Vol.2017-IOT-38 No.5
2017/6/24

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装置の開発

小川康一[†1, †2]　吉浦紀晃[†1, †2]

概要：大学では，利用者が自身の裁量で多種多様なネットワーク機器を導入し利用している．しかし，利用者のネットワーク機器が引き起こすトラブルの対処は，ネットワーク管理者の大きな負担となっている．これらの課題に対し，我々はネットワーク管理者の目の代わりとなる画像処理技術を活用することによって，利用者のネットワーク機器を監視する新しい手法を提案している．本論文では，提案手法の発展として小型コンピュータを活用した監視装置の開発と具体的な監視手法について述べる．提案手法を実装，評価し，利用者のネットワーク環境における状態監視に有効であることを実証した．

キーワード：ネットワーク障害，利用者対応，ネットワーク監視，画像処理技術

Development of monitoring equipment for network devices by using Small Computers and Image-Processing Technology

KOHICHI OGAWA[†1, †2]　NORIAKI YOSHIURA[†1, †2]

1. はじめに

大学などの教育機関において，学術情報基盤は教育研究を支える重要な要素のひとつである．中でも学内ネットワーク（以下，ネットワーク）は，インターネットや学内サーバへの接続に必須であり，教職員や学生が講義や文献検索，研究・実験などに利用している．このため多くの大学では，ネットワークを情報センターが独自に環境を整備している．大学は企業と比べて自由度の高い利用環境にある．各大学で情報セキュリティ規則や倫理規定が定められているが，一般的に企業ほどの接続制限はなく，利用者の裁量に委ねられている．このため利用者は，各自が用意したネットワーク機器を利用してネットワークに接続している．このような環境では，利用者の不注意によりネットワーク障害を引き起こすことが多い．

利用者の環境におけるネットワーク障害は，ネットワークに繋がらなくなる場合や，極端にネットワークが遅くなるなどの現象によって発覚する．この障害の多くは，イーサネットスイッチのループ，ネットワーク機器の故障，LAN ケーブルの抜け，ネットワーク機器の電源抜けなど，利用者の不注意が原因である．しかし，利用者は障害の原因を特定できないため，ネットワーク障害を解決することは難しく，情報センターに問い合わせる．ネットワーク管理者が個別に利用者の部屋に出向き，障害対応せざるを得ないのが現状である．

このようなネットワーク管理者の負担を軽減するため，我々はシステムの管理運用の自動化を検討する中で，ネットワーク管理者の障害解決の際の振る舞いに着目した．管理者が管理機能のない機器の異常を調べる方法＝「目視」によって機器が正常か異常かをリアルタイムに判別する方法[1]を提案している．

本論文では，[1]の手法を発展させ，監視装置を設計・開発した．監視対象のネットワーク機器が運用するネットワークは障害の可能性があり，監視に影響が懸念されることから3G/LTEの回線を利用した．提案手法の適用例として，著者がネットワーク管理を行っている埼玉大学（以下，本学）で利用されているメディアコンバータの監視実証実験を通して本提案手法の有用性を述べる．

2. 研究背景

本章では本研究の背景について利用者環境の障害とその対応について述べる．

2.1 障害対応の人員不足とコストの問題

大学という教育研究機関の特性上，ネットワークは平日だけでなく休日深夜においても定常的に利用される．しかし，障害対応やサポートにあたる職員の勤務時間は定められており，対応時間帯が限られている．障害対応や利用者サポートを外部企業に委託する場合でも問題がある．専用のサポートデスクを設置する場合は年間で相当額の費用がかかる．保守費用にかけられる財源は年々減少傾向にある．また，障害対応にかかる時間が対応ごとに異なるため，費用の見積が困難で対応を定型化して依頼できない．

本学をはじめ，少数の管理者でネットワークを管理しているような大学では，利用者のネットワーク環境を手厚く

†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

支援するまで手が行き届かないのが現状である．これまで利用者の障害対応は，サポートを実施する人員の確保や，障害の内容が多種多様であるため，コスト面からも積極的な対応が行われてこなかった．

しかし，利用者が抱える障害を放置し，障害が継続することによって大学のネットワーク全体へ悪影響を及ぼすこともある．たとえば，ネットワーク機器自体が行き先不明なパケットを大量に送出する例[2]や複合的に障害を発生させる例[3]が報告されている．これらの障害の原因は，機器の異常やループ構成によるもので，大量のブロードキャストを送出する「ブロードキャストストーム」と呼ばれている．ブロードキャストストームを防ぐ機能を持つ基幹スイッチも存在しているが，一定量のストームを監視してブロックするだけで障害自体を追跡できないため，障害の原因を確認不能である．ネットワーク全体の不具合は教育研究に与える影響が大きく，利用者への対応は急務となっている．このような背景から，利用者のネットワークの状況を的確に把握する必要性がある．

2.2 利用者のネットワーク機器の管理機能不足

通常，ネットワーク障害の検知は，管理ツールを用いた監視システムで行われる．監視システムとしては，オープンソースソフトウェアでは Nagios[4]や Zabbix[5]が，商用ソフトウェアでは JP1[6]や Tivoli[7]などがよく利用されている．監視システムでネットワーク機器を監視する場合，ネットワーク機器が備えた管理機能として，SNMP（Simple Network Management Protocol）[8]といった監視プロトコルを用いて，機器の死活監視やネットワークの状態を確認する方法が用いられる．しかし，利用者が用意するネットワーク機器は，機能が最小限であり，SNMP などの管理プロトコルや管理機能を備えていない場合が多い．利用する機器は利用者の裁量で決まるため，メーカーや機器も多種多様で入れ替えも不定期に行われる．ネットワーク管理者が初めて目にするネットワーク機器も存在する．このため監視システムの利用は難しい．仮に，利用者に管理機能を持つネットワーク機器を利用しても，監視対象となるネットワーク機器が多く，管理が立ち行かなくなる．一部，ループを検知する機能をもつスイッチも発売されているが，利用者が積極的に導入する必要性がない．

2.3 継続監視の必要性

利用者が，自身の行動が原因で障害を引き起こした場合，その事実を受け入れがたい場合がある．しかし，ネットワーク管理者が継続して利用者の環境を注視することは困難である．結果，事象の時系列的な変化を確認できず，利用者が納得するような解決方法の提示に至らない．慢性的に障害が頻発するような環境では，障害対応が長期間に及ぶこともある．このような場合，障害が発生する部屋に何度も通う必要がある．コストや場所の問題等，様々な理由から障害を未然に防ぐ対策を実施できない場合がある．このことから継続して利用者のネットワーク環境を監視しておく必要がある．

3. 提案手法

本章では，利用者のネットワーク機器を監視するための提案手法について述べる．

3.1 画像処理による監視手法

通常，ネットワーク管理者は障害の状況を把握する際に，問題がありそうなネットワーク機器を目視によって確認している．そこで，管理者の目の代わりとなるカメラを用いる．カメラは小型コンピュータに接続して得た画像を保持，分析に用いる．ネットワーク機器の状態を自動的に把握する方法を提案する．手法は図 1 に示すような流れとなる．

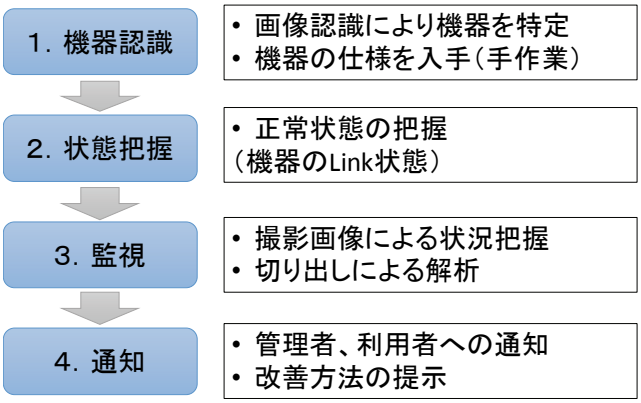

図 1　提案手法の処理手順

整理すると，利用者環境でのネットワーク機器の障害時の動作には大きく分けて 2 種類がある．ひとつはメディアコンバータのようにあらかじめ障害の状態を把握できる場合である．これは例えばネットワーク機器のマニュアルを参照して LED ランプのつき方を知ることで把握可能である．もうひとつはあらかじめ状態が正常か異常かを定義できない場合である．ネットワークの障害が頻発する部屋に設置するとき，ネットワーク障害が発生せずに通常通り利用できる状態を正常状態として仮定し，ネットワーク障害が発生して利用者がネットワークを利用できない状態を異常状態として定義することができる．この情報を利用して差分から変化を見ることができる．清掃等で空きポートに LAN ケーブルを接続してループを構成してしまう例が多発している．本手法は，このようなループを構成する場合でも状態監視に利用可能である．

3.2 定常的に状態を把握可能な場合

ネットワーク機器の状態を確認するためには，ネットワ

ーク機器の LED ランプを確認し，正常な場合と異常な場合を保持しておく．本提案手法では，点灯表示の違いを利用する．たとえば，本学で利用している利用者側のメディアコンバータ[9]には，4 つの LED が備わっている．それぞれ，電源ランプ，光のリンクアップ，イーサネットポートのリンクアップ，リンクスピードランプである．図 2 のようにすべての LED が点灯している状態が正常状態である．

図 2　メディアコンバータの正常状態

一方で，図 3 に示すように LED が不完全な状態で点灯する場合がある．これらは利用者に故意に電源や，LAN ケーブルもしくは光ファイバを抜いたことに起因する．これらの LED をオブジェクトとして識別して異常を検知する．

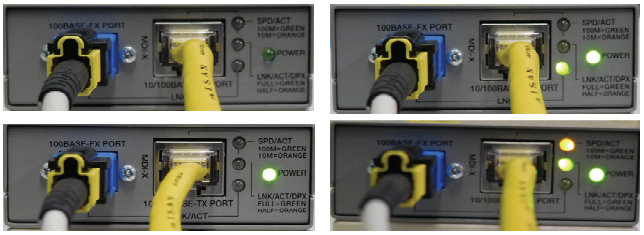

図 3　メディアコンバータの異常時の LED 表示パターン（左上：電源断，左下：電源接続のみ，右上：光リンクのみ，右下：LAN 接続のみ）

メディアコンバータは，LED の点滅状態を把握することで 5 つのパターンに分けることができる（表 1）．

表 1　メディアコンバータの状態パターン一覧

パターン	異常				正常
	1	2	3	4	5
電源	×	○	○	○	○
光Link	×	×	○	×	○
LAN	×	×	×	○	○

メディアコンバータがどのような状態かを把握するために，情報を収集する必要がある．具体的には，LED の点灯位置を把握するために，LED のみを検出したい．そのためには撮影した画像を白黒で表現する二値化処理を行い，LED の位置を検出しやすくする．二値化処理では，画像の明るい部分は白く，暗い部分は黒く表示される．二値化処理後，LED が白く一連の繋がった要素として表示されるようになる．この一連繋がった要素を「ブロブ」（Blob）という．ブロブの把握によって，物体の大きさや姿勢状態の把握が可能である．画像処理では，ブロブの形状を数値化で表現する．また，1 つの画像内に複数のブロブが存在するときは，各ブロブに一意の識別子をつけて個々のブロブを識別するラベリング処理が行われる．このように，LED をブロブとして認識し，個々の LED をラベリング処理によって識別可能であれば，イーサネットスイッチの状態を把握できる．図 4 はメディアコンバータの LED 点灯位置を把握するための処理の流れを示したものである．

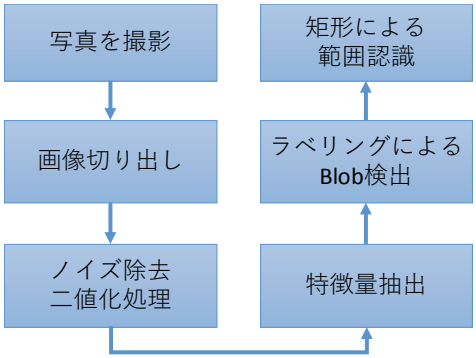

図 4　LED 点灯位置の把握処理の例

図 5 にラベリング処理によってメディアコンバータの LED を検出する例を示す．矩形によって LED の範囲が正確にプロットされていることがわかる．しかし，メディアコンバータでは，表 1 のパターン 3 とパターン 4 は同じく 2 つの LED が点灯状態である．この場合は，画像認識で取得した位置情報により 2 つの状態の差を認識する．

図 5　ラベリング処理による LED 検出の例
（左：元画像，右：LED 検出後の画像）

本提案手法はイーサネットスイッチやブロードバンドルータなど LED を持つネットワーク機器にも利用可能である．ループは高速な点滅を動きとして捕らえる必要があり，[1]で用いた Intel NUC などの高速な機器が監視装置として必要となるが，本手法は Raspberry Pi などの機器で十分処理することが可能であるため，障害の初期状態を把握する上で利用可能である．

4. 提案手法による監視システムの実装

本章では前章で述べた提案手法に基づき，画像処理を用いた監視システムの実装について述べる。

4.1 監視システム構成

監視システムは，監視装置と監視情報集約サーバ，管理者端末で構成される．システムの構成概要を図 6 に示す．

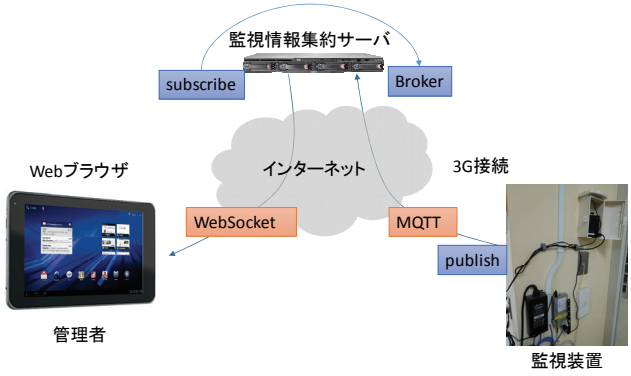

図 6　監視システムの構成概要

システムの処理の流れは以下のとおりである．

1. 監視装置は，接続された USB カメラで LED ランプの画像を定期的に（5 分ごと）撮影し保存する．
2. 取得した画像データから画像処理プロセスにより LED ランプの数を取得する．
3. 監視措置は手順 2 で得た LED ランプの数を監視情報集約サーバに送信する．
4. 監視情報集約サーバでは，収集したデータをデータベースに蓄積するとともに，管理者のブラウザに最新の情報を送信する．

4.2 通信方式

監視装置と監視情報集約サーバ間，監視情報集約サーバと管理者端末間の通信には，MQTT(Message Queue Telemetry Transport)[10]を利用する．MQTT は IoT 向けのメッセージキュープロトコルである．出版購読モデル（Publish/Subscribe モデル）を採用しており，プロトコルヘッダーが小さいという特徴がある．実装には，MQTT の参照実装として広く利用されているオープンソースソフトウェアの Mosquitto[11]を採用する．

MQTT は，データを送出する Publisher，データを受信する Subscriber，Publisher と Subscriber の間を取り持つ中継の役割を担う Broker で構成される.Publisher は事前にデータを必要とする配信先を知る必要がなく，自分の範囲内の Broker にデータを送信するだけで良い．

本システムにおける各機器の MQTT における役割は，監視装置は Publisher，監視情報集約サーバは Broker と Subscriber を兼務する．監視情報集約サーバは内部で定期的に Broker よりデータを購読することで監視データを取得する．各機器間の通信フローを図 8 に示す．

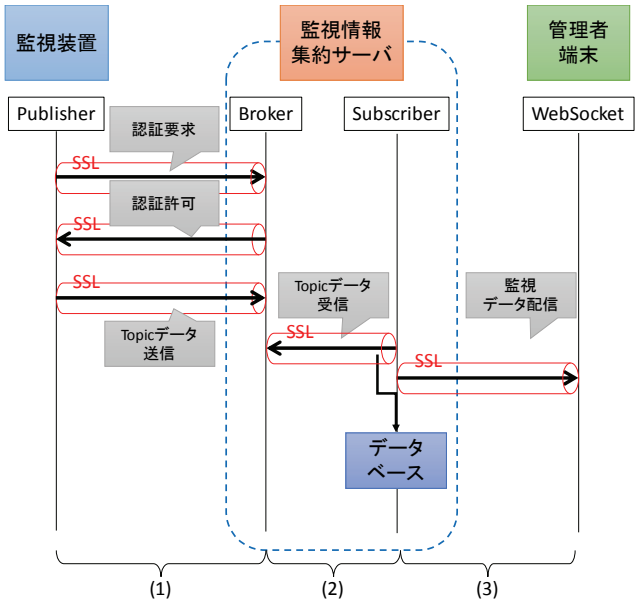

図 7　各機器間の通信フロー

監視装置と監視情報集約サーバ間の通信は 3G/LTE 回線を利用する．監視情報集約サーバと管理者端末の通信は大学構内のネットワークの利用する．監視情報は，WebSocket で直接リアルタイムに管理者の操作端末に送信する．

MQTT では，メッセージはトピックという単位で定義される．取得したいデータがあれば，トピックを指定して購読する．監視装置の主要プログラムは Python で実装するため，MQTT のパッケージである Paho-mqtt[12]を利用した．時系列のログを収集して，監視情報を追跡可能にするために MySQL データベースを用意する．Broker に対して Subscribe するのと同時にデータベースのテーブルへ書き込みを行う．この時管理者端末への WebSocket の通信も同時に実施する．WebSocket の実装には Python の Web フレームワークである Tornado[13]を採用した．

MQTT の通信は，セキュリティ面を考慮し SSL により通信を暗号化している．ログイン認証を必要とすることで一定のセキュリティを確保している．管理者端末から監視情報集約サーバへのアクセスは Apache の SSL 通信により暗号化する．ブラウザが読み込んだ HTML ファイルに対して Javascript を用いて WebSocket により監視情報の更新を Push 配信する．この時，Tornado の SSL 暗号化により通信のセキュリティを確保する．

4.3 監視装置

監視装置は，LED のランプを監視するため，Raspberry Pi に USB 接続の Web カメラを取り付け，監視を行う．監視装置について図 8 に設置例を示す．不特定多数の利用者が

出入りするオープンスペースには，ウオルボックスを設置して施錠する．実装には画像ライブラリとして OpenCV[14] を用いた．プログラミング言語は Python 2.7 を用いた．OpenCV のブロブ検出機能を利用してブロブを特定し，LED ランプのつき具合を把握する．障害を想定した場合，監視対象のネットワークは利用不能となるため，監視装置の通信には 3G/LTE 回線を利用する．監視装置の仕様は以下の通りである．

- コンピュータ：Raspberry Pi 3
- カメラ：BSWHD06MBK
- データ通信端末：L-02C
- OS：Rasbian Jessie

図 8　監視装置の設置例

4.4 Web カメラのアタッチメント

ネットワーク機器の LED を的確に監視するためには，LED ランプを監視しやすい位置に取り付ける必要がある．たとえば，本学のメディアコンバータは壁に常設しており，最適な位置に Web カメラを固定可能である．そこで，金具を組み合わせたアタッチメントを設計，取り付けた(図 9)．

図 9　Web カメラのアタッチメントによる装着例

監視対象であるメディアコンバータの組み込みネジを利用して，メディアコンバータ自体に金具を直接取り付ける方法を検討した．この方法により，監視を開始する際にメディアコンバータの電源を停止する必要がなく，短時間で装着ができ，既存の環境へ与える負荷を最小限に抑えることができる．部屋などに固定されていないイーサネットスイッチの監視には，ある程度の範囲で移動可能な Web カメラを設置することとした．

4.5 監視情報集約サーバ

監視装置と監視情報集約サーバの接続はインターネットを介して行われる．セキュリティの確保は SSL 証明書とユーザ認証により実現する．SSL 証明書は国立情報学研究所が提供する UPKI サービスのサーバ証明書を利用した．サーバは学内のサーバ室に収容し，固定 IP アドレスを付与した．サーバの仕様は以下のとおりである．

- HP ProLiant DL320G6
- CPU：Intel Xeon 2.53 GHz Dual Core
- メモリ：8GB
- HDD：SATA 600GB × 2 (RAID1)
- OS：Ubuntu Server 14.04.4

4.6 管理者端末

管理者端末のユーザインターフェースには，OS やハードウェアを問わず利用可能なブラウザを採用した．ただし，管理者から能動的にブラウザを読みこんで確認する方法は監視方法として望ましくない．そこで，リアルタイムに画面の情報が更新される WebSocket を採用した．実際の画面を図 10 に示す．

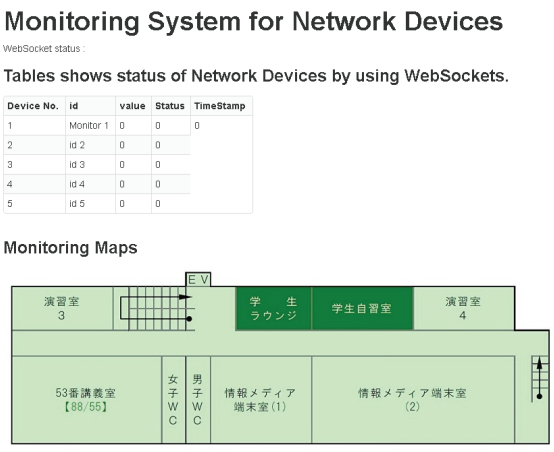

図 10　管理者端末の画面

5. 監視システムによる実証実験

本章では，前章で述べた監視システムを用い，埼玉大学で実運用しているメディアコンバータを利用した実証実験について述べる．

5.1 メディアコンバータの障害検知実験

実験は埼玉大学工学部講義棟 2F に設置されているメデ

ィアコンバータを利用して実施した．監視装置の通信に用いるアクセス回線として 3 種類の回線を準備した．表 2 に仕様の一覧を示す．このような SIM カードを提供する企業は，仮想移動体通信事業者（Mobile Virtual Network Operator，以下 MVNO）と呼ばれ，回線設備は持たずに他の回線業者の回線を間借りしてサービスを提供する企業を指す[15]．いずれも NTT ドコモの MVNO 提供業者である．

表 2　実験で使用した SIM カードの仕様一覧

MVNO企業名	楽天モバイル	nuromobile	OCNモバイル
SIMの種類	データSIM	データSIM	データSIM
SIM通信速度	200Kbps	500MBまで0円	1日110MBまで以降200Kbps
料金（月額）	525円	0円～1600円	972円
付与IPアドレス	プライベート	プライベート	グローバル

実験方法は，各部屋に設置されたメディアコンバータに対して，それぞれ異なる条件を設定して故意に障害を発生させることで実現した．検証条件を下記に示す．

■検証条件
1. 光ファイバを抜く（ランプ 2 つ点灯）
2. LAN ケーブルを抜く（ランプ 2 つ点灯）
3. 電源を抜く（全 LED ランプ消）

今回は各検証条件において同一の日時でプログラム内，もしくはログを利用して処理にかかる時間を計測する．連続して 5 回の計測し，平均を算出した．表 3 に検証条件 1 の場合の実験結果を示す．

表 3　検証条件 1 の場合の経過時間
（表内番号は図 8 中に準拠）

実験回数	(1)[s]			(2)[s]	(3)[s]
	楽天	Nuro	OCN		
1	0.025469	0.024824	0.024978	0.250	0.016
2	0.025991	0.024864	0.024467	0.175	0.023
3	0.024924	0.025193	0.026003	0.240	0.015
4	0.02658	0.025599	0.026737	0.321	0.031
5	0.02605	0.025758	0.025662	0.204	0.032
平均	0.025803	0.025248	0.025569	0.238	0.023

(1)の 3G/LTE 回線を使用した場合でも，各社ともに 0.025s 程度の通信でデータをサーバへ送信できることがわかった．これは MQTT のプロトコルヘッダーが小さく，少量のデータを細かく転送する通信に向いているためと推測できる．(2)では受信した監視データをサーバ上で処理するために多少時間がかかっていると推測される．(3)は WebSocket のデータ受信のみの計測で平均して 0.023s で通信できることがわかった．

6. 考察

本章では，提案手法による評価実験を元に課題に対する考察について述べる．

6.1 人手不足に対する解決

提案手法により実装した監視装置を用いることにより，ネットワーク機器が設置された場所に赴かなくとも状態を把握することが可能である．また，3G/LTE を活用することによりネットワークがない箇所でも監視を維持可能である．メディアコンバータやイーサネットスイッチをはじめ，LED 表示部をもつネットワーク機器は状態変化を把握可能である．

通信が不能になった場合でも，監視装置自体の障害か故障，と判断ができる．監視情報集約サーバに蓄積された情報から時系列で順を追って状況変化を把握可能で，監視装置が最後の監視情報を報告するまでさかのぼって状態を追跡可能である．本提案手法の欠点としては，LED 表示部を持たないネットワーク機器に関しては監視不能なことがあげられる．

6.2 利用者ネットワーク環境の把握

本提案手法は，SNMP などの監視プロトコルや管理機能がなくとも利用者のネットワーク機器の状態を監視可能であることを示した．このことにより利用者の部屋へ駆けつける前に症状を把握可能である．障害のある部屋へ向かわずとも，利用者に対して電話やメール等で解決方法をアドバイスすることも可能となる．

現段階で，本手法は LED ランプの点灯状態を元に管理者が判断する方法をとっているが，監視情報集約サーバの機能を強化することにより，ネットワーク管理者でなくとも，アドバイスが可能な支援ツールとなりうる．

本提案手法により，定量的な判断が可能となるため，利用者に対して同一の品質を提供可能可能である．今回の実装では，ネットワーク機器の LED ランプの状態を 5 分ごとに撮影したが，撮影の粒度に関しては適切かを検討する必要がある．データベースを活用することにより，管理者用のインターフェースを充実させることで長期的な監視が可能である．監視装置自体が不具合により停止しても，それまでの記録は監視情報集約サーバに蓄積される．このことから，研究室で発生する障害について時系列で障害発生の原因を追跡可能であるといえる．

7. 本研究における課題

本章では本研究における課題について述べる．

7.1 回線準備の費用

近年，比較的安価な MVNO の SIM カードや IoT デバイス用途に特化した SIM カードが出回っているが，多数の監視装置に実装する場合にコストと準備の手間がかかる．少数であれば問題になりえないが，本学のようにメディアコ

ンバータが 1,800 箇所にあるような場合，すべての場所の監視は不可能である．アクセス回線がない状況において，監視装置の情報をいかに情報収集していくかが課題となる．

7.2 監視装置の専用機器化

今回開発した監視装置は，簡単に入手可能な小型コンピュータの Raspberry Pi を採用している．OS を含め，監視システムとして実用性を考慮した場合，メンテナンス性や耐久性が低い．高フレームレートで稼働するカメラを用いるとともに，FPGA などを利用して監視用途に特化した専用ボードを開発することも検討している．

8. 関連研究

通信の監視の研究では，北口ら[16]は無線 LAN アクセスポイントの状態評価を目的として通信の品質や性能などの指標に基づいて, Raspberry Pi を用いて監視を行う手法を提案している．しかし，当該研究では監視用の有線ネットワークが必要である点など、本研究とは異なる．大垣内ら[17]は，大学ネットワークを構成するエッジスイッチに OpenBlocks を接続してモニタリングする手法を提案しているが，利用者のネットワーク環境自体の監視ではなく，本研究の視点とは異なる．

ネットワークの障害を検知する方法には，ネットワーク機器から収集したログを機械学習やデータマイニングの手法で分析する方法[18]がある．しかし，ネットワーク機器のログ出力機能や SNMP などの監視プロトコルを備えた機器に限定され，利用者が普段使う簡易的な機器を網羅できない．一部市販のルータはログを出力する機能を持つが，継続してログを収集するためには別途サーバが必要であり，異常を検知した際には管理者へ通知する監視専用のネットワークが必要となる．また，端末から流れるパケットを分析することにより異常検知を行う方法[19]や，端末にエージントを導入して監視する方法があるが，障害の原因が端末にある場合に限られ，この方法ではネットワーク機器に問題があるとは判断できない．

SNMP により監視を行うことで，利用者のバーストトラフィックやブロードキャストストームを検知する研究[20]もある．しかし，全体への影響を防ぐことに重点が置かれており，利用者が引き起したネットワーク障害自体の解決には利用できない．

以上の点から，本研究のような利用者のネットワーク環境を直接監視し状態を把握するような研究例は希少である．本研究は LED インジケータを持つ機器であれば監視可能であり，他の研究分野での応用も十分期待できる．

9. おわりに

本論文では，利用者のネットワーク環境に焦点を当て，ネットワーク管理者の行動に着目し，画像処理技術を活用する提案手法を示した. 実装には小型コンピュータを用い，監視装置を開発し，監視システムによる実証実験を通して提案手法の有効性を確認した．今後は継続して実証実験を継続するともに課題解決を進め，監視装置の情報収集手法について別の解決策も検討していく予定である．

謝辞

本研究の一部は，JSPS 科研費 17H00371 の助成によるものである．

参考文献

[1]小川康一，吉浦紀晃：画像処理を用いた汎用性の高いネットワーク機器監視システムの提案，研究報告インターネットと運用技術（IOT）, pp.1-7(2016).
[2]インターネット接続解説ブログ KAGEMARU-info, https://www.akakagemaru.info/port/cannotinternet-hub.html
[3]日経ネットワーク，トラブルシューティング「気づき」のヒント，2012 年 6 月.
[4]Nagios, https://www.nagios.org/
[5]Zabbix, http://www.zabbix.com/jp/
[6]JP1, http://www.hitachi.co.jp/Prod/comp/soft1/jp1/index.html
[7]Tivoli, http://www.ibm.com/software/tivoli
[8]Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework, The Internet Society, 2003.
[9]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，研究報告インターネットと運用技術（IOT），2015-IOT-30，pp. 1-8(2015).
[10]MQTT, http://mqtt.org/
[11]Mosquitto, https://mosquitto.org/
[12]Paho-mqtt，https://pypi.python.org/pypi/paho-mqtt
[13]Tornado，http://www.tornadoweb.org/en/stable/
[14]画像ライブラリ OpenCV，http://opencv.org/
[15]MVNO に係る電気通信事業法及び電波法の適用関係に関するガイドライン（再改定） (プレスリリース), 総務省, (2008 年 5 月 19 日) 2017 年 5 月 31 日閲覧.
[16]北口善明，石原知洋，高嶋健人，田川真樹，田中晋太朗：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).
[17]大垣内 多徳,半田 憲嗣,澤田 雅子,福井 一俊,山下 芳範：福井大学学内ネットワークシステムの設計と構築，研究報告インターネットと運用技術（IOT），pp.1-6（2010）.
[18]木村達明:Syslog 分析による大規模ネットワーク故障診断技術 -ネットワーク機器ログデータの活用-, 電子情報信学会会誌, Vol.98, No.9, pp.823-828(2015).
[19]國吉賢吾,森井昌克：端末監視によるホームネットワーク異常検知システム，研究報告ユビキタスコンピューティングシステム（UBI），pp.39-46(2009).
[20]村井 秀聡,砂田 英之,牧 和宏:SNMP を利用したバーストトラフィック検知方式の提案，情報科学技術フォーラム講演論文集，FIT2014，pp.175-178(2014).