

生徒の操作ミスをいち早く発見するための一手法 - 操作ミスの前からやり直しができるシステムの提案 -

森田賢太*1・山下泰輝*2・森田直樹*1・高瀬治彦*3

Email: morita.k@star.tokai-u.jp

*1: 東海大学情報通信学研究科情報通信学専攻

*2: 東海大学情報通信学部通信ネットワーク工学科

*3: 三重大学大学院工学研究科電気電子工学専攻

◎Key Words グローバルフック, 仮想マシン, 操作復元

1. はじめに

本研究ではコンピュータを用いた演習の授業において、操作ミスをいち早く発見し、操作ミス前の状態からやり直しができるシステムを開発したので報告する。本研究の対象とする操作ミスとは、受講者が講師の指示したマウス操作やキーボード操作とは違う操作をしてしまうことである。

コンピュータを用いた演習の授業では、1名の講師と複数のティーチングアシスタント (TA) で受講者をサポートすることが多い。著者はTAとして「ここをクリックしても動かない」と質問を受けたことがある。経験をつんでいる講師は、受講者のそのときの画面を見ただけでミスの原因を判断することができるが、知識が浅いTAは、画面を見ただけでは判断できず、試行錯誤しながら受講者が行なった動作を確認する。

本研究の目的は、試行錯誤せずにいち早く操作ミスを発見するために、受講者が行なった操作の過程を簡単に振り返ることができるようにする。また、操作ミスを発見した時、その画面まで戻してあげる必要があるため、復元の手助けを行なうことができるようにすることである。

本稿では、操作過程がわかる事と適切な箇所から操作をやり直せるシステムを提案する。

2. 先行研究

操作過程を知るためのシステムに片岡ら[1]や前田ら[2]

のシステムがある。[1]はプログラミングの過程を記録し、画像としてみせるシステムである。このシステムはプログラミングを対象としており、特定のアプリケーションのみにしか対応していない。[2]はキー入力などの操作をアプリケーションに入力することで、操作過程をみることができる。そのため、操作過程をみるためには、アプリケーションを適切な状態にする必要がある。また、[1]と[2]はいずれも操作をやり直せる機能を持っていない。

このように操作過程と操作を復元することの両方を行えるシステムはない。

3. アイデア

最初に操作過程がわかるアイデアを述べる。本稿で提案する操作過程の確認方法は、キャプチャした画像をコマ送りすることで、操作過程を確認する手法をとる。具体的には図1のような操作の遷移画像をみることである。

次に操作を復元するためのアイデアを述べる。操作の復元には仮想化技術を利用する。図1はプログラミングを行なう準備の画面遷移である。(C)の画面では保存をすることができないため、手順を間違えた場合、(A)と(B)をやり直さなくてはならない。そこで、仮想化技術を用いることで、(C)のような画面など、通常は保存できない状態でもバックアップとして保存できるようにする。

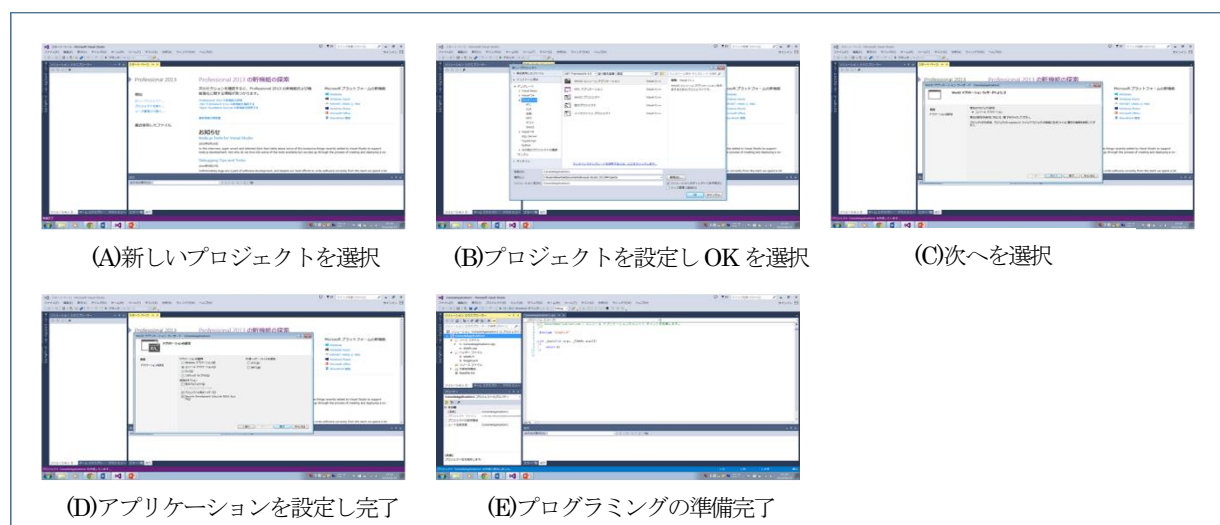


図1 プログラミングを行なう準備の過程

4. グローバルフックを用いた操作過程の可視化

本システムでは、受講者がパソコンをどのように操作されたかを知る為に、操作時ごとに画面のキャプチャを行う必要がある。パソコンの操作が行われたことを認識するために、ウィンドウへメッセージを横取りすることができるグローバルフックを用いる。これにより既存のアプリケーションに手を加えずに操作を監視することができる。また、キャプチャされた画像をそのまま見るのではなく、操作が分かりにくいのでキャプチャした画像に加工を施す。

4.1 キャプチャのタイミング

Windows では、アプリケーションの処理に図2のようにメッセージを用いる。Windows などのオペレーティング・システムは、マウス操作やキーボード操作が行われると、行われた内容を含むメッセージを発行する。発行されたメッセージはメッセージキューと呼ばれるところに格納される。アプリケーションはメッセージキューからメッセージを習得し、そのメッセージの内容に合う処理を行う。グローバルフックは本来処理を行うアプリケーションより先に、メッセージを取得することができる。そのため、OS からキーボードによる文字の入力の内容や、マウス操作によるメニューバーへのクリックなど細かい情報を取得し、取得したタイミングでキャプチャを行なう。

4.2 キャプチャした画像

キャプチャされた画像を撮影された順番にパソコンのディスプレイに映すことにより、今まで行われた操作の過程を見ることができる。しかし、キャプチャされた画像はマウスやカーソルが表示されないことがあり、どこの編集を行なっているのかわからないことがある。そこでキャプチャした画像に図3のような加工を用いる。図3の赤い四角い枠で囲まれている部分が操作をおこなわれた範囲である。これにより、操作された箇所がわかりやすくなる。

5. 仮想マシンによるバックアップ

本章では仮想マシンを用いたバックアップ機能について説明する。通常アプリケーションは対応できる画面でないと保存することができない。また、その保存対象はテキストファイルなどである。そのため、図4のようなアドインの設定などアプリケーションの構成を変更している最中にバックアップをとることができない。

本稿では仮想化ソフトウェアの一つである Virtual Box を用いる。

5.1 仮想マシンの運用

図5は仮想マシンを使用しているようすである。このように仮想マシンは、既存のオペレーティング・システム上に追加のオペレーティング・システムを動かすことができる。つまり、現実のコンピュータ上に他のコンピュータがある。既存のプログラムオペレーティング・システムをホスト OS、追加のオペレーティング・システムをゲスト OS と呼ぶ。

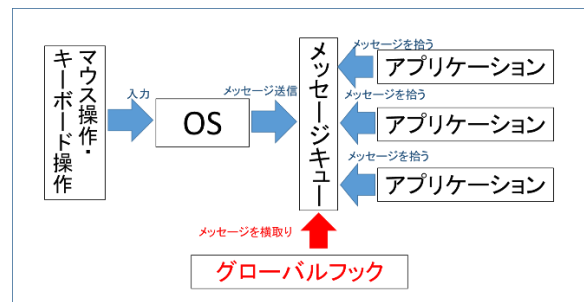


図2 メッセージの流れ

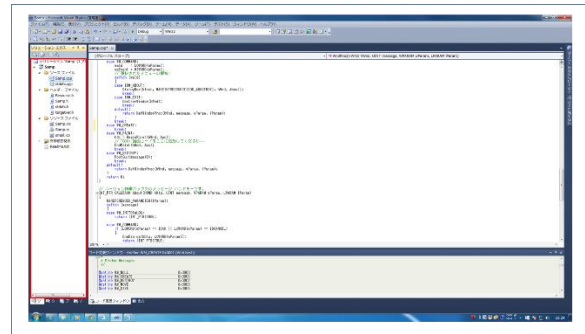


図3 キャプチャした画像の加工例

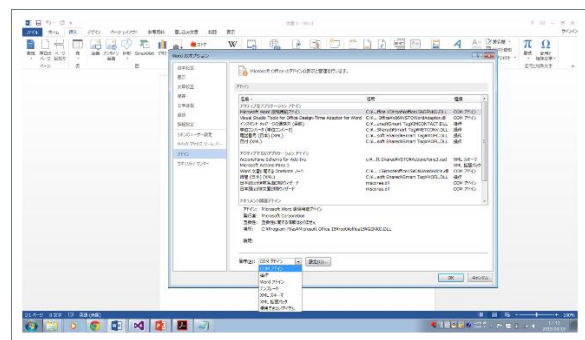


図4 Word のアドイン設定

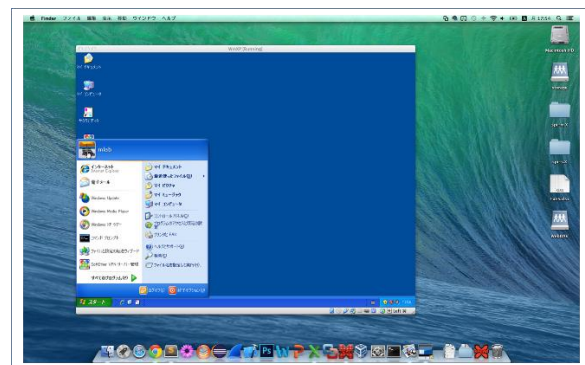


図5 Mac OS とその中にある Windows OS

5.2 仮想マシンのスナップショット

仮想マシンのスナップショットとは、特定の時点におけるゲスト OS の状態を保存する機能である。

スナップショットを利用することにより、任意のゲスト OS の状態へ戻ることができる。

操作を復元できるのはゲスト OS での操作のみなので、操作はゲスト OS で行なうが、仮想マシンのスナップショット機能の問題として、スナップショットを実行できるのはホスト OS であるため、ゲスト OS での操作でスナップショットを取ることはできない。

そこで本システムは、ゲスト OS とホスト OS で TCP 通信を行うことにより、ゲスト OS での操作からスナップショットを行う。

このスナップショットのファイルサイズは前回のスナップショットファイルとの差分になるため、さまざまであり 1 つの容量が 1GB になることがある。いらなくなった古いスナップショットのファイルを削除または外部に移動することで解決できる。

6. 本システムの実装

本システムの構成は図 6 のようになっており、実際に受講者が操作を行うのはゲスト OS である。

図 7 は開発したゲスト OS 用のシステムである。キャプチャ開始ボタンを押すと、4 章のグローバルフックを用いた操作過程の可視化が行われ、操作ごとに画面のキャプチャが指定のフォルダに保存される。仮想マシンによるバックアップについては、接続ボタンでゲスト OS からホスト OS へ接続され、保存ボタンを押すとホスト OS へスナップショットをとる要求が送られる。要求を受け取ったホスト OS は、コマンドラインにて Virtual Box のコマンドラインインターフェースを用いて、ゲスト OS のスナップショットを行う。保存ボタンを押す以外にも設定により、スナップショットの時間間隔を設定し、バックアップをすることができる。

7. システムの運用例

実装したシステムの運用例を説明する。

最初に講義開始までの流れを示す。(1)ホスト OS にてサーバープログラムを立ち上げる。(2)Virtual Box のゲスト OS を起動させる。(3)ゲスト OS にてクライアントプログラムを立ち上げる。(4)クライアントの接続ボタンを押す。(5)クライアントプログラム開始ボタンを押す。(6)講義を開始する。

次に受講者に質問を受けた際の講師または TA のパソコンの操作の流れを示す。(1)講師または TA は学生のパソコンのキャプチャされた画像の保存ファイルを開く。(2)学生と共にサムネイルから大まかに操作過程を見る。(3)気になった操作過程の画像ファイルを開きコマ送りで見ている、赤い四角い枠で囲まれた箇所を見る。(4)操作ミスを見つめる。

最後に、操作を復元する流れを示す。(1)講師または TA が、操作ミスをした時間を画像ファイルのキャプチャされた時間から求める。(2)ホスト OS にコントロールを戻す。(3)Virtual box 上の復元させたいゲスト OS を一時停止させる。(4)一時停止させたゲスト OS を戻したいタイミングのスナップショットファイルで復元を行なう。(5)コントロールをゲスト OS に戻し、受講者に操作をやり直させる。

これにより、操作の過程を確認し操作ミスを見つけることと、操作ミス前の状態へ復元することができる。

8. 試用実験

本システムを用いて 2 つの実験を行った。1 つ目なるは、グローバルフックを用いた操作過程の可視化により、受講者の操作ミスの助けになるかを検証した。2 つ目は、仮想マシンによるバックアップの有効性を検証した。

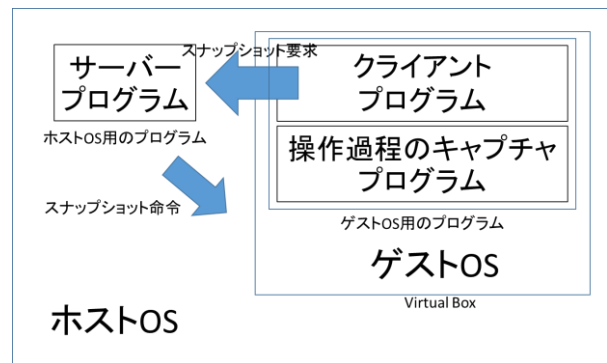


図 6 本システムの構成図

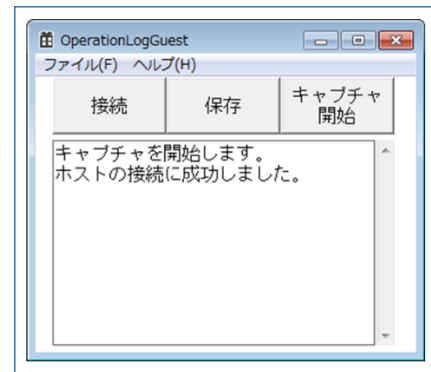


図 7 ゲスト OS 用のプログラム

8.1 操作過程の可視化による操作ミスの発見

大学 1 年生のプログラミング入門において、本システムを使用して頂き、講師、TA、受講者からコメントを頂いた。

プログラミング入門の内容は、統合開発環境 Visual studio2013 を用いて C 言語の多次元配列の演習を行うものである。授業の時間は 1 コマ 90 分授業の 2 コマ連続の授業である。

また、システムが定期的に 5 分間隔でのスナップショットを行なう設定にした。

キャプチャされた画像のファイルサイズは約 130KB で操作によりことなるが、800 枚ほど作成された。

はじめに、TA のコメントについて述べる。「受講者の操作過程を画像として見れることで、間違いの発見に役に立った」「直接的に画面で本来の原因ができなくても、それ以外の原因に間違いやミスがあるなどの判断材料として、役に立つと感じた」「ファイルを誤って消してしまった時など、誤作動前に復元が可能なのが良い」とコメントを頂いた。

次に、講師からのコメントについて述べる。「受講者が操作した様子画像として見るので、どこで受講者がつまづいたのかが分かる。しかし、全てのキー操作をキャプチャすると、かえってログが多く使いにくい」「欲しい情報は、何を押したかという細かい情報ではなく、どういう風に状態が変化したかであるので、1 行の入力単位くらいで時系列として見えるくらいが良い」「入力している文字を見ることができるので、パスワードなどのセキュリティ面を考える必要がある」「ほかのアプリケーションでも使えそう」とコメントを頂いた。

最後に、受講者からのコメントを述べる。「覗いたところの画面を振り返って見ることができるので、講師の説

明がわかりやすい」「バックアップにかかる時間はそこまでかからないが、もう少し早くして欲しい」「急いで文字を打っている時に、バックアップが実行されてしまうと、操作ができなくなってしまうため、イライラする」とコメントを頂いた。

8.2 仮想マシンによるバックアップの有効性

仮想マシンによるバックアップの有効性を調査するために、ダイアログによる設定を多く行う授業を対象にした。

本システムを使用して、大学一年生を対象に、統合開発環境の eclipse にてアンドロイドアプリを作成する講義をして頂いた。作成したアンドロイドアプリはボタンを押すと名前が画面に出力される簡単なものである。

本システムの設定について、ファイルの操作が少なく、システムの設定が多いため、1節のバックアップを行う時間間隔を設定するのではなく、好きなタイミングでバックアップを行うようにした。例えば、講師が「ここから操作が難しくなるので、バックアップをとるように」と言うと、受講者は開発したシステムの保存ボタンを押すことで容易にバックアップを行う。

本システムを使用した結果を述べる。図 8 は作成したアンドロイドアプリをパソコン上で動かすための、エミュレータの作成画面である。アンドロイドの設定を行うために、必要項目をいくつか入力したあと、作成ボタンを押すと、設定されたエミュレータが作成される。

数名の受講者は、作成ボタンを押さずに閉じてしまったため、エミュレータは作成されなかった。これらのエミュレータ作成用の画面はアプリケーションが状態の保存に対応していないため、設定の入力中の状態を保存することができない。本来は、エミュレータを作成できなかった受講者は、設定項目を再度埋めてから、エミュレータ作成ボタンを押さなければならないため、手間がかかる。今回は、本システムを使用していたので、設定を埋めている最中にバックアップを取ることができた。その結果、作成ボタンを押すだけでエミュレータの設定の再入力を省き、エミュレータを作成することができた。

このように、手間がかかるのに、保存することができない状態でも、仮想マシンを用いることで保存して、操作ミスをした際に、そこからやり直すことで、手間や時間の短縮できるため、本システムの仮想マシンによるバックアップ機能は有効だと考える。

8.3 本システムの考察

最初にグローバルフックを用いた画面キャプチャ機能について述べる。実験から、操作時に画面をキャプチャすることで、TA と講師は受講者の躓いた箇所を知ることができること、受講者は躓いた箇所に対する理解がしやすいことから、操作時に画面をキャプチャすることは操作ミスの発見に役に立つと考えられる。しかし、キャプチャされた画像が多いと目的の画像の発見が大変になるため、1行の入力単位でキャプチャを行うなど、キャプチャされた画像の提示の仕方を工夫が必要である。

また、すべての入力を監視できるため、パスワードの入力の時などセキュリティ面を考える必要がある。

次に、仮想マシンによるバックアップ機能について述

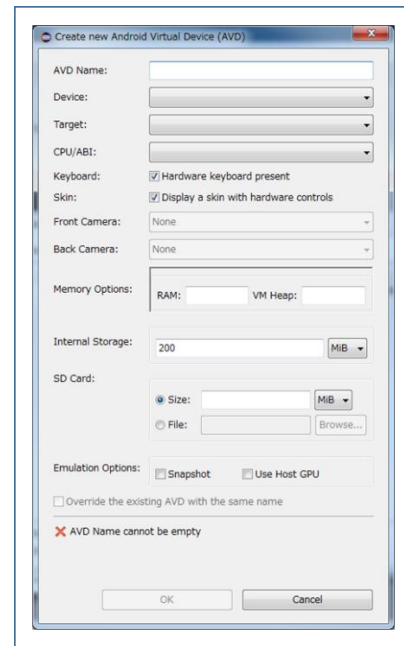


図 8 エミュレータの作成画面

べる。1節から、5分間隔など一定の時間間隔でバックアップをすると、受講者の操作中にバックアップが行われることがある。そのため、時間間隔でバックアップを行う際には短い時間でバックアップを行えるようにするなど工夫が必要である。また、2節より、プロジェクト作成などの通常は保存できない操作が多い授業に対して、より有効であると考えられる。

9. おわりに

本稿では、操作過程がわかる事と適切な箇所から操作をやり直せるシステムを提案した。本システムを使うことで、受講者の操作過程を見ることができ、操作ミスを見つける手助けができた。また、適切な操作ミスの前に戻すことができた。

今後の課題として、キャプチャされた画像の提供方法、パスワード入力時などのセキュリティ面とバックアップにかかる時間の短縮が挙げられる。

参考文献

- (1) 片岡由裕, 立岩祐一郎, 山本大介, 高橋直久: “受講者の操作履歴の分析機能を用いたプログラミング指導者支援システムの実現”, 信学技報, ET2009-83,p181-186, (2009).
- (2) 前田恵三, 中野靖夫: “コンピュータ操作過程の再現システム”, 日本教育工学雑誌, 16(4), 185-195(1993).