

携帯端末を利用した無線 LAN システムの 運用管理支援手法の開発と実装

小川康一^{1,a)} 吉浦紀晃²

概要：通常、集中管理型の無線 LAN システムは、専用の管理システムで運用する。しかし、無線 LAN アクセスポイントと接続する有線ネットワークの障害や、利用者からアクセスポイントの接続不具合についての苦情など、管理者が意図しない状況が考えられる。この場合、管理者は各アクセスポイントの設置箇所へ赴いて動作状態を確認する必要がある。この運用負担を軽減する解決策として、本稿では、人が携帯端末を持って無線 LAN アクセスポイントの設置箇所を巡回し、情報収集する方法を提案する。具体的には、携帯端末のカメラアプリケーションで無線 LAN アクセスポイントを撮影することによって、アクセスポイントの LED インジケータから状態を把握する。撮影した動画を画像として分割し、深層学習の識別器で識別させる。本手法を Android スマートフォンに実装し、大学内での実験により提案手法の有効性を確認した。実験の結果、以前の移動ロボットで巡回してアクセスポイントの LED インジケータを撮影する方法に比べて識別精度が向上した。

キーワード：無線 LAN アクセスポイント障害、携帯端末、ネットワーク監視、電波状態監視、深層学習

A Method of Management of Wireless LAN System using Mobile Devices

KOHICHI OGAWA^{1,a)} NORIAKI YOSHIURA²

1. はじめに

無線 LAN は、スマートフォンやタブレットの普及に伴い、社会において広く利用されている。大学においても、教育研究を行う様々な場面での活用が進んでいる。多くの大学では、情報センターが無線 LAN システムの整備を行っている。

埼玉大学（以下、本学）では、学内に 300 箇所の無線 LAN アクセスポイント（以下、アクセスポイント）を設置している。運用は本学の情報センターが行っている。常時約 20 箇所のアクセスポイントが何らかの原因により停止し、障害となっている。主な障害の原因は、メディアコンバータの電源の抜け、パワーインジェクタの電源の抜け、アクセスポイント自体の故障である。障害の発生源のほとんどは、図 1 に示す電源を利用者が故意に電源を抜いていることによる[1]。



図 1 アクセスポイントの電源や周辺機器

本学ではこの対策として、耐震改修を実施した建物では、アクセスポイントの設置箇所を利用者が届かない上部へ移設する対策をとっており、一定の効果をあげている（図 2）。しかし、このような建物は本学の構内全体の一部であり、まだほとんどのアクセスポイント設置箇所では利用者の手の届く位置に電源コンセントやメディアコンバータなどの機器があり、たびたびネットワーク障害が発生している。

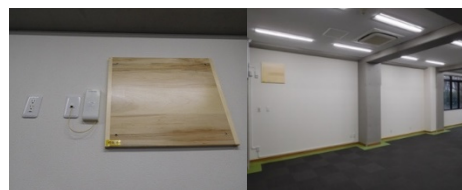


図 2 アクセスポイントを高所に設置する例

大学や企業では、管理対象のアクセスポイントが多数となるため、無線 LAN スイッチを利用した集中管理方式が採用されている。運用管理は、専用の管理システムで行われている。たとえば、Cisco 社のシステムでは WLC[2]が、HP 社のシステムでは AirWave[3]が管理システムとして用意されている。管理システムでは、各アクセスポイントの状態や接続端末数、電波状態などを把握可能である。しかし、この管理システムを利用しても、前述のような障害の詳細な状況や原因を把握することはできない。本学では、以下

1 埼玉大学情報メディア基盤センター
Information Technology Center, Saitama University

2 埼玉大学大学院理工学研究科
Graduate School of Science and Engineering, Saitama University
a) kogawa@mail.saitama-u.ac.jp

のような課題がある。

- ①管理システム上ではアクセスポイントの稼動状態は正常と表示されているが、実際のアクセスポイントの動作に異常がある。
- ②管理システムの不具合によりアクセスポイントの状態を確認できず、不具合の原因が特定できない。
- ③利用者から無線接続に不具合があると苦情を受けるが、管理システム上の表示では異常がなく、実際のアクセスポイントの状態が明らかでないため信頼できない。
- ④障害の根本原因は、現地に赴いて調査するまで明らかでなく、都度管理者が各アクセスポイントの設置箇所に赴いて動作状態の確認が必要となっている。

本学の情報センターでは、管理者が窓口で利用者の対応も行っており、情報センターの事務室に常駐しなければならない。管理者の数も少なく、アクセスポイントの障害を巡回して解決することが大きな負担となっている現状にある。

この課題に対し、我々は、監視装置と移動ロボットに搭載されたカメラを利用することにより、アクセスポイントの状態を LED インジケータ（以下、LED）の表示を読み取り識別する方法[4]を考案、監視業務を自動化する方法を提案している。しかし、この方法では、アクセスポイントの状態を識別するために良好な画像を取得する必要があるが、移動ロボットで巡回しながら撮影位置を決定することは困難であった。カメラで取得した画像が小さいために少しの位置のずれで識別がうまくいかないこともあった。

そこで、本稿では、人が携帯端末を持ち、アクセスポイントの設置箇所を巡回することによって、アクセスポイントの情報を収集する手法を提案する。具体的には、携帯端末のアプリケーションを利用し、アクセスポイントの設置箇所を巡回し、カメラでアクセスポイントの LED の撮影を行うことにより、アクセスポイントの正常性や電波状態を把握可能とする。本手法により、IT の専門知識がない人でも巡回してアクセスポイントの状態把握が可能となる。本手法を監視システムに実装し、大学内での実験により提案手法の有効性を確認した。

本稿の構成は次のとおりである。第 2 章では、本稿の前提となる我々の研究について述べる。第 3 章では、提案手法について述べる。第 4 章では、提案手法に基づいた監視システムの実装方法について述べる。第 5 章では実装した監視システムを用い、実験とその考察を行う。第 6 章で課題について述べ、第 7 章で関連研究について述べる。第 8 章で本稿をまとめる。

2. 研究背景

本章では本稿の背景として、これまでに我々が提案している提案手法、監視装置によるネットワーク機器の監視と移動ロボットによる情報収集について述べる。

我々は、一般の利用者のネットワーク機器が SNMP[5]などの管理プロトコルをサポートしていないために、ネットワーク障害を監視できない問題に対して、ネットワーク機器の LED の表示に着目し、画像処理により表示の変化を分析する方法を提案している。これを Web カメラと小型コンピュータを利用して監視装置として実装し（図 3 左）、移動体通信回線により監視情報を送信する方法を提案している[6]。この方法では、監視装置を複数展開する場合、移動体通信回線を複数用意する必要がある。そこで、監視装置と移動ロボット間の通信は無線 LAN 通信で行い、移動ロボット（図 3 右）により情報収集する方法を提案している[7]。



図 3 監視装置(左)と移動ロボット（右）の例

3. 提案手法

本章では、携帯端末を利用したアクセスポイントの監視方法について述べる。

3.1 携帯端末を利用したアクセスポイントの監視

以前の研究[6]では、監視対象として、これまでにメディアコンバータとイーサネットスイッチを監視している。メディアコンバータやイーサネットスイッチなどのネットワーク機器と同様に、アクセスポイントは、多くの場合管理者に状態を知らせる LED を備えており（図 4）、この表示パターンを分析することで状態を把握可能である。メーカーにより表示方法や表す意味は異なるが、LED の色を変化させることによって状態を示している[8,9]。よって LED の色や表示パターンを分析し、識別可能であれば、アクセスポイントの状態を把握可能である。

我々は、監視装置と移動ロボットに搭載されたカメラを利用することにより、アクセスポイントの状態を LED から判別する方法[4]を提案し、監視業務を自動化する方法を提案している。しかし、この方法では、判別のために最適な撮影位置を探索することや、画像内の LED 部分が占める割合が小さいために少しの位置のずれで識別に失敗するなどの課題があった。

そこで、本稿では、携帯端末を利用して、人が巡回することによりアクセスポイントの情報を収集する手法を提案する。具体的には、IT の専門知識がない人でも携帯端末の

アプリケーションを利用し、アクセスポイントの設置箇所を巡回し、カメラでアクセスポイントのLEDの撮影を行うことにより、アクセスポイントの正常性や電波状態を把握可能とする。本稿では、アクセスポイントの設置箇所を巡回して確認する人を巡回者と呼ぶ。

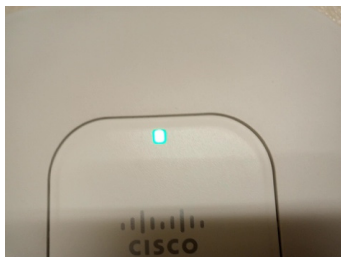


図4 アクセスポイントのLED

3.2 提案手法の動作手順

提案手法の動作手順を図5に示す。なお本稿では、アクセスポイントの大まかな設置箇所については巡回者があらかじめ知っているものとする。動作手順は以下の通りとなる。(以下手順は、図中の番号に一致)

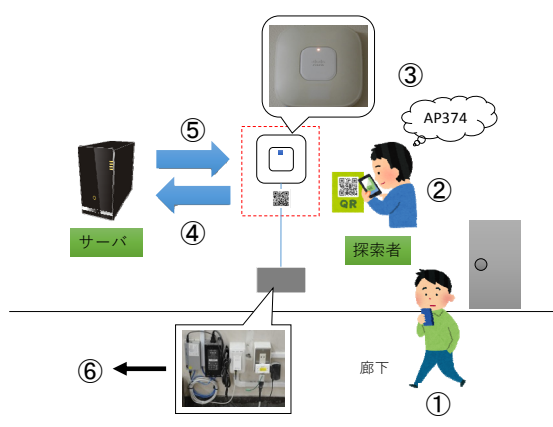


図5 提案手法の動作概要図

- ①巡回者は、携帯端末を持ち、アクセスポイントの設置箇所に徒歩で廊下などを移動する。
- ②アクセスポイントの設置箇所のQRコードを読み取り、アクセスポイントの属性情報を得る。
- ③アクセスポイントのLEDを撮影する。
- ④アクセスポイントの情報とともにLEDの撮影データ、電波状態の情報をサーバへ送信する。
- ⑤サーバでLEDの撮影データをもとにアクセスポイントの状態と障害の状況を判定し、巡回者と管理者に通知する。
- ⑥巡回者は、次のアクセスポイントの設置箇所へ移動する。

3.3 アクセスポイントの監視方法と予備実験

アクセスポイントの動作状況を把握するために、アクセスポイントのLEDの点滅状況を把握する方法を用いる。こ

れまでの我々の提案手法に、改善・改良を加えることで実現する。アクセスポイントは、メーカーごとに異なるが、LEDによって管理者に機器の状態を知らせる。例えば、Cisco社のアクセスポイント1140シリーズでは、ステータスLEDを備えており、状況により以下のように点灯する[8]。

- 正常状態（利用者無）・・・緑（点灯）
- 正常状態（利用者有）・・・青（点灯）
- 異常状態・・・緑、赤、黄の連続点灯

このパターンに着目し、アクセスポイントの状態を把握可能である。そこで、このパターンを取得可能な撮影時間を把握するために予備実験を行った。アクセスポイントに監視装置を取り付け、故意に障害を発生させる。カメラで一定期間LED部分の動画を撮影する。アクセスポイントの状態把握の実験は以下の順で行う。

1. 一定時間毎に動画（画像）を取得する
2. 矩形のうち一番面積が大きな矩形を抽出する
3. 矩形の中心座標情報を得る（x, y）
4. 座標上の画素色を取得する（BGR）
5. BGRのパターンが連続する場合に障害

予備実験の結果を図6に示す。30fpsの状態動画で撮影し、400フレームを収集した。それぞれのフレームに対して前述の動作手順を適用した。予備実験の結果、異常状態では、繰り返される特徴があることがわかった。

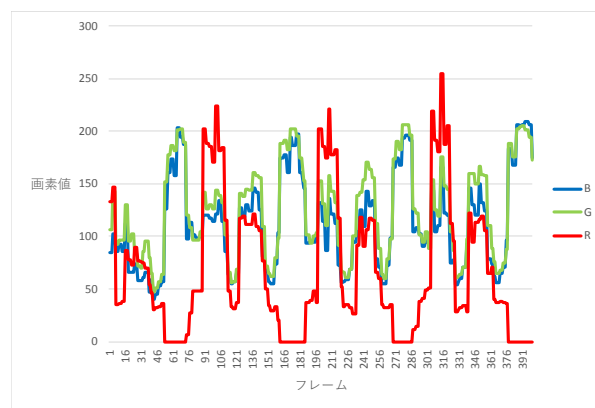


図6 異常状態の画素（BGR）の値

具体的には、50フレームから150フレーム、150フレームから250フレームの2つに着目すると、最初のピークでは、Rの値が0であるときにBGは高い値を示している。2つ目のピークでは、Rだけが高くなる。3つ目のピークではBGRともに2つ目のピークのBGとほぼ同値を示す。この繰り返しである。そしてパターンの繰り返しの間隔は約10秒である。この予備実験の結果から、アクセスポイントの異常状態を把握するために必要な時間間隔を獲得した。

4. 実装

本章では、前章で述べた提案手法に基づき、具体的な実

装方法について述べる。

4.1 携帯端末を利用した監視方法

アクセスポイントは、電波のカバー範囲を広範囲とするため、ある程度の高さに設置されている。本学のアクセスポイントは、床から平均で約 220cm の高さに設置されている。そのため、そのまま携帯端末のカメラで撮影した場合には画像が小さく、LED の状態を詳細に取得できない可能性がある。そこで、スマートフォンの自撮り棒を活用する。

また、巡回者により撮影の個人差をある程度なくするため、アプリケーション内のカメラ機能にガイドを表示する。これにより撮影した動画の品質を確保する。携帯端末は、Android スマートフォンの UMIDIGI A3 Pro Updated Edition を用いる。実現する監視システムは、図 7 に示す。

監視システムは、巡回者が持つ携帯端末の Android スマートフォンと表示端末のタブレット、監視情報を収集する監視情報集約サーバ、後述するアクセスポイントの LED の識別を行う画像処理サーバ、および管理者の表示端末で構成する。Android スマートフォンとタブレットの通信は、ネットワーク障害の可能性も考慮し移動体通信回線を利用する。監視情報は、アクセスポイントの識別番号、MAC アドレス、動画データ、RSSI (Received Signal Strength Indicator) 値である。

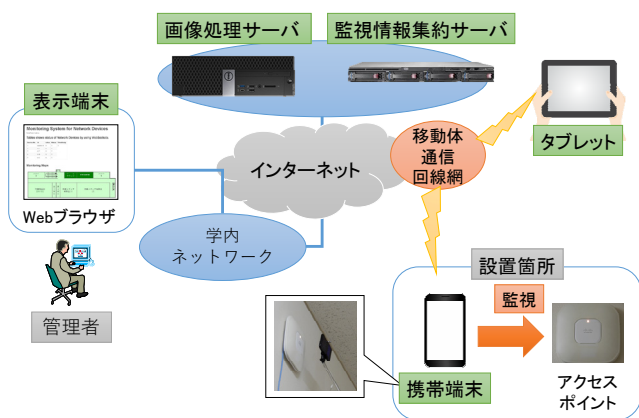


図 7 監視システム構成の概要

Android スマートフォンと監視情報集約サーバ間の通信には IoT 向けの通信プロトコルである MQTT[10]を利用する。MQTT は、出版購読モデルを採用している。Android スマートフォンで取得した監視情報は、Publisher により送信する。監視情報集約サーバをデータ中継の役割を持つ Broker とする。監視情報集約サーバ内で監視情報を収集するプログラムは Subscriber とする。Android スマートフォンの Publisher の実装には paho-mqtt[11]を用いる。監視情報集約サーバの Broker の実装には mosquitto[12]を用いる。管理者の表示端末とタブレットの監視情報の表示は、Web ブラウザを利用する。監視情報を表示する部分の実装は、以前

の研究[6]で開発した WebSocket による管理画面のプログラムを利用した。

4.2 アクセスポイントの状態特定方法

アクセスポイントの状態の特定は、前述の通り LED の色をもとに検出を行えば把握可能である。しかし、その特徴は人が考えて抽出を行う必要がある。そのため、複数箇所にアクセスポイントが設置されており、設置箇所の環境の特徴がそれぞれ異なるような状況では即座に対応が困難である。そこで本稿では、LED の点灯状態の特徴を柔軟に捉えるため、以前の研究[4]と同様に、機械学習を用いる。

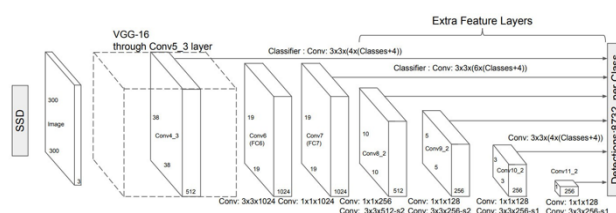


図 8 SSD の CNN 構造[13]

アクセスポイントの状態特定に用いる物体識別は、畳み込みニューラルネットワーク（以下、CNN）を採用した。CNN により、カメラから得られる動画のフレームを直接分析し、アクセスポイントの状態を特定する。本稿における CNN の利用目的は、あくまでもアクセスポイントの状態特定であり、特定物体検出に位置付けられる。よって、一般物体検出のように多数の物体を同時に特定する必要がなく、処理が軽く高い FPS が期待できる Single Shot MultiBox Detector（以下、SSD）[13]を採用した。SSD の CNN 構造を図 8 に示す。SSD の処理には高速な GPU が必要なため、GPU を搭載したコンピュータを用意した。

SSD の実装には、深層学習フレームワークの Tensorflow[14]を用いる。LabelImg[15]で撮影した写真中のアクセスポイントを囲み、annotation を作成する。実装方法は、公式サイト Custom Object Detection Tutorial[16]を参考にした。アクセスポイントの LED の識別は、SSD で学習したモデルを利用し、画像ライブラリの OpenCV[17]で行う。SSD の学習と識別を高速化するため、GPU 搭載の画像処理サーバを用意した。サーバの仕様は以下のとおりである。

- コンピュータ：Dell OptiPlex 7050
- CPU：Intel Core i7-7700 3.60GHz
- メモリ：8GB
- HDD：1TB
- GPU：Nvidia GeForce GTX 1050Ti (4GB)
- OS：Ubuntu 18.04.02 LTS

4.3 Android カメラアプリケーションの開発

携帯端末のアプリケーションの開発は、Android を採用した。Android の SDK バージョンは 27 を用いた。LED の状態把握を行うカメラアプリケーションの開発にあたっては、Android Developers のサイト[18]を参照した。具体的には、同サイト内で紹介されている Camera 2 Video[19]を参考に、改良を加えることで実現した。これに描画ライブラリの Canvas でプレビュー画面に白い四角を表示させ、撮影の目安とした。これは、カメラでの撮影に、人によってアクセスポイントが異なる位置で撮影されることを避けるためである。カメラによる LED の取得例を図 9 に示す。



図 9 LED 状態の取得例

巡回者はカメラアプリケーションのスタートボタンを押下後、10 秒のカウントダウンで動画を取得する。アプリケーションでは、撮影した動画データを含む監視情報は自動的に監視情報集約サーバへ転送され、その後動画データは監視情報集約サーバから画像処理サーバへ転送する。

4.4 アクセスポイントの通信状態の把握方法について

巡回者は、アクセスポイントの通信状態を把握するために、LED を撮影するのと同時に Android スマートフォンで無線 LAN の電波強度である RSSI を収集する。アクセスポイントが発する SSID に基づき RSSI を収集する際、近くに同じ SSID のアクセスポイントが多数設置されている状況ではアクセスポイントの判別ができない。そこで、監視対象のアクセスポイントのもののかどうかの判断は、各アクセスポイントの MAC アドレスをあらかじめ把握しておき、該当するアクセスポイントであるか確認する。

5. 監視システムによる実験

本章では、前章で実装したシステムを用い、実験のために設置したアクセスポイントを利用した実証実験について述べる。

5.1 実験の概要

実験は、本学情報メディア基盤センター2F で実施した。情報収集対象のアクセスポイントは2箇所である。実験環境の概要図を図 10 に示す。実験に使用したアクセスポイントは Cisco 社の Aironet1142 である。実験は、以下の2

つの目的を持つ。

1. アクセスポイントの LED を色別に識別でき、その分析により障害の検知が可能かを確認
2. 移動ロボットで撮影した以前の研究[4]との識別精度の比較

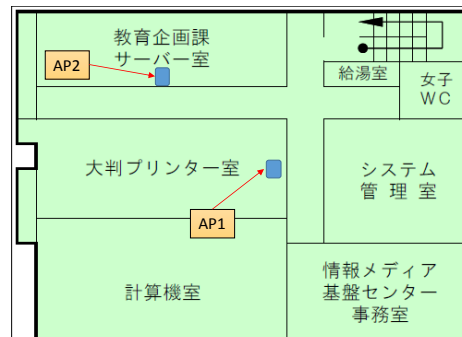


図 10 実験環境の概要図

以上の実験の目的に基づき、以下2つの実験を行う。

1. 2つのアクセスポイントで5回の監視を行い、正しく LED の色が識別できることを確認
2. 2つのアクセスポイントで5回の監視を行い、学習における評価を実施

あらかじめ、異常状態にしたアクセスポイントを巡回し、異常状態の動画を撮影する。これを教師データとする。画像の総数は500枚用意した。これを画像処理サーバで学習し識別器を作成する。学習におけるパラメータとして、バッチサイズはリソースの関係で1とし、ステップ数は20,000ステップとした。学習には約2時間を要した。



図 11 撮影の様子

実験に用いるアクセスポイントの設置高さは、本学の平均値に合わせて約220cmとした。実験は、アクセスポイントを故意に異常状態にする。実験に用いる検証用の動画は、計5回、10秒間の撮影をしたものを利用する。撮影の様子を図 11 に示す。それぞれの動画の大きさは約12MBであった。各動画のフレームレートは14fpsである。動画は、動画を編集・変化するプログラムである FFmpeg[20]を利用し、mp4 ファイルを JPEG 画像に分割する。

5.2 LED の色識別の実験

検証用に撮影した動画の画像を用い、識別器により LED の色を正しく把握できるかを実験により確認した。この実験では、たとえば黒の点灯が連続する場合は、点灯の回数にかかわらず「黒」と考えるものとする。Aironet1142 の場合、障害の状態は、図 12 のように「黒・緑・黒・赤・黒・黄」のパターンで繰り返される。図 13 に例として識別後の画像を示す。

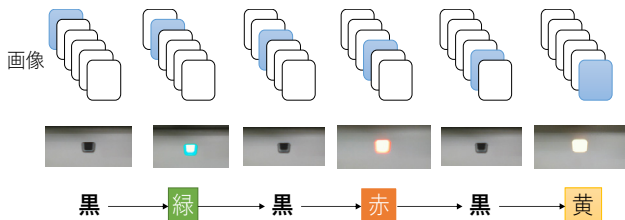


図 12 障害の状態



図 13 アクセスポイントの識別画像
(左から LED は黒、緑、赤、黄)

異常状態の場合の識別結果を表 1 と表 2 に示す。AP1 では、以前の研究[4]と本稿の提案手法ともに識別は成功しており、結果に違いは見られない。しかし AP2 では、例えば 1 回目の識別では「緑・黒・赤・黒・黄・黒」と識別すべきところが、「緑・黄・黒・赤・黒」と誤識別している。このことは、以前の研究[4]では、動画での識別方法を用いていたために識別精度が悪かったことが原因と考えられる。

表 1 異常状態の場合の識別結果比較 (AP1)
(上: 以前の研究[4], 下: 本稿)

1	黒	緑	黒	赤	黒	黄	黒	緑	黒	赤
2	緑	黒	赤	黒	黄	黒	緑	黒	赤	黒
3	赤	黒	黄	黒	緑	黒	赤	黒	黄	黒
4	黒	黄	黒	緑	黒	赤	黒	黄	黒	緑
5	黒	緑	黒	赤	黒	黄	黒	緑	黒	赤

1	赤	黒	黄	黒	緑	黒	赤	黒	黄	黒
2	緑	黒	赤	黒	黄	黒	緑	黒	赤	黒
3	黒	黄	黒	緑	黒	赤	黒	黄	黒	緑
4	黒	赤	黒	黄	黒	緑	黒	赤	黒	黄
5	黄	黒	緑	黒	赤	黒	黄	黒	緑	黒

表 2 異常状態の場合の識別結果比較 (AP2)
(上: 以前の研究[4], 下: 本稿)

1	緑	黄	黒	赤	黒	緑	黒	赤	黒	赤
2	黒	赤	黒	黄	黒	緑	黒	赤	黒	緑
3	赤	黒	黄	黒	緑	黒	黄	黒	緑	黒
4	黄	黒	黄	黒	緑	黒	赤	黒	黄	黒
5	黄	黒	緑	黒	赤	黒	緑	黒	赤	黒

1	黄	黒	緑	黒	赤	黒	黄	黒	緑	黒
2	赤	黒	黄	黒	緑	黒	赤	黒	黄	黒
3	黄	黒	緑	黒	赤	黒	黄	黒	緑	黒
4	赤	黒	黄	黒	緑	黒	赤	黒	黄	黒
5	緑	黒	赤	黒	黄	黒	緑	黒	赤	黒

本稿の提案手法では、画像で分割することで識別精度を向上させることができた。いくつかの誤識別が見られたが、色の組み合わせに影響する誤識別はなかった。

5.3 学習における評価

検証用に撮影した動画の画像を用い、識別器の性能評価を行った。学習における評価方法は、F 値を用いた。本稿で用いている方法は多クラス分類である。よって、性能評価ではクラスごとに正しく分類できるかを確認する。識別できた場合を TP とし、実際のクラスに識別できなかった場合 FN とする。表 3 に混同行列の評価一覧表を示す。

表 3 混同行列の評価一覧表

		実際の分類結果			
		黒	緑	赤	黄
アルゴリズムによる判定	黒	TP	FN	FN	FN
	緑	FN	TP	FN	FN
	赤	FN	FN	TP	FN
	黄	FN	FN	FN	TP

表 3 の混同行列の値を集計した後、Precision, Recall, F 値を求めた。式は、それぞれ、

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F-measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (3)$$

を用いる。

アクセスポイントごとの各 LED の検出結果は表 4 の通り

である。この表では、1に近い値が良い結果を示している。SSD を利用した識別器で識別にかかる時間は、平均で約139秒である。以前の研究[4]と比較して、スマートフォンによりアクセスポイントのLEDを的確に捉えられたことと、識別する対象を画像としたことが識別精度の向上につながったと考えられる。以前の研究[4]では、無点灯の場合に緑と識別することや、緑の場合に無点灯と識別する場合があった、黄と赤は誤識別が多くみられたが、本稿の提案方法では誤識別は少なかった。

本実験の結果から、移動ロボットを用いた巡回監視と比較して、人がスマートフォンでLEDを撮影し、SSDの識別器により監視する手法は、識別精度を向上でき、アクセスポイントの状態を把握可能であることがわかった。

表4 各LEDの検出結果比較（上：AP1，下：AP2）

	Precision		Recall		F-measure	
	研究[7]	本稿	研究[7]	本稿	研究[7]	本稿
黒	0.93	0.98	0.92	0.98	0.93	0.98
緑	0.93	1.00	0.86	0.98	0.89	0.99
赤	0.83	0.80	0.81	0.80	0.88	0.80
黄	0.78	0.97	0.88	0.97	0.83	0.97

	Precision		Recall		F-measure	
	研究[7]	本稿	研究[7]	本稿	研究[7]	本稿
黒	0.97	0.98	0.88	0.98	0.92	0.98
緑	0.89	0.95	0.80	0.95	0.84	0.95
赤	0.93	0.96	0.77	0.96	0.97	0.96
黄	0.71	0.97	0.97	0.97	0.82	0.97

6. 課題

本章では本稿における課題について述べる。

6.1 巡回の頻度とスケジュール

本稿の提案手法では、巡回者がアクセスポイントの設置箇所を巡回する。本稿では、この巡回の頻度とスケジュールについては未検討である。本学の場合は、アクセスポイントが300箇所設置されており、建物や設置箇所同士の距離が離れている。全ての設置箇所を一人の巡回者が一日で巡回することは現実的ではない。よって、複数の巡回者で手分けして巡回を行うことや、ある程度巡回する箇所を絞り込んでおくなどの工夫が必要であると考えられる。

たとえば、頻繁に障害が発生する設置箇所の統計を収集しておき、障害の発生頻度に応じて巡回ルートやスケジュールを計画することが考えられる。今後、監視システムを大学において実運用し、データが蓄積された段階で巡回のための実装を進める予定である。

6.2 LEDの識別方法

本稿では、以前の研究[4]と同様に、アクセスポイントの状態把握を目的として、LEDの色を識別するため、深層学習を利用した。深層学習は、識別対象の特徴を人が抽出する必要がないメリットがある一方で、教師データの準備に時間がかかるなどのデメリットがある。また、学習の方法や識別対象のデータによっては十分な識別が行えない可能性も考えられる。

本稿では、動画を分割して画像を識別対象とすることで識別精度を向上させることができた。しかし、誤識別は依然として発生している。これについては、教師データを拡充することや、ぶれのない撮影方法を検討することで改善が考えられる。ただし、識別器の精度を向上させた場合であっても、環境によっては誤識別が発生する可能性がある。そこで、誤識別があっても色と色の間の関係性を把握できるようなアルゴリズムを検討する必要があると考えている。本稿では、異常状態の検知に重点を置いたが、誤識別を防ぐ観点から正常状態の監視もあわせて行う方法も比較検討したいと考えている。また、深層学習を利用せずに識別する方法の可能性についても検討を進めて行く予定である。

本来、巡回者がITの専門知識がない人であっても、アクセスポイントの点灯パターンが単純な場合は確認や判断が可能である。しかし、点灯パターンが複雑な場合に何が起きているかの判断が難しいことや、巡回時に各アクセスポイントの状態について記録をとる必要があり、監視対象のアクセスポイントの数が多い場合は負担となる。そこで、本稿で提案するようなシステムの支援により巡回者の負担を軽減できると考えている。

7. 関連研究

本章では本稿の関連研究について述べる。画像処理を用いて、サーバやネットワーク機器のLEDをに状況を把握する手法にFalcon Eye[21]がある。当該研究は、データセンターやサーバ室などのラックを固定したカメラで監視する手法である。しかし、当該研究はサーバ室のようなまとまった一箇所で有効であるが、本稿が対象とするような学内に分散したアクセスポイントでは利用できない。

機械学習を用い、ログを分析することでネットワーク障害や機器故障を検知する研究[22]がある。また、無線LANにより位置を特定するために機械学習を利用している研究[23]がある。しかし当該研究は、動画像ではなく、無線LANの電波強度であるRSSIを用いており、アクセスポイントそのものを直接確認する方法ではない。

ネットワーク機器を視覚的に捉える点ではARマーカーを利用した研究[24]があるが、これは管理者にネットワーク機器の状態を教示することが目的であり、本稿の視点とは異なる。

無線LANの監視やネットワークの動作状況の把握につ

いては、北口らの情報収集手法[25]がある。当該手法は、定常的なデータの収集が行える点が優れているが、収集したデータを送信するためのネットワークが必要であり、固定的にその箇所に設置しなければならない点が本稿の手法と異なる。本稿の提案手法と同様の研究に Pazl[26]がある。Pazl は、モバイルクラウドベースの屋内無線 LAN 監視システムで、携帯端末を利用してアクセスポイントの状態を把握し、センサーの活用により移動した位置を地図上に図示可能である。当該研究は、アクセスポイントのカバレッジ範囲を最適化することが目的であり、提案手法は電波状態の調査に留まっており、アクセスポイントの状態を把握する本稿の目的と異なる。

本稿が対象とする、アクセスポイントの表示部の LED から状態を特定し、システムの運用管理に利用する研究は、我々が知る限りいまだ行われていない。

8. おわりに

本稿では、携帯端末を利用して、人が巡回することによりアクセスポイントの情報を収集する手法について述べた。IT の専門知識がない人でも、携帯端末のアプリケーションを利用し、アクセスポイントの設置箇所を巡回し、アクセスポイントの LED の撮影を行うことにより、アクセスポイントの正常性や電波状態を把握可能とする。本提案手法を監視システムに実装し、提案手法の有効性を大学内での実験により確認した。

本稿の提案手法を応用することにより、将来的には提案手法を実装したアプリケーションを利用者に配布し、利用者から運用に必要な情報の提供を受ける展開も可能となる。また、本稿の提案手法は、監視のみにとどまっており、障害対応についても取り組む予定である。今後は、課題解決を進めるとともに、監視システムとして完成度を高め、実運用での検証を継続していきたい。

参考文献

- [1] 小川康一, 吉浦紀晃: 埼玉大学における光ファイバ直収型ネットワークの運用経験について, 情報処理学会研究報告インターネットと運用技術 (IOT), 2015-IOT-30, pp.1-8(2015).
- [2] Cisco Wireless Control System (WLC) データシート (オンライン), 入手先 <https://www.cisco.com/web/JP/product/hs/wireless/wcs/prodlit/wcs41_ds.html> (2019-08-25 参照).
- [3] HP Aruba Networks Aruba AirWave ネットワーク管理(オンライン), 入手先 <<https://www.arubanetworks.com/ja/products/networking/management/airwave/>> (2019-08-25 参照).
- [4] 小川康一, 吉浦紀晃: 移動ロボットと監視装置を利用した無線 LAN システムの運用管理支援手法の提案, 情報処理学会研究報告インターネットと運用技術 (IOT), 2019-IOT-46, pp.1-8(2019).
- [5] Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework, The Internet Society, 2003.
- [6] 小川康一, 吉浦紀晃: 小型コンピュータと画像処理技術を活用したネットワーク機器監視手法の提案と実装, 情報処理学会論文誌, Vol.59, No.3, pp.1026-1037(2018).
- [7] 小川 康一, 吉浦 紀晃, “移動ロボットと小型コンピュータを活用したネットワーク機器監視手法”, 情報処理学会論文誌 60(2), pp.668-679(2019).
- [8] Cisco Aironet 1140 シリーズ Autonomous アクセス ポイント (オンライン), 入手先 <https://www.cisco.com/c/ja_jp/td/docs/nma/works/works4windows/gs/001/ap1140aut-getstart.html#60734> (2019-09-15 参照).
- [9] Aruba 300 シリーズワイヤレスアクセスポイント設置ガイド (オンライン), 入手先 <https://www.arubanetworks.com/assets/_ja/support/ig/IG_AP-300Series.pdf> (2019-09-15 参照).
- [10] MQTT(online), available from <<http://mqtt.org/>> (accessed 2019-09-06).
- [11] Paho-mqtt(online), available from <<https://pypi.python.org/pypi/paho-mqtt>> (accessed 2019-09-06).
- [12] Mosquitto(online), available from <<https://mosquitto.org/>> (accessed 2019-09-06).
- [13] Wei Liu, Dragomir Anguelov, Dumitru. Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg, “SSD:Single Shot MultiBox Detector,”, European Conference on Computer Vision, ECCV2016, pp.21-37(2016).
- [14] TensorFlow (オンライン), 入手先 <<https://www.tensorflow.org/>> (2019-08-25 参照).
- [15] LabelImg (オンライン), 入手先 <<https://github.com/tzutalin/labelImg>> (2019-08-25 参照).
- [16] TensorFlow Custom Object Detection Tutorial (オンライン), 入手先 <<https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/index.html>> (2019-08-25 参照).
- [17] OpenCV (オンライン), 入手先 <<https://opencv.org/>> (2019-08-25 参照).
- [18] Android Developers Camera API (オンライン), 入手先 <<https://developer.android.com/guide/topics/media/camera>> (2019-05-21 参照).
- [19] android-Camera2Video (オンライン), 入手先 <<https://github.com/googlesamples/android-Camera2Video>> (2019-09-15 参照).
- [20] FFmpeg (オンライン), 入手先 <<https://ffmpeg.org/>> (2019-09-15 参照).
- [21] K. W. Park, W. Hwang and K. H. Park.: FalconEye: Data Center Status Extraction via Vision Transformation Techniques, *IEEE 34th Annual Computer Software and Applications Conference Workshops*, pp. 388-392(2010).
- [22] T. Kimura, K. Ishibashi, T. Mori, H. Sawada, T. Toyono, K. Nishimatsu, A. Watanabe, A. Shimoda, and K. Shiimoto: “Spatio-temporal Factorization of Log Data for Understanding Network Events,” *Proc. of IEEE INFOCOM2014*, pp.610-618(2014).
- [23] 竹内 裕幸, 平川 豊, 大関 和夫: 機械学習を用いた無線 LAN 屋内測位における精度評価, 第 80 回情報処理学会全国大会講演論文集, pp.133-134(2017).
- [24] 松村 洋志, 笹井 一人, 北形 元, 木下鉄男: ネットワーク情報の監視支援のための自動編纂手法の提案, 第 79 回情報処理学会全国大会講演論文集, pp.159-160(2017).
- [25] 北口善明, 石原知洋, 高嶋健人, 田川真樹, 田中晋太郎: Raspberry Pi を用いた無線ネットワーク状態評価手法の提案, 研究報告インターネットと運用技術 (IOT), 2014-IOT-25, pp.1-6(2014).
- [26] V. Radu, L. Kriara and M. K. Marina: Pazl: A mobile crowdsensing based indoor WiFi monitoring system, *Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013)*, pp.75-83(2013).