

# 操作ログを用いた着脱可能 Web チュートリアル の作成支援手法

湯浅 駿平<sup>1,a)</sup> 小形 真平<sup>1,b)</sup> 小林 一樹<sup>1,c)</sup> 岡野 浩三<sup>1,d)</sup>

**概要:** Web チュートリアルは、Web ページの操作学習対象に直接埋め込まれたチュートリアルである。Web チュートリアルでは、従来のスライド形式などのチュートリアル資料と比べ、操作者が資料と Web ページを見比べる必要がないことから、操作学習に対し有利であることが期待される。しかし、導入には Web ページの編集が必須であり、開発知識の乏しい資料作成者には容易でない。そこで、本稿では、資料作成者が効率的に Web チュートリアルを作成でき、既存の Web アプリケーションに容易に導入できるように、着脱可能 Web チュートリアルおよび操作ログを用いたその作成支援手法を提案する。手法を提案する。本研究では、つぎの3つの機能を提供するブラウザの拡張機能が実現される。1) 資料作成者の操作ログを記録する。2) 操作ログから Intro.js による Web チュートリアルをほぼ生成する。3) 生成したチュートリアルを Web アプリケーションに埋め込む。提案手法と電子文書形式による従来法との比較評価の結果、Web チュートリアルが操作の正確性を向上でき、提案手法が作成効率を向上できる見込みを得た。

**キーワード:** 操作学習支援, Web チュートリアル, Web ブラウザ, 拡張機能, 操作ログ

## Support Method for Creating Pluggable Web Tutorial with User-Operation Log

SHUNPEI YUASA<sup>1,a)</sup> SHINPEI OGATA<sup>1,b)</sup> KAZUKI KOBAYASHI<sup>1,c)</sup> KOZO OKANO<sup>1,d)</sup>

**Abstract:** Web tutorials are tutorials embedded into Web pages as learning targets of user-operation. The tutorial is expected to offer an advantage in operation learning because it does not require learners to compare Web pages with any separated tutorial materials in traditional formats such as slides. However, introducing the Web tutorial is difficult for tutorial creators who do not always have knowledge of development because the tutorial requires them to edit existing Web pages directly. In this paper, a pluggable Web tutorial and a support method of creating it with user-operation log are proposed so that such creators can create the Web tutorial efficiently and introduce it to an existing Web application easily. In this research, a browser extension providing three functions as follows is actualized: recording tutorial-creator-operation logs to a Web application; mostly generating a Web tutorial in Intro.js from the logs; embedding the generated tutorial into the Web application. As the result of experiments comparing the proposed method with a traditional method utilizing an electronic document, prospects that the proposed method can offer two advantages as follows were obtained: a Web tutorial enhances the correctness of user-operation consistent with the tutorial; the proposed method makes tutorial creation efficient.

**Keywords:** Operation learning support, Web tutorial, Web browser, Extension, Operation log

<sup>1</sup> 信州大学  
Shinshu University

a) 14tm542c@shinshu-u.ac.jp

b) ogata@cs.shinshu-u.ac.jp

c) kby@shinshu-u.ac.jp

d) okano@cs.shinshu-u.ac.jp

## 1. はじめに

近年、急速な IT 技術の発展に伴い、様々な場面で多く  
の人が利用している Web アプリケーションは、社会イン

フラの一部として機能している。そのため、アプリケーションのユーザ同士がアプリケーションの操作について教える、教わる関係になる場合が多く存在する。例えば、教育現場では学習者の理解を促進するために Web アプリケーションが活用される。そこでは、学習者は授業という限られた時間内で授業内容と共に Web アプリケーションの操作も理解することを求められ、教育者もまた、学習者のために利用するアプリケーションの操作説明資料を作成する必要がある。そのため、双方に少なからず負担が生じる。さらに、教育者は理解の追い付かない学習者に対し個別に指導する時間を取られ、結果、授業の進行が遅れるなど重大な問題が生じる可能性がある。このことから、操作者が初めてアプリケーションを操作する場合に効率的に操作方法を学習できる支援が重要である。

従来では文章や画像、動画、音声など多様な形式を用いてスライド形式や文書形式などの操作説明資料が作成され、操作者に提供することで操作方法の理解を支援している。そのひとつである Web チュートリアルは、Web アプリケーションに直接埋め込んだチュートリアルであり、実際の操作箇所をハイライトして操作の指示や説明をステップバイステップで表示する [1]。そのため、Web チュートリアルでは、操作者は説明と操作対象のアプリケーションを見比べる負担が小さく、直感的に理解しやすく感じると考えられる。

しかし、チュートリアルを既存の Web アプリケーションに手で埋め込むことは容易ではない。とはいえ、Web チュートリアル以外の紙媒体や電子文書形式の資料を別途作成する場合、画像の切り出しや図形の挿入、位置調整などに時間を割く必要がある。

そこで、本稿では資料作成者の効率的な Web チュートリアル作成と容易な Web チュートリアルの導入を支援するため、下記の 2 つの特長を持った作成支援手法を提案する。

- (1) 資料作成者が操作説明資料を効率的に作成できるように、Web アプリケーション上で一連の操作を行うことで Web チュートリアルの雛形を自動生成できる。この雛形は操作の説明文を付加することで完成する。
- (2) Web アプリケーションへの Web チュートリアルの導入が容易になるように、Web チュートリアルを Web ブラウザの拡張機能から汎用的かつ自動的に着脱できる

また、提案手法を実現するツールを Google Chrome の拡張機能 [2] として試作し、大学で実際に運用されている、学生向けの求人検索システムに適用することで評価を行った。本評価では、文章と画像を用いた電子媒体の操作説明資料と、試作ツールによる Web チュートリアルの間で、資料作成者の作成効率と操作者の操作効率を比較した。

## 2. 諸概念

### 2.1 ブラウザ拡張機能

ブラウザ拡張機能とは、ブラウザに独自の機能を拡張するための枠組みであり、複数のブラウザがサポートしている。たとえば、Google Chrome では、“Extensions” と呼ばれ、その拡張機能の大きな特徴として HTML, CSS, JavaScript などのウェブ標準だけを利用して作成できることがある。また、拡張機能から Chrome 内部の情報や機能にアクセスすることも可能である [3]。

### 2.2 Intro.js

Intro.js[4] は実際に操作を行いながら操作に対する指示を受けられる視覚的な Web チュートリアルを Web ページに追加する OSS の JavaScript ライブラリである。このライブラリを読み込み、HTML に属性を付加するだけで簡単に Web チュートリアルを導入できる。図 1 に示すように、HTML 要素をハイライト表示することで操作箇所を注視させることが可能で、次の操作や前の操作への誘導も容易である。本研究では、操作者にステップバイステップの Web チュートリアルを提供するために、本 OSS を試用する。

Intro.js では、任意の HTML 要素に対し以下の 2 つの属性を付加することで任意の HTML 要素への説明文の追加を行う。

(1) data-intro=“任意の説明文”

(2) data-step=“任意のチュートリアルステップ番号”

また、“IntroJs.start()”によってチュートリアルの開始を制御できる。

Intro.js を利用して HTML ソースが手元にある Web ページに対し Web チュートリアルを導入する場合、下記の手順が必要となる。

- (1) 任意の HTML 要素に対する説明文を作成する
- (2) (1) の要素に対し、既定の属性を追加する
- (3) (1) から (2) をステップの数だけ繰り返す
- (4) IntroJs.start() を Intro.js が存在するページ上の HTML 要素に追加する



図 1 Intro.js の HTML 要素への適用例

### 3. Webアプリケーションの操作説明資料における問題点

Webアプリケーションの操作を説明する資料は、文章や画像、動画、音声など多様な形式や媒体によって作成されるが、操作者の見やすさや作成者の作りやすさを考慮し選択すべきである。本章では、基本的な操作説明資料の形式や媒体の差について触れた後、その作成における問題点について述べる。

#### 3.1 資料の形式や媒体の差が操作者の操作に及ぼす影響

Webアプリケーションの操作について説明する際、一般的には紙の文書や電子媒体のスライド、またはWebページで文章を用いて記述される。また、画像や図形を加えることで操作を行う画面や状況をより的確に操作者に伝えることが可能である。しかし、Webアプリケーションの操作手順はステップバイステップであるため、操作者がこのような文書やスライドによる資料を利用する場合、対象のWebアプリケーションと操作説明資料を見比べながら内容を確認し、操作しなくてはならない。この動作は操作を行う箇所を間違えたり、入力値を忘れてしまったりと誤操作、誤入力を誘発してしまう可能性がある。

また、操作の手順と内容を一度に直感的に把握させる方法として動画を用いた説明資料があるが、操作者は、動画の内容を全て覚えた上で操作するか、再生を止めつつ随時操作する必要がある。この場合、手順や内容を把握しやすくなるが、やはり操作者は操作説明資料と対象アプリケーションを見比べなくてはならない。

このように、従来では操作画面と操作説明資料を見比べながら操作を行わなくてはならない問題があるが、この問題を解決する方法のひとつとして、実際の画面で目的的操作を行うと同時に操作に関する説明を受けることができるWebチュートリアルがある。そのため、本研究ではWebチュートリアルに着目する。

#### 3.2 文書またはスライド形式の操作説明資料を作成する際の問題点

操作説明資料において一般に用いられる文書形式で、分かりやすい資料を作成するため、詳細な説明を加えていくと資料の作成時間は必然的に増えていく。例えば、画像を用いた図2の操作説明資料について作成手順を考えると、1つの操作に対する説明だけでも最低限以下の5つの手順が必要になる。

- (1) 操作画面のスクリーンショットを撮影する
- (2) スクリーンショットを挿入する
- (3) スクリーンショットを加工する
- (4) スクリーンショットに図形を挿入する
- (5) 説明文を記入する

このように、複数の操作について説明を行う必要があり、かつ操作者の分かりやすさを重視し、手をかける場合、資料作成には多くの時間と労力が必要となる。教育現場においては教員が資料作成に時間を取られると、授業の進行に影響を及ぼす危険性があるように、資料作成者が資料を効率的に作成できることが望ましい。



図2 画像を用いた操作説明資料の例

#### 3.3 Webチュートリアルによる操作説明資料を作成する際の問題点

第1章や3.1節で述べてきたように、Webチュートリアルは操作者に効果的な操作学習を提供できると期待される。しかし、資料作成者が対象のアプリケーションに対し直接変更を加えてWebチュートリアルを導入することは、Web技術や対象アプリケーションの開発仕様などの専門知識を要するため非常に困難と言える。

### 4. 提案手法

#### 4.1 提案手法の概要

Webアプリケーションの操作に対する説明資料は操作者にとって直感的に分かりやすく、かつ資料作成者にとって作りやすいことが求められる。また、前章で述べたように、Webチュートリアル作成者がWebチュートリアルを完成されたアプリケーションに対し手動で導入することは困難であることから、Web技術に関する知識のない一般的な資料作成者であっても、容易にWebチュートリアルをアプリケーションに対し導入できることが望ましい。そこで、本稿では資料作成者が操作者に期待する操作を、対象のWebアプリケーション上で行うことでWebチュートリアルの雛形を自動生成する方法と、生成した雛形をブラウザの拡張機能を利用してWebアプリケーションに着脱する方法からなるWebチュートリアル作成支援手法を提案する。

提案手法は下記の3つの機能を、Webブラウザの拡張機能により実現する。

- (1) 資料作成者の操作ログを記録する

- (2) 操作ログから Web チュートリアル雛形 (後に資料作成者が編集することで完成する) を自動生成する
- (3) 完成したチュートリアルを Web アプリケーションに自動的に埋め込む

図 3 は提案手法の全体像とその動作を示したものである。資料作成者と操作者はそれぞれ拡張機能を扱える Web ブラウザを導入していることを前提とする。

資料作成者は Web ブラウザに導入した拡張機能を用いて操作ログから Web チュートリアル雛形を作成する。この際、Web チュートリアルで必要となる各ステップにおける説明文を拡張機能から入力することでチュートリアル雛形はチュートリアルデータとなり、完成する。

一方、操作者は Web ブラウザで目的の Web アプリケーションを開き、拡張機能を用いて Web チュートリアルを閲覧する。この拡張機能により、資料作成者が作成したチュートリアルデータが Web アプリケーションに自動で埋め込まれ、操作者は Web チュートリアルを閲覧しながら目的の操作を行うことができるようになる。

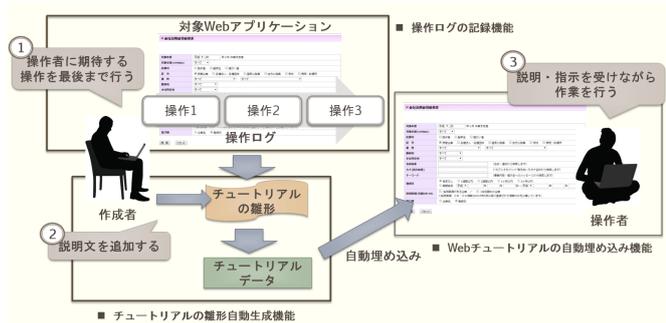


図 3 提案手法の全体像

## 4.2 操作ログを用いた Web チュートリアル雛形の自動生成手法

### 4.2.1 提案手法における操作ログとその生成

本稿では、資料作成者が説明したい操作箇所を特定するためのものとして、操作ログを取得する。操作ログとは、図 4 で示すように“資料作成者が操作した HTML 要素の DOM ツリーパスのリスト”と定める。そして、その HTML 要素は、マウスクリックイベントの発生源を指す。尚、属性の有無によらず HTML 要素を特定するために、提案手法では DOM ツリーを取得し、そのパスをテキストとして保持する。DOM ツリーパスの取得と、テキスト化の概要を図 5 に示す。

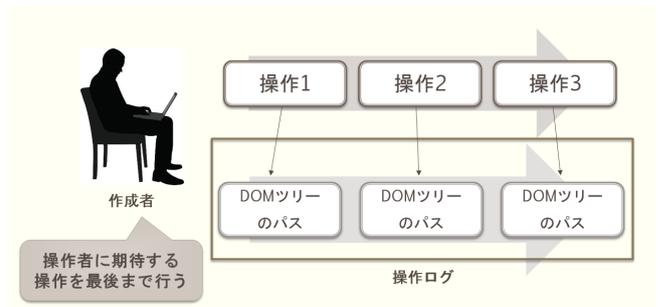


図 4 提案手法における操作ログ生成の概要

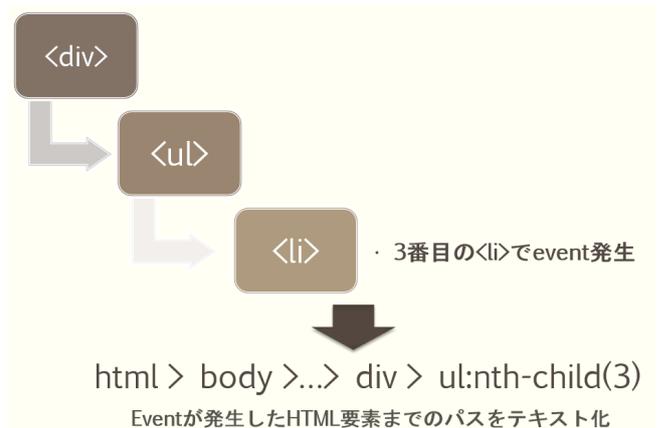


図 5 DOM ツリーの取得とそのテキスト化の概要

### 4.2.2 Web チュートリアル雛形生成

提案手法で扱う Web チュートリアルは、任意の 1 つの HTML 要素における操作について説明した“ステップ”が複数集まり、“チュートリアル”を形成している。1 つのステップを構成する要素は下記の通りである。

- (1) 操作画面
- (2) 操作を行う HTML 要素
- (3) 説明テキスト
- (4) ステップ番号

そこで、提案手法では Web チュートリアル雛形を“操作画面、操作を行う HTML 要素、ステップ番号から成るデータ”と定める。また、この雛形に資料作成者が後から編集した説明テキストを追加したものをチュートリアルデータとする。

## 4.3 Web チュートリアル Web アプリケーションへの自動埋め込み手法

提案手法では、資料作成者が対象の Web アプリケーションを手動で直接変更することなく、汎用的かつ自動的に Web チュートリアルを導入し、操作者に提供できる必要がある。そこで、拡張機能を介し、Web ブラウザで解釈した HTML ソースに対し変更を加えることで Web チュートリ

アルの汎用的かつ自動的な導入を実現する。図6に概要を示す。

Web チュートリアルデータの埋め込みは、資料作成者が作成したチュートリアルデータから DOM ツリーのパスを読み込み、埋め込み先を特定した後、説明文とステップ番号を追加することで行われる。Web チュートリアルの自動埋め込み手順を下記に示す。

- (1) 拡張機能が起動されたページの URL に紐づいたチュートリアルデータを読み出す
- (2) チュートリアルデータから DOM ツリーのパスを取得し、ハイライト表示を行う HTML 要素を特定する。
- (3) チュートリアルデータから説明文とステップ番号を取得し、特定した HTML 要素に対し Intro.js の属性としてこれを挿入する。

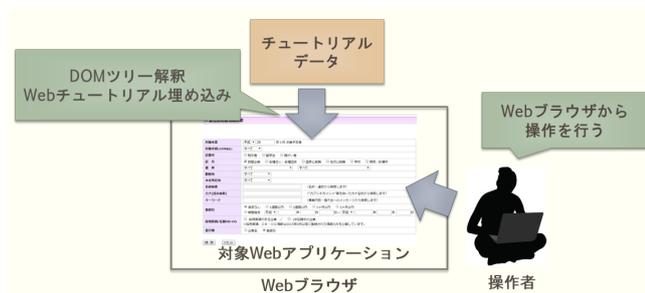


図6 Web チュートリアルの自動埋め込み手法の概要

## 5. Web チュートリアル作成支援ツールの試作

提案手法の実現可能性の評価と、第6章で述べる有効性を評価するため、Web チュートリアルの作成支援ツール(以下ツールとする)を Google Chrome の Chrome 拡張機能を用いて試作した。本節ではツールの実装方法について述べる。

### 5.1 ツールの概要

本ツールでは、Web ブラウザとして Google Chrome を使用し、操作ログの取得とチュートリアルの雛形の自動生成、Web チュートリアルの自動埋め込みを Chrome 拡張機能で実装した。また、Web チュートリアルとして2.2節で述べた“Intro.js”を利用した。さらに、チュートリアルデータを保存するデータベースとして、JavaScript と親和性が高く、データを JSON 形式で保存できる mongoDB を使用した。

### 5.2 チュートリアルの雛形の自動生成

資料作成者が用いるツールとして、提案手法におけるチュートリアルの雛形を自動生成する、Chrome 拡張機能を試作した。ツールのアーキテクチャを図7に示す。本

ツールにおいてはマウスクリックのみを操作のイベントとし、イベントが発生した HTML 要素を検知し、DOM ツリーをログとして取得している。これは、Intro.js を埋め込む際、資料作成者が Web チュートリアルを埋め込みたい HTML 要素さえ特定できれば良いためである。取得した DOM ツリーパスのログは、操作画面の URL と共に JSON 形式でチュートリアルの雛形として mongoDB に保存する。

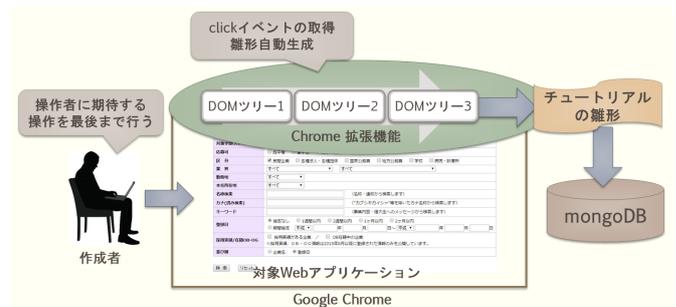


図7 雛形自動生成ツールのアーキテクチャ

### 5.2.1 試作ツールにおけるチュートリアルの雛形

本ツールにおけるチュートリアルの雛形はイベントが発生した HTML 要素が存在する URL と、要素の DOM ツリーパスを JSON 形式で表したものである。その一例を図8に示す。これは Yahoo 検索システムのロゴ部分について、チュートリアルの雛形を生成しコンソールに表示したものである。紙面の都合上、改行しているが、本来は一行で扱われる。

```
[{"url": ["http://www.yahoo.co.jp/"], "tree": [{"html": "<body> <div#wrapper> <div:nth-child(1)#header> <div:nth-child(1)#masthead> <h1> <a> <img>"}], "step": [1]}
```

図8 チュートリアルの雛形の例

### 5.2.2 試作ツールにおけるチュートリアルデータの完成

本ツールでは、Web チュートリアルの雛形をチュートリアルデータとして完成させるために、mongoDB 上で直接、操作説明文の編集を行う。そこで、JavaScript を読み込むことができるという mongoDB の特性を用いる。資料作成者はテキストエディタ上でシェル操作と同時に説明文を記述した js ファイルを作成し、mongoDB に読み込ませることで、チュートリアルデータを完成できる。図9に js ファイルの例を示す。

```
use test;
db.tutorial.update({"step":1, {$set: {"text": "説明文"}});
```

図9 js ファイルの例

### 5.2.3 ツールの利用手順

- (1) 資料作成者は Web ブラウザで対象の Web アプリケーションを開く
- (2) 資料作成者は拡張機能を起動し、開始ボタンを押す (図 10)
- (3) 操作者に操作してほしい要素をクリックする
- (4) ツールは URL とイベントが発生した HTML 要素を取得し、チュートリアル雛形を生成する
- (5) 資料作成者は説明文を js ファイル内に MongoDB のシェル操作と共に記述し、MongoDB に読み込ませ、チュートリアルデータを完成する

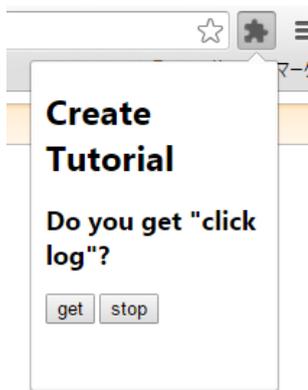


図 10 ツールの POP

## 6. 評価実験

提案手法を評価するため、第 5 章における試作ツールを大学で実際に運用されている、就職支援に関するシステムに対して適用し、評価実験を行った。本実験では、資料形式としての有効性を評価する操作説明資料閲覧実験と、資料の作成効率を評価する操作説明資料作成実験の 2 つの実験を行った。

### 6.1 操作説明資料閲覧実験

#### 6.1.1 実験概要

提案手法において Web チュートリアルを用いることの有効性を示すため、操作効率について、スライド形式の操作説明資料との比較を行った。

#### 6.1.2 実験条件

本実験では、Web チュートリアルとして Intro.js を提案手法により“会社説明会情報検索システム”に埋め込んだものを用いた。対象アプリケーションでのチュートリアルのステップ数は 9 ステップであった。また、スライド形式の操作説明資料は、Microsoft PowerPoint2013 を用いて作成し、pdf に変換した。実験環境は、OS は Microsoft Windows7、Web ブラウザは Google Chrome ver.48 とする。実験参加者 (以降、単に参加者) は情報工学を専攻する

学生 4 名で、対象となる Web アプリケーションを操作した経験を持たない。

操作説明資料としての質が操作効率に影響を与えにくいように、提案手法と従来手法間で質が同等になるよう条件を定めた。以下にその条件の趣旨を示す。

- (1) 操作手順は、提案手法ではステップを切り分け、従来手法ではスライドを切り分けて、操作箇所ごとに説明する
- (2) 操作画面を視覚的に理解できるように、システム画面またはその画像を用いる
- (3) 操作箇所を注視させるように、提案手法では Intro.js の注視機能を利用し、従来手法では図形ツールによる強調を行う
- (4) 操作に対する説明は単一の吹き出しを設けて、その中に 20 文字以内で記述する
- (5) 説明文の吹き出しは操作箇所に隣接して配置する
- (6) 入力が生じる場合、入力項目名と入力値と説明文内に必ず含める

本稿では、これら 6 つの条件を質的条件と呼び、これを満たす提案手法の説明例を図 11 に示し、従来手法の説明例を図 2 に示す。



図 11 提案手法の説明例

実験方法としては、参加者 4 名を Web チュートリアルによる操作説明資料を用いて操作する学生 2 名、スライド形式の操作説明資料を用いて操作する学生 2 名に分けた。参加者は操作説明資料に記述された、目的の検索結果を得るまで操作を行った。操作効率を測るため、その操作時間を測定し記録した。また、参加者の操作の様子を確認するため、操作に影響を与えない動画キャプチャソフトを用いて録画した。

### 6.1.3 実験手順

実験は概要の説明を行った後、下記の手順で行った。

- (1) 実験の概要、操作の目的の説明
- (2) 操作説明資料の配布
- (3) 操作説明資料に沿った対象アプリケーションの操作
- (4) アンケート

### 6.1.4 実験結果

Web チュートリアルとスライド形式の操作説明資料間で操作時間を比較した結果、どの参加者も、提案手法に比べ従来手法の方が操作時間が短くなった。しかし、従来手法では、誤操作や操作すべき場所を一瞬迷う様子が見受けられ、提案手法ではこのようなことはなかった。また、アンケートでは“資料を確認しながらの操作に面倒なところがあれば具体的に書いてください”という問いに対し従来手法について、「pdf と実際の操作画面を何度も往復することが面倒だった」という回答が得られた。

### 6.1.5 考察

提案手法に比べ従来手法の方が操作時間が短くなった原因として、今回扱った検索システムでは入力要素や画面遷移が少なかったことが考えられる。しかし、どの参加者も提案手法では誤操作や操作箇所を見失うことがなかったことから、操作の正確な誘導という面では提案手法の方が優位性が見込まれた。チュートリアルにおいて操作者が誤操作などにより操作に行き詰る場合、資料作成者にとって個別対応が必要となるため、正確な操作を誘導できるという性質は、操作効率に比べて重要である。このことから、Web チュートリアルは説明媒体として有用であると考察する。

## 6.2 操作説明資料作成実験

### 6.2.1 実験概要

提案手法の有効性を示すため、試作ツールで生成する操作説明資料と、スライド形式の操作説明資料の手動作成(以下、従来手法)でその作成効率を比較した。本実験では、参加者は提案手法と従来手法の両方により資料作成を行った。

### 6.2.2 実験条件

従来手法として、画像の加工、図形の挿入などの機能を有した Google スライド [6] を利用した。実験環境は、閲覧実験と同一とした。対象の Web アプリケーションは“求人情報検索システム”である。実験参加者は、情報工学を専攻する学生 4 名である。

本評価では作成プロセスにおける効率性の比較に重点を置く。よって、操作説明資料としての質の差が作成時間に影響を与えにくいように、提案手法と従来手法間で資料の質が同等になるよう質的条件を閲覧実験と同一とした。この条件は紙媒体の資料にして参加者に配布した。

また、操作説明資料の作成プロセスは、完成した資料の確認作業の有無や細かな修正など、個人によって差異があり、作成時間に影響する。よって、この実験では作成の終

了条件を“全ての操作について説明を作成し終えたとき”と定めた。参加者が資料の作成を終えた後、実験者が条件を満たしているかどうかを確認し、満たしていなかった場合、資料の修正を要求した。本実験における資料作成時間は初期作成作業と修正作業にかかった時間の合計とし、参加者がストップウォッチで測定、記録を行った。また、参加者の操作の様子を確認するため、操作に影響を与えない動画キャプチャソフトを用いて録画した。

### 6.2.3 実験手順

実験は概要の説明を行った後、下記の手順で行った。この際、本実験では個人の資料の作成能力を評価するものではないことを強調した。

- (1) 利用ツールの説明とその練習操作
- (2) 操作説明資料の作成条件の配布、説明
- (3) 対象 Web アプリケーションの説明とその練習操作
- (4) Web アプリケーションの操作シナリオの配布、説明
- (5) シナリオに沿った操作説明資料の作成
- (6) 条件を満たしていなかった場合、作成した資料の修正
- (7) 両手法に対するアンケート

### 6.2.4 実験結果

提案手法と従来手法における各参加者の資料作成時間を図 12 に示す。提案手法を用いた場合、どの参加者も従来手法より作成時間が短くなった。また、4 名のうち 3 名の参加者は従来手法に比べ作成時間が半分以下まで短縮された。また、アンケートにおいてそれぞれの手法について利用したツールでの資料作成のしやすさを聞いたところ、提案手法について、4 名中 3 名が作成しやすかったと答え、従来手法について、4 名全員が面倒に感じたことと答えた。その理由としては、提案手法では作成の手順が少ないこと、従来手法ではスクリーンショットの撮影や図形の挿入、Web ブラウザと利用ツールの往復が多いことが挙げられた。

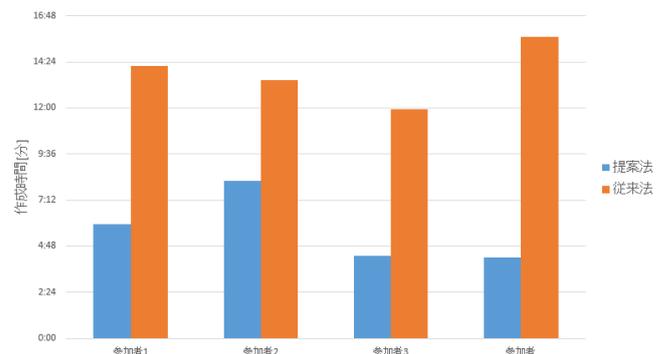


図 12 被験者別の操作資料作成時間の比較

### 6.2.5 考察

この結果から、資料作成効率の面において、操作ログを用いてチュートリアルの雛形を自動生成する提案手法が有効であることが確認できた。また、アンケート結果から、

資料の作成手順が多いことは資料作成者の負担になることが確認できた。

## 7. 関連研究

本研究と類似する研究として、Windows 環境を前提とし、教材作成者が操作したアプリケーションの操作ログから PC 初心者が行うべき操作を示す教材を自動生成する手法がある [1]。本手法では、MSAA(Microsoft Active Accessibility) を用いて、Windows アプリケーションへの操作イベントを詳細に取得し、これを教材の自動生成に活かしている特徴がある。

提案手法と比較して、資料作成者の操作ログに基づくチュートリアル自動生成や、操作箇所の注視機能、画面への資料の埋め込みと言う点で類似する。一方、決定的に異なる点としては、前提となる実行環境が挙げられる。MSAA は Windows に特化した API を提供しているため、Linux 系の OS で利用できない。大学教育における情報工学系の講義・演習を例に挙げると、学習者の操作する PC は OS やそのバージョンまでを含めると多様であり、Windows 環境を前提とすることが現実的には困難なケースもある。その点では、Web ブラウザベースで、OS 非依存なアプローチを採っている提案手法が優位であると考えられる。

当該研究との比較の結論としては、Windows 環境であれば当該手法が優位であり、OS が不統一だが Web アプリケーションを対象とするならば提案手法が優位であるため、適切な使い分けによって、同様な趣旨の手法を適用できる範囲が拡充できたものと言える。当該研究では他に、PC 初心者の誤操作を検出し、正常操作に復帰する支援方法も提案しており、これは提案手法の今後の課題としたい。

操作ログを利用した学習支援の研究として、操作時間間隔により学習者の行き詰まりを判定する手法がある [5]。当該手法では、学習者の操作ログとして得た操作間の時間間隔に対して、移動平均や移動分散を算出することによって、行き詰っている学習者の特定支援をねらうものである。

当該手法は課題学習を対象としており、アプリケーションの操作学習を対象としていない点で、本研究の論点との明確な違いがあるが、チュートリアルが分からずに操作に行き詰っている学習者が存在しう点では共通性がある。今後、当該手法を応用することによって、提案手法で作成した Web チュートリアルの低質な箇所を操作者の操作ログに基づいて発見支援する方法も検討していきたい。

## 8. まとめ

本稿では、Web アプリケーションの操作を説明する方法として Web チュートリアルに着目し、効率的な Web チュートリアル作成支援手法を提案した。提案手法では、資料作成者の操作ログから Web チュートリアルの雛形を自動生成し、かつ Web ブラウザの拡張機能から Web チュートリ

アルを着脱を可能とした。

評価実験において、提案手法と従来の電子文書形式による資料で、その作成における効率性を比較し、提案法を用いた場合どの参加者も従来の資料に比べ作成時間が大幅に短縮されたことから、提案手法が資料作成効率に対し貢献する見込みを得た。また、Web チュートリアルとスライド形式の資料を用いた場合における操作効率を比較することで Web アプリケーションに対する操作説明資料としての有効性を検証した。本実験におけるシナリオのシンプルな問題から提案手法の方が不利な結果になったが、Web チュートリアルは操作のシナリオに対する正確性という面で電子文書形式の資料に比べ、優位性があるという見込みを得た。

今後の課題として、試作ツールにおける説明文の追加の半自動化が挙げられる。現段階ではデータベースを直接変更する必要があるが、資料作成者がどの HTML 要素に対して説明分を追加するか視覚的に確認しながら説明文を編集できることが望ましい。また、提案手法における操作ログではマウスクリックのみを取得しているが、入力値などを取得することで操作者がどこでどのように入力を間違えたかという情報を検出し、資料作成に役立てる機能の作成が挙げられる。

## 参考文献

- [1] 入部百合絵, 藤原真, 安田孝美, 横井茂樹, 学習者の操作プロセスに適応した対話型ソフト学習システム, 信学論 D 情報・システム, J91-D(2), pp.269-279, 2008.
- [2] Google, “GoogleChrome”, <https://www.google.com/chrome/browser/desktop/index.html>
- [3] 吉川徹 他, 開発者のための Chrome ガイドブック, インプレスジャパン, 2013.
- [4] afshinm, “Intro.js”, <http://usablica.github.io/intro.js/>
- [5] 中村喜宏, 赤松則夫, 桑原恒夫, 玉城幹介, 操作時間間隔の変動に着目した CAI 学習の行き詰まり検知方法, 信学論 D-I, J85-D-I(1), pp.79-90, 2002.
- [6] Google, “Google スライド”, <https://www.google.co.jp/intl/ja/slides/about/>