

対話型電子白板連動システム

坂東 宏和, 大即 洋子, 加藤 直樹, 中川 正樹
東京農工大学工学部

1. はじめに

近年, 教育の情報化の必要性が訴えられる中で, 各学校に積極的にパソコンが導入され, 授業に利用されるようになってきた. また, 最近では, 様々な利点を持つ対話型電子白板 (以下電子白板) が注目され, 各学校に導入されはじめています.

我々は, 以前から電子白板を授業内で利用することを提案し, それにより従来の黒板とチョークによる一斉授業の利点と, 情報化による利点とを融合できる可能性を示した¹⁾. しかし, 電子白板には, 従来の黒板と比較して板書面が狭い, 複数の入力を同時に処理できないといった問題があった.

そこで本稿では, Windows のマルチモニタ機能を利用することで板書面を広げ, さらに, 電子ペンによる同時 2 入力を可能にすることを目的とした, 対話型電子白板連動システムの設計と試作について述べる.

2. 基本設計

2.1 ハードウェア構成

システムのハードウェア構成を図 1 に示す. 本システムでは, 複数枚の電子白板の画面を連動させる基本的な方法として, Windows のマルチモニタ機能を採用する. ネットワークを介して表示データを交換する方法も考えられるが, システム上で動作するアプリケーションの開発が容易なことから, マルチモニタを採用する.

プライマリモニタ側の電子白板からのペン入力データを COM1 に, セカンダリモニタ側の電子白板からのペン入力データを COM2 に入力する. また, 電子白板は, 背面投影方式のものを利用することも可能である.

2.2 同時 2 入力に関する考察

現在市販されている電子白板は, ハードウェア

的な制約から同時に複数の電子ペンによる入力を受け付けられないが, マルチモニタ機能を利用して 2 枚の電子白板を 1 枚の画面のように利用することにより, 各電子白板で 1 つずつの入力を受け付け, 画面全体で 2 つの入力を同時に受け付けることが可能になる.

しかし, Windows の制約から, 通常の方法ではアプリケーションレベルで 2 つの入力を同時に処理できない. そこで, 本システムでは, 専用の電子白板ドライバを実現することにより 2 つの入力を同時に処理できるようにする. 同時 2 入力を実現することにより, 電子白板を板書に利用した場合や, 対戦型アプリケーションなどにおいて, 2 人で同時に板書・解答できる利点が生まれる.

2.3 ソフトウェア構成

ソフトウェア構成を図 2 に示す. 専用電子白板ドライバは, 電子白板の初期化と, 電子白板から送信されてくるデータをマウスイベントに変換・送信する処理を行う. この処理は, 特別なコード

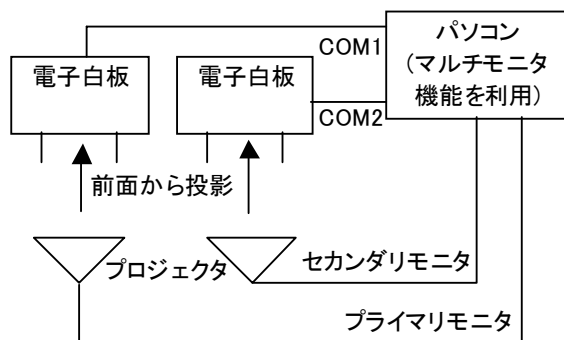


図1 ハードウェア構成(前面投影型電子白板)

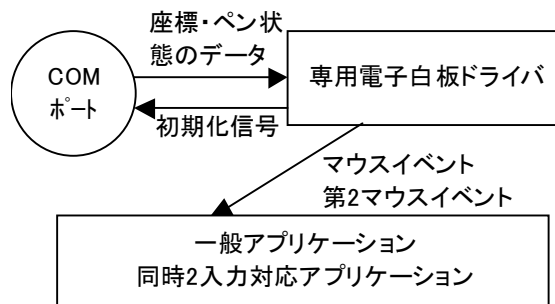


図2 ソフトウェア構成

を記述していない一般的なアプリケーションでも問題なく動作できるよう配慮する。次に、マウスイベントへの変換処理の手順について述べる。この処理は、電子白板からのデータを受信した時に行う。

(1) 座標系の変換

COM1 へ送られてきた座標データをプライマリモニタの座標系に、COM2 へ送られてきた座標データをセカンダリモニタの座標系に変換する。

(2) ペン状態からマウス状態への変換

ペン先や裏側、サイドスイッチのアップダウン状態を基に、生成するマウスイベントのアップダウン状態を決定する。

(3) イベントの送信

変換した座標とマウス状態を基に、適切なイベントを生成し、アプリケーションへ送信する。2本のペンが同時に入力されていない場合には、すべてのイベントを Windows 標準のマウスイベントとして送信する。同時に入力されている場合には、基本的に、最初に入力された方をマウスイベントとして送信し、後に入力された方を独自に定義した第2マウスイベントとして送信する。

また、イベントはマウスボタンダウンからアップまでを1まとまりとし、そのまとまりの途中でイベントの種類が変わることはないものとする。たとえば、まとまりの最初のマウスボタンダウンイベントを第2マウスイベントとして送信したら、もう片方のペンの状態変化に影響されることなく、まとまりの最後のイベントまで第2マウスイベントとして送信する。

これらの処理により、少なくとも1つのペンの動きが標準的な電子白板ドライバと同様のマウスイベントとして送信されることになるので、第2マウスイベントを処理していない一般的なアプリケーションでも、問題なく動作する。さらに、第2マウスイベントを処理することで、同時2入力を実現することができる。

なお、マウスイベントの送信は、マウスイベントを発生させる Windows の API を用いて行う。第2マウスイベントの場合は、マウス座標に位置するウインドウのハンドルを取得し、そのウインドウにイベントを送信する。この時、一般的なアプリケーションの場合、マウスボタンダウンからマウスボタンアップまでのイベントをキャプチャ

一することが多いので、1まとまりの最初のイベントを受信したアプリケーションが、そのまとまりの全てのイベントを受信するものとし、処理を簡略化する。

3. 対応アプリケーションの一例とその試用

同時2入力機能に対応した教育用アプリケーションの一例を図3に示す。このアプリケーションは、画面に表示された漢字だけを用いて単語を作成し、それを手書き文字で解答していくことにより、単語学習を行うことを目的としている。本システムを利用して対戦型とすることにより、児童の競争心を刺激し、楽しみながらより効果的に学習できることが期待される。

このアプリケーションを、東京農工大学科学技術展'2000において試用した。その結果、片側のユーザの入力に対して手書き文字認識などの重い処理を行った場合に、もう片方のユーザの入力に対する処理に遅延が発生することが分かった。この問題については、アプリケーション側の処理を工夫することで対処できると考える。また、同時2入力に対応していない一般的なアプリケーションについても、問題なく動作することがわかった。

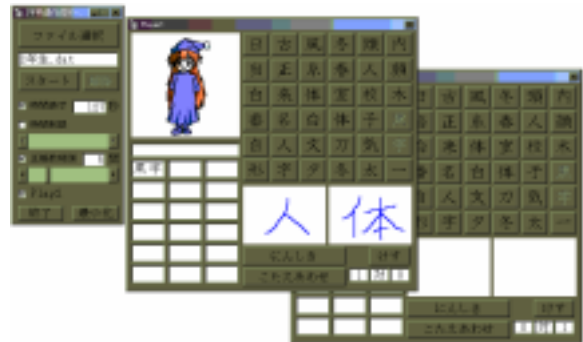


図3 教育用アプリケーションの一例

4. おわりに

対話型電子白板連動システムの利用により、電子白板の板書面の狭さを改善し、また電子ペンによる同時2入力を実現することができた。今後は、同時2入力機能を活かした教育用アプリケーションの開発を進めるとともに、それらのアプリケーションの詳細な評価を行い、本システムの有用性について検討していきたい。

参考文献

- 1) 坂東宏和, 根本秀政, 澤田伸一, 中川正樹: 黒板の情報化による教育ソフトウェア, 情報処理学会研究報告, 2000-CE-56, pp. 63-70 (2000).