社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE

携帯端末向け動画・3D グラフィックス統合システム

井上 大亮† 甲藤 二郎† 金子 格‡ 堤 純也†† マーク キャロウ††

†早稲田大学大学院理工学研究科 〒169-8555 新宿区大久保 3-4-1
‡早稲田大学理工学総合研究センター
††株式会社エイチアイ

E-mail: †{inoue,katto}@katto.comm.waseda.ac.jp

あらまし 近年，多くの携帯電話端末に３D グラフィックス表示機能や Java 実行環境が備わり，さらには MPEG４ビデオの符号化・再生機能も備えられている．本稿では，これらの機能を統合することで，より魅力的なコンテンツを作成・提供することを目的とする，携帯端末上の動画・グラフィックス統合システムに関する報告を行う．本検討の特徴として，端末に大幅な機能拡張を求めることなく，既存の機能を活用するシステム構成を提案するとともに，PC 上で行ったシミュレーション結果を報告する．

キーワード 3D グラフィックス，ストリーミングビデオ，携帯端末，メディア統合

An Integration System of Video and 3D Graphics for Mobile Terminals

Daisuke Inoue† Jiro Katto† Itaru Kaneko‡ Junya Tsutsumi†† and Mark Callow††

† Graduate School of Science and Engineering, Waseda University

3-4-1 Okubo Shinjuku-ku Tokyo, 169-8555 Japan
‡ Advanced Research Institute for Science and Engineering, Waseda University

†† HI Corporation

E-mail: †{inoue,katto}@katto.comm.waseda.ac.jp

Abstract Recently, many cellular-phone terminals are equipped with 3D graphics rendering engine, Java execution environment and, furthermore, MPEG4 compression/decompression capability. In this paper, we report media integration system of video and graphics on the cellular-phone terminal aiming at creating and supplying more attractive contents. We propose a system architecture without requiring drastic change on current configuration, and report simulation results performed on PC.

Keyword 3D Graphics, Streaming Video, Cellular-Phone Terminals, Media Integration

1. はじめに

近年，携帯電話端末の発展は著しく，端末は高性能化，高機能化している[1]．それに伴い，携帯電話で実現可能なコンテンツも数え切れないほど増加し，その幅も広がっている．そのなかで CG アニメーションも可能な高速な 3D グラフィックス機能を備えた携帯電話が急速に広まっている．

3D グラフィックスは人気アニメのキャラクターを 3D アニメーション表示するサービス等に広く使われている．そればかりでなく，今後は携帯機器のグラフィカル・ユーザー・インターフェース(GUI)において，小さな画面でも見やすくするために 3D アニメーションを利用することも可能だろう．将来 3D グラフィックスは狭い画面にてより直感的な見やすい GUI を構築するための重要な機能となると考えられる．

一方，携帯電話では写真やオーディオ・ビデオの再生機能も標準的に備るようになってきている．しかし，現状では 3D グラフィックス機能と写真やオーディオ・ビデオ機能を組み合わせて利用することができない．これらを同時に組み合わせての表示を可能とすることで，これらの機能の応用分野がいっそう広がると期待される．例えば，企業等のホームページに掲載する道案内を，ビデオと 3D グラフィックスを使ってよりわかりやすいものにすることが可能である．また，3D グラフィックスの一部にビデオをテクスチャとして埋め込むことで，より目をひく映像を作ることが可能である．

本稿では，携帯電話端末に現存する機能を活用し，3D グラフィックス(アニメーション)と動画を統合するシステム構成と，携帯通信環境を想定したシミュレーション結果の報告を行う．

2. 提案システム

2.1 システム構成

まず，図 1 に，本システムの受信端末の構成を示す．受信端末は，(1) 現在多くの携帯電話端末に搭載されている 3D 描画エンジンである Mascot Capsule Engine，(2)ビデオデコーダ，(3) 3D 描画エンジンとビデオデコーダを制御する Java アプリケーション，から構成される[2].

(1) Mascot Capsule Engine

Mascot Capsule Engine は株式会社エイチアイの開発した，専用ハードウェアを必要とせずに 3D アニメーションを可能にするソフトウェアレンダリングエンジンである[3]．このエンジンには幾つかバージョンがあるが，今回は主として携帯電話等の小型デバイスで 3D グラフィックス表示を行うための Micro3D を用いた．市販 3D ツールで作成したデータをモデルデータ，アニメーションデータ，テクスチャデータの 3 つの専用フォーマットへコンバートし，それらのデータを統合してアニメーション表示を行う．

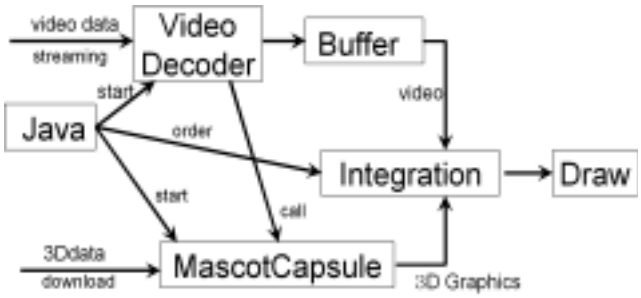

図 1　ビデオ・グラフィックス統合受信端末の構成

Fig.1　Structure of a receiving terminal enabling integration of video and graphics

(2) ビデオデコーダ

ビデオデコーダとしては様々なものが利用可能である．例えば，H.263，MPEG-4，H.264 等が挙げられる．H.263 は，ITU-T において，アナログ電話網のビデオ圧縮方式として勧告化され，その後インターネット用途の H.263+ へと拡張された．MPEG-4 は，ISO/IEC において，主にモバイル用途，インターネット用途のビデオ圧縮方式として標準化された．最後に H.264 は，MPEG-2 以来の ITU-T と ISO/IEC のジョイント勧告として，主にインターネット用途のビデオ圧縮方式として勧告化された．

(3) Java アプリケーション

Java アプリケーションは，グラフィックスエンジンとビデオデコーダの制御を担当し，セッション開始時の初期化，起動，処理の監視，グラフィックスとビデオデータの合成・表示，を担当する．

関連する技術として，MPEG-4 で標準化されている MPEG-J (MPEG-4 端末を制御する一連の Java API 群)が挙げられる．筆者らは，文献[1]において，本提案と MPEG-J との整合性について言及している．もちろん，MPEG-J に縛られない実装も可能である．

次に，受信端末の動作について述べる．本稿が前提とするのはワイアレス IP 網であり，受信端末が送信サーバにアクセスし，グラフィックス，ならびにビデオデータが IP 転送され，合成・表示される．

まず Java アプリケーションが，セッション開始時にビデオデコーダと Mascot Capsule Engine を起動させる．

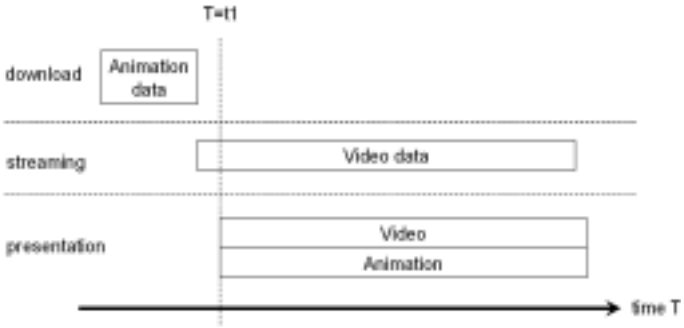

図 2　データ伝送・再生スケジュール
Fig.2　Schedule of data transmission and presentation

次に,3D オブジェクトを描画させるのに必要なテクスチャデータ，ポリゴンデータ，アニメーションデータの 3 つのファイルを，通信相手のサーバからダウンロードする．その後，エンコードされたビデオデータをサーバから受信しながら，ビデオデコーダが随時ストリーミング再生を行う．並行して，Mascot Capsule Engine はすでに受信したデータ群から 3D グラフィックスを作成する．ビデオデコーダは，ビデオの復号タイミングに合わせて Mascot Capsule Engine に 3D グラフィックスデータを送るように指示を出し,最終的に,Java アプリがビデオデータとグラフィックスデータを合成して，描画する．

2.2 データ伝送方式

図 2 には，送信サーバからのデータ転送スケジュール，および受信端末の再生スケジュールを示す．

誤りの許されないグラフィックスデータは，TCP を用いて事前にダウンロードするものとする．一方，ビデオデータは，グラフィックスデータのダウンロード終了後に転送を開始し，アニメーションと同期しながらのストリーミング再生を行う．ビデオデータの転送には，TCP，UDP のいずれかを用いる．

3 シミュレーション実験

3.1 シミュレーションモデル

図 3 のようなサーバ／クライアントモデルを想定してシミュレーション実験を行った．

サーバとクライアントの間に Cloud という TCP/IP シミュレータを配し,ワイアレス IP 環境の模擬を行った．このシミュレータを用いることで，帯域幅やパケットロス率，転送遅延等の各種パラメータを変化させることができる．そして，さまざまな条件下でストリーム受信実験を行い,表示レート,プレイアウト遅延,等の評価を行った．

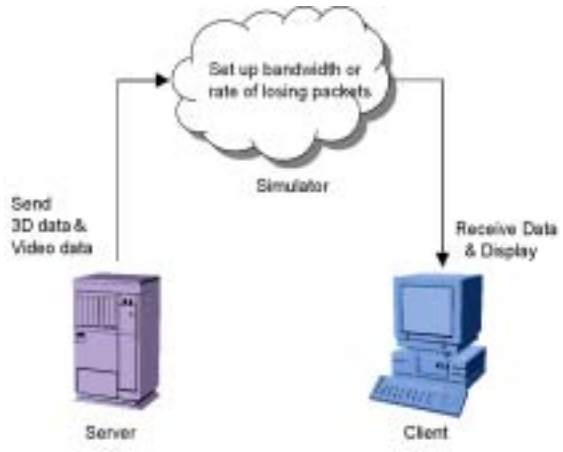

図 3　シミレーションモデル
Fig.3　Simulation model

図 4　グラフィックスとビデオの統合例
Fig.4　An example of an integration of graphics and video

まず，帯域幅を 64 kbit/s，及び 384kbit/s，パケットロス率を 0.01%，0.1%，1%，10%，パケットサイズを 512 バイト，1024 バイトと変化させた場合の実験結果を下記に示す．

なお，ビデオデコーダには H.263 を用いた．フォーマットは CIF(Common Intermediate Format，352×288)としている．また，データ転送は，3D グラフィックスデータは TCP，ビデオデータは TCP と UDP の二通りで評価を行った．

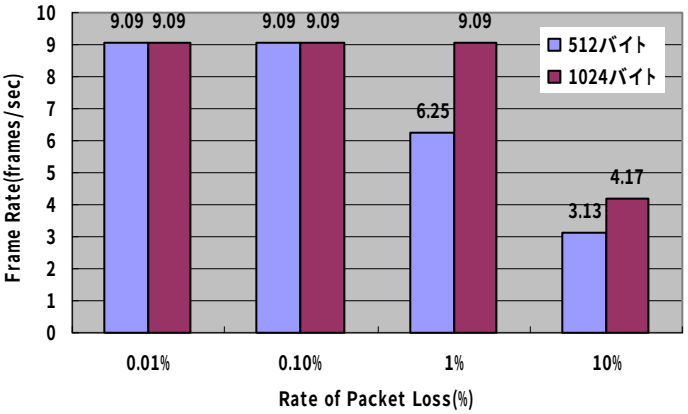

図 5　フレームレート(64kpbs，TCP)
Fig.5　Frame fate(64kbps，TCP)

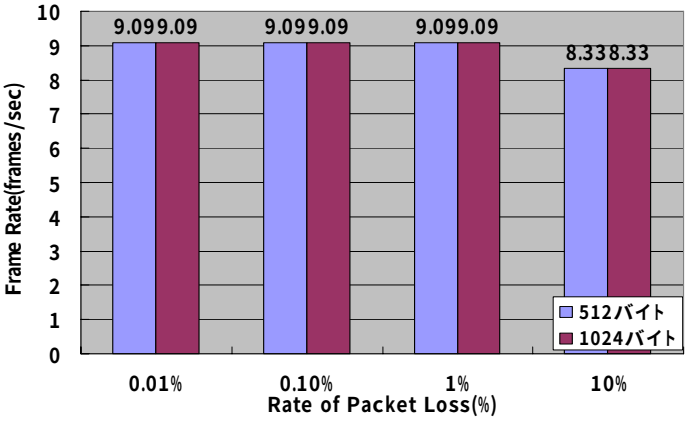

図 7　フレームレート(64kpbs，UDP)
Fig.7　Frame fate(64kbps，UDP)

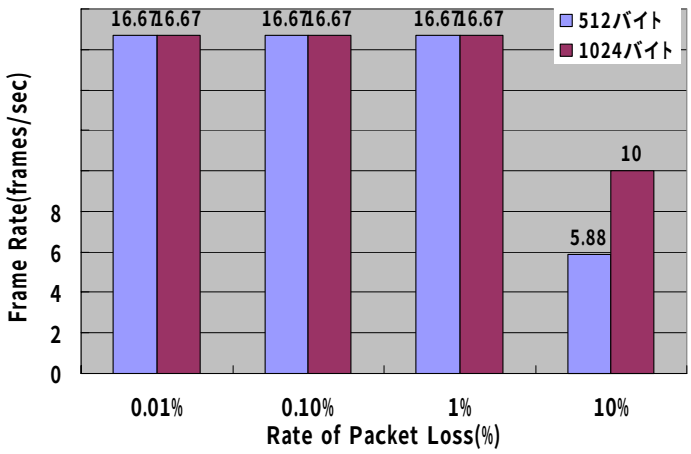

図 6　フレームレート(384kbps，TCP)
Fig.6　Frame rate(384kbps，TCP)

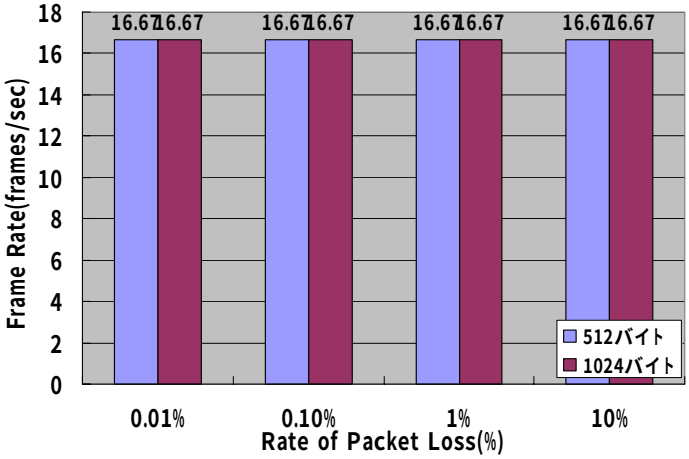

図 8　フレームレート(384kbps，UDP)
Fig.8　Frame rate(384kbps，UDP)

3.2 結果・考察

3.2.1 グラフィックス・ビデオ統合

図 4 にシミュレーション実行時の合成表示画面を示す．本稿では，さらに 3D グラフィックスとしての特長を生かし，ユーザインタラクション機能を追加している．今回はキー操作により，キー押下時のオブジェクトの拡大／縮小，回転機能等を追加し，滑らかなインタラクションを実現した．3D グラフィックスとビデオ再生の統合については，後述する極端に大きなパケット廃棄が発生しない限り，ほぼ期待通りの動作を確認している．

3.2.2 TCP 配信実験

図 5,図 6 に 3D グラフィックスデータ，ビデオデータ共に，転送プロトコルとして TCP を用いた時のビデオ再生に対するパケットロスの影響に関するシミュレーション結果を示す．

前述のように，グラフィックスとの同期再生において，グラフィックスデータやアニメーションプログラムのパケットロスは致命的となる．そこで，グラフィックスデータの転送には TCP を使用する．また，本節では，ビデオデータの転送も TCP で行った．これによって，データ転送の信頼性は保証されるが，見返りに遅延の増大が予想されるため，前述の TCP/IP シミュレータを用いて，ネットワーク上のパケットロスが与える影響を調べた．

なお，本実験ではあらかじめ 64kbps, 10FPS (frames/sec) でエンコードした圧縮ビデオデータを用意し，フレーム単位で配信を行いながら受信端末で再生可能なフレームレートの評価を行った．すなわち，図 5，図 6 のフレームレートはネットワークを介して配信しうるデータ量を表している (10FPS が 64kbps のストリーミング転送に相当する)．

本結果の第一の考察として，図 5，図 6 共に，パケットロス率が 1%以下であれば，安定した再生が可能

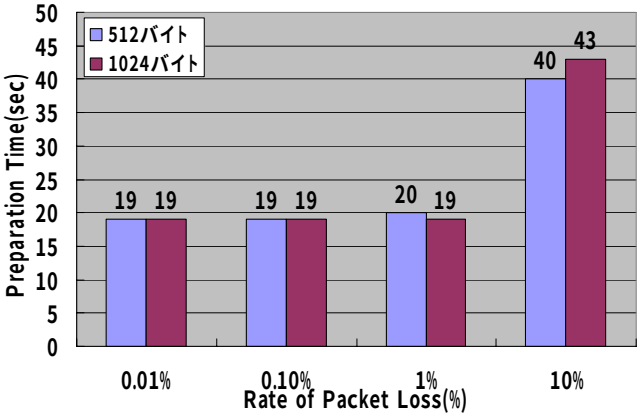

図 9　プレイアウト遅延(64kbps)
Fig.9　Preparation Time(64kbps)

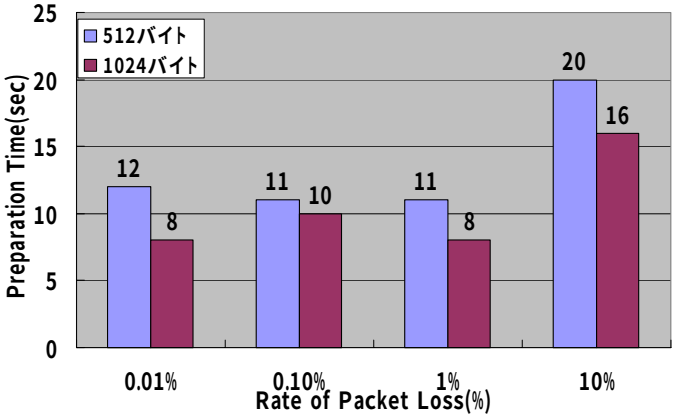

図 10　プレイアウト遅延(384kbps)
Fig.10　Preparation Time(384kbps)

であることがわかった．現状では，帯域が 384kbps の時に約 100kbps の転送量と十分なスループットを得られていないが，これについては TCP の送受信配信スケジューリング，受信端末の CPU パワー，表示能力の問題が考えられるため，現在，原因を調査中である．その一方で，15FPS 以上の再生を実現できれば主観品質も向上するとの報告もあり[4]，64kbps よりも 384kbps のほうがよりなめらかな再生品質を提供できている．

第二に，パケットサイズに関する考察として，廃棄率が小さい場合はほとんど差がなかったのに対し，廃棄率が上がると長いパケットを用いたほうが，特性が改善している．これは，送信パケット数が削減することで，廃棄率の影響が相対的に削減されたためと考えられる．すなわち，TCP ではパケットロスが増えると，再送も増える．従って，遅延も増大し，結果としてフレームレートが下がる．実際に，再生中に何度かデータ転送待ち状態に陥り再生が中断した．したがってビデオを含めて全体を TCP により伝送する方法では，転送容量に余裕がない場合に正常に動作しないことがわかる．

3.2.3 UDP 配信実験

図 7，図 8 には，3D グラフィックスデータの伝送には TCP，ビデオデータの伝送には UDP を用いた場合の実験結果を示す．ビデオデータに UDP を用いた場合，パケットロス率が高い場合でも，比較的安定した再生を行うことができた．これはパケットロスに対する再送が行われないため，遅延が起きにくいからだと考えられる．しかし，ロスしたデータはそのまま画像に現れてしまうため，パケットロス率が高い場合には，何度か画像が乱れるシーンがあった．なお，ビデオデータのパケットロス対策としては，スライス構造を用いた再同期デコードのみを行っている．

3.2.4 プレイアウト遅延

次に，サーバとクライアント間の通信開始から，実際に再生が始まるまでの時間であるプレイアウト遅延を計測した．結果を図 9，図 10 に示す．再生までにかかる主な時間は 3D グラフィックスデータのダウンロードとビデオデータのバッファリングである．

まず，3D グラフィックスデータのサイズは 3 種類の合計で，約 130k バイトである．また，ビデオデータのバッファリングは，受信バッファに 10 フレーム受信した時点でデコード開始とした．よって，今回用いた圧縮データの場合，約 8k バイトとなる．このため，今回の実験では，プレイアウト遅延はグラフィックスデータの受信完了時間が支配的となる．前述したように，3D グラフィックスデータは誤りが許されないので，データ伝送には TCP を用いる．そのため，パケットロス率が高く再送遅延が増えると，その分再生までに時間がかかってしまう．図 9，図 10 の比較としても，チャネル速度の増加と共にプレイアウト遅延が改善されていることがわかる．

4 まとめ

本稿では，携帯端末向けのグラフィックス・ビデオ統合システムの提案とシミュレーション実験を行い，評価を行った．

3D グラフィックスとビデオ再生の統合に関してはビデオ再生と平行してスムーズなグラフィックス描画が可能であることが確認された．本手法によりすでに携帯電話に備わっている機能を利用して，3D グラフィックスとビデオの同期再生が可能であることが実証できた．ビデオストリーミングに関しては，TCP 方式と UDP 方式の比較を行い，それぞれの方式の問題点を明らかにした．

今後は，　3D グラフィックスデータやビデオデータ

の配信スケジューリングの改善に関して検討を加える．特に，ユーザインタラクションをトリガとする，ビデオ，CG アニメーションのスムーズなシーンチェンジを実現することを目的とする．また，CG アニメーションの UDP ストリーミング [5] についても検討を加える予定である．

文　献

[1] 山尾泰，梅田成視，大津徹，中嶋信生，“第 4 世代移動通信の展望-無線システムを中心とした課題について-，”信学論(B)，vol.J83-B，no.10，pp.1364-1373，Oct.2000.

[2] Itaru Kaneko, Junya Tsutsumi, Mark Callow, Jiro Katto and Daisuke Inoue: “Lightweight Graphics Adaptation, integration of portable 3D graphics in MPEG-4., “ISO/IEC JTC1/SC29/WG11 MPEG2003/9827 Trondheim, Norway, July 2003.

[3] “Mascot Capsule”, http://www.mascotcapsule.com/.

[4] 山崎達也，佐藤範之，福永茂，“MPEG-4 動画像を用いた主観品質評価実験とその検討，”信学技法，CQ2000-83，MVE2000-99(2001-02)，pp.23-29.

[5] J. Katto and M. Ohta: "System Architecture for Synthetic/Natural Hybrid Coding and Some Experiments," IEEE Transactions on Circuit and Systems for Video Technology, Vol.9, No.2, pp.325-335, 1999.