

K-001

## HTML5 を用いた動画ヘルプの作成とネット共有手法 Creating and sharing method of Animation Help by HTML5

松崎 広起<sup>†</sup> 納富 一宏<sup>†</sup>  
Hiroki Matsuzaki Kazuhiro Notomi

### 1. はじめに

オフィス系アプリケーションの操作手順の習得は、静止画と文字で構成されたマニュアルを参照するのが一般的な方法であるが、直感的な理解を促すとは言い難い。そこで、我々は直感的な理解を促す動画ヘルプシステム的设计・開発を行った<sup>[1]</sup>。開発した動画ヘルプシステムは、カーソルポインタもしくは別のポインタを用いて、オートパイロット的な自動操作を行うことによって、実際の操作手順を追うことで直感的な理解を促すシステムである。しかし、開発したシステムを実際に運用する上で、操作手順の録画から利用者への受け渡しまでに時間がかかることが問題点であった。そのため、本研究では、作成した動画ヘルプの簡易的な共有を目的として、Web 上での動画再生を可能とするシステム的设计・開発を行った。

本稿では、実装した動画ヘルプシステムの構成と動作確認した結果について述べる。

### 2. システム構成

本システムは、マウス操作やデスクトップのスクリーンショットを撮影・保存する録画用と、録画された動画を Web 上で再生する再生用の 2 種類のアプリケーションで構成されている。

#### 2.1 本システムにおける動画の録画方法

本システムでは、操作中のマウスの移動とデスクトップ画面の更新を中心に録画を行なっている。最初にデスクトップ画面の画像を取得する。その後、操作中のマウスの座標を記録するが、その間のデスクトップ画面は取得しない。マウスでのクリック操作が行われた場合に、デスクトップ画面の取得を行い、クリック操作と取得したデスクトップ画面を記録する。記録されたデータの例を表 1 に示す。

表 1 記録されたデータの例

命令文	意味
Move,200,200,100	(200,200)に移動後、 100ms 待機
Click,190,190,50,1.bmp	(190,190)をクリック後、50ms 待機。 クリック直後のデスクトップ画面は 1.bmp である

動画の描画には HTML5 で新しく策定された Canvas を用いた。Canvas とは、Web ページ上の矩形領域に JavaScript を使用することで、任意の図形を描画できる。HTML5 をサポートした Web ブラウザであれば、利用するコンピュータ環境に関係なく視聴可能であるというメリットがある。

#### 2.2 録画アプリケーション

録画アプリケーションが、動画ヘルプを作成する流れを

<sup>†</sup> 神奈川工科大学大学院工学研究科 Graduate School of Engineering, Kanagawa Institute of Technology

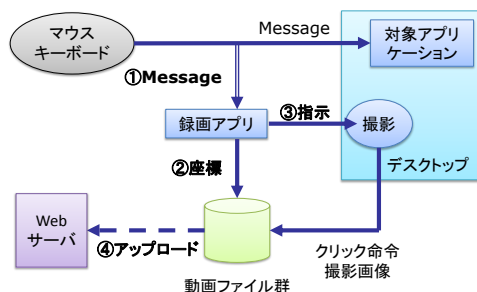


図 1 録画アプリケーションの構成図

図 1 に示す。本システムは、録画開始ボタンが押されることで、マウス操作に対して、グローバルフックをかける

(図 1 ①)。グローバルフックとは、OS からアプリケーションに伝達されるメッセージを取得する技法であり、アプリケーションがバックグラウンドであってもメッセージを受け取ることができる。グローバルフックからメッセージを受け取った上で解析を行い、操作中のマウスの座標を記録する(図 1 ②)。また、録画中にマウスでのクリック操作が行われた場合、デスクトップのスクリーンショットの撮影を行う(図 1 ③)。録画終了ボタンが押されることで、記録したマウス操作のログや撮影したデスクトップのスクリーンショットを動画ファイル群として、Webサーバへアップロードを行う(図 1 ④)。

#### 2.3 再生アプリケーション

動画の再生は、Web 上で行う。再生アプリケーションの構成図を図 2 に示す。

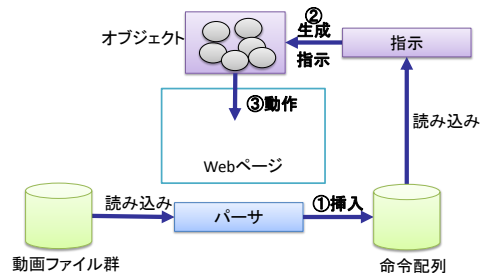


図 2 再生アプリケーションの構成図

利用者から再生指示を受け取ると、まずは動画ファイル群からマウス操作のログを読み込み、正規表現による簡易なパーサに通し、命令群に変換する(図 2 ①)。次に、命令群を先頭から順次読み込むことで、命令文に基づいてオブジェクトの生成や指示を行う(図 2 ②)。オブジェクトの例として、吹き出しの表示、クリックエフェクトの描画、マウス軌跡の描画といった種類があり、Webページの矩形領域内で動作する(図 2 ③)。命令文には待機時間が存在し、設定した値に基づいて待機した後、次の命令文を読み込む。

### 3. 実験

本システムの動作確認と、作成した動画の共有にかかる時間の計測を行った。本稿では、動画の共有にかかる時間の目安として、操作説明を動画に変換する時間とその動画の容量を比較した。人がマウスやキーボードの操作を行うと、クリックや入力時間などにばらつきが出るため、録画開始および終了操作を除く手動操作の自動化を行った。比較対象として、eラーニングオーサリングソフトウェアである Adobe Captivate 5 と、Web 上で利用可能な組み合わせとして有力なコーデック 3 種類を対象とした<sup>[2]</sup>。実験で対象とした操作説明の概要を表 2 に示す。

表 2 操作説明の概要

ソフトウェア	Microsoft Excel 2010
操作内容	ソルバーの簡単な説明
操作時間	1 分 16 秒
マウスクリック回数	32 回
ウィンドウサイズ	852×658

#### 3.1 動作確認

表 3 に示した機種や Web ブラウザを対象に Web 上で動作する再生アプリケーションの動作確認を行った。

表 3 動作確認結果

機種 / Web ブラウザ	バージョン	動作
iPhone	4	○
Firefox	4.01	○
Google Chrome	11.0.696.68	○
IE	8.0.6001.18702	○
Opara	11.11	○
Safari	5.0.5	○

動作確認を行った機種と Web ブラウザでは、正常に動作することを確認した。しかし、iPhone での再生速度は非常に遅く、本来の再生時間よりも長く再生されていることが確認された。

#### 3.2 動画変換実験

録画中の操作説明が終了してから、Web 上にアップロード可能なファイル形式が完成するまでの時間を計測した。Adobe Captivate 5 については、録画停止ボタンを押してから編集画面に移行する時間と、編集内容をファイルに書き出す処理を合計した時間とした。コーデックについては、録画されたファイルを読み込み後、各コーデックを用いてエンコードが開始される時間から終了までの時間とした。録画の設定は色数 32bit、30fps とし、キャプチャソフトには AmaRecCo<sup>[3]</sup>を用いた。エンコードには Miro Video Converter を使用した。各方法で変換にかかった時間を表 4 に示す。

表 4 動画変換にかかった時間

変換方法	時間[秒]
本システム	0.5
Adobe Captivate 5	11.3
H.264 + AAC + MP4	93.3
Theora + Vorbis + Ogg	180.0
WebM	206.3

本システムの方式では、変換にかかる時間が小さいことが確認された。

#### 3.3 容量比較実験

操作説明中の画面を各方法で録画し、容量の比較を行った。なお、Adobe Captivate 5 はデフォルトの設定で録画を行った。コーデックについては、キャプチャソフトを用いて録画し、各コーデックを用いてエンコードを行った。実験結果を表 5 に示す。

表 5 容量の比較

変換方法	容量[kB]
Adobe Captivate 5	1116
H.264 + AAC + MP4	1868
本システム	2549
Theora + Vorbis + Ogg	6326
WebM	8058

本システムでは、容量が小さくならないことが確認された。

### 4. 考察

動作確認から、今回実験を行った機種 Web ブラウザで動画が正常に再生されることを確認した。しかし、環境によって、再生時に負荷が大きくなった。これは、各環境における JavaScript の実行速度が大きく影響していると考えられる。

動画変換実験の結果から、本システムは比較対象よりも動画変換にかかる時間が少ないことが示された。要因として、本システムで採用している動画形式は、マウス操作のログとスクリーンショットの組み合わせだけで構成されているためだと考えられる。容量比較実験の結果から、本システムで採用している動画形式では、容量が比較的大きくなるため、動画のアップロードおよびダウンロードに時間がかかることが示唆された。要因として、マウスをクリックする度にスクリーンショットを保存する本システムの方法は、スクリーンショット間の時間軸差分圧縮ができないため容量が大きくなったと考えられる。

### 5. まとめ

本稿では、Web 上で動画再生および編集を行うシステムを設計・開発し、システムの動作確認やスペックについて実験を行った。その結果、録画した動画の変換にかかる時間は短い、容量が比較的大きくなることが示された。また、視聴環境によって、再生の負荷が大きく異なることが判明した。

今後は、動画ファイル群の容量削減を行うとともに、操作説明の再現性の実験や人の手による共有にかかる時間を計測する実験を行う予定である。

#### 参考文献

- [1] 松崎広起, 西尾高典, 納富一宏, “Windows 初心者向けアニメーションヘルプシステムの UWSC による試作と方式の検討”, 画像電子学会第 38 回年次大会 S4-5 (2010).
- [2] 入門 HTML5, PP.91-100, 株式会社オライリー・ジャパン (2011).
- [3] アマレココ公式ホームページ, <http://amamaman.hp.infoseek.co.jp/>