

Vine Linux 5 による
担任用ネットワークカメラシステム
(NCSFCT)

A Network Camera System for a Class Teacher
(NCSFCT) with Vine Linux 5

国立茨城工業高等専門学校・電子情報工学科

小飼 敬, 滝沢 陽三, 杉村 康

1. はじめに (1/3)

最近、単眼カメラを持つWebサーバであるネットワークカメラが、安価に手に入る様になった。

又、教育環境の高度化に伴い、教室等への無線LANの導入が進んでいる。

その様な環境では、ネットワークカメラを教室に設置さえすれば、遠隔の教員室のWebブラウザより、教室内を覗くことは可能。

1. はじめに (2/3)

しかし、主な用途はブラウザによる目視にあるため、多量の画像を遠隔のパソコン上への保存や参照の機能が無い。

そのため、多量の画像を過去に遡って、再度繰返し見ることが困難。

又、教室の様な出入りが容易な場所では、悪戯等によるカメラ障害発生の可能性も有り、その警報システムの実現も必要。

1. はじめに (3/3)

当論文では、

担任の先生からのヒアリング結果に基づいて、

ネットワークカメラを新規開発プログラムより無線

LAN経由で駆動する等により、上記の問題点等

を解決する「担任用ネットワークカメラシステム」

を提案.

当システムを使用することによって、

より充実した担任業務等を実現することが可能.

2. システムへの要求条件(1/1)

- (1) 教室内を**死角無**で、遠隔の教員室から視認可.
- (2) カメラ設置に**大規模な工事は不要**.
- (3) **24時間録画・録音**. 但し、**音声無時**で、**真っ暗時**又は**画像に変化無時**は、**録画・録音無**.
- (4) カメラへの**悪戯等の監視と通知**.
- (5) **最低1年分**の録画・録音で**メンテナンスフリー**.
- (6) オンライン録画・録音と並行して、**過去の画像等を参照可**.
- (7) 運用時の**操作が簡単**.
- (8) **無償**のOS,ライブラリ等を使用し、**GPLで開放**.
- (9) LAN**回線の使用率は、10%未満**.

3. 実現手法(1/2)

- (1) 文献[1]の4台のネットワークカメラを教室等に設置し、それらを、遠隔の教員室等のパソコンより駆動するプログラムを新規開発.
- (2) 文献[5]の移動体検出アルゴリズムを応用して、24時間録画・録音. 音声無時で真っ暗時又は画像に変化無時は、録画・録音無.
- (3) 4秒に1回、各カメラにアクセス. 異常検出時は、指定の携帯電話のメールアドレス等に通知.
- (4) 1.5TBのUSB HDDへの録画・録音により、2年分のデータを保持. HDDの使用率 $\geq 90\%$ 時に最古月日のデータを自動消去により、メンテナンスフリー.

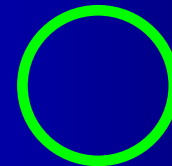
3. 実現手法(2/2)

- (5) オンライン録画・録音と過去画像等の参照処理を独立プロセスで実施により、**並行実行**.
- (6) 運用時の以下(A)~(C)の**操作は、全て画像又はボタンのクリックで実現**.
 - (A) 現在の画像や指定月日の指定時間の画像の表示.
 - (B) 画像の表示位置の変更.
 - (C) 画像の拡大表示や拡大画像の保存.
- (7) OSは無償のVineLinux5. コンパイラやライブラリ等も、該OS提供の無償のgcc, GTK+, Imlibを使用し、**新規 開発プログラムも、GPLで無償開放**.

3.1 実現手法の詳細

3.1.1 ネットワークカメラの選定 (文献[8]より)

B・K社 CS-W02G	C社 CG- WLNCPTG	P社 BL-C31	S社 SNC-M3W
実売 2.98万円	実売 2.98万円	実売 3.98万円	実売 4.8万円
640×480 ドット	640×480 ドット	640×480 ドット	640×480 ドット
最低照度 約5ルクス	最低照度 2.5ルクス	最低照度 1ルクス	最低照度 4.5ルクス



3.1.2 システムの実現方式 (1/2)

方式1.

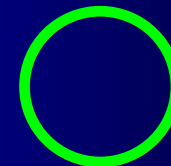
使用カメラの人感センサーや画像変化感知による画像のファイルサーバへの転送機能の
利用して、新規プログラムで
ファイルサーバを擬似(主導権
はネットワークカメラに有る.)

別途、障害チェック方式の併
用が必要

方式2.

使用カメラのWebブラウザ
によるオンライン画像表示
機能の利用して、新規プロ
グラムでWebブラウザを擬
似(主導権は新規開発プロ
グラムに有る.)

アクセス自体が障害チェ
ックを兼ねることが可能



3.1.2 システムの実現方式 (2/2)

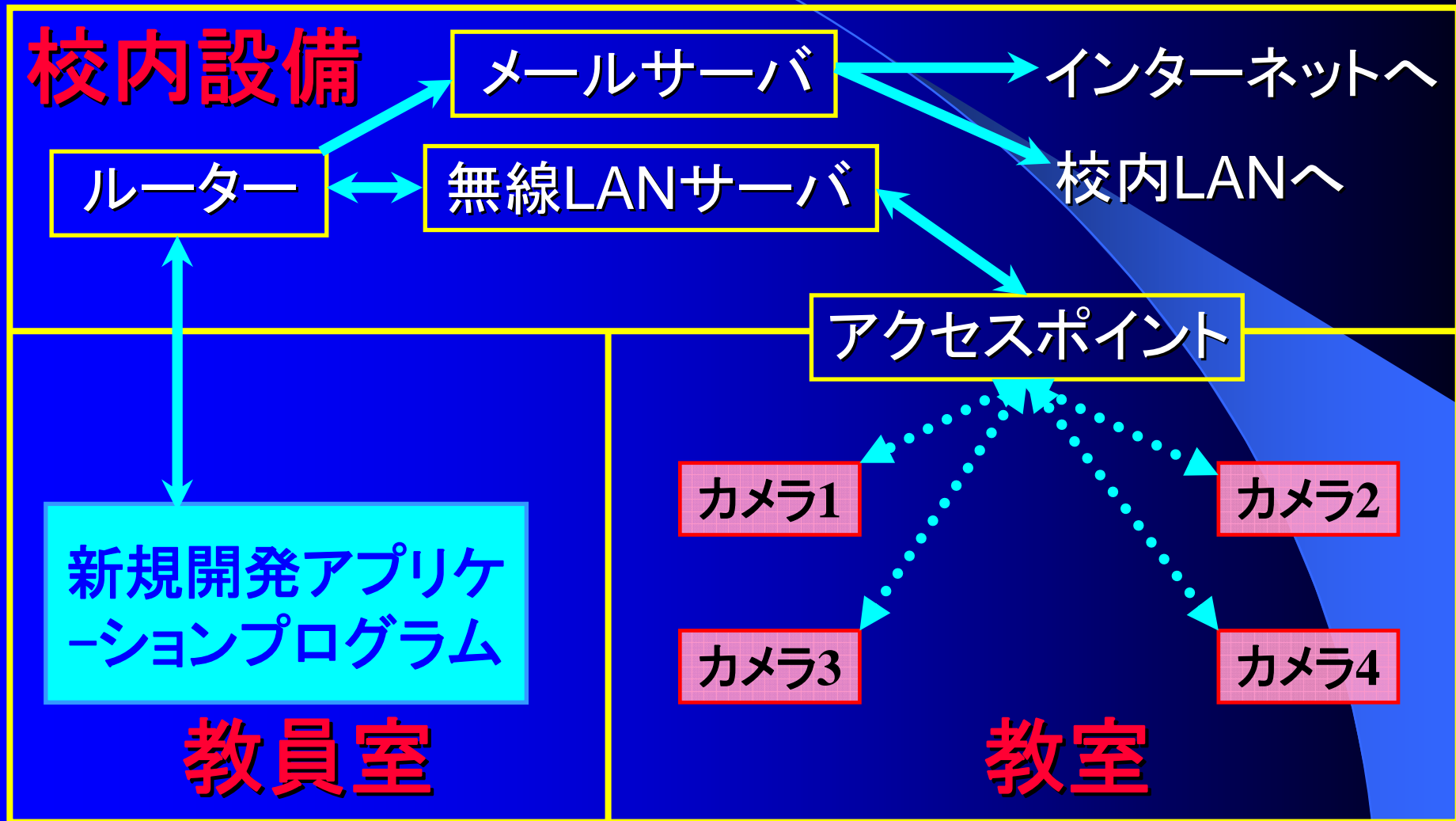


図7. 当システムの構成概念図

3.1.3 ネットワークカメラのインタフェース の解析方法 (1/4)

(1) tcpdumpによる電文のダンプ情報の取得

/* 以下をスーパーユーザ状態で行う */

script↵ **/* 標準出力をtypescriptへ書込み*/**

tcpdump tcp -x -X -s1000↵

**/* tcp電文の先頭1000バイトを16進数と
文字形式で標準出力へ出力. */**

Webブラウザでネットワークカメラを操作.

ctrl + C **/* tcpdumpの停止 */**

exit↵ **/* typescriptへの書込みを停止 */**

3.1.3 ネットワークカメラのインタフェース の解析方法 (2/4)

(2) 電文の送信元・受信先IPアドレスより電文を特定

(3) IPヘッダとTCPヘッダの除去

(A) IPヘッダ長(電文先頭1バイトの右4ビット×4[B])
分のIPヘッダを除去/* その直後がTCPヘッダ */

(B) TCPヘッダ長(TCPヘッダの12B目の左4ビット×
4[B])分のTCPヘッダを除去 /* その直後がハイ
パー・テキスト・トランスファー・プロトコル(HTTP)
の電文 */

(4) HTTP電文の解析(1/2)

(A) 空行[0x0d(CR)0a(LF)だけの行]以前の部分
(リクエスト行orレスポンス行+メッセージヘッダ部分)

(a) CRとLFと空白以外は全て印字可能文字(目視が可能)

(b) ブラウザからサーバへの要求電文では、要求内容を示すリクエスト行+要求の補足事項を示す複数のメッセージヘッダ行からなる。

(c) サーバからブラウザへの応答電文では、応答状態を示すレスポンス行+応答の補足事項を示す複数のメッセージヘッダ行からなる。

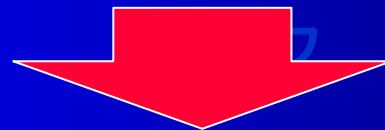
(4) HTTP電文の解析(2/2)

(B) メッセージボディ(空行の直後にあるデータ)

今回の解析対象電文では、応答電文内にのみ
に存在し、

(a) 画像データ、又は(b)音声データ.

解析では、上記(A)のリクエスト行とレスポンス行
とメッセージヘッダ行を目視.



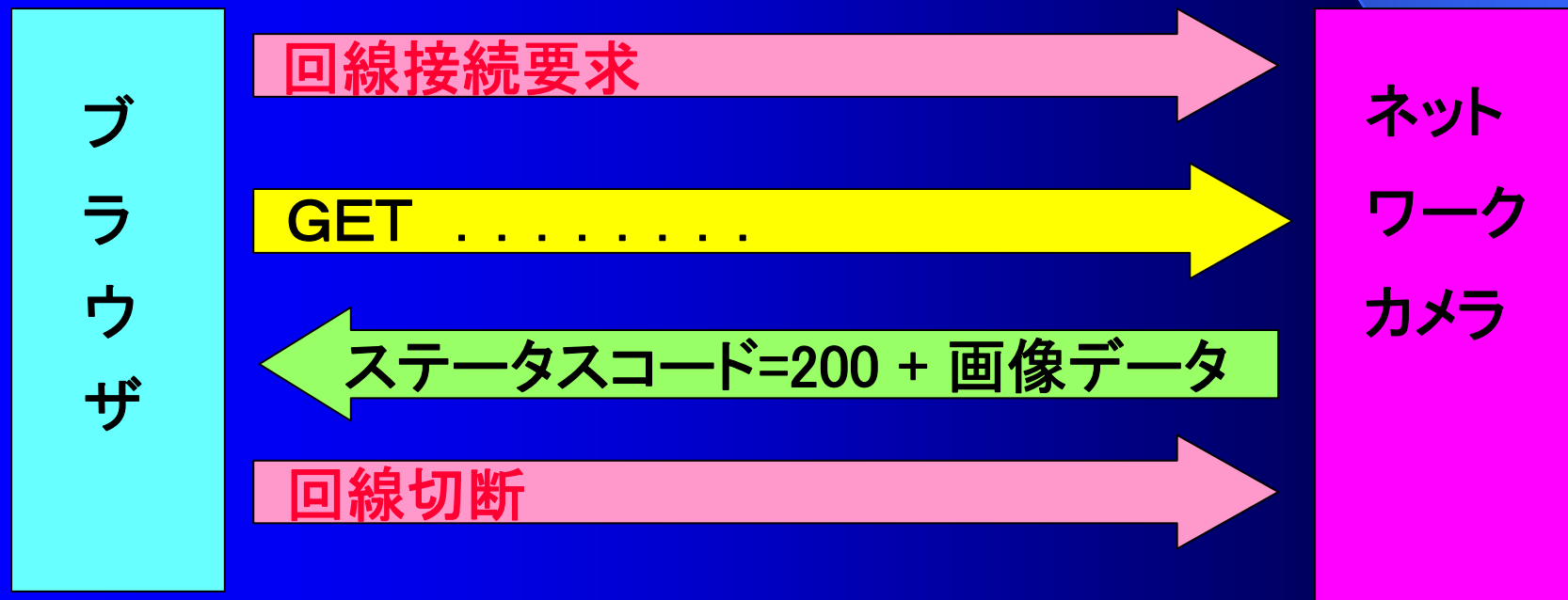
要求方法とメッセージボディの内容を推定.

3.1.4 ネットワークカメラのインタフェース 解析結果概要 (1/5)

(1) 画像データ取得では、

1画像の要求・応答の直前・直後に、

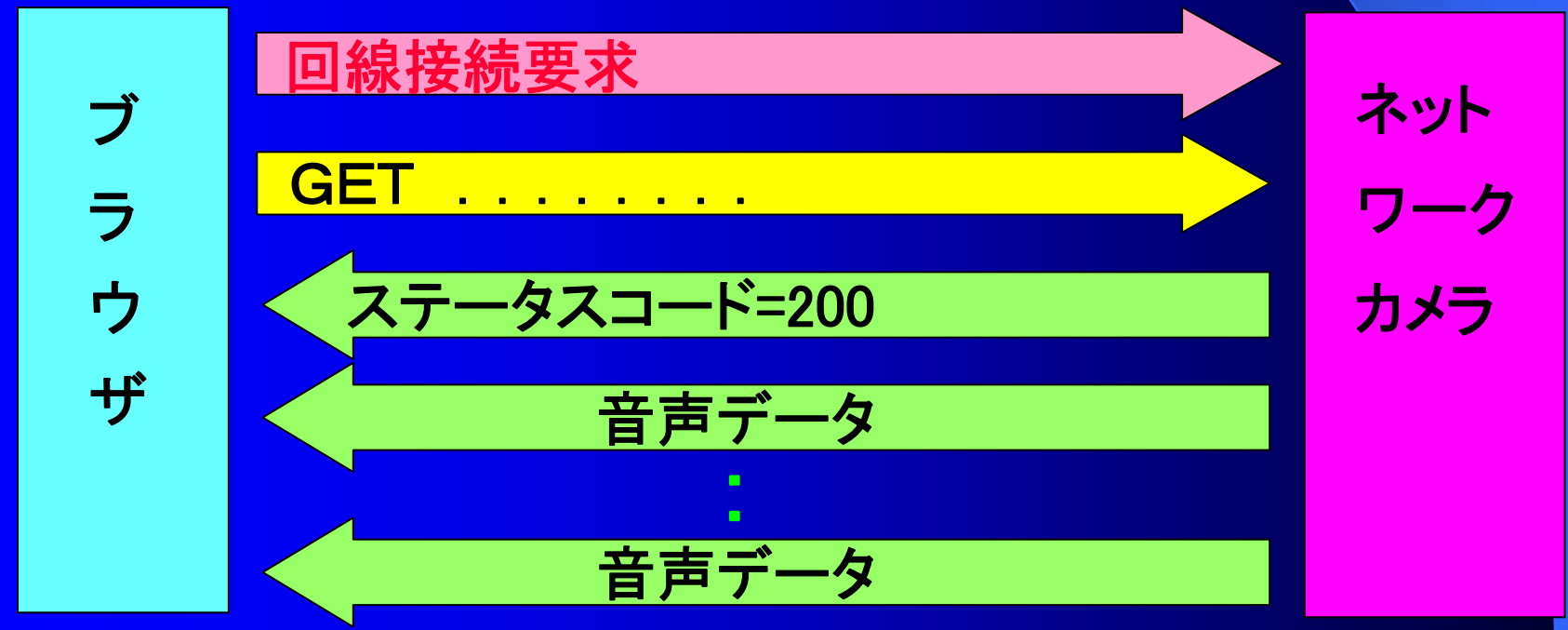
回線接続要求と回線切断が行われる。



3.1.4 ネットワークカメラのインタフェース 解析結果概要 (2/5)

(2) 音声データ取得では、

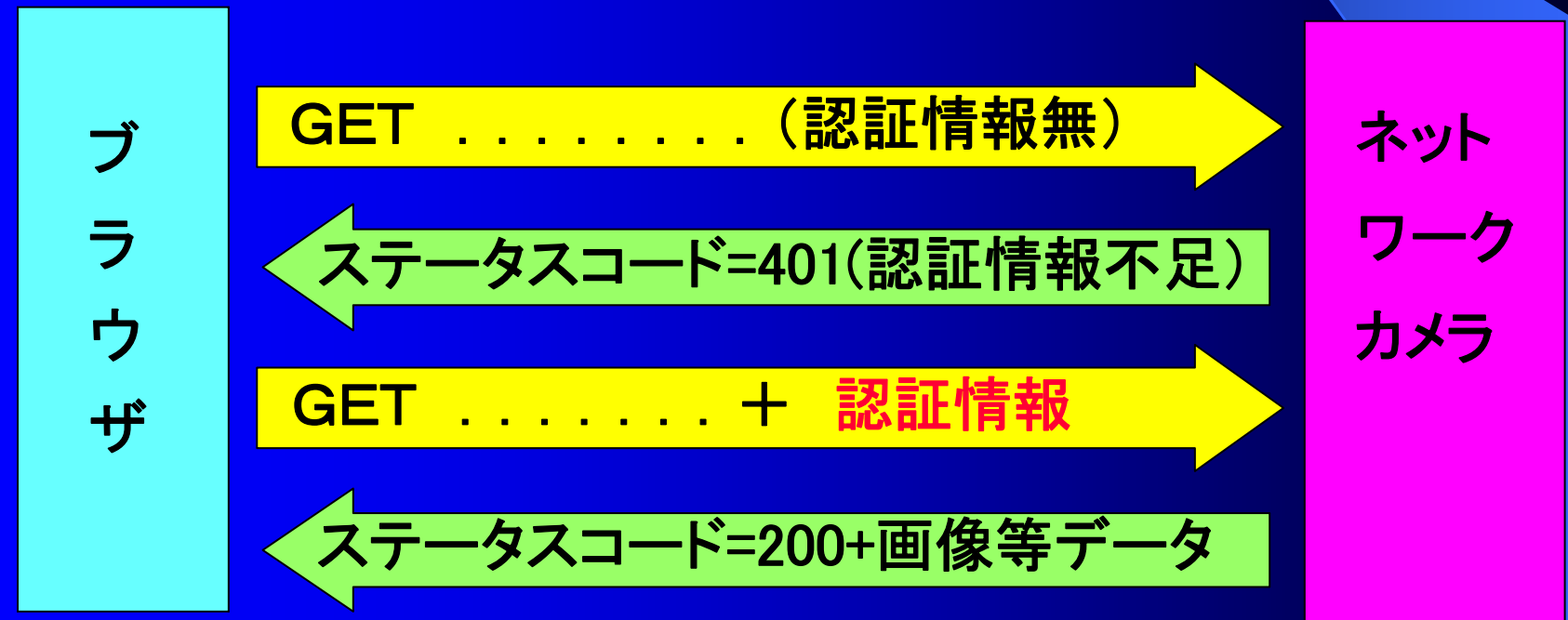
1回のみ回線接続の要求と音声の要求が行われ、その後、凡そ250ミリ秒毎に、音声データのみの応答が繰返される。



3.1.4 ネットワークカメラのインタフェース

解析結果概要 (3/5)

(3) **画像と音声の要求**では、
ユーザ名とパスワードを使用した**基本認証**が行わ
れるので、要求電文内にAuthorizationメッセージ
ヘッダを使って暗号化された**認証情報**を付加。



3.1.4 ネットワークカメラのインタフェース 解析結果概要 (4/5)

(4) **画像の要求**では、以下を送信.

```
GET△/SnapshotJPEG?Resolution=  
640x480&Quality=Clarity&Count=99  
99△HTTP/1.10x0d0a  
/* 0x0d0aはCRLF */
```

(5) **画像の応答**では、メッセージボディ内に圧縮された**JPEG画像**が返される.

メッセージボディの**バイト数**は、Content-Lengthメッセージヘッダ行で示される.

3.1.4 ネットワークカメラのインタフェース 解析結果概要 (5/5)

(6) **音声の要求**では、以下を送信.

```
GET /nphaudio HTTP/1.00x0d0a  
/* 0x0d0aはCRLF */
```

(7) **音声の応答**では、メッセージボディー無の応答が1回のみ最初に返され、その後、1000[B]のメッセージボディーのみの**ADPCM32kbps形式 *1**の音声の応答が、250ミリ秒毎に、繰返される.

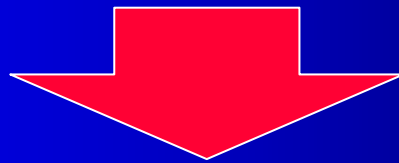
*1; Adaptive Differential Pulse Code Modulation;
適応的差分パルス符号変調(文献[11]参照)

3.1.5 ネットワークカメラの台数と配置

カメラの画角は43度.

教室で死角を無くすためには最低3台が必要.

人物特定のためには、人物を複数の角度から撮影することがベター.



4台のカメラを教室の左前、左後、右後、右前に置く。 (音声収集は、右前のカメラのみ。)

3.1.6 画像と音声の録画・録音(1/5)

(1) 録画・録音用ディレクトリ名: `d_yymmdd`
西暦下2桁 月 月 日 日

(2) 録画ファイル名: `cxx_hhmmss.jpg`
カメラ番号 時 時 分 分 秒 秒

(3) 録音ファイル名: `ghhmmssmmm.g726`
時 時 分 分 秒 秒 ミリ秒

(4) 障害ファイル名: `dorolog`

3.1.6 画像と音声の録画・録音(2/5)

(5) 画像が真っ暗で音無時は録画・録音せず.

音声の最大値 $\leq 3,000$

コントラスト(1/768のエリアの各輝度
合計の最大値-最小値) $\leq 20,000$

(6) 直前の4画像と同じで音無時は録音・録画せず.

画像を768個に分割した微小エリア(20×20ピクセル)
の全てについて、直前の画像と相関関係がある
場合(文献[5]の移動体検出アルゴリズムの応用)

3.1.6 画像と音声の録画・録音(3/5)

(7) 2時点の画像の相関関係のチェック方法(1/3)

- (A) 直前の画像1と現画像2のアドレスをpt1及びpt2が保持.
- (B) 各画像を768個の20×20ピクセルの微小エリアに分割.
=横32個×縦24個
- (C) sabun_goukei[0~767]=「各微小エリアの400*3個の各輝度情報」の画像1と画像2の差分の絶対値の「各微小エリア毎の合計」.
- (D) sabun_goukei[0~767]が一定値(=24,000)より大きい(画像が変わっている可能性がある)微小エリアについて、次頁(a),(b)の様に、ベクトル相関をチェックし、ベクトル相関が無い微小エリアが、4台のカメラの画像について、一つでもあれば、相関関係無し(録画が必要)とする.

3.1.6 画像と音声の録画・録音(4/5)

(7) 2時点の画像の相関関係のチェック方法(2/3)

- (a) 微小エリア内の各画素(ピクセル)について、直前と現在の何れかに輝度255のものがあれば、色が飛んでいるので下記(b)の一部と見なさない。
- (b) 上記(b)以外のピクセルについて、(ア)直前画像と現画像の微小エリアの正規化ベクトルを求め、(イ)それより直前画像と現画像の正規化距離を求め、(ウ)それが一定値(=0.04)を超えていれば、その微小エリアは、ベクトル相関無しとする。

(7) 2時点の画像の相関関係のチェック方法(3/3)

(ア) $20 \times 20 = 400$ 個のピクセルの各赤(R), 緑(G), 青(B)の輝度を、400個の x, y, z の3次元ベクトルと考えると、その合成ベクトルの大きさ; $|V| =$

$$\sqrt{(V_{x0}+V_{x1}+\dots+V_{x399})^2 + (V_{y0}+V_{y1}+\dots+V_{y399})^2 + (V_{z0}+V_{z1}+\dots+V_{z399})^2}$$

単位ベクトルの x, y, z 成分 V_x, V_y, V_z は、

$$V_x = (V_{x0}+V_{x1}+\dots+V_{x399}) / |V|$$

$$V_y = (V_{y0}+V_{y1}+\dots+V_{y399}) / |V|$$

$$V_z = (V_{z0}+V_{z1}+\dots+V_{z399}) / |V|$$

(イ) 画像2と画像1の合成単位ベクトルの各成分を、 $V_{2x}, V_{2y}, V_{2z}, V_{1x}, V_{1y}, V_{1z}$ とすれば、正規化距

$$\text{離}; W = \sqrt{(V_{2x} - V_{1x})^2 + (V_{2y} - V_{1y})^2 + (V_{2z} - V_{1z})^2}$$

3.1.7 ディスプレイ解像度と表示ウィンドウ(1/2)

(1) 使用カメラの解像度=640×480ドット

(2) 使用カメラの台数 = 4台

(3) (640×2=)1280×960(=480×2)以上の解像度のディスプレイ(例;SXGA+)では、そのまま表示可.



図9.ウィンドウ形式例1
(解像度=1280×1024ドット)



図10.ウィンドウ形式例2
(解像度=1280×1024ドット)

3.1.7 ディスプレイ解像度と表示ウィンドウ(2/2)

(4) $(640 \times 2 =) 1280 \times 720 (=480+240)$ 以上の解像度のディスプレイ(例;WXGA)では、図11.の様に表示可.

(5) $(640+320 =) 960 \times 720 (=480+240)$ 以上の解像度のディスプレイ(例;XGA)では、図12.の様に表示可.

これらを制御文(詳細後述)で選択可.

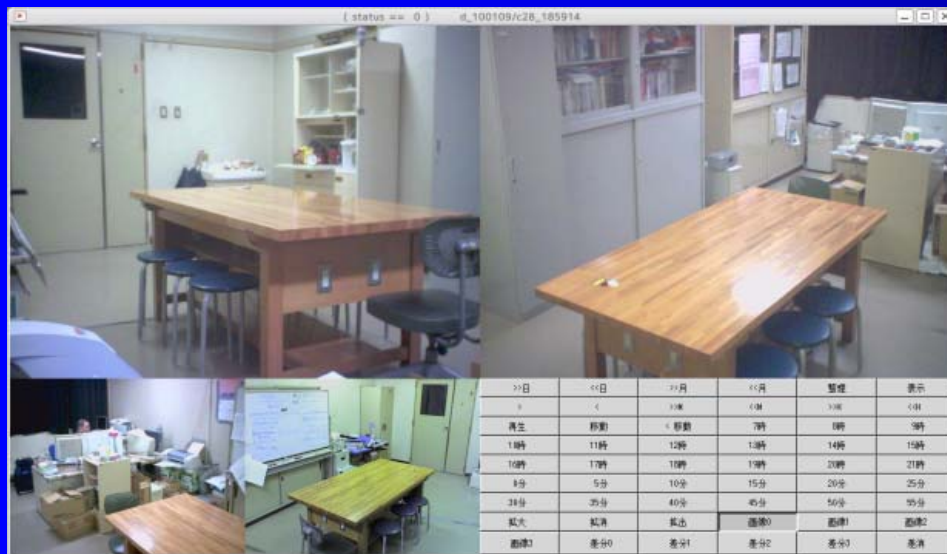


図11.ウィンドウ形式例3
(解像度=1280×720ドット)



図12.ウィンドウ形式例4
(解像度=960×720ドット)

3.1.8 システムの状態(1/13)

大別して、下記の二つの状態を実現.

① 現画像状態:

- (A) 4秒に1回、画像・音声を収集
- (B) 録画・録音を実施
- (C) ウィンドウ上に現画像を表示
- (D) スピーカに現音声を再生

② 過去画像状態:

- (A) 4秒に1回、画像・音声を収集
- (B) 録画・録音を実施
- (C) ウィンドウ上に過去録画を表示
- (D) 音声の再生は「再生」ボタンが押された時のみ.

3.1.8 システムの状態(2/13)

(1) 当システム起動直後は、

(A) WXGAやXGAでは、**現画像状態(現在画像)**

(a) ウィンドウには、**現在の画像とボタン**を表示.

(b) ウィンドウのタイトルに以下を表示

(**status== 0**) d_yymmdd cxx_hhmmss

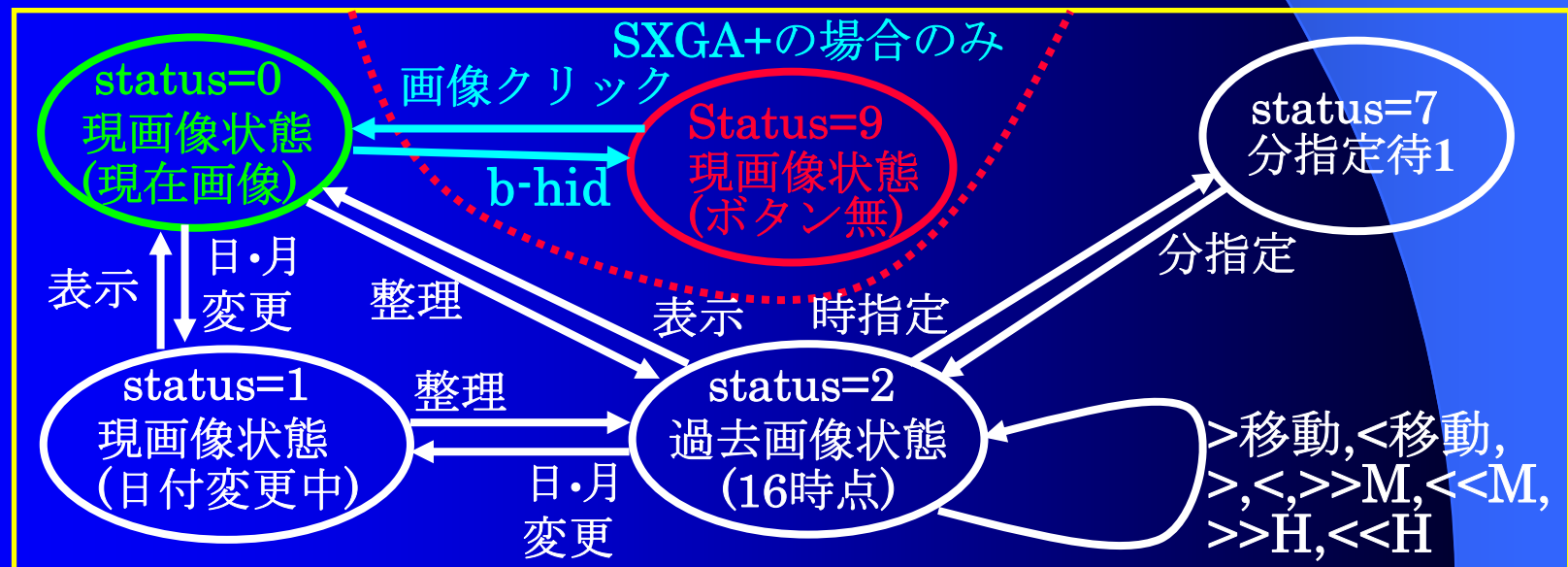
(B) SXGA+では、**現画像状態(ボタン無)**

(a) ウィンドウには、**現在の画像**を表示.

(b) ウィンドウのタイトルに以下を表示

(**status== 9**) d_yymmdd cxx_hhmmss

図14-1.
ボタン
操作等
と状態
遷移と
の関係



3.1.8 システムの状態(3/13)

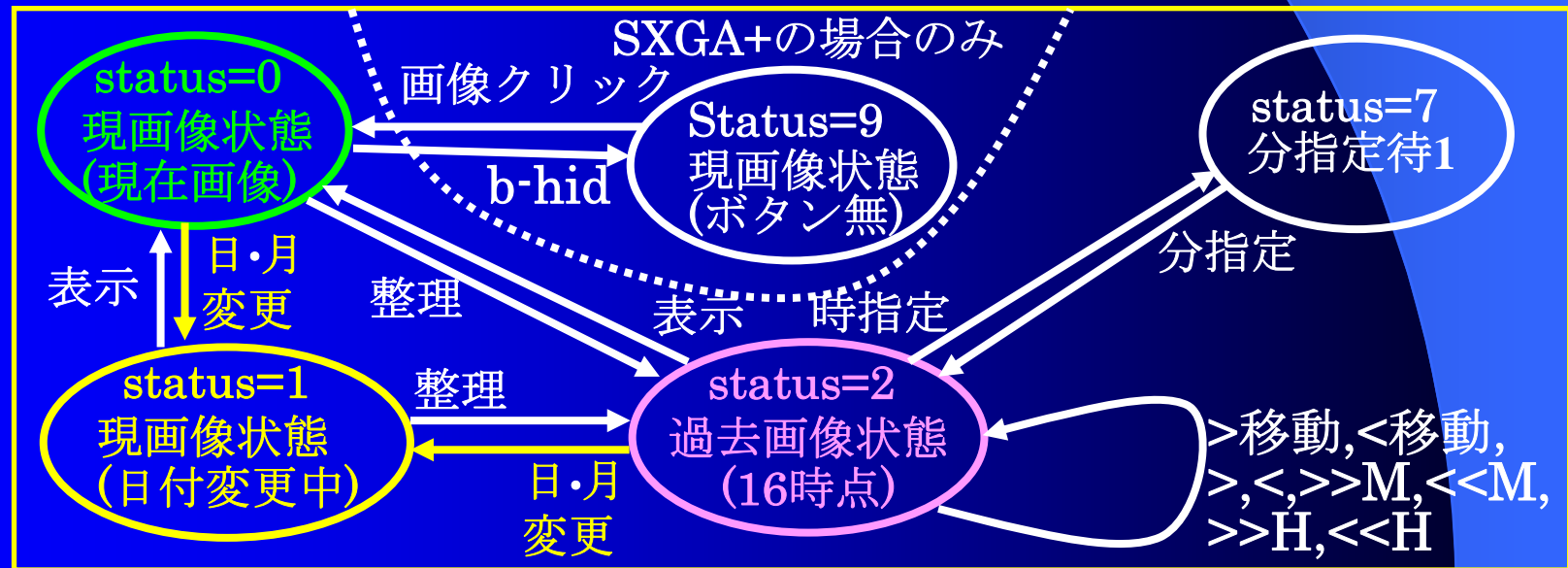
(2) 現画像状態(現在画像)又は過去画像状態(16時点)で、「>>日」、「<<日」、「>>月」、又は「<<月」(日・月変更)ボタンを押すと、
 現画像状態(日付変更中)となり、

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	< 移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	抵消	拡出	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	抵消

(a) ウィンドウには、
 現在の画像とボタン
 を表示。

(b) ウィンドウタイトル
 には、以下を表示。
 (status== 1)
 yymmdd

図14-2.
 ボタン
 操作等
 と状態
 遷移と
 の関係



3.1.8 システムの状態(4/13)

(3) 現画像状態(現在画像)又は現画像状態(日付変更中)で、

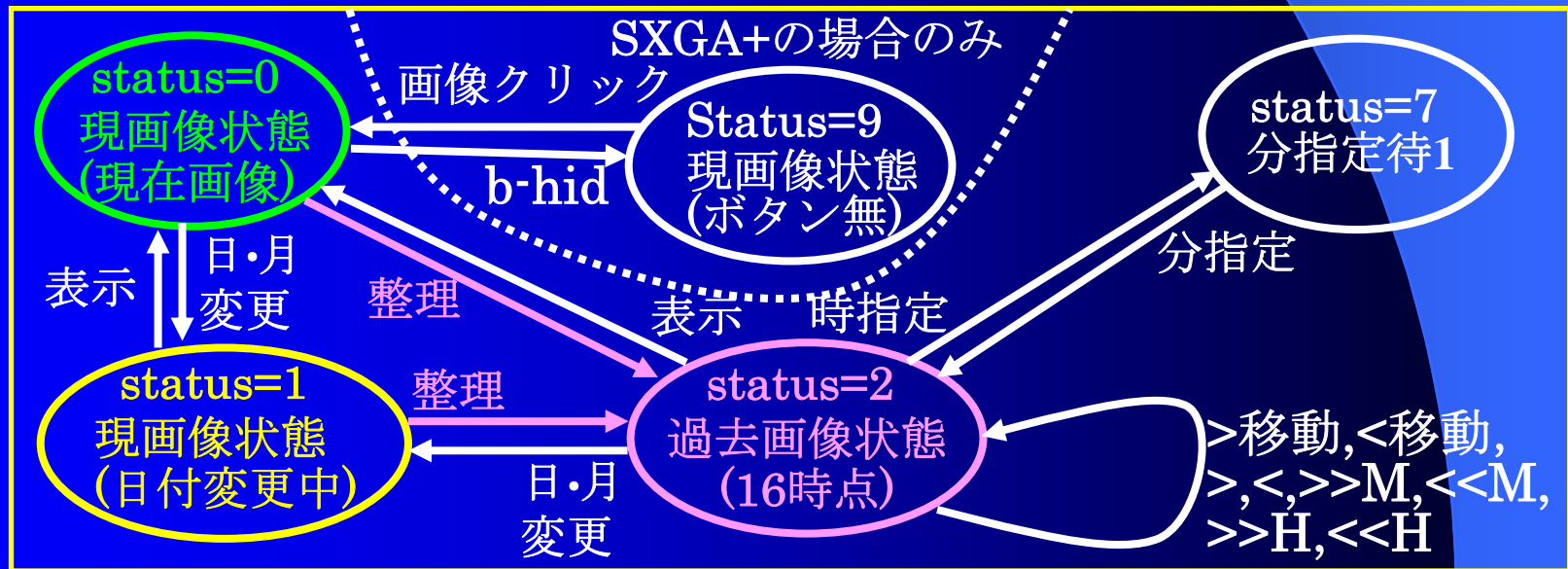
「整理」ボタンを押すと、過去画像状態(16時点)となり、

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	< 移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	拡大	拡大	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	差消

(a) ウィンドウには、指定した日の最も古い16時点の画像とボタンを表示.

(b) ウィンドウタイトルには (status== 2) d_yymmdd cxx_hh mmssを表示

図14-3. ボタン操作等と状態遷移との関係



3.1.8 システムの状態(5/13)

図3. 現画像状態
表示例
(WXGA ;
1280 × 800ドット)

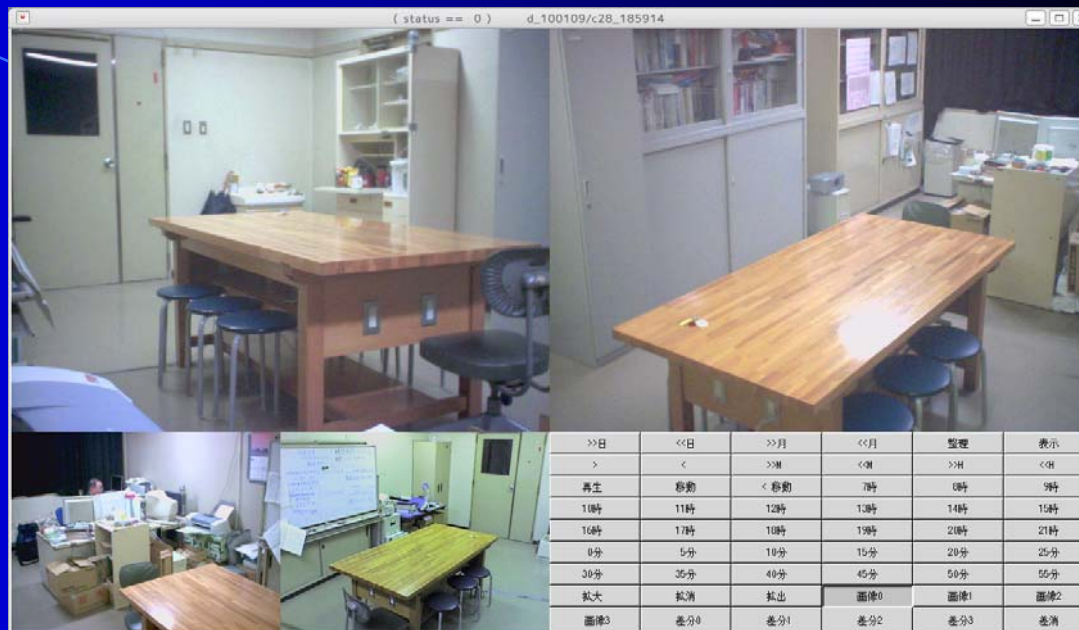


図4. 過去画像状態
(16時点)
表示例
(WXGA ;
1280 × 800ドット)



3.1.8 システムの状態(6/13)

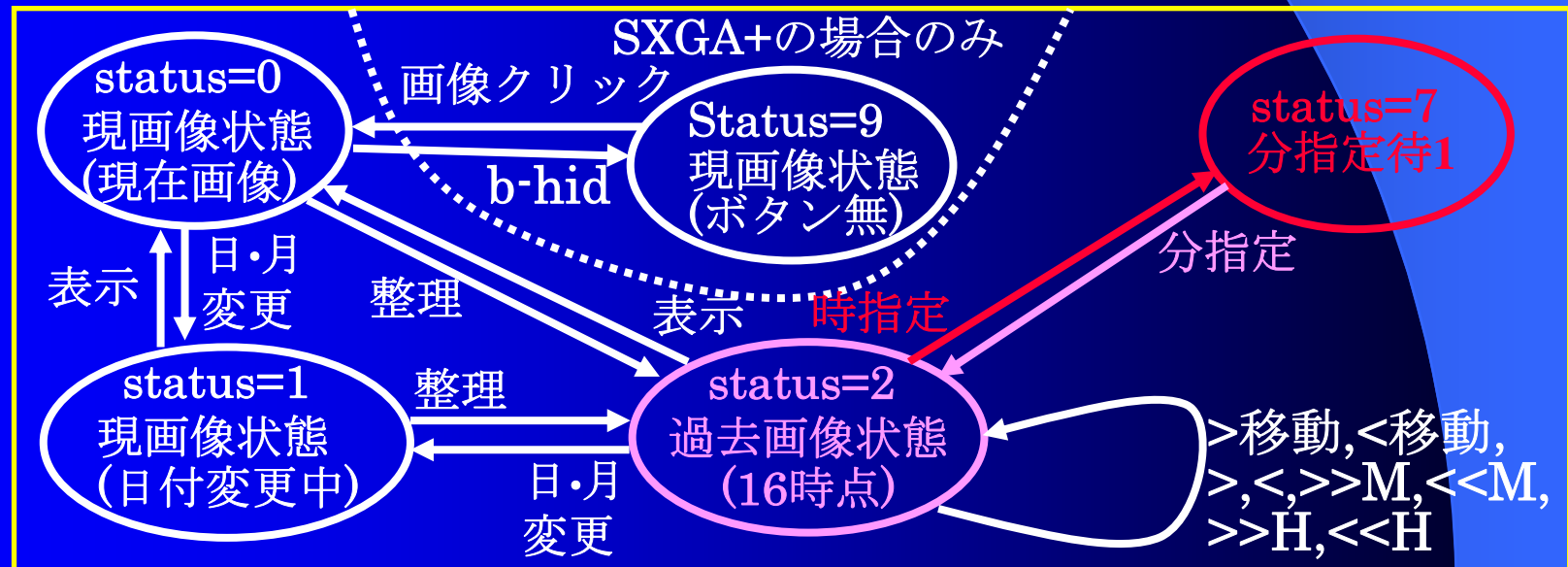
(4) 過去画像状態(16時点)で、「XX時」ボタンを押すと、(status=7)分指定待1となる。

(5) 分指定待1の状態では、「xx分」ボタンを押すと、指定の時間以降の16時点の画像を表示すると共に、(status=2)過去画像状態(16時点)となる。

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	< 移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	抵消	拡出	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	差消

16時点の画像を表示すると共に、(status=2)過去画像状態(16時点)となる。

図14-4. ボタン操作等と状態遷移との関係



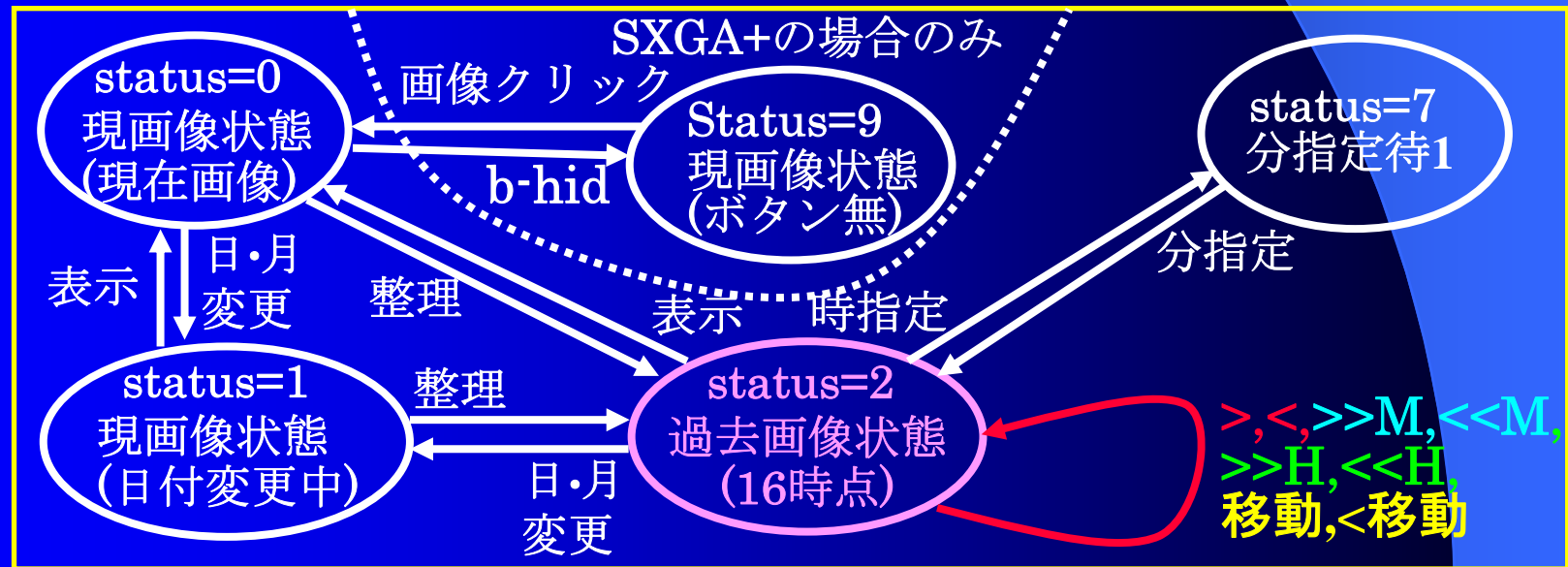
3.1.8 システムの状態(7/13)

(6) 過去画像状態(16時点)で、「>」or「<」ボタンを押すと、1画像分だけ新しいor古い16時点の画像を表示。「>>M」or「<<M」ボタンを押すと16画像分だけ新しいor古い16時点の画像を表示。「>>H」or「<<H」

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	<移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	抵消	拡出	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	抵消

ボタンを押すと、960画像分だけ新しいor古い16時点の画像を表示。「移動」or「<移動」を押すと一つ後or一つ前の障害時点以降の16時点の画像を表示。

図14-5. ボタン操作等と状態遷移との関係



3.1.8 システムの状態(8/13)

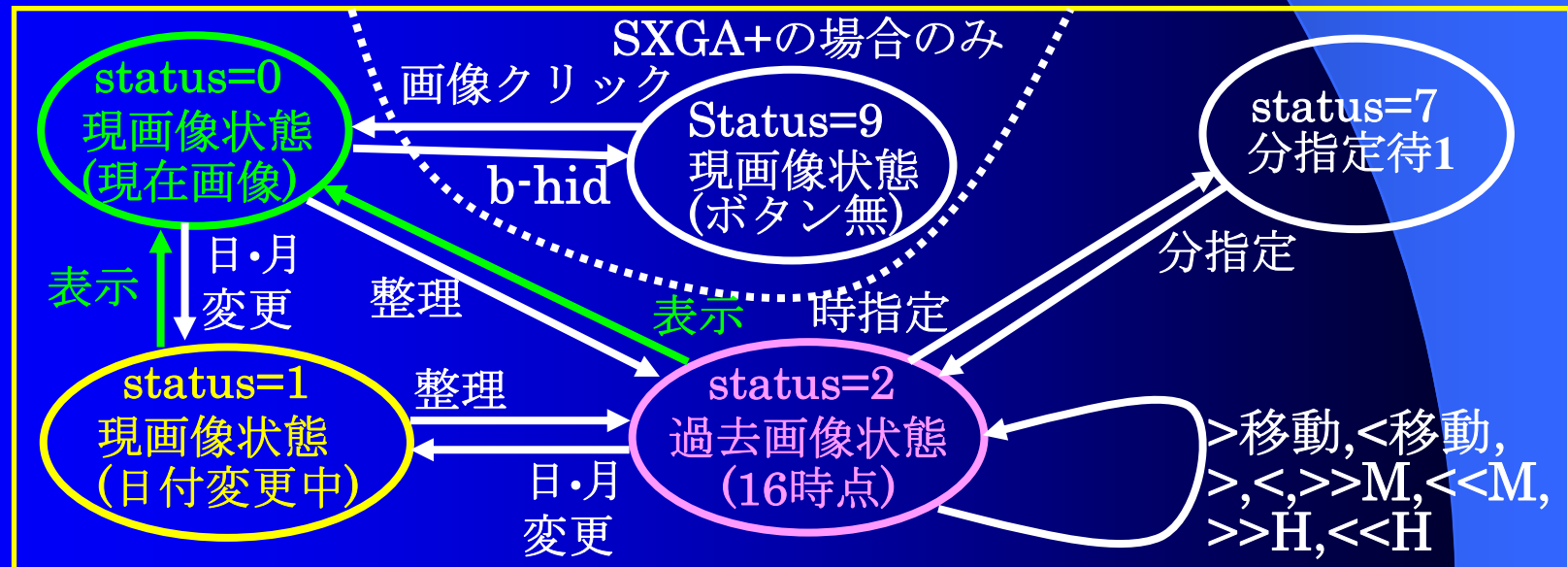
(7)現画像状態(日付変更中)又は過去画像状態(16時点)で、「表示」ボタンを押すと、現画像状態(現在画像)となり、

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	< 移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	拡大消	拡大出	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	差分消

(a) ウィンドウには、現在の画像とボタンを表示。

(b) ウィンドウのタイトルに以下を表示
(status== 0)
d_yymmdd
cxx_hhmmss

図14-6. ボタン操作等と状態遷移との関係



3.1.8 システムの状態(9/13)

Linux Conference 2010 (36/47)

(8)過去画像状態(16時点) → 画像クリック → 過去画像状態(1時点)

(9)過去画像状態(16時点) ← 画像クリック ← 過去画像状態(1時点)

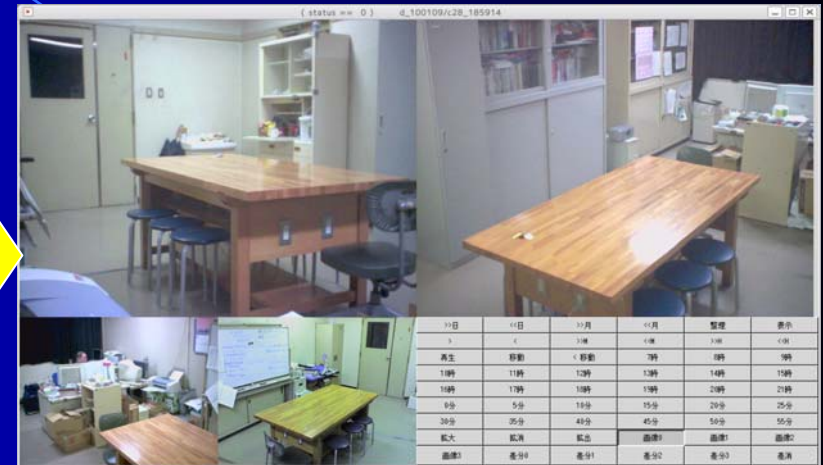
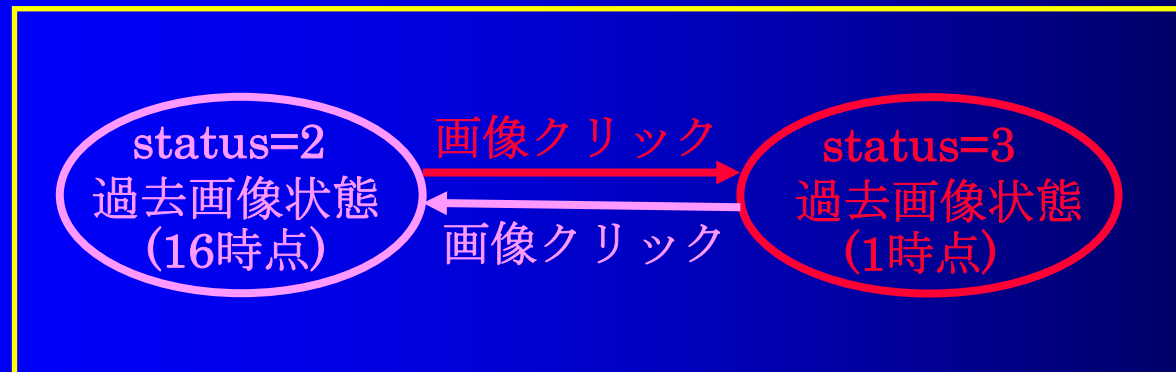


図4-1. 過去画像状態(16時点) 表示例(WXGA;1280×800ドット)

図4-2. 過去画像状態(1時点) 表示例(WXGA;1280×800ドット)

図14-7. ボタン操作等と状態遷移との関係

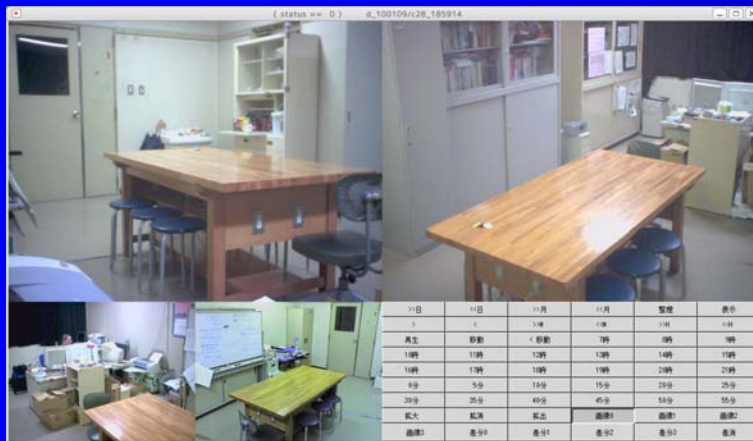


3.1.8 システムの状態(10/13)

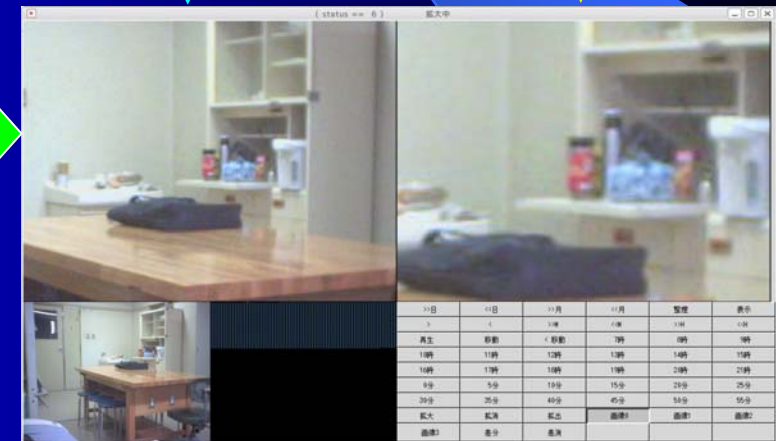
- (10) 過去画像状態(1時点) → 「拡大」 → (status=5)場所指定待
- (11) 場所指定待 → 画像クリック → 4倍と16倍に拡大中(status=6)
- (12) 拡大中 → 「拡大消」 → 過去画像状態(1時点)

4倍に拡大した画像

16倍に拡大した画像



「拡大」+
画像クリック



「拡大消」

図4-2. 過去画像状態(1時点)表示例(WXGA;1280×800ドット)

拡大前の画像

図5. 過去画像拡大中例(WXGA;1280×800ドット)

3.1.8 システムの状態(11/13)

Linux Conference 2010 (38/47)

(13) 拡大中 → 「抽出」 →

拡大された画像をファイル出力

(A) 4倍拡大のファイル名:

カメラ番号 画像の中心のx座標 y座標

k2yyymmdd_hhmmss_wxxxyyy.ppm

西暦下2桁 月月 日日 時時 分分 秒秒

(B) 16倍拡大のファイル名:

k4yyymmdd_hhmmss_wxxxyyy.ppm

3.1.8 システムの状態(12/13)

Linux Conference 2010 (39/47)

(14) 過去画像状態(16時点) → 「差分0~3」

→ 最も古い2画像の相関関係チェック結果

→ 「差消」 → 過去画像状態(16時点)

11日	10日	9日	8日	整理	途中
再生	移動	移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	縮小	表示	画像1	画像2	画像3
画像1	差分	差消			

図13. 相関関係チェック結果表示例
(解像度=1280×1024ドット)

3.1.8 システムの状態(13/13)

Linux Conference 2010 (40/47)

尚、トグルボタンでウィンドウの左上に表示する画像のカメラ番号を変更可

>>日	<<日	>>月	<<月	整理	表示
>	<	>>M	<<M	>>H	<<H
再生	移動	< 移動	7時	8時	9時
10時	11時	12時	13時	14時	15時
16時	17時	18時	19時	20時	21時
0分	5分	10分	15分	20分	25分
30分	35分	40分	45分	50分	55分
拡大	拡大	拡大	画像0	画像1	画像2
画像3	差分0	差分1	差分2	差分3	差分

「画像0」; 教室左前のカメラ

「画像1」; 教室左後のカメラ

「画像2」; 教室右後のカメラ

「画像3」; 教室右前のカメラ

3.1.9 dorodata内制御文概要

Linux Conference 2010 (41/47)

- (1) pass文: 基本認証データ.
- (2) cipn文: (n=0~3) 4台のカメラのipアドレス.
- (3) mail文: 障害検出メッセージの送り先メールアドレス.
- (4) msip文: メールサーバのIPアドレス.
- (5) msop文: メールサーバのポート番号.
- (6) fils文: 現ディレクトリがあるHDDの装置名.
- (7) disp文: ディスプレイの解像度.
(0=1280×960, 1=1280×720, 2=960×720)

3.1.10 プロセス構成

Linux Conference 2010 (42/47)

当システム(NCSFCT)は、以下の5つのプロセス等の
並行実行により実現する。

- (1) 親プロセスのタイマー駆動ルーチン: 関数getg726で、
250msec毎に音声を読み込んで、1秒毎に、適宜、保存。
- (2) 第一子プロセス: 関数getgamenhで、カメラより4秒
に1回画像を読み込んで、適宜、ファイルへ保存。
- (3) 親プロセスの画像表示ルーチン: 関数getgamenで、
、上記(2)と同期をとって、画像を表示。
- (4) 第二子プロセス: 関数erasefile内で、60秒に1回、
画像の格納装置のエリア使用率をチェック。90[%]を超える
場合は、最古年月日のファイルを消去。
- (5) 第三子プロセス: 格納済の音声ファイルを適宜、再生。

4. 試用による評価結果(1/2)

Linux Conference 2010 (43/47)

(1) 以下の点で担任業務に非常に役に立った.

(A) 学生の登校・帰宅時間や教室の消灯等を教員室より確認できた.

(B) 席表との照合により、出欠の確認を教員室よりできた.

(C) 画像と音声により、授業の進行状況が確認でき、休み時間での学生への連絡等が能率的に行えた.

(D) カメラに悪戯されることは無かったが、先生が誤ってカメラの電源を抜いてしまった時に、警報をメールで受信して、その証拠(画像)を得ることができて、迅速な対応がとれた.

4. 試用による評価結果(2/2)

Linux Conference 2010 (44/47)

(2) 使用ファイル容量等は以下の通り.

約2年分のデータをメンテナンスフリーで保持可.

(A) 4秒当り216KB. 1日12時間分で2.332[GB].

1カ月当り22.1日分で51.65GB.

よって1.5TBのHDDで26.2カ月分の録音・録画が可.

(3) 10MbpsのLANでは、回線使用率 \div 4.32[%]

(B) 4秒間で40Mbpsの転送速度で4秒間に216KB.

$$\begin{aligned} \text{回線使用率} &= (216 \times 8 \times 1000 / (40 \times 1000,000)) \times 100 \\ &= 4.32 [\%] \end{aligned}$$

(4) 平均CPU使用率 < 5[%]

(CPU=Intel Duo P9500; 2.53GHzの場合)

5. ソース等の公開場所等

Linux Conference 2010 (45/47)

(1) 作成規模: 5Kstep (C言語記述)

(2) 公開用ホームページ: (GPL)

[http://www.ibaraki-ct.ac.jp/ece/yas/
ncsfct/index.htm](http://www.ibaraki-ct.ac.jp/ece/yas/ncsfct/index.htm)

doro003.c : 当システム(NCSFCT)のソースプログラム

ccdoro : ソースプログラムのコンパイル用コマンドファイル

decode: 音声データデコーダ(文献[13]のデコーダの
ロードモジュール.)

dorodata : NCSFCTの制御文例

env.pdf : NCSFCT用VineLinux5の初期設定方法例
(ImlibやGTK+用設定例)

dlan.pdf: NCSFCT用無線LAN等の初期設定例

6. 今後の予定

今年度末までに、Ubuntu Linux等への移植方法も検討し、上記Webサイトでの成果報告を予定している。

7. 謝辞

素晴らしいLinuxを実現して下さったLinus Torvalds様と仲間の皆様、並びにGPLの元で種々のプログラムや貴重な情報を提供して下さったVineLinuxメーリングリストの皆様に、この場をお借り致しまして深謝致します。

又、当システムで「サンマイクロシステム株式会社によって、所有権が放棄され無償で一般に広く公開されたライセンス(パブリックドメイン)で開放されている文献[13]のデコーダ」を使用している、ことにつきましてサンマイクロシステム株式会社に深謝致します。

[完]

- (1) `gtk_widget_ref()`; オブジェクトのリファレンスカウンタを1に設定して、ウィジェット除去可とする。
- (2) `gtk_container_remove()`; コンテナからウィジェットを除去する。
- (3) `gtk_container_add()`; コンテナにウィジェットを乗せる。
- (4) `gtk_widget_show_all()`; ウィジェットをウィンドウ上に表示する。
- (5) `gtk_toggle_button_set_active()`; トグルボタンを、押or非押の状態に設定する。
- (6) `gtk_window_set_title()`; ウィンドウのタイトルバーに文字を表示する。
- (7) `gtk_main_quit()`; プログラムを停止する。
- (8) `gtk_set_locale()`; ロケールの設定を行う。
- (9) `gtk_init()`; GTK+の初期化を行う。
- (10) `gtk_rc_parse()`; スタイルの初期設定を行う。
- (11) `gtk_window_new()`; 新しいウィンドウを作る。
- (12) `gtk_window_set_default_size()`; ウィンドウの大きさを設定する。
- (13) `gtk_drawing_area_new()`; ウィンドウ用の描画エリアを作成する。
- (14) `gtk_drawing_area_size()`; ウィンドウ用の描画エリアの大きさを設定する。
- (15) `gtk_signal_connect()`; イベント発生時のコールバック関数を登録する。
- (16) `gtk_widget_set_event()`; ウィジェットにイベントのマスク等を設定する。
- (17) `gtk_button_new_with_label()`; ラベル付きのボタンを生成する。

- (18) `gtk_toggle_button_new_with_label`; ラベル付きのトグルボタンを生成する。
- (19) `gtk_hbox_new()`; 水平パッキングボックスを生成する。
- (20) `gtk_vbox_new()`; 垂直パッキングボックスを生成する。
- (21) `gtk_box_pack_start()`; パッキングボックスにウィジェットを乗っける。
- (22) `gtk_container_add()`; コンテナにウィジェットを乗っける。
- (23) `gtk_timeout_add()`; タイマー起動のコールバック関数を登録する。
- (24) `gtk_main()`; イベントを待ち合わせるためのメインループに入る。

付録2. 使用gdkのライブラリの概要

- (1) `gdk_imlib_init()`; Imlibの初期設定を行う。
- (2) `gdk_window_get_pointer()`; マウスのボタンがクリックされた時のマウスポインタの座標を、指定のエリアに設定。
- (3) `gdk_imlib_render()`; 指定のピクスマップ上に、描画を行う。
- (4) `gdk_pixmap_unref()`; ピクスマップのレファレンスカウンタを1減らす。0になれば、ピクスマップを開放する。
- (5) `gdk_imlib_move_image()`; 画像データをピクスマップ上に移動する。
- (6) `gdk_draw_pixmap()`; 描画されたピクスマップをウィンドウ上に表示する。
- (7) `gdk_imlib_kill_image()`; 画像データを開放する。
- (8) `gdk_imlib_create_image_from_data()`; 画像データをピクスマップ形式へ変換。