社団法人　情報処理学会　研究報告
IPSJ SIG Technical Reports

2005－ＡＶＭ－51（10）
2005／12／13

負荷分散型マルチプラットフォーム対応発話訓練システムの実装

梅田　耕佑†　村井　慶史††　鱒渕　俊児††　上野　歩美†　山川　仁子†
萱木　禎史††　宇佐川　毅††

† 熊本大学 大学院 自然科学研究科
†† 熊本大学工学部
〒 860-8555 熊本市黒髪 2-39-1
E-mail: †{kousuke,keishi,masu,uayumi,jin}@hicc.cs.kumamoto-u.ac.jp, ††{chisaki,tuie}@cs.kunamoto-u.ac.jp

あらまし　日本語を第二言語とする学習者を対象にした，自動音声認識と発話画像の取り込み機能を有する，日本語発話訓練システムの開発を進めている。本システムは，サーバ・クライアント方式で構成されており，従来顔画像の分離抽出等を行う画像処理部はサーバ側に配置し，C 言語で実装されていたが，クライアント数が増加すると，通信帯域とサーバへの負荷が増大し，応答時間が長くなる等の問題が生じる。この問題を解決するために，クライアント側に画像処理部を移行した。また，マルチプラットフォーム対応のため，Java により実装を行った。従来の処理系と新たに実装した処理系で，クライアント数を変動させて応答時間等の性能を比較し，その有効性が確認された。また、本システムは発話音声と発話画像のフィードバック機能を有しており，画像と音声の同期ずれについて検討した結果，知覚限界内に十分に収まることが確認された。さらに，フィードバックされる発話画像について，発話訓練システムでの有用性について主観評価し，有用性についても若干の検討を加えたので報告する。
キーワード　発話訓練システム，顔画像処理，マルチプラットフォーム，Java，負荷分散

Implementation of load-balancing type multi platform utterance training system

Kousuke UMEDA†, Keishi MURAI††, Shunji MASUBUCHI††, Ayumi UENO†, Kimiko YAMAKAWA†, Yoshifumi CHISAKI†† and Tsuyoshi USAGAWA††

† Graduate School of Science and Technology, Kumamoto University
†† Faculty of Engineering, Kumamoto University　2-39-1 Kurokami, Kumamoto, 860-8555 Japan
E-mail: †{kousuke,keishi,masu,uayumi,jin}@hicc.cs.kumamoto-u.ac.jp, ††{chisaki,tuie}@cs.kumamoto-u.ac.jp

Abstract　An utterance training system of Japanese which equips an automatic speech recognition and face image caption is developing in order to assist Japanese learners as the second language. Although this system was originally designed as server-client system and the image processing part which extract a face image was implemented on server-side by the C language, the response performance became a problem when the number of clients were increased due the network band width as well as performance of server-side system. In order to overcome this problem, an image processing part is implemented on client-side by Java language aiming at multi-platform system. A performance of a new implementation is examined while the number of clients is varied, and the results shows the effectiveness of a new system. Also the synchronization of image and speech is examined and the average time difference is sufficiently smaller than the perceptual threshold. Results of simple subjective evaluation supports the effectiveness as the image feedback part of the utterance training system.
Key words　utterance training system, facial image processing, multi platform, Java, load balancing

1. まえがき

近年，インターネットの普及により，ネットワークを介したe-Learning システムを用いた学習が盛んに行われており，様々な種類の e-Learnig システムが開発されている。また，国内外を問わず日本語学習者は増加する傾向にあり，日本語教師の人的不足が問題となっている。そこで，この問題を解決するために，日本語学習を支援する e-Learnig システムが必要となる。現在日本語学習者向けの発話訓練システム [1] の開発を進めており，この訓練システムでの発話画像の活用を目指した研究を行っている。

一般的に，発話訓練において，口唇の動作を学習者に提示することで，効率的な学習が可能であるとされている [2]。そのため，本システムは，音声認識の結果と共に，発話音声と発話画像 (口唇領域が抽出された画像) をユーザにフィードバックする。

本システムは，サーバ・クライアント方式で構成されており，従来保存した画像に対して顔画像の分離抽出等の処理を行う画像処理部はサーバ側に配置し，C 言語で実装されていた。そのため，クライアント側で発話画像を録画した後，サーバ側へ送信し，サーバ側でその画像を処理していた。システム利用者の増加に伴い，サーバ側の負荷が増大し，画像データを送信することで，ネットワークの負荷が増大する。そこで，画像処理部の処理をクライアント側で実現する必要がある。また，マルチプラットフォーム対応のため，Java により実装を行った。

画像処理部について，従来の処理系と新たに実装した処理系で，負荷を変動させてクライアント側の応答時間やサーバ側の処理時間等を比較し，その有効性を確認する。また，本システムの有する発話音声と発話画像のフィードバック機能において，自分の発話内容の理解のために，両者は同期している状態でフィードバックする必要がある。そこで，画像と音声の同期ずれについて定量的に検討する。

さらに，画像処理部で処理された発話画像について，発話訓練システムでの有用性について主観評価し，有用性についても検討する。

2. 発話訓練システム

本章では，システムの使用例と構成について述べる。

2.1 システムの使用例

本システムは，ユーザの発話音声と発話画像を保存し，保存した音声と画像に対して音声認識と，画像処理を行い，ユーザに音声認識結果と共に発話音声と発話画像をフィードバックする機能を有している。尚，本システムは，単語の発話訓練システムであるため，発話時間は 1 秒程度を想定している。ユーザがシステムを使用している様子を図 1 に示す。本システムの使用時のユーザとシステムの流れを以下に記述する。

(1) セッティング

ユーザは図 1 に示すような状態に機器をセッティングする。本システムを使用するにあたり，インターネットに接続された計算機の他に，計算機に接続するヘッドセットとカメラを必要とするが，カメラの使用/不使用は選択可能である。

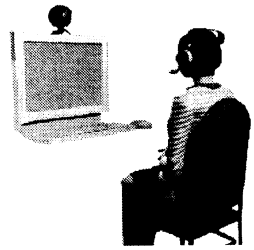

図 1 ユーザがシステムを使用している様子

(2) 学習問題の選択

ユーザはシステムにログインし，学習したい単元の聞き取り訓練，発話訓練などの訓練モードを選択し，問題に対して解答する。

聞き取り訓練を選択した場合，聞こえた単語を選択肢から選択する多項選択方式を用いている。発話訓練を選択した場合，問題に対する解答をマイクロホンに発話する。解答の際には，ユーザが任意のタイミングで発話ボタンを押してから，解答する。ユーザが発話ボタンを押すと，システムの録画と録音が開始する。発話終了の判断は，音声区間が a $[ms]$ 連続した後，非音声区間が b $[ms]$ 連続した場合とする。ここで，発話音声のレベルが設定した閾値を超える連続する区間を音声区間とし，域値に満たない連続する区間を非音声区間とする。今回は，$a = 200$ $[ms]$，$b = 500$ $[ms]$ とする。

(3) 発話データの保存

ユーザの発話データをシステムが保存する。発話訓練の場合は発話音声と発話画像をそれぞれ保存する。

(4) 発話データの処理

システムが，発話データの処理を行う。音声データはサーバ側に送信され，サーバ側で音声認識される。画像データはクライアント側で画像処理され，ユーザの顔領域または口唇領域が抽出される。

(5) 認識結果の表示

クライアント側に発話音声に対する音声認識の結果が送信される。

(6) 発話データの参照

ユーザは発話音声と処理された発話画像を参照することが可能である。また，教師による正しい発話音声と発話画像を参照することも可能である。

(7) 次の問題に進む。または，学習を終了する。

2.2 システム構成

サーバと通信帯域の負荷分散のため，システム構成の変更を行った。変更前のシステム構成を図 2(a) に示す。変更前は，音声と画像はサーバ側で処理していたため，クライアント数が増加すると，サーバに負荷が集中して応答時間が長くなるという問題点があった。本システムの利用環境は，CALL 教室や学習者の自宅など，インターネットに接続されている計算機環境を想定している。想定環境下においては，通信帯域が狭い場合が考えられる。そのために，サーバ・クライアント間の通信量を

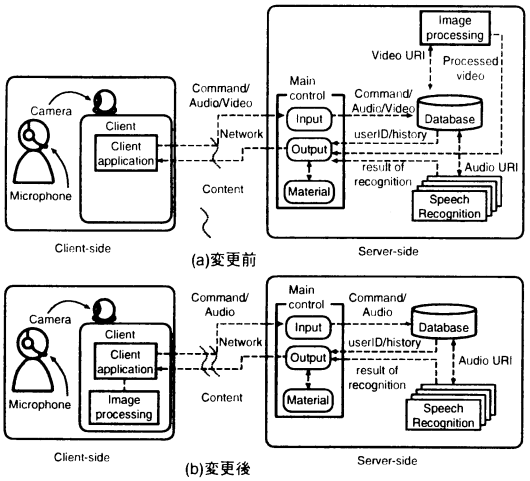

図 2　システム構成 (a):変更前, (b) 変更後

可能な限り削減する必要がある。そこで, 画像処理部をサーバ側からクライアント側へ移行した。これにより, サーバとネットワークの負荷分散が可能であると推測できる。変更後のシステム構成を図 2(b) に示す。

3. 画像処理部

本章では, まず, 画像処理部における口唇領域の推定法について述べ, システムの画像処理部の全体の処理の流れとプログラミング言語の変更について述べる。

3.1 口唇領域の推定

ユーザの利用環境を考慮すると, CPU の性能が低い計算機を用いて, システムを利用する場合が考えられる。そこで, 画像処理部の処理時間を可能な限り短縮しなければ, クライアント側の待ち時間が長くなる可能性がある。

1 章で述べたように, 口唇領域は発話訓練のための重要な要素の一つであり, その抽出については様々な研究がなされている。その中には, 色情報を用い閾値を決め抽出する方法 [4] や, エッジ情報を用いて抽出する方法 [5] がある。しかし, その色情報が肌色情報と酷似しているため, ユーザの個人性, 訓練環境に十分対応することは困難である。これに対応するために, 部分的テンプレートを用いる方法 [6], などが提案されているが, テンプレートの大きさを変更しながらのマッチングとなるので, その演算量が増大してしまう。

他にも発話時のフレーム間差分により, 口唇領域を抽出する方法も考えられるが, 口唇領域の動きと同時に他の部位も不確定で動く可能性が高く, 遠隔学習支援システムでの応用を考えた場合, その実用性は低い。そこで, 口唇領域を抽出するのではなく, 複雑な背景下でも照明条件の影響を受けずに比較的安定に検出できる両目を先に検出し, その位置情報から口唇領域を推定する手法を用いる [7]。

3.2 画像処理の流れ

画像処理の流れを図 3 に示す。

（1） RGB 画像から YUV 画像への変換

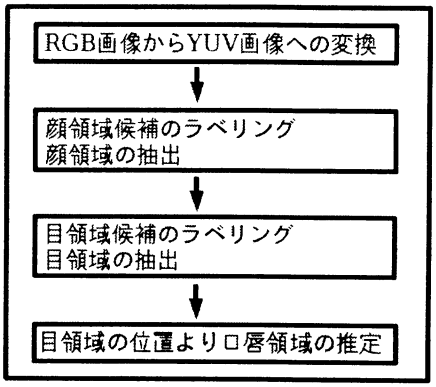

図 3　画像処理の流れ

顔画像候補の抽出のために, キャプチャーされた RGB 画像から比較的照明の影響を受けにくい YUV 画像へ変換する。YUV とは輝度信号 (Y) と、輝度信号と赤色成分の差 (U), 輝度信号と青色成分の差 (V) の 3 つの情報で色を表す形式である。変換式を以下に示す [3]。

$$Y = 0.2990R + 0.5870G + 0.1140B \quad (1)$$

$$U = 0.1684R - 0.3316G + 0.5000B \quad (2)$$

$$V = 0.5000R - 0.4187G - 0.0813B \quad (3)$$

指定した UV 値の域値を参照し, 肌色領域と背景を分離する。YUV 変換によって背景と分離された領域を顔領域候補と呼ぶことにする。

（2） 顔領域候補のラベリング

UV の域値により分離された顔領域候補をラベリングする。ラベリングとは, 連結する全ての画素 (以下連結成分) に同一のラベル (番号) をつけ, 異なった連結成分には異なった番号をつける処理のことである。次に, ラベリング処理によって連結された領域の面積を算出し, 最大の面積の領域を顔領域とする。

（3） 口唇領域の推定

口唇領域は直接検出するのではなく, まず, 目領域をの対を抽出し, その位置関係から口唇領域を推定する。

（a） 目領域候補のラベリング

人間の顔特徴の位置と, 計算コストの削減を考慮して, 抽出された顔領域の上半分の領域に対して顔領域と判断されていない領域のラベリングを行なう。ラベリングにより連結された領域の大きさの閾値を設定し, 目領域候補を抽出する。

（b） 目領域の決定

目領域候補として抽出された任意の 2 点の領域の中心座標に対して, 以下の 2 つの条件を満たす対を目領域の対とする。

- 2 点のなす角度を θ とすると, $-10^\circ < \theta < 10^\circ$ である。
- 顔領域の横幅に対する 2 点間の距離の比率を ω とすると, $0.2 < \omega < 0.7$ である。

眉の対を目の対と誤認識してしまう可能性があるため, 目の対が複数個検出された場合には, 最も下の対を目領域とする。

（c） 口唇領域の推定

検出された目領域の位置関係より, 大まかな口唇領域を推定

する。また，目領域の対が検出されなければ，顔領域の下半分を口唇領域とする。

3.3 Java 言語での実装

画像処理部は，以前は C 言語で記述していた。しかし，C 言語は汎用性が低いため，動作環境が限られてしまう。実際，本システムの画像処理部は Linux 環境下で C 言語で開発していたため，Linux 系の OS でしか動作していなかった。そこで，画像処理部を C 言語から汎用性の高いプログラミング言語である Java 言語への移植を行った。

Linux 系の OS をもつユーザが本システムを利用すると仮定した場合，画像処理部をサーバ側からクライアント側に移行すると，クライアントの計算機の設定を変更し，カメラのドライバをインストールしてカメラを認識させる必要がある。その設定をユーザが行う事で，手間と時間を要してしまう。ユーザが気軽にシステムを利用できないため，利用ユーザが減少してしまう可能性があり，設定変更に失敗すると，システムにカメラを利用できない。そこで，画像処理部を Java と QuickTime for Java を用いて記述した。

QuickTime for Java は，ストリーミングの音声や画像などのマルチメディアを作成するマルチプラットフォーム API のセットを提供する。QuickTime for Java を使用することにより，QuickTime ムービーを録画、再生したりするような Java プログラムの作成が可能となる。これにより，クライアントシステムをダウンロードすれば，Java と QuickTime for Java が動作する環境であれば，カメラのドライバをインストールするだけで，本システムの利用が可能となる。

Linux 系の OS では，QuickTime for Java が動作しないため，カメラを使用してシステムの利用はできないが，Windows, Mac OS ではカメラの動作が可能となる。また，カメラを使用しないのであれば，Java が動作する環境であれば，Windows, Mac OS, Linux 系の OS において本システムを利用することが可能である。これにより，本システムの利用にあたり，ユーザの負担の軽減が期待される。

3.4 Java 言語の欠点

Java は実行コードを中間コードにすることによって、Java 実行環境（Java virtual machine）上でインタプリタ形式で動かしているため，実行速度が遅くなってしまうという欠点がある。実行速度は CPU に依存するところが大きいため，解決策の一つとしては，CPU の高性能化が挙げられる。しかし，プログラムのコードを効率的に記述することによっても，実行速度の高速化が可能であると考えられるため，無駄無く効率的にプログラムを記述する必要がある。

4. 音声と画像の同期

本システムは発話訓練の機能を有しているため，フィードバックされた音声と画像が同期していなければ，ユーザが自分の発話の内容を正しく理解できない可能性がある。そのため，音声と画像は同期している状態でユーザにフィードバックされる必要がある。

音声と画像のタイミングについては，アナウンサがニュース原稿を読む時の同期ずれの検知限は音進み約 45 [ms]，音遅れ約 125 [ms]，許容限は音進み約 90 [ms]，音遅れ約 185 [ms] という報告がある [8]。音進みとは，画像よりも音声の方が進んでいる状態，音遅れとは，画像よりも音声が遅れている状態，検知限とは，同期ずれが 分かる・分からない の境界，許容限とは，同期ずれが分かるが，気になる・気にならない の境界である。

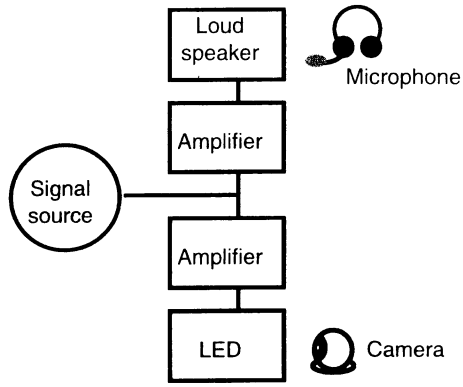

図 4　音声と画像の同期ずれの測定に用いた回路構成

そこで，本システムを使用して取得した音声と画像の同期ずれについて定量的に計測するため，図 4 に示すような回路構成で実験を行った。実験において，あらかじめ 2 チャンネルのサイン波の音声ファイルを作成しておく。両チャンネルのサイン波の on-set は同時刻にしておく。図 4 に示すように，一方のチャンネルからの音声はスピーカから出力する。他方のチャンネルは発光ダイオードを制御する信号として利用し，発光ダイオードが接続されたアンプに入力する

実験の手順としては，まず，2.1 章に示す手順でシステムにログインし，発話訓練モードを選択する。問題に対する解答のために，発話ボタンを押してから録画と録音が開始している状態で，あらかじめ作成した 2 チャンネルの音声を出力する。システムへの入力音声はスピーカから出力された音声とし，入力画像は発光ダイオードをキャプチャーした画像とし，それぞれを保存する。それをオフラインでフレーム単位で解析する。音声信号が入力されるタイミングとダイオードが発光するタイミングを比較し，音声と画像の同期ずれを定量的に計測する。
本システムでは，ヘッドセットに付属したマイクロホンへの音声入力を想定しているため，マイクロホンとスピーカーとの距離は，50 [mm] に設定した。

実験の計算機環境を表 1 に示す。この計算機環境において，画像のサイズは 320 × 240 [pixel]，フレームレートは 30 [fps]，音声のサンプリング周波数は 16 [kHz]，量子化ビット数は 16 [bit] に設定した。

音声と画像の同期ずれを 30 回計測し，その平均を求めた結果，両者の同期ずれは，音遅れが 43.5 [ms] であった。これは，参考文献 [8] における検知限の範囲内に値である。これにより，本システムにより録画・録音した発話データを，ユーザにフィードバックするのに問題ないことが推測される。

表 1　計算機環境

OS	Mac OS X
OS version	10.3.9
CPU	PowerPC G4 1.25[GHz]
Memory	256[MB]
Java version	1.4.2_09
QuickTime for Java version	6.1

表 2　使用した機器

機器名	メーカー	製品名
カメラ	Logitech	QCam Orbit
ヘッドセット	sony	STEREO HEADSET DR-220

5. 評価実験

本章では，以下の 2 つの評価実験について述べる。

評価実験 1　システムの処理系の変化による性能評価

評価実験 2　画像処理されてユーザにフィードバックされる画像の評価

5.1 実験環境

実験に使用する機器を表 2 に，示す。評価実験 1 の計算機環境を以下に示す。

クライアント側

- ハードウェア環境
- CPU : Pentium4 2.0[GHz], Memory : 512[MB] × 10 台
- ソフトウェア環境
- OS : RedHat Linux
- OS version : 7.3

サーバ側

- ハードウェア環境
- CPU : Pentium4 2.0[GHz], Memory : 512[MB] × 1 台
- CPU : Pentium4 1.6[GHz], Memory : 512[MB] × 7 台
- ソフトウェア環境
- OS : RedHat Linux
- OS version : 7.3

尚，クライアント側の計算機とサーバ側の計算機は 100baseTx で接続されている。

5.2 実　　験

5.2.1 評価実験 1

システム構成の変更前と後でユーザ数の変化による様々なデータを比較する。比較する項目を以下に示す。

- サーバ側のプログラムサイズ
- サーバ・クライアント間の一秒あたりの通信データ量
- サーバ側の処理時間
- クライアント側でユーザの発話が終了してから，音声認識の結果が出力されるまでの応答時間

システム構成の変更前は，保存した発話音声と発話画像をクライアント側からサーバ側へ送信し，サーバ側で音声認識処理と画像処理を行う。変更後は，保存した発話音声はクライアント側からサーバ側へ送信し，サーバ側で音声認識処理を行い，保存した発話画像はクライアント側で口唇領域の抽出処理を行う。

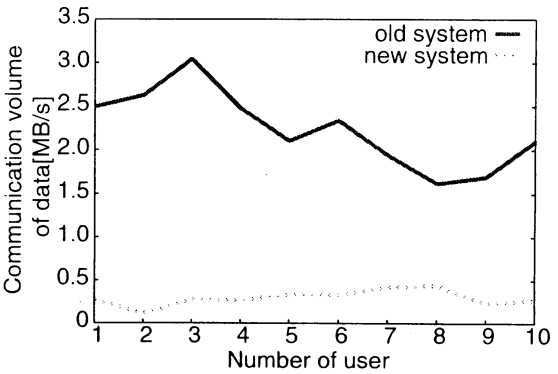

図 5　ユーザ数の変化によるサーバ・クライアント間の 1 秒あたりの通信データ量の推移

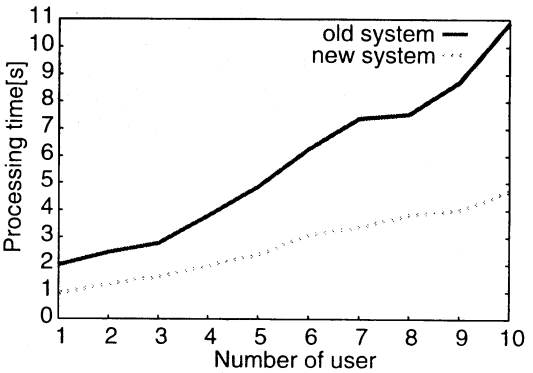

図 6　ユーザ数の変化による処理速度の推移

本システムは単語の発話訓練システムであるため，発話音声と発話画像は想定時間内の 1.0 [s] のデータを用意した。データサイズは，音声が 32 [kB]，画像が 6.8 [MB] である。

実験は，実験用に作成したクライアント側のアプリケーションをクライアント側の計算機で動作させることにより行った。クライアントアプリケーションを実行すると，自動でログインし，教材情報を取得し，用意した発話データをサーバ側に送信し，結果を取得する。クライアント側がデータを送信し終ってから，結果を取得するまでの時間を応答時間とし，サーバ側がデータを受け取ってから音声認識処理が終了するまでの時間処理時間とする。また，サーバ側に最大限の負荷をかけるため，複数台のクライアントマシンから同時にクライアントアプリケーションを実行する。同時に使用するクライアントマシンを 1 台 ～ 10 台と変化させ，それぞれの場合においてサーバ側で 100 回処理し，通信データ量，応答時間，処理時間それぞれの平均を算出した。結果のグラフを図 5，6，7 にそれぞれ示す。また，サーバ側のプログラムサイズは，92% に削減された。

5.2.2 評価実験 2

ユーザにフィードバックする画像について，発話訓練に役立つかという視点で，以下に示すそれぞれのパターンにおいて評

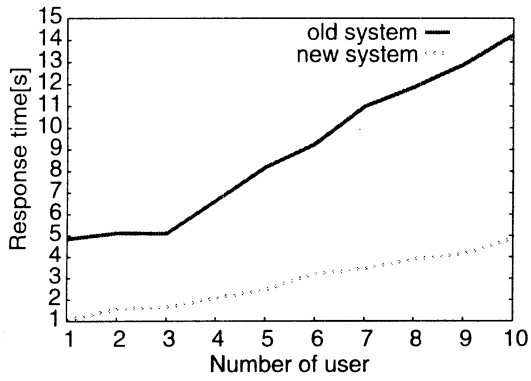

図 7 ユーザ数の変化による発話が終了してから結果が出力されるまでの応答時間の推移

図 8 出力画像の例 (左から 入力画像, 顔領域を抽出した画像, 口唇領域を抽出した画像)

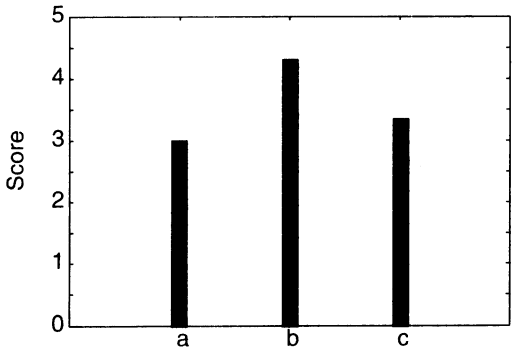

図 9 出力画像の評価 (a):入力画像, (b):顔領域を抽出した画像, (c):口唇領域を抽出した画像

価実験を行った。被験者は, 成人大学生 16 人 (男性 11 人, 女性 5 人) で, 評価する画像サイズは, 320 × 240 [pixel] である。

- 入力画像 (処理していない画像)
- 顔領域を抽出した画像
- 口唇領域を抽出した画像

それぞれの画像の例を図 8 に示す。

評価は 1 - 5 の 5 段階であり, その内容を以下に示す。

- 1 : 悪い
- 2 : どちらかと言えば悪い
- 3 : どちらとも言えない
- 4 : どちらかと言えば良い
- 5 : 良い

実験結果を図 9 に示す。実験結果より, 顔領域を抽出した画像が抽出した画像が最も良い結果となった。口唇領域を抽出した画像は, 高い得点をつける人と低い得点をつける人とで分かれてしまったため, 結果的に得点が下がってしまったと思われる。しかし, フィードバックする画像は, 学習開始時に選択可能とすることで, ユーザが自分にあったフィードバック画像で学習を進めることが可能となると考えられる。

6. ま と め

日本語発話訓練システムの開発を進めており, システム利用者の増加への柔軟な対応のためシステムの処理体系の変更を行った。

本システムはサーバ・クライアント方式で構成されており, 通信帯域の負荷分散を目的として, 以前はサーバ側で位置していた画像処理部をクライアント側へ移行した。この変更により, 保存した画像データはクライアント側で画像処理される。また, マルチプラットフォーム対応のため, C 言語で実装されていた画像処理部を Java により実装を行った。画像処理部について, 従来の処理系と新たに実装した処理系で, 負荷を変動させてサーバ側の処理時間やクライアント側の応答時間等を比較し, その有効性を確認した。

また, 本システムは発話音声と発話画像のフィードバック機能を有しており, 画像と音声の同期ずれについて定量的に検討した結果, 知覚限界内に同期ずれが収まっており, 本システムにより録画・録音した発話データを, ユーザにフィードバックするのに問題ないことが示された。

さらに, 画像処理部で処理された発話画像について, 発話訓練システムでの有用性について主観評価を行い, 結果より, 画像処理部の有用性が示された。

今後の課題として, クライアントアプリケーションをダウンロードする必要が無くなるように, クライアントアプリケーションをブラウザで利用可能にすることが挙げられる。

文 献

[1] 上野 歩美, 岡田 航生, 梅田 耕佑, 山川 仁子, 苣木 禎史, 宇佐川 毅, "発話評価フィードバックを有する遠隔日本語訓練システムにおけるユーザインタフェースの検討," 信学技報, ET2004-92, (2004).

[2] 文化庁, "日本語教育指導参考書 1 音声と音声教育," 大蔵省印刷局, 東京, (1998).

[3] 永山 貴司 他, "指先軌道形状による視覚インタフェースの試み," 第 9 回画像センシングシンポジウム講演論文集, pp. 329 - 332, (2003).

[4] 寺田 賢治, 山中 理聖子, 大恵 俊一郎, "口のカラー動画像を用いた音韻認識," 電学論 D, vol119, No.1, pp. 37 - 43, (1999).

[5] Brunelli and T.Possio, "Face Recognition: Features versus Templates," IEEE Trans PAMI, vol. 15, no. 10, pp. 1042 - 1052, (1993).

[6] 宋欣光, 李七雨, 徐剛, 辻三郎, "部分特徴テンプレートとグローバル制約による顔部品特徴の自動抽出," 信学論, vol. J80-D-II, no. 8, pp. 2178 - 2185, (1997).

[7] 中島 雄平, 梅田 耕佑, 苣木 禎史, 宇佐川 毅, "日本語発話訓練のための顔画像処理 -顔データベースの構築-," 信学論, vol. 104, no. 494, pp. 73 - 78, (2004).

[8] "http://www.nhk.or.jp/strl/publica/dayori/dayori9705/kaisetsu1-j.html ", 赤井田 卓郎, 黒住 幸一, 岡田 清孝, 林 俊一, 深谷 崇史," リップシンク～映像と音声のタイミング～," NHK 技研だより, (1997).