

モバイル・ビデオ会議システム - アーキテクチャ

4W-3

梶谷浩一, 小林真, 野田晴義, 坂入隆, 篠崎雅英, 前田潤治

日本アイ・ビー・エム(株) 東京基礎研究所



図 1: システムの外観

1 はじめに

昨今のノートブック PC の普及で, 出先で PC をネットワークに接続して使うという新しい使用形態が一般化しつつある。企業を例にとると, 現在は, 出先からオフィスのサーバにアクセスして, ファイルをダウンロードしたり, データベースを検索したりするのが主流である。しかし, オフィスにいる社員と同じ資料を見ながらインタラクティブに意見を交換しあったり, アドバイスを受けたりすることの出来るビデオ会議システムは, より創造的な計算機の使用形態である。

われわれはノートブック PC(CPU は PowerPC 603e 100MHz) 上で動作するビデオ会議システムを作成した。

音声, 画像の圧縮伸長, データと音声のマルチプレキシング (Voice-Over-Data) を全て CPU で実行する。

ノートブック PC を接続するネットワークとして「普通の電話回線」を使用する。理由は, 世界中のどこにいても「普通の電話回線」が存在する可能性が (ISDN などよりは) 高いから, である。

LAN と異なり, 電話回線では接続要求があった時点で誰からかかってきたかシステムにはわからない。したがって, 必要ならば接続時に, 顔写真などのユーザ情報を交換する, 等の工夫が必要だった。また, バ

Mobile video conference system - Architecture
Kohichi Kajitani, Makoto Kobayashi, Haruyoshi Noda,
Takashi Sakairi, Masahide Shinozaki, Junji Maeda
IBM Research, Tokyo Research Laboratory

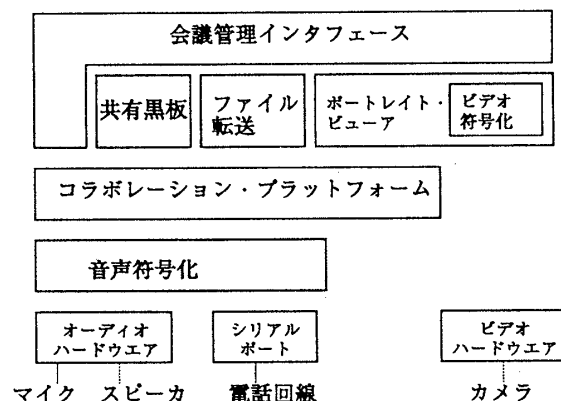


図 2: システムの構成

ンド幅が狭いので, 効率的な音声, ビデオ圧縮法を採用すると同時に, 共有黒板はなるべくデータ転送量を抑えるようにつくられており, かつ, ファイル転送中は, ビデオや音声の通信量を動的に抑える仕組みを付加してある。

現時点で用意したアプリケーションは, 「共有黒板」, 「ファイル転送プログラム」, 「ポートレイトビューア」(ユーザのビデオ画像を表示), 「会議管理インタフェース」(会議情報とユーザ情報を管理) である。

2 システム構成

図 1 にシステムの外観を示す。液晶パネルのすぐ上に CCD カメラが装着されている。28.8Kbps の V.34 モデム PC カードを電話回線に接続している。マイクはキーボードと液晶パネルの間に, スピーカーはキーボードの手前に装着されている。

CPU で音声と画像の圧縮を行うので, PC カードモデムとカメラ以外のデバイスを必要としない。

図 2 にシステムの構成を示す。

本システムは AIX 4.1 上で動いているが, 会議システムは, 本来, 様々なプラットフォーム上で稼働し, かつ, 互いに通信可能であることが望ましい。そのため, 本システムのアプリケーションは, AIX, OS/2, Windows 3.1, Windows NT, Windows 95 上で, ソースコード互換であるように書いた。

以下に, 図 2 の音声符号化, ビデオ符号化, コラボレーション・プラットフォームについて説明を加える。

2.1 音声符号化

マイクからの音声入力は、8KHz、8ビットでサンプリングされる。これをGSM方式で、9.6Kbpsに圧縮する。実際は、無音区間を検知することで、9.6Kbps以下の通信量しか必要としない。またエコーキャンセルの機能を備えているので、スピーカーホンとして使用できる。

音声の圧縮伸長を行うプロセスの優先度は他のプロセスよりも高く設定されている。また、音声データの“転送の優先度”は、他のアプリケーションデータのそれよりも高く設定されているので、大きなデータの転送が音声データの転送を妨げることがない。これらの工夫により、設計上、圧縮伸長による音声の“遅れ”は約60ms(ミリ秒)に抑えられる。

音声とビデオの同期機能は備えていない。

2.2 ビデオ符号化

カメラから得られたNTSCのビデオ信号は、16ビットのYCrCb 4:2:2のフォーマットでデジタル化され、最終的にメインメモリに4:1:1のフォーマットで転送される。この部分はQCIFサイズ(176x144)で30フレーム/秒のバンド幅がある。メインメモリに転送された画像データは、MPEG-1方式で圧縮される。MPEG-1を採用した理由は、カメラから入力される動画像も、CD-ROM等に蓄積されている動画像も統一的手法で扱うため、である。

DCTをPowerPC上で効率よく実行するために“LinzerとFeig”法[1]を用いた。これは、PowerPC内の整数ユニットと浮動小数点ユニットを常時並列に使う方式である。通常の方式に比べて約2倍の性能を得ることができた。

また、動き補償を行う際のフレーム間類似度の判定に関して新しい方式[2]を導入した。

2.3 コラボレーション・プラットフォーム

コラボレーション・プラットフォームは以下のサービスを包括的に行うための層である。

- セッション管理
- アプリケーション管理
- チャンネル管理
- トークン管理
- プロセス間通信

われわれの作成したアプリケーションはすべてこの層の上に作られている。

この層は、会議用のアプリケーションを構築するためのライブラリであるLakes[3]の上にC++で我々独自のAPIとして実装してある。この層を導入した理由は、(1)Lakesの提供する機能が多岐に渡りAPIが複雑であったこと、(2)将来Lakes以外のAPI(例

えば、T.120[4])上で動かす余地を残しておきたかったこと、の二つである。

実際この層はLakesに比べて少ない数のAPIで構成されているにもかかわらず、われわれのアプリケーションを作るのに十分な機能を有している。また、この層は会議アプリケーションを作るために本質的に必要な概念を簡潔に表現しているため、Lakes以外のライブラリ上に載せ代えるのも容易である。

本システム自体は2人のユーザを前提としているが、コラボレーション・プラットフォーム層とビデオ以外のアプリケーションは3人以上のユーザによる会議の場合も実際に正しく動作する。

以下に各サービスの説明を行うが、各サービスは、(1)ライブラリが用意している機能を利用するためのメンバー関数と(2)システムからのイベント通知を受けるためのcallback関数からなる。

セッション管理は、会議の開始、終了を行う。またあるユーザ(ノード)の会議への途中参加、途中退席もサポートしている。

アプリケーション管理は、共有アプリケーションの相手ノードでの起動や、共有開始、共有終了などを行う。

チャンネルは、アプリケーションがデータを送受信するための論理的な通信路である。チャンネル管理は、チャンネルの作成、削除及びアプリケーション・データの送受信を行う。

トークンは、アプリケーションが排他制御のために用いる。本システムでは共有黒板が、同時に入力できるのは一人だけであることを保証するために使用している。トークン管理は、トークンの作成、消去、獲得要求、解放を行うモジュールである。

プロセス間通信は同じノードにある異なる二つのアプリケーション間での通信を容易にするための機能である。

参考文献

- [1] E. Linzer and E. Feig, “New Scaled DCT Algorithms for Fused Multiply/Add Architectures,” IEEE Proceedings of ICASSP-91, vol.3, pp.2201-2204, May 1991
- [2] 前田: 「雑音に強い動画像符号化のためのフレーム間類似度判定法」, 情報処理学会第51回全国大会予稿集 vol.1 pp.229-230, 1995
- [3] IBM Lakes - An Architecture for Collaborative Networking, R Morgan Publishing (1994)
- [4] ITU-T Draft Recommendation T.120: Transmission Protocols for Multimedia Data (1995)