_t v_{t+1} \leq ropeMax$ を満たすなら $E(v_t), E(v_{t+1})$ にそれぞれ v_{t+1}, v_t を追加する満たさないなら v_t と v_{t+1} の間に集約点 v_{t+2} を追加し、 $(v_{t+2}.pos)$ を $\frac{(v_t.pos + v_{t+1}.pos)}{2}$ で初期化し、集約点集団 V に追加する $E(v_t), E(v_{t+1})$ にそれぞれ、 v_{t+2}, v_{t+2} を追加し、 $E(v_{t+2})$ に v_t, v_{t+1} を追加する
- (6) ステップ1に戻る

4. シミュレーション実験

提案手法によるロープの設置方法の有効性を評価するために、シミュレーションを作成し、シミュレーション実験を行った。畑と面している山地を想定しフィールドの大きさを決定し、フィールドに監視点を設定したマップを構築し、そのマップに対して、集約点を求めることにより、ロープの設置位置を決定する。本実験では、提案手法との比較のために、k-means法により最初に、集約点を全て決定した後、各集約点に対し、ロープを2本設置する方法(以下、k-means法という)を提案手法と同じ条件下で実行し、両者に必要な集約点数、ロープの本数を比較することで、提案手法の有効性を評価した。

4.1 シミュレーション設定

実験では、 $40[m] \times 32[m] \times 10.5[m]$ のフィールドを想定し、98個の監視点を2通りの方法で配置したマップ1,2を用意した。マップ1では監視点を規則正しく設置し、隣の監視点との間隔は約4[m]としている。マップ2では獣道な

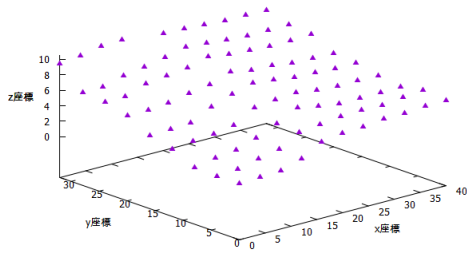


図 7 マップ 1:規則正しく設置した監視点

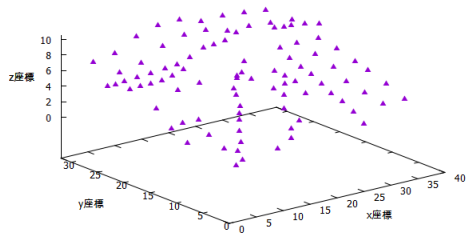


図 8 マップ 2:獣道を想定し設置した監視点

どの害獣が現れやすい領域を想定し、ある範囲では、監視点を密に設置し、その他の部分では疎に設置した。密な範囲の監視点は約 2[m] 間隔で並んでおり、疎の方では約 5[m] 間隔で並んでいる。マップ 1,2 で配置した監視点をそれぞれ図 7, 図 8 に示す。マップ 1,2 それぞれに対し、集約距離係数 q の値を $q = 1, \frac{3}{4}, \frac{1}{2}, \frac{1}{4}$ と変更し、提案手法, k-means 法を適用し、ロープの設置位置を求めた。k-means 法の初期クラスタ数 k は同条件下での、提案手法による最適解となる集約点数とする。シミュレーション実験において、設定したパラメータを表 1 にまとめる。

表 1 シミュレーション実験のパラメータ

フィールドサイズ	40[m] × 32[m] × 10.5[m]
監視点の数	98
監視カメラの撮影可能距離	10[m]
ロープの限界距離	10[m]
集約距離係数 q	1.0, 0.75, 0.5, 0.25
k-means 法の初期クラスタ数 k	同条件下での提案手法の集約点数

表 2 マップ 1 での提案手法, k-means 法別集約点数, ロープの本数

提案手法	集約距離係数 q			
	1	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
集約点数	16	18	44	-
ロープの本数	15	17	43	-
k-means 法				
集約点数	18	22	44	-
ロープの本数	21	27	51	-

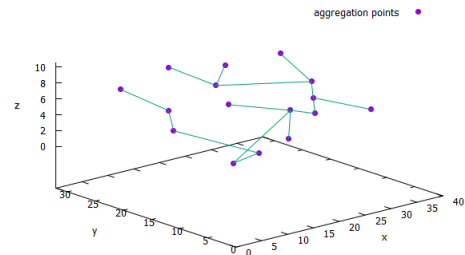


図 9 マップ 1, 提案手法, $q=1$ での出力結果

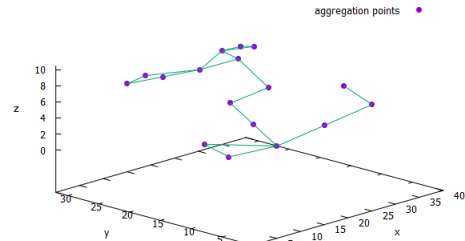


図 10 マップ 1, kmeans 法, $q=1$ での出力結果

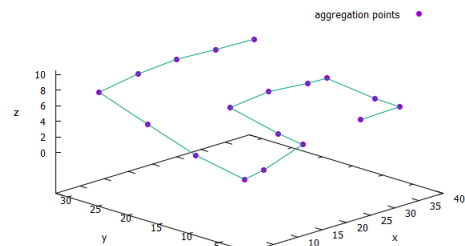


図 11 マップ 1, 提案手法, $q=3/4$ での出力結果

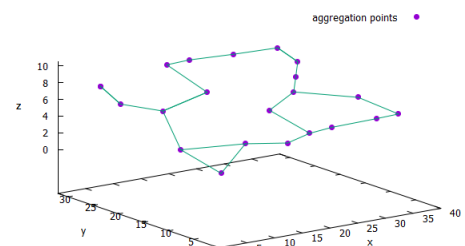


図 12 マップ 1, kmeans 法, $q=3/4$ での出力結果

4.2 実験結果

4.2.1 マップ 1:規則正しく監視点を設置した環境での結果

マップ 1 に対して、集約距離係数 q を変化させたときの提案手法, k-means 法の出力結果を図 9 ~ 図 14 に示す。出力結果は、集約点と、設置されるロープを表す。また、マップ 1 では、集約距離係数 q の値を $\frac{1}{4}$ に設定すると、集約距離に対して、隣接する監視点までの距離の方が大きくなるため、設定した監視点そのまま集約点となった。表 2 は

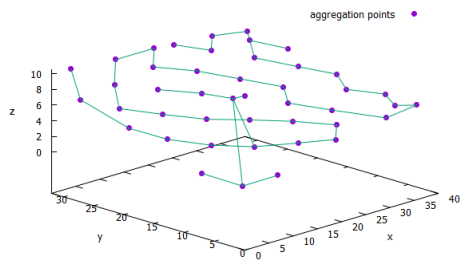


図 13 マップ 1, 提案手法, $q=1/2$ での出力結果

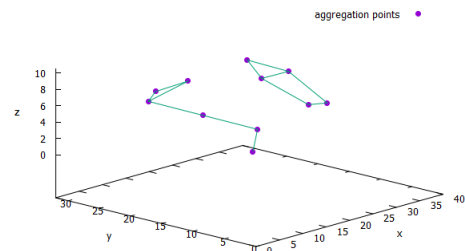


図 16 マップ 2, kmeans 法, $q=1$ での出力結果

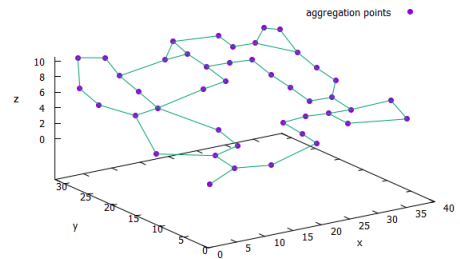


図 14 マップ 1, kmeans 法, $q=1/2$ での出力結果

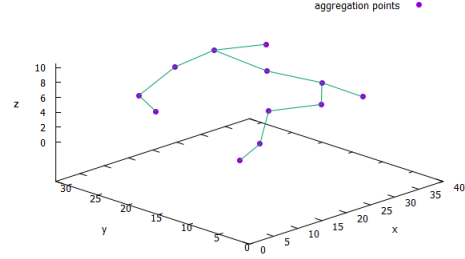


図 17 マップ 2, 提案手法, $q=3/4$ での出力結果

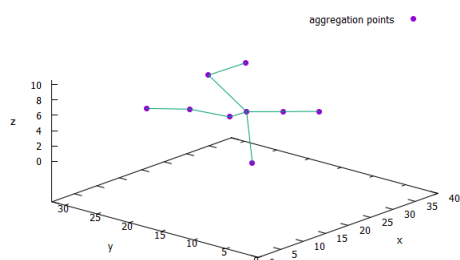


図 15 マップ 2, 提案手法, $q=1$ での出力結果

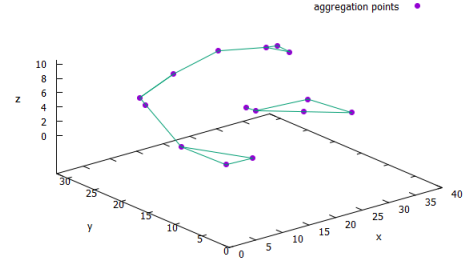


図 18 マップ 2, kmeans 法, $q=3/4$ での出力結果

マップ 1 に対して集約距離係数 q での提案手法, k-means 法を適用した結果, 必要となる集約点の数とロープの本数である。ロープの本数は, 必要となる監視ノードの数と同じである。必要となる監視ノードは提案手法の方が少なくなった。また, 集約距離係数 q の値を 1 から $\frac{3}{4}$ にしても, 必要となる集約点数, ロープの本数共にあまり変化が見られなかった。しかし, 集約距離係数 q の値を $\frac{1}{2}$ にした場合は, 必要となる集約点数, ロープの本数共に大幅に増加した。

4.2.2 マップ 2: 獣道を想定した環境での実験結果

マップ 2 に対して, 集約距離係数 q を変化させたときの提案手法, k-means 法の出力結果を図 15 ~ 図 22 に示す。出力結果は, 集約点と, 設置されるロープを表す。マップ 2 では, 集約距離係数 q の値を $\frac{1}{4}$ に設定しても, 監視点が密に存在する領域は, 集約距離に対して, 隣接する監視点までの距離の方が小さくなり, 監視点を集約できる結果となった。表 3 はマップ 2 に対して集約距離係数 q での提案

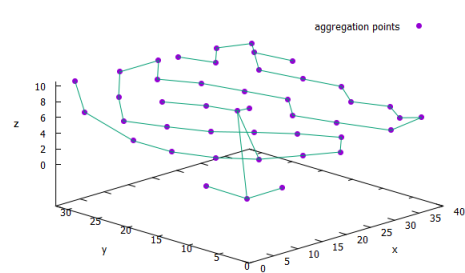


図 19 マップ 2, 提案手法, $q=1/2$ での出力結果

表 3 マップ 2 での提案手法, k-means 法別集約点数, ロープの本数

提案手法	集約距離係数 q			
	1	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
集約点数	9	12	24	63
ロープの本数	8	11	23	62
k-means 法				
集約点数	11	15	27	72
ロープの本数	13	17	36	90

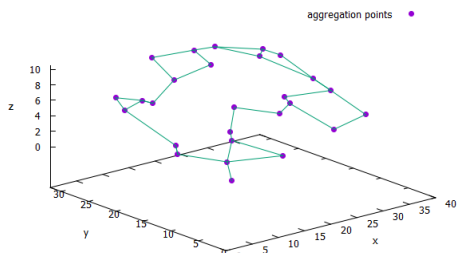


図 20 マップ 2, kmeans 法, $q=1/2$ での出力結果

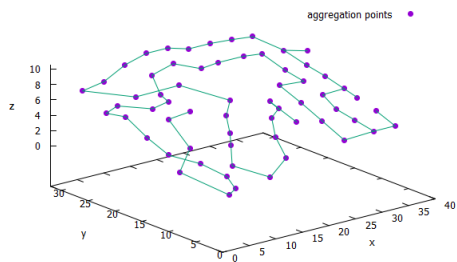


図 21 マップ 2, 提案手法, $q=1/4$ での出力結果

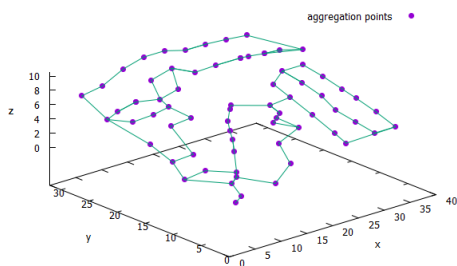


図 22 マップ 2, kmeans 法, $q=1/4$ での出力結果

手法, k-means 法を適用した結果, 必要となる集約点の数とロープの本数である. ロープの本数は, 必要となる監視ノードの数と同じである. マップ 1 のとき同様, 必要となる集約点数, ロープの本数は提案手法の方が少なくなった. 監視点が密な領域があるため, 必要となる集約点数, ロープの本数は, 同条件でのマップ 1 と比べ少なくなった. やはり, 集約距離係数 q の値を 1 から $\frac{3}{4}$ にしても, 必要となる集約点数, ロープの本数共にあまり変化がなく, 集約距離係数 q の値を $\frac{1}{2}$ にすると, 必要となる集約点数, ロープの本数共に大幅に増加している.

4.3 実験結果の評価

今回想定した 2 つのマップ, 監視点を規則正しく配置する場合と, ある領域には密に配置する場合の両方ともに, 提案手法を適用することで, k-means 法と比べると, 監視ノードの数を少なくすることが可能であった. ロープの長さには上限があるため, k-means 法で全集約点を決定してからロープの設置位置を決定すると, 距離が大きくなってし

まい, ロープの上限を越すため, 中間に集約点を設置しなければいけなく, ロープの本数が増えることが原因だと考えられる. 提案手法では, 集約点を決定した後に, ロープの長さを考慮し, 次の集約点を決定するため, ロープウェイ式害獣検知システムでは, 有効的であるといえる. また, 集約距離係数 q を 1 から $\frac{3}{4}$ に変えても, 必要となる集約点数, ロープの本数にあまり変化が見られなく, 一方で, ロープ上を移動可能な距離は q の値を 1 に設定した際は, $1[m]$ も移動できない監視ノードがあるが, q の値を $\frac{3}{4}$ に設定した場合は, 最も移動距離が小さい監視ノードでも, $2.5[m]$ は保証されている. 今回のマップでは, 集約距離係数は $\frac{3}{4}$ が一番適していた.

5. まとめ

本稿では, ロープウェイ式害獣検知システム導入時において, 複数の監視点をまとめて監視することができる集約点を求めることにより, 必要となる監視ノードの数を減らす手法を提案した. シミュレーション実験を行い, k-means 法と比較することで, 提案手法の有効性を検証した. その結果, k-means 法を用いて全集約点を決定してから, ロープの設置位置を決定するよりも, 提案手法のように, 集約点を決める処理と, ロープの設置位置を決める処理を交互に行う方が, 必要となる監視ノードの数を少なくすることができることを確認した. 今後は, ロープウェイ式害獣検知システムにおいて, 他の監視ノードと協調移動することで, システム全体の撮影領域を最大化する手法を開発したい.

参考文献

- [1] 農林水産省 “鳥獣被害の現状と対策”: 入手先 <http://www.maff.go.jp/j/seisan/tyozyu/higai/attach/pdf/index-15.pdf>
- [2] 堀内 史郎, 今井 乃里子, 高柳 和宜, 今井 康仁: 野生ニホンザルが農山村住民に及ぼす生活被害の指標化: サルの出現率, 畑の被害レベル, 作物の総収穫件数の分析, ” *J. of Naturalistae*, No.3, pp.9–18, (2009).
- [3] 武山 絵美, 九鬼 康彰, 東口 阿希子, 奥村 啓史: 中山間水田農業地域における農地周辺バッファゾーンの空間特性と害獣対策, ” *J of Rural Planning Association* 30, pp.405–410, (2011).
- [4] アイニックス株式会社 “害獣捕獲監視システム わなタグ”: 入手先 http://www.ainix.co.jp/products/biometrics_service/monitoring/
- [5] 小川勇治: “ロボット監視カメラによる獣害被害防止の対策技術開発”, 第 19 回生物学技術研究会・第 30 回生理学技術研究会予稿集, pp.39
- [6] J.MacQueen: Some methods for classification and analysis of multivariate observations, ” *Proc. Fifth Berkeley Symp. on Math. Statist. and Prob.*, Vol. 1, pp.281–297, (1967).