

## 聴覚障害学生にコンピュータ操作を教示する支援ツール

### — SZKIT : SynchroniZed Key points Instrution Tool —

筑波技術大学 保健科学部 情報システム学科<sup>1)</sup> 産業技術学部 総合デザイン学科<sup>2)</sup>

小林 真<sup>1)</sup> 鈴木拓弥<sup>2)</sup>

**要旨：**聴覚障害学生を対象とした授業で利用するため、教員のニーズをもとにコンピュータ操作を教示する教育支援ツールソフトウェア、SZKIT (SynchroniZed Key points Instrution Tool) を作成した。作成したソフトウェアは、マウスカーソル脇に説明文およびマウスのクリック状態・特殊キーの押下状態を表示するもので、Adobe 社の Illustrator や Photoshop 等、複雑なマウス操作が必要になるグラフィックデザイン系ソフトウェアの使い方を教える際に利用することを想定している。文章を表示する領域は、自由に入力できるモードと予め用意しておいたファイルを読み込んで逐次表示させるモードの2つのモードを持つ。また、教える対象のソフトウェアの操作を妨げることはないよう、支援ツールの操作はゲームパッドを用いて行うことができる。

**キーワード：**聴覚障害学生, グラフィックデザイン系ソフトウェア, 特殊キー

#### 1. はじめに

本学の産業技術学部には総合デザイン学科があり、聴覚障害学生がデザインに関する知識と技術の習得を目指して修学している。彼らにとって Adobe 社の Illustrator や Photoshop といった業界標準グラフィックデザイン系ソフトウェアの使い方を習得することは必要不可欠である。そのため同学科の授業体系には、関連する実習科目が盛り込まれている。そしてそれらの操作は、マウスクリックやドラッグひとつとってみても、Shift キー、Ctrl キー、Alt キーといった特殊キーを使用しなくてはならない場面が多く、複雑である。特殊キーの組み合わせによって操作の意味が異なってくるため、その教示は困難を極める。

例えば、Illustrator において描画オブジェクトをドラッグすると、そのオブジェクトは移動する。ここで Alt キーを押しながらドラッグすると、オブジェクトはコピーされる。また、コピー操作中に Shift キーを押すことで、コピー元オブジェクトと水平方向や垂直方向など 45 度ごとに揃った位置にコピー先を指定することもできる。このように Shift キーは位置を揃える場合に押されるため、往々にして操作途中で押されることがある。すなわち、揃った位置にオブジェクトをコピーする場合には、「Alt キーを押しながらオブジェクトをドラッグし、途中で Shift キーを押して、目的の位置でマウスのボタンを離し、その後特殊キーを離す」という操作が行われることになる。

このような操作を教える場合、一般的には操作している

画面の様子を提示しながら「ここで Alt キーを押しながらドラッグします」「さらに Shift キーを追加して押します」と音声で説明を加える。しかし聴覚障害学生は音声情報を正確に受け取ることが困難であるため、主に画面表示からのみ情報を受け取ることになり、操作の理解が難しい。例として、図1に通常のドラッグ操作と、Alt キーと Shift キーを押したドラッグ操作の画面状態を示す。マウスカーソルの形の変化からかろうじて Alt キーが押されたことは分かるもの

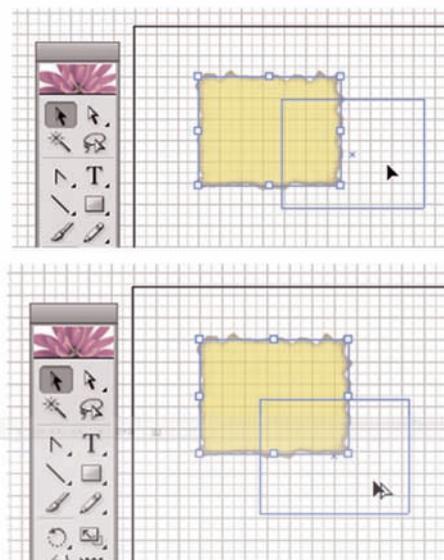


図1 通常のドラッグ操作 (上) と Alt キー・Shift キーを押しながらのドラッグ操作 (下)

の、その差は微々たるものである。また Shift キーに関しては、オブジェクトの位置や動きを良く観察していないと全く理解できない。

こうした操作を聴覚障害学生に教える手法としては、教科書・参考書等の教材に加えて、詳しく解説が書かれたプレゼンテーションコンテンツや、適切な解説などを字幕挿入したマルチメディア教材を提示する教示方法などが考えられる [1]。しかし実際には、学生のペースに合わせた説明が必要だったり、学生からの質問に答えたりするために、リアルタイムに操作を説明する必要が生じてくる。

## 2. チャットを用いた解決策と新たな問題点

このリアルタイムでのソフトウェア操作説明を目的として、グラフィックデザイン系ソフトウェアの授業担当者でもある共著者の鈴木は、学生らに操作説明を文章でリアルタイムに示すことを試みた [2]。これはチャットを利用するもので、細かい操作方法をその場で具体的に指示できるという利点があり、一定の効果が得られた。しかし注視するマウスポインタ周辺と、チャット画面とが離れていることから、学生は2つの画面エリアを交互に注視しながら操作方法を学習しなくてはならないという欠点もあった。一方で、提示する説明文を、定型のパターンで構成できる可能性も見出された。さらに、学生は説明されたマウス操作時に押されている特殊キーの理解が不足していることなども分かってきた。

そこで、得られた知見を生かした次のステップとして、操作説明文と押されている特殊キー表示が「マウスポインタに追従して動く」支援ツールの開発を共同で進めることになった。このツールは教員が操作のデモをする際に利用することを想定しており、説明文や特殊キー押下といった、操作のキーポイントを同期して提示することから、「SynchroniZed Key points Instruction Tool: SZKIT」と名付けた。

## 3. SZKITの要件と実装方法

### 3.1 SZKITの要件

SZKITに関して、鈴木から寄せられた要件を整理すると以下ようになる。

1. マウスポインタに追従するテキスト領域に操作説明やコメント等を表示したい。
2. 一連の説明文を予めファイルで用意しておき、それらを読み込んでテキスト領域に順に表示したい。
3. いくつかの多用する説明文は、ワンアクションで呼び出して表示したい。
4. マウスクリック、ドラッグ、Shift・Alt・Ctrl キーの状態（押されているのか離されているのか）を視覚的に提示したい。

5. 自由記述で書き込めるモードも欲しい。

それぞれの要件について、実装しつつ改良を進めたので、以下順に説明する。

### 3.2 マウスポインタに追従するテキスト領域の実装

まず要件1については、これまで「でかポインタ」や「あんだーまうす君」といったマウスポインタに追従する弱視者用のソフトウェアを作成してきている [3][4] ため、同様の手法を用いることにした。これは 50ms 毎にマウスポインタの座標値を取得し、ウィンドウ座標を追従させつつ、画面最上位に描画するものである。コンピュータには相応の負荷がかかるものの、現在のハードウェア性能では使用感に変化が現れるほどのものではないことが確認されている。

基本的にこの仕組みを用いて、要件2に対する機能も組み込み、支援ツールを試作した。これは予め用意したファイルの文章を1行ごとに切り替えて表示するもので、Windows+CtrlとWindows + Shiftのキー操作で説明文を切り替える仕様とした。

### 3.3 ゲームパッドインタフェースの採用

SZKITはシステムに常駐する形で動作し、対象となるグラフィックデザイン系ソフトウェアと同時に実行され、操作状態や説明文を表示する。上記試作ツールでは、キーボードに説明文の切り替え機能を割り当てたが、要件3を満たすような操作を、すべてキーボードで行うには限界がある。例えばCtrl+F1などといった特殊キーとファンクションキーの組み合わせにワンアクション呼び出し機能を割り当ててしまうと、説明対象のソフトウェアに備わるショートカットキーと重なってしまう恐れがある。そこで、今後の機能の増加を考慮し、教員にどのようなインタフェースで操作させるかを考えた結果、「ゲームパッド」を利用することにした。ゲームパッドは汎用ハードウェアでありながら、通常のグラフィックデザイン系ソフトウェアではまず用いられることがなく、SZKITのような常駐型ツールの操作に適している。そしてゲームパッドを左手に持ち、右手でマウスを操作することを想定すると、パッドの左半分のインタフェースを積極的に用いるのが良策であるため、これらに機能を割り当てることにした。図2にゲームパッドの外観と左側に配置されたボタンの拡大図を示す。俗にプレステ型と称されるゲームパッドの場合、左半分には、十字キーと5番と7番のボタンがあるのが一般的である。

ここで、ゲームパッドの操作を検出するにはDirectInputを利用する手法もあるが、簡便さからWindowsAPIを利用することにした。そして、SZKITによる負荷を減らすために、ポーリングではなく、イベント駆動型のスタイルで進めることにした。この場合、フックできるイベントは十字キーおよび1～4のボタンの操作時に発生するもののみであり、通常



図2 ゲームパッドの外観（上）と左側のボタン群（下）

5番や7番ボタン操作時にイベントを発生させることはできない。しかし、十字キー操作時に、5番や7番ボタンが押されているかどうかのチェックはできるため、2ビット4種の状態を区別できることから、十字キーの上下左右とそれらボタンの組みあわせにより、合計16種の機能が割り当てられる。これは今回の支援ツールにおいては十分な数であると判断した。

機能の割り当てに関しては、要件2と要件3を満たすべく、十字キーの上下にファイルで用意した説明文の送りと戻しを割り当て、左に先頭に戻る機能を割り当てた。そして5番と7番のボタンを押しながら十字キーを操作する12種の操作には、ワンアクション呼び出しを割り当てた。しかし後述するように最終的にこれらの設定は変更した。

### 3.4 マウスと特殊キーの状態表示の実装

次に要件4にあるマウスと特殊キーの状態表示の実装について述べる。クリックとドラッグは、左ボタンが押された場合にアイコンを表示すれば、表示されつつ動いている場合はドラッグであると学生が理解できるので、特に両者を区別することなくマウスの左ボタンの押下のみを検出することにした。そしてShift・Alt・Ctrlキーについても状態を監視して、それぞれのキーが独立してアイコンを表示するようにした。

マウスやキーの状態を表示するソフトウェアは、AutoHotKeyを用いるosdHotkey[5]やプレゼンテーション支援用のNullmass TeamによるTarget[6]などが存在す



図3 SZKITの外観

る。しかしそれらは固定ウィンドウ内に文字で状態を提示するものであり、視線移動を伴ううえ直感的ではない。また、Targetのマウスドラッグ表示はマウスポインタと重なった位置に丸印を表示するため、作業を示す今回の用途には適していない。それらを踏まえSZKITでは、それぞれの状態表示がマウスポインタに重ならないようポインタの上側に左から「マウスクリック」「Shiftキー」「Altキー」「Ctrlキー」の順に配置することにした。そして表示するアイコンの作成は鈴木が担当し、直感的に分かりやすいよう、マウスの形とキーボードの形にした。最終的に作成されたSZKITの外観を図3に示す。図ではすべてのアイコンを表示させているが、実際の利用時には、キーやマウスのアイコンは押されたタイミングで表示され、押されていないときは表示されない。

### 3.5 自由記述エリアの実装

図3のマウスポインタ右下の領域が説明文の表示領域である。ここに要件5を満たすため、モード切り替えをすることで自由記述を行うことができるようにした。ファイルに用意された説明文の表示と自由記述文の表示の切り替えの機能は、ゲームパッドの十字キー右方向に割り当てた。

## 4. 追加機能の要望とブラッシュアップ

以上のような実装を進め、試用を進めるうち、鈴木から次のような要望が上がってきた。

- ・操作説明時に邪魔になることもあるため、説明文エリアの表示・非表示機能を追加したい。
- ・後ろの学生から文字が読めないという意見があるので、文字サイズを変更したり、太字にしたりする機能が欲しい。
- ・自由記述の入力後に操作説明をし、再度自由記述を追加する、という作業が繰り返されるため、キーボードショートカットで自由記述入力モードに移行したい。
- ・自由記述で書きためたものをログとして残したい。

さらに、当初は重要であると考えていた要件2と3（ファイルで用意された文章表示・ワンアクション呼び出し）よりも現状では自由記述の方が、利用頻度が高いことも分かってきた。そこで以上の要望を満たすため、機能を追加し、キーやゲームパッドの機能割り当てを変更した。

まずゲームパッド十字キーの上下方向に文字サイズの拡大・縮小を割り当て、左方向に太字と標準の切り替え、右

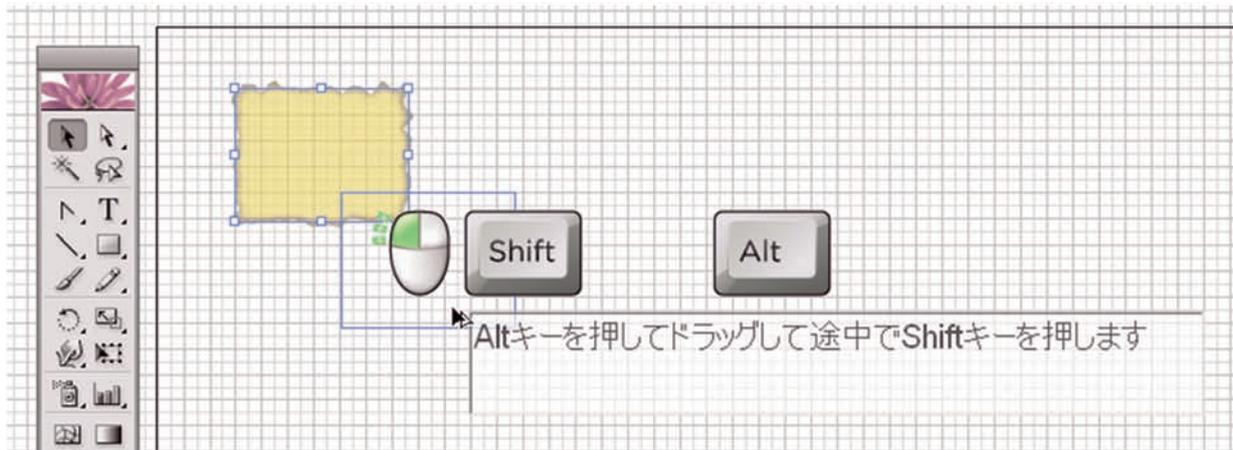


図4 SZKITを用いて図1下と同じ操作をしている様子

方向に表示・非表示を割り当てた。そしてキーボード操作については、Windows+Ctrlで非表示、Windows+Shiftで表示とし、SZKITがアクティブになっている状態でCtrl+"["を押すと文字サイズ拡大、Ctrl+"]"を押すと文字サイズ縮小を行い、Ctrl+Bを押すと太字と標準の切り替えを行うようにした。更にCtrl+上下矢印に説明文表示領域の透過率の変更を、Ctrl+左右矢印に白黒反転機能を割り当て、文字表示の見やすさを容易に変更可能ようにした。そしてログ記録については、Ctrl+Sを押すことでテキストファイルに記録できるようにした。

また、表示モードへの切り替え時にSZKITをアクティブにすると同時に自由記述用のテキストボックスにフォーカスをあてるようにしたので、教える対象となるソフトウェアの操作時にWindows+Shiftを押すと、直後に説明文の追加入力とこれら表示に関する変更ができるようになった。

## 5. まとめ

図4に、SZKITを用いて図1と同様の操作をした画面を示す。操作のポイントが文章とアイコンでリアルタイムに表示され、学生の理解を助けていると思われる。今後は、学生の意見を聞きつつ、定型文の作成のために自由記述のデータをログとして収集していく予定である。

## 参考文献

- [1] 村上裕史、皆川洋喜：聴覚障害者のためのマルチメディア教育システム、電子情報通信学会技術研究報告、ET、143-148、2001.
- [2] 鈴木拓弥：聴覚障害学生を対象としたデザイン実技演習支援に関する研究、筑波技術大学テクノレポート、18(2)、2011.
- [3] 小林 真：視野狭窄者用マウスカーソル探索支援ソフトウェア、第28回感覚代行シンポジウム講演論文集、28、75-78、2002.
- [4] 小林 真：マウスカーソル探索支援ソフトウェアの製作と評価、弱視教育、40、16-19、2003.
- [5] <http://www.romeosa.com/osdHotkey/help.html>
- [6] [http://download.cnet.com/Target/3000-2075\\_4-10915620.html](http://download.cnet.com/Target/3000-2075_4-10915620.html)

## Support Software which Instructs Hearing Impaired Students of Computer Operation — SZKIT: SynchroniZed Key points Indication Tool —

KOBAYASHI Makoto<sup>1)</sup>, SUZUKI Takuya<sup>2)</sup>

<sup>1)</sup>Department of Computer Science, Faculty of Health Sciences,

<sup>2)</sup>Department of Synthetic Design, Faculty of Industrial Technology,  
Tsukuba University of Technology

**Abstract:** An educational support software which instructs computer operation was developed to be utilized in lectures for hearing impaired students from a need of a teaching staff. We named the support software “SynchroniZed Key points Indicating Tool (SZKIT).” The SZKIT displays instruction texts, states of mouse button and modifier keys besides a mouse pointer. It is assumed that the SZKIT is intended to teach graphic designing software like Illustrator or Photoshop. These designing software require complex mouse operation with modifier keys with which hearing impaired students will struggle. The instruction text area has two modes. In one mode of them, user can input arbitrary text to the area. In the other mode, the software reads information from a text file which is prepared in advance and displays them sequentially. The teaching staff can control the sequence of that text information and other operation through a game pad. Using the game pad prevents disturbing the graphic design software.

**Keywords:** Hearing impaired students, Graphic design software, modifier key