

ネットワークを介した枠無し手書き認識のための クライアント・サーバ方式

依藤 充範 櫻田 武嗣 加藤 直樹 中川 正樹

東京農工大学工学部

1.はじめに

近年、PDA(Personal Digital Assistants)の普及や e-pen、アノトペン、Microsoft 社が発売した Windows XP Tablet PC Edition を搭載した端末などペン入力を基本とした機器が登場している。これらの機器には、標準の入力インタフェースとしてペン入力を用いられており、ペン入力はキーボード、マウスに代わる入力インタフェースとして注目されている。

ペン入力環境ではペンを用いて手書きで文字や図形を入力することができるが、ユーザが手書きで入力した内容を清書するには手書きパターンをコンピュータで認識する必要がある。ペン入力環境における文字の入力方法としてユーザの入力した手書き文字パターンを文字コードに変換する手書き文字認識技術があり、手書き文字認識では記入枠を固定した枠有り文字認識の他に枠にとられない自由な入力が可能である枠無し文字認識がある。しかし、現在の PDA などは CPU の性能やメモリ容量といった制約から枠無し文字認識の実装は困難である。

最近ではネットワーク環境の整備が進み通信速度の高速化やどこでもインターネットに接続できる環境が整いつつある。それによってネットワークで繋がっている他の機器の処理能力を利用するといったことが可能となっている。

そこで本稿は上述の問題点の解決策としてネットワークを用いたクライアント・サーバ方式の手書き認識手法について述べる。

2.基本設計

2.1 既存の手書き文字認識システム

現在 PDA などに搭載されている手書き認識システムは CPU の性能やメモリ容量といったリソースの制限からユーザの満足する認識結果を提供できない場合がある。また、枠無し文字認識などは実装することが困難である。さらに一般的な文字認識システムは一度システムに組み込んでしまうと以後システムの変更ができない場合が多い。また、アプリケーション内で手書き文字認識を利用したい場合個々のアプリケーションで文字認識システムを実装するか、アプリケーションを利用する機器に文字認識システムを組み込まれていなければならないといった問題点がある。

2.2 提案するシステム

そこで我々は CPU の性能やメモリ容量などのリソースに左右されることなく利用できるネットワークを用いたクライアント・サーバ方式の手書き文字認識システムを提案する。システムの概念図を図 1 に示す。

本システムはクライアントの動作を手書きパタンの入力および認識候補の表示・確認などに限定し、負荷の高い手書きパタンの認識処理をサーバで行うことでクライアントとなる機器の能力に依存しない認識システムを提供することができる。また、サーバ側の認識システムを変更することで多くのクライアントにシステムの変更を適応することができる。新たにアプリケーションへ文字認識システムを組み込む際、認識処理を行うプログラムや認識辞書などを用意する必要がないため組み込みの手間を省くことができる。また、クライアントはサーバと通信できればよいため、一般的なアプリケーションだけでなく Web ブラウザなどから手書き認識を利用できる。サーバはユーザからの手書きパタンの入力とそれに対する認識結果の対応を管理することができるため、ユーザの許可を得た上で手書き文字パタンのデータベースの自動拡張が可能[1]である。

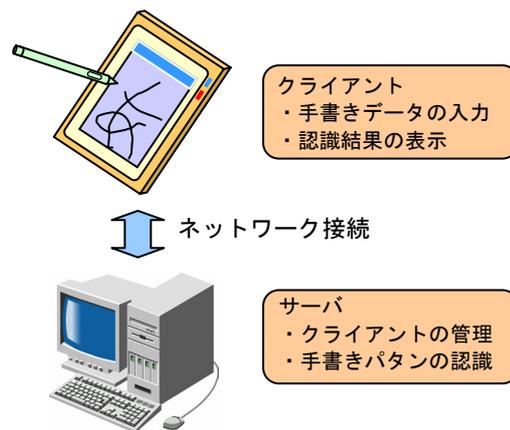


図 1 システム概念図

2.3 システム構成

本システムはクライアント、サーバの 2 つで構成する。

2.3.1 クライアント

クライアントはユーザからの手書きパタンの入力を受け持ち手書きパターン、認識候補の変更の情報などをサーバへ送る。また、サーバから受け取った認識候補の表示を行い、誤った認識結果が表示された場合は正しい認識候補を得るために誤認識修正操作などを行う。

2.3.2 サーバ

サーバは複数のクライアントからの接続を受け付けクライアントとの接続を管理し、クライアントから受け取った手書きパターンを認識エンジンへ渡す。サーバへ組み込まれた認識エンジンは受け取った手書きパターンをもとに認識処理を行い認識候補を取得しサーバを通じてクライアントへ送る。またデータベースなどと連携し手書きパタンの収集などを行う。

A prototyping of a client-server scheme for on-line recognition of free format handwriting through networks
Mitsunori Yorifuji, Takeshi Sakurada, Naoki Kato,
Masaki Nakagawa

2.4 実装方針

2.4.1 接続方法

本システムは世界各地の不特定多数の人が利用できるようにするため、クライアント、サーバ間の接続方法にTCP/IPを用いたインターネットおよびLANを用いる。

2.4.2 拡張性を確保した通信プロトコルの設定

手書き認識は文字だけでなく図形などの認識も今後可能であると考えられるため、クライアント、サーバ間でやりとりするデータの通信プロトコルは拡張性の高いものを考案する。

2.4.3 通信データ量の削減

インターネットを用いた通信システムでは個々のネットワークの通信速度が異なるため、通信速度の遅い回線を用いている場合ネットワーク間のデータの送受信に時間がかかりスタンドアロン型の認識システムに比べて動作速度が遅くなってしまいう可能性がある。そのため送受信するデータ量を少なくすることでデータ通信にかかる時間を減らしシステム全体の動作速度を高速にする。

3. 詳細設計

3.1 筆点列データ送信時のデータ量の削減方法

本システムではクライアントで入力した手書きパタンの筆点列データを文字列としてサーバへ送信しサーバで認識処理を行う。この際、筆点列データをそのままの形で送ってしまうと送信文字列が長くなりデータの量が多くなってしまいうため送信に時間がかかる。本システムでは各筆点列は、筆点列の始点を基準点した相対座標に変換して送信する。相対座標を使うことで、基準点からの変化量が少なくなるため筆点列データの値が小さくなり文字列に変換した筆点列データ量が小さくなる。

3.2 誤認識修正時の送信データの削減方法

手書き文字認識ではユーザの意図しない認識候補が認識システムから返ってくることもある。その場合ユーザは誤認識に対して希望する認識結果を得るための修正操作[2]を行う必要がある。また、誤認識修正時には再認識を必要とするものがあるため、クライアント、サーバ間で誤認識修正に必要な新たなデータの送受信が必要となる。しかしユーザの入力した誤認識修正操作で得られたデータをそのまま送受信するのではクライアント、サーバ間で多くのデータをやりとりすることになってしまう。多くのデータをクライアント・サーバ間でやりとりすると再認識結果を得るまでに多くの時間がかかるため、送受信するデータは小さくするべきであると考えられる。データ量を少なくするためには、クライアント側で修正に必要な最小限のデータを作りそれをサーバへ渡す方法が考えられる。

ユーザからの誤認識修正操作によって得られるデータとしては、囲み操作や区切り操作による分割位置変更のための筆点列データなどがある。クライアントは修正操作から得られた筆点列データをもとに誤認識修正の対象となる筆点の番号をサーバへ送信することで、誤認識修正時にクライアントからサーバへ送信するデータを少なくすることが可能である。

4. システムの試作

ネットワークを利用したクライアント・サーバ方式の手書き認識システムの実現可能性を確かめるためシステムの試作を行った。各システムはWindows2000で動作確認を行った。システムで使用した文字認識エンジンは筆者の所属する東京農工大学中川研究室で作られたオンライン枠無し手書き文字認識エンジン[3]である。作成したクライアントプログラム、サーバプログラムの実行画面を図2に示す。

本試作システムでは手書きパタンのストロークデータをクライアントからサーバ間へ送信する際、各点列の始点を中心とした相対座標系を用いて点列を再構成し送信することで、サーバへ送信する点列の文字数を減らす実装としている。

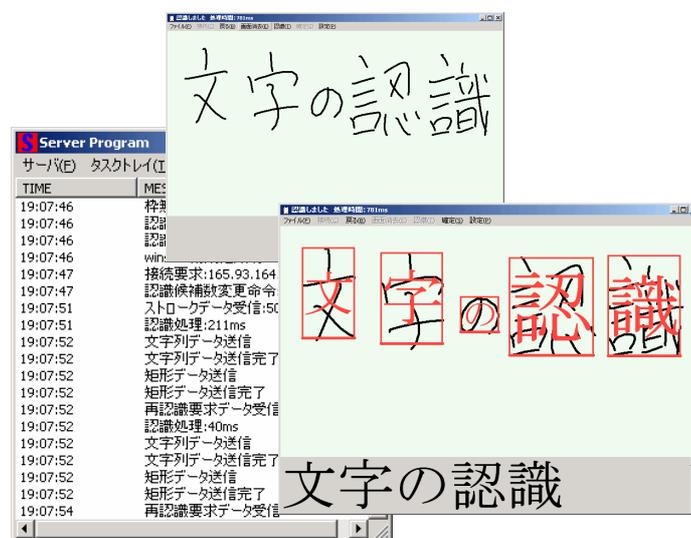


図2 クライアント、サーバの実行画面

5. まとめ

本稿ではクライアント・サーバ方式の手書き認識システムの方式設計について述べ、試作を通してシステムの実現可能性を示した。

本方式を用いることにより、CPU性能やメモリ容量の制約のあるPDAなどのモバイル機器でも高速・高精度なオンライン手書き文字認識が利用可能になると考えられる。また、今後各種の認識エンジンなどへの対応、拡張、評価実験などを行うことが必要であると考えられる。

参考文献

- [1] 櫻田他：ネットワークを介したクライアント・サーバ方式の枠有り文字認識の方式設計，ヒューマンインタフェースシンポジウム2002，pp.687-688(2002.9).
- [2] 坂東他：枠なし手書き文字列認識における誤認識訂正インタフェース，情報処理学会論文誌，vol.43，No.6，pp.1996-2005(2002.6).
- [3] 小沼他：筆記方向及び文字方向自由の枠なしオンライン手書き文字認識とそのインタフェース，情報処理学会第64回全国大会論文集，vol.2，pp.119-120(2002.3).