

HIの高度化を目指したメディア変換機能統合利用環境の構築 ～HIウェアの開発～

辻本修一、杉山博史、長谷川保、半田豊、橋本美奈子、竹林洋一
(株) 東芝 研究開発センタ
情報・通信システム研究所

コンピュータの使い勝手を決めるHIという観点からパターン認識や自然言語処理などの機能を簡単にかつ有機的に利用できる高度なヒューマンインタフェース環境(HIウェア)を開発している。HIウェア構築のために、OCRシステムや機械翻訳システムなどに組み込まれたエンジンを切り出し、それらを統一的に扱えるインタフェース仕様を策定し実装した。これにより種々のメディア変換機能を統一した方法で有機的に組み合わせて利用することが可能となった。本報告では、HIウェアを用いて開発したパッケージおよびHI応用システムについても紹介し、HIウェア利用の有効性を確認する。

Human Computer Interface Environment to Integrate MultiMedia Transformation Functions ～ HI ware ～

S. Tsujimoto, H. Sugiyama, T. Hasegawa, Y. Handa, M. Hashimoto, and Y. Takebayashi
R&D Center, Toshiba Corp.

e-mail: tsuji@eel.rdc.toshiba.co.jp

We have been developing human-computer interface environment where various kinds of media transformation functions are easily and organically available to develop advanced HI. The environment called HI ware consists of the servers provided with a variety of services including character recognition, speech recognition / synthesis, machine translation and so on, and the standardized interface specification which keeps consistency among the functions so that various kinds of HI applications can incorporate them in the common manner. In this paper, the feasibility of the development of applications and systems utilizing HI ware is also shown.

1. はじめに

近年、インターネットやPC利用者の普及によりオフィスや家庭でマルチメディア情報を受信したり編集したりする機会が増加し、マルチメディアへの夢と期待が高まってきている。しかしこれまでのマルチメディアシステムは複数のメディアデータ（テキストや音声）を単に入出力（記録／再生）しているに過ぎず、利用者の負担が増加しているのが現状である。今後、マルチメディア情報機器やマルチメディア応用システムの使い勝手を決めるヒューマンインタフェース（HI）を高度化し、利用者にとって使い勝手の良い情報機器を実現していくためにはマルチメディア情報のビットレベルの処理に加えて、内容に基づいた処理が必要となり音声認識や機械翻訳、文字認識といったメディア変換処理機能が必須となる。

しかしながら、これらのメディア変換処理技術はこれまでOCRや機械翻訳システムといった特定のシステムに結びつけられて開発が行われてきたため、他の応用システムへの考慮が十分になされなかった。その結果、オープン性に欠け、汎用のシステムに組み込んで利用することが困難となり用途が限定されていた。

一方、マルチタスク環境下における音声認識機能をサーバ化した使い勝手の良いマルチモーダルインタフェース[1]やネットワーク環境下における音声認識サーバ[2]、機械翻訳サーバ[3]、オンライン文字認識サーバについての報告がなされている。また、最近ではパッケージ化やOCXなどによる部品化や、MSやIBMによる音声処理のための標準インタフェース仕様（Speech API）の提唱が行われている。

しかし、これらはいずれも個々のメディア変換機能のサーバ化や部品化であり種々のメディア変換処理機能を包括的に扱おうとするものではなかった。

「HIウエア」[4, 5]は、文字認識や音声認識などの種々のメディア変換機能を様々なマルチメディア応用システムから簡単に利用できる環境を構築し、システムの使い勝手（HI）を向上させることを目的とする。

本報告では、まず、「HIウエア」の設計方針を述べ、次いで開発状況の現状を報告する。更に、これまでに構築したパッケージ及び応用システムを紹介すると同時にその評価を行う。

2. 方針

HIウエアの全体像を図1に示す。HIウエア構築のために、OCRシステムや機械翻訳システムなどに組み込まれたメディア変換機能エンジンを切り出し、

モジュール化を行う。メディア変換機能には連続／離散メディア処理、同期／非同期処理などがあるが、それらを統一的に扱えるインタフェース仕様を策定し、実装を行う。これにより、種々のメディア変換機能を統一した方法で有機的に組み合わせてアプリやシステムより利用することが可能となる。また共有辞書によりメディア変換機能の辞書を一本化することができ、辞書のメンテナンス及び性能を向上させることができる。

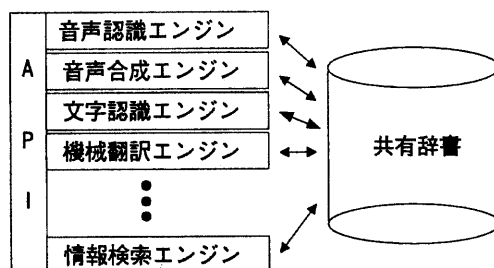


図1：HIウエア

2. 1 部品化

パターン認識や自然言語処理の技術は、これまで特定の製品に結び付けられて開発が行われてきた。例えば文字認識技術の場合で説明すれば、当社の文字認識技術は郵便番号読取装置で始まり、帳票などの定型書類を入力する光学式文字読取装置（OCR）としてオフィスで使われるようになり、更により汎用的なドキュメントリーダー[6]として一般的に使われるようになってきた。これは、高い認識率を得るために、認識対象を限定し、認識対象に合わせたきめ細かい辞書や処理のカスタマイズを行い、更に処理を高速化するために専用のハードウェアを必要としてきたからである。

しかしながら今後、より広い場面でパターン認識や自然言語処理の技術が利用されていくためには、特定システムに組み込まれて開発されてきた技術を一般化して部品化し、様々なシステムやアプリから簡単に利用できるようにする必要がある。そのためには、従来、場合毎にカスタマイズしてきた内容を分析し、例えば認識対象に応じて変更してきた辞書や内部データの構造を包含できるようなフォーマットを決め、対象毎に前処理／後処理の内容を切り替えられるように処理モジュールを切り分け、必要なパラメータの洗い出しなどを行うなどの一般化を行い、それに基づいた処理モジュールの実装を行わなければならない。

一般化については次節でさらに詳しく説明する。

2. 2 標準API

HIUエアは、様々なメディア変換機能を、ユーザやアプリ開発者が自由に組み合わせて簡単に利用できる環境を提供することを目的としている。このような環境を実現するためには、メディア変換機能を内包する従来の専用システム（機械翻訳システム等）からメディア変換機能モジュール部分を単に切り出し、コンポーネント化するだけでは不十分であり、アプリとメディア変換機能モジュールとの間に、メディア変換機能を利用するための標準化されたAPI（Application Programming Interface）を定義する必要がある。この標準化されたAPIは、認識、変換、合成処理を行うエンジン部分、学習を管理する辞書管理部分、非同期リアルタイムの入力データをエンジンに送るデバイス管理部分などメディア変換機能モジュールを構成する各処理部分の間にも同様に定義すべきである。

このように標準化されたAPIにより、アプリ開発者はメディア変換機能の種類（機械翻訳、文字認識等）毎に、個別のメディア変換機能モジュールに依存しない一定の決められた手順でそれらのメディア変換機能を利用することが可能となるため、異なる開発者が提供する複数のメディア変換機能モジュールをサポートするアプリを開発する場合、それぞれのメディア変換機能モジュールに依存したプログラム記述をしなくてもすむようになり、生産性を著しく向上させることができる。更に作成したアプリは、この標準APIに従って実装された任意のメディア変換機能モジュールと、プログラムの変更なしに自由に組み合わせて動作させることが可能になる。

一方、メディア変換機能エンジンの開発者をはじめとするメディア変換機能モジュール開発者は、標準APIに従ってメディア変換機能モジュールを実装することにより、このAPIを利用する全てのアプリからそのメディア変換機能を利用してもらえるようになる。加えてメディア変換機能モジュールの改良や変更をアプリとは独立して行うことができるため、そこで行われた改良の効果は標準APIを利用するすべてのアプリに反映されることができる。

標準APIによりユーザが得られる利益としては、メディア変換機能対応アプリ及びメディア変換機能モジュールの選択肢の増加、それらの自由な組み合わせ、そして競争によるメディア変換機能モジュールの精度向上の促進などが挙げられる。

上述した利点を考慮し、HIUエアではメディア変換機能をサポートする標準APIを提供することとし、

現在その策定作業を進めている。この標準APIは、以下の条件を満たすように策定する必要がある。

一般化： 特定の認識、翻訳、合成の方式（アルゴリズム、パラメータ）に依存する部分と依存しない部分を分離し、特定の方式に依存する部分では従来採用されてきた主要な方式に対応し、それらに特有のデータ、制御をサポートする。パラメータはできる限り一般的なデータ構造、単位、値域をもつ形式とする。

業界標準技術の採用： 分散オブジェクト技術、ソフトウェアコンポーネント化技術には、業界標準の技術を採用する。これにより、他のソフトウェアコンポーネントとスムーズに連携動作することが可能になる。また、一般的に使われているプログラム開発環境を利用し、標準APIに準拠したメディア変換機能モジュールや、この標準APIを利用するアプリを容易に開発することを可能になる。

マルチベンダー、マルチプラットフォーム対応： 複数のハードウェアプラットフォーム、ソフトウェアプラットフォームをサポートする。またマルチベンダー対応のオープンなAPIとする。

複数機能間の一貫性： 異なるメディア変換機能においても、インタフェースの機能粒度、記述形式をできる限り揃え、同じような動作制御やパラメータ操作を、同じような流儀で記述できるような一貫性のあるAPIとする。

高レベルAPIと低レベルAPIのサポート： メディア変換機能を簡単に利用するための高レベルAPIと、詳細な処理が行える低レベルAPIの両方をサポートする。

複数機能間の連携、辞書共有に関する新規機能のサポート： 複数のメディア変換機能の連携した実行、及び修正処理機能をサポートする。また、複数のメディア変換機能間での学習、辞書の共有もサポートする。

以上の条件を満たすAPIの策定を段階的に進めていくとともに、順次公開を行い、本APIの普及、業界標準化を目指していく予定である。

2. 3 共有辞書

共有辞書とはHIUエアに搭載されている複数のメディア変換機能の個別辞書を一本化した辞書で、あるメディア変換機能の辞書情報に生じた変更を他のメディア変換機能の辞書情報に自動的に波及させ、さらに波及させる辞書の範囲を制御することを可能とする。

従来、メディア変換機能はワープロやOCRといった個別のシステムとして実現され、システム毎に個別辞書を持っていた。これらの個別辞書の情報は異なるシステム間では共有されないため、辞書登録をはじめとする辞書のメンテナンス作業を個別辞書毎に何度もおこなう必要があった。辞書の量と質がメディア変換機能の品質を決定すると言っても過言ではないため、辞書の開発とメンテナンスには高い専門的知識に基づく膨大な作業が必要となる。従って、これらの辞書開発に係る作業の手間を軽減することは重要である。

そこで辞書メンテナンスの向上とそれに伴うメディア変換機能辞書の性能向上を目指し、個別辞書を一本化した共有辞書を開発する。図2に示すように、ユーザは共有辞書に対して辞書登録などの作業をおこなう。共有辞書は、あるメディア変換機能の辞書情報に生じた変更を他のメディア変換機能の辞書情報に自動的に波及させ、更に波及させる辞書の範囲を制御することにより、共有辞書への登録結果を必要に応じて各個別辞書に波及させることができる。その結果、例えばかな漢字変換辞書に登録した新語を、音声合成辞書へ自動的に登録することができるようになり、辞書登録作業の手間を軽減することができる。

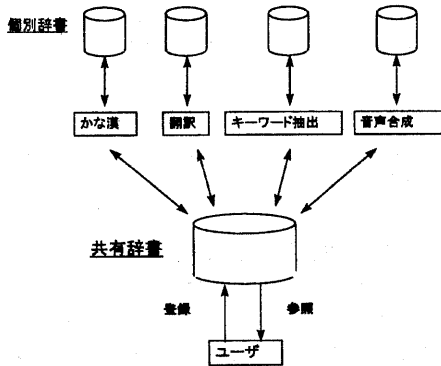


図2：共有辞書

3. H I ウエア

前章で述べたように、パターン認識や自然言語処理技術を標準化された機能モジュールにすることにより、既存の様々なシステムやアプリからそれらの機能を容易に組み込み、それらの機能を利用した高度なH I環境を容易に開発することができるようになる。

現在、筆者らは、前節で述べた方針に基づき開発を進めており、その第一段階として以下の機能について部品化を済ませている。

H I ウエアエンジン群： 音声認識、音声合成、文字認識（日本語／英語）、機械翻訳（英日／日英）、キーワード抽出、自動タグ付け、オンライン文字認識、暗号、抄録

それぞれは、Windows95 / NTを対象としたDLL、OCXとして実装されている（図3参照）。基本的に認識処理や自然言語処理のためのモジュールであり、必要に応じてパラメータ設定や後修正などのユーザインタフェースを提供するモジュールも含んでいる。

APIについては、現在は当社独自の認識／言語処理アルゴリズムに合わせたAPIの仕様になっており、

- ・エンジンを初期化するインタフェース
- ・エンジンをオープンするインタフェース
- ・エンジンのパラメータを設定するインタフェース
- ・エンジンを実行するインタフェース
- ・エンジンをクローズするインタフェース

を基本的なインタフェースとして持ち、必要に応じて

- ・前処理／後処理のインタフェース

なども提供している。現時点では、他の認識／自然言語処理アルゴリズムも考慮したAPIの標準化には到っていない。したがって、前節で述べた他のエンジンモジュールや辞書との互換性というユーザ利益を提供することは実現されていない。しかしながら、DLLやOCXの形態での部品化が行われたことにより、

- ・市販アプリから利用（マクロ機能からDLLやOCXを利用）
- ・ソフトウェア開発キットとして新規アプリの開発に利用

といった方法で種々のメディア変換機能を利用したアプリの開発が容易に行えるようになった。実際の開発事例などについては次節で紹介する。

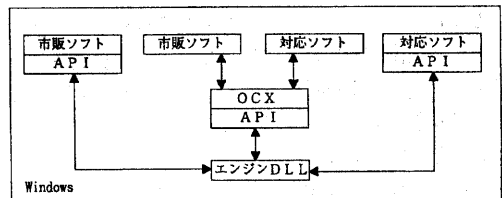


図3：システム構成

なお、共有辞書に関してフォーマットを決定し、現在、かな漢、音声合成、機械翻訳、キーワード抽出の辞書の統合が完成し、H I ウエアへ搭載中である。

4. 応用

H I ウエアの利用形態として、クライアントアプリとしてのパッケージとサーバアプリとしてのH I 応用システムがある。次にその一例を示す。

4. 1 パッケージ

種々のメディア変換機能を既存アプリから自由に呼び出して利用するためのマクロ集とメディア変換機能を組合せて利用するためのツールバーを作成した。

前者により通常利用しているお気に入りのアプリから種々のメディア変換機能を簡単に利用することができ、H I による付加価値化が可能となる。

後者は、従来、複数のメディア変換アプリの組み合わせによって実現していた複雑な連携機能を手軽に実行することを目指したものであり、例えば英文FAX画像の文書を日本語に翻訳して音声出力させたい場合、ドキュメントリーダー、英日翻訳、音声合成の各ボタンを順番に押したり、あるいは出力形態として日本語音声か欲しいと指定するだけで自動的にメディア変換を行ってくれる。

上記のマクロ集およびツールバーは現在、筆者らが所属する所内で評価中であるが、これまでにアプリとメディア変換機能との一体感があり、短い手順で直感的な操作が実現できているとの評価を受けている。ある試用者からは、通常、ある目的および明確な意図をもって何らかのアプリなどを動かしている際に、翻訳や音声合成などの他の機能を利用したいと思った時、従来ならそれらの機能アプリを探し出し、立ち上げる必要があったため、新しい行動をとるためのコストが高かったが、H I ウエアの利用により、新しい機能利用を行うための物理的および心理的コストが低くなったとの感想があった。つまり、H I ウエア利用により、少ない動作で新しい行動に気楽に移ることができるようになったため、新しい行動の変更に対するためらいや意識を低減することができ、その結果、行動のパターンの幅広くかつ複雑なものへの発展を実現できたと評価できる。これはH I 的な観点から捉えれば、利用者が現実世界で無意識的に行っている自然な振る舞いの一部を実現することができたと言える。

4. 2 知識情報共有システム

知識情報共有システム[7]では図4に示すように個人

や組織にストックされた知識や情報[8]をオンデマンドで検索・利用することを目的として開発した。知識情報共有システムは、図5に示すようにコンテンツを入力するためのマルチメディア情報構造化、ストック情報の理解・活用の鍵となる知識ベース、利用者の質問に応じてストックされた情報から検索を行うマルチデータベース検索および利用者との対話制御を行う対話インタフェースから構成されている。

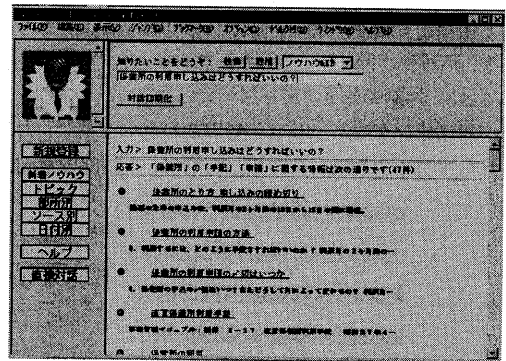


図4：知識情報共有システムのGUI

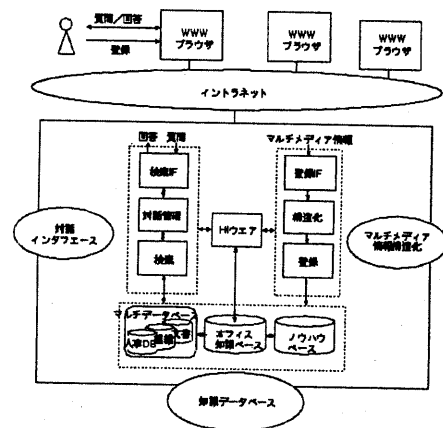


図5：知識情報共有システムの構成図

(1) マルチメディア情報構造化

利用者が登録したノウハウや報告文書などのテキスト、イメージ、映像情報などの多種多様なマルチメディア情報は検索・活用しやすいように構造化され蓄積される。入力されたマルチメディア情報はH I ウエアに搭載されたドキュメントリーダーや音声認識、英日翻訳などの機能を用い日本語テキストに変換され、キーワード抽出やタグ付け機能により構造化される。

(2) 知識データベース

知識データベースの情報を利用してH I ウエアの機能向上をはかることが可能となる。例えば、キーワード抽出はオフィス用語辞書を参照し、文書の種類毎に抽出キーワードの優先順位を決めるルールを用いて行っている。優先順位は出現頻度の語の重要性の重み付けをして計算する。オフィス用語は重要後として重みを高くし、一般名詞は重みを低くし、さらに一般的な名詞でキーワードとして相応しくない語を不要語として登録しておき、重みを0として計算する。計算の結果、評価値の高いものからキーワードとして抽出する。

(3) 対話インタフェース

対話インタフェースは、利用者がキーボード入力した自然言語より利用者の意図を抽出し、質問の意図に適合したデータの蓄積源(知識ベース/ノウハウベース)を検索して回答を返す。入力された自然言語はH I ウエアに搭載されたキーワード抽出や自然言語の解析処理機能により意図理解が行われ、データベースとの検索により回答が決定される[9]。

知識情報共有システムは現在、組織内で実践評価中であり、これまでに下記の3点の有効性が確認されている。

(1) 開発コストの低下

従来、メディア変換機能を利用した応用システムの構築には機能モジュールの作成だけでなく種々のノウハウが必要であったが、H I ウエアでは機能がコンポーネント化されているためシステムへの組み込みが容易であり開発コストを大幅に下げることができた。

(2) コンテンツ入力促進

知識情報共有システムが有効に機能するかどうかは枠組みの善し悪しではなくコンテンツの充実が重要であるが、H I ウエアによる各種メディア変換機能の活用により、利用者はテキスト形式以外に音声、紙の書類、映像情報のままでコンテンツを入力を行うことができるようになりコンテンツ入力コストを大幅に下げることができ、その結果、より多くのコンテンツを集めることができた。

(3) システムの利用範囲の拡大

H I ウエアの活用により利用者の質問形式の多様化が実現できた。本システムでは、自然言語処理機能により自由な表現を許しているが、今後、翻訳や音声認識機能の活用により英語による問い合わせや電話を経由しての質問入力が可能となり、より柔軟なシステムの構築が可能となった。

5. おわりに

H I ウエアはH I を向上させる機能モジュールを統合したミドルウェアである。本報告では、H I ウエアがマルチメディア情報をより効果的に分かりやすく入出力するための環境を提供し、Ask&Tellなヒューマンインタフェースをより簡単に開発できることを示した。今回開発したH I ウエアのAPIはアプリとメディア変換機能間の連携をおこなうものであり、アプリからのメディア変換機能のシームレスな呼び出しおよび利用を実現するものであった。今後はアプリとメディア変換機能間のより密接な連携を実現する標準化されたAPIの開発を推し進める。また、共有辞書を開発を加速し、標準APIや共有辞書による種々のメディア変換機能の統合化を進め、複数のメディア変換機能の密接な連携によるシナジー効果を狙う。

謝辞 知識情報共有システムの開発は、東芝研究開発センタの中山康子、真鍋俊彦他の諸氏との共同研究によるものであり、御支援と御助言を頂いた関係各位に深謝致します。

参考文献

- [1] 山田他, “実時間動作を考慮した音声認識サーバ”, 音響学会講演論文集, 2-8-2, 1994.
- [2] Rudnicky, A. et al., “Spoken language recognition in an office management domain”, IEEE, CH2977-7, 1991.
- [3] 伊藤他, “クライアント/サーバ型機械翻訳システムの学習方式”, 情処全大第 50 回, 5R-10, 1995.
- [4] 杉山他, “コモンH I サービス環境の開発”, 情処全大第 52 回, 2W-1, 1996.
- [5] 長谷川他, “コモンH I サービス環境の応用”, 情処全大第 52 回, 2W-2, 1996.
- [6] Tsujimoto, S. et al., “Major Components of a Complete Text Reading System”, Proc. of IEEE, Vol.80, No.7, 1992.
- [7] 中山他, “知識情報共有システム (Advice / Help on Demand) の開発と実践 ~オフィス知識ベースとノウハウベースの構築~”, 情処学会 HI 研究会インタラクシオン'97, 103-110, 1997.
- [8] 福井他, “コミュニケーション支援のための個人情報公開システム (PIP)”, 情処学会 HI 研資, HI64-8, 1996.
- [9] 真鍋他, “マルチデータベース日本語インタフェースの試作”, 情処全大第 51 回, 2D-2, 1995.