

DIPS2 における新たなプログラミングの可能性

酒井 由
国立音楽大学
sayu@kcm-sd.ac.jp

本稿では、Max インターフェースを採用したリアルタイム映像処理・生成環境である「DIPS」について、主にバージョン 2 において追加・強化された機能を用いたプログラミングの実例を紹介し、さらなる応用の可能性を述べる。

A new possibility in the DIPS2 programming

Yu Sakai
Kunitachi College of Music
sayu@kcm-sd.ac.jp

The DIPS (Digital Image Processing with Sound) is a set of Max objects that handle the realtime visual image processing. In this paper, I would like to introduce several possibilities employing new functions of DIPS version 2.

1. 導入

1.1 DIPS から DIPS2 へ

DIPS (Digital Image Processing with Sound) は、Max の視覚的なインターフェースを用いたリアルタイム映像処理・生成ソフトウェアであり、IRCAM で開発されている GUI プログラミング環境 jMax のエクスターナルオブジェクトの形で実装されている。このソフトウェアは現在 Mac OS X と Linux プラットフォームで稼働しており、3次元グラフィックライブラリ「OpenGL」を用いて様々な映像の描画を行う他、ピクセル演算による2次元エフェクト、ビデオカメラからの映像の取り込みや解析、OBJ形式のファイルから読み込んだモデルの変形なども可能である。それらの機能を組み合わせることにより、容易に音と映像のインタラクションを生み出すことができる。DIPS オブジェクトは機能によって7つのオブジェクト群に分けることができる(表1)。DIPSは1997年に松田周が開発を始め、2000年にICMC(国際コンピュータ音楽会議)において発表が行われ、さらに2002年にはバージョン2となる「DIPS2」がリリースされた。DIPS2では、

jMaxの最新バージョンであるjMax4に対応するとともに、プラットフォーム間のソースコードの一本化、映像処理の効率化、さらにいくつかの機能追加がなされた。本稿では、自作品でのDIPSの使用法と、今後のプログラミングの可能性について考察を述べる。

オブジェクト群	機能
DIPS	ウィンドウの生成やビデオカメラからの映像の取り込みなどを行う
DGL	OpenGLの関数をそれぞれ実装
DGLU	OpenGL Utility Libraryを実装
DGLUT	OpenGL Utility Toolkitを実装
DPX	ピクセル演算による2次元エフェクト処理を行う
DPXA	画像の解析を行う
D3D	3次元モデルの読み込みやその変形、描画等を行う

表1 DIPS オブジェクト群

1.2. DIPS におけるプログラミング

前述の通り、DIPS は jMax のエクスターナルオブジェクトの集合であるので、そのプログラミングは個々のオブジェクトをパッチコードでつないでいく、という手法で行われる (図 1)。DGL、DGLU、DGLUT に分類されるオブジェクトにはその名前に対応した OpenGL 関数がほぼそのままの形で実装されているため、OpenGL の知識があれば非常に容易にプログラムを組むことができる。

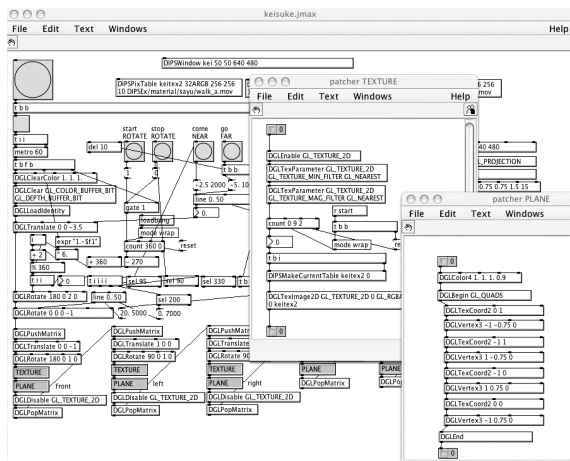


図 1 DIPS のパッチ例

1.3 DIPS2 における新機能

次に、DIPS2 へのバージョンアップに伴って追加された機能に関して述べる。

まず DIPS2 ではウィンドウに関するいくつかの機能が強化され、これまで不可能であった open、close、move、resize などの動作や透明度の設定ができるようになった。また、ウィンドウ上でのマウスイベントの取得も可能となり、up、down などの動作の他にその xy 座標も数値としてパッチ内で用いることができる。さらに DGL パッケージに glReadPixels という OpenGL 関数が実装されたことにより、DIPSWindow に描画されている画像を再び DIPSPixTable に取り込むことができるようになった。これにより 2 次元映像リソースの再利用性を高めることができ、さらに複雑な映像表現の可能性が得られる。また、前述のように新たに DPXA オブジェクト群に分類される DPXATrim、DPXABrightness、DPXADifference という 3 つのオブジェクトが開発され、これらはビデオカメラが捉えた画像の解析を行い、あらかじめ指定された 2 次元画像あるいは直前のフレームとの差分を計算するこ

とにより物体の動きを検出する。これらのオブジェクトにより、音声から映像だけでなく映像から音声へのインタラクションも可能となった。

2. 自作品での DIPS の使用

筆者は 2003 年 9 月にピアノと Max/MSP、DIPS のためのライブ作品「Glow」を制作した。使用したのは OpenGL の基本的な機能が主であり、具体的には線や面などのプリミティブの描画 (図 2) や DGLUT オブジェクトを用いて描画された球体の表面へのテクスチャマッピングを行った。さらに、Maya により作成された 3 次元モデルを D3D オブジェクトにより DIPS 上に取り込み、そこに動画を同じくテクスチャとしてマッピングするなどの手法も用いた。

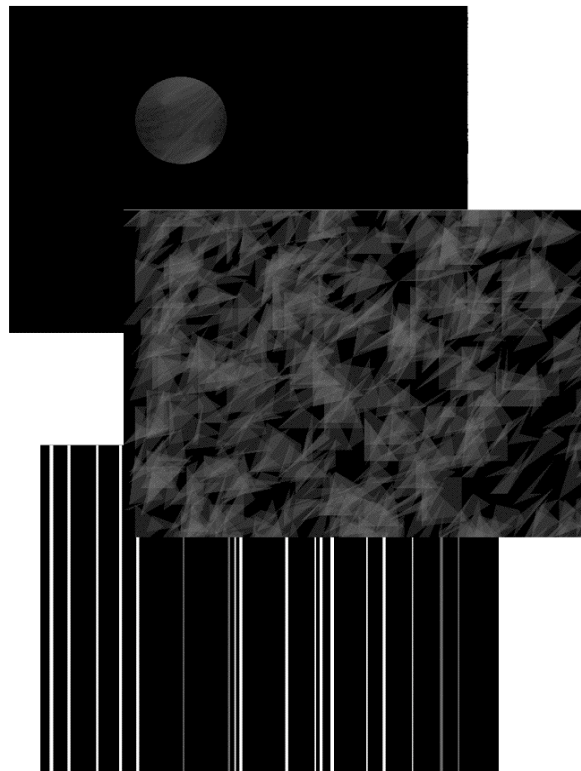


図 2 「Glow」での映像

Max/MSP、DIPS それぞれを稼働させるコンピュータは Ethernet で接続され、CNMAT の開発による OpenSoundControl プロトコルを用いて任意のデータを Max/MSP 側から DIPS へ送ることができるようになっている。この作品では、演奏されたピアノの音量のデータを送り、その数値を描画される線の色や数、照光処理な

どのパラメータとして使用することにより、映像を変化させ、音と映像とのインタラクションを実現した。

続いて2004年2月にフルートとMax/MSP、DIPSのためのライブ作品「Monochrome」を制作した。前作での反省点として、音との関連性が分かりにくいことと映像が平面的であることが挙げられると考え、本作品では第一にその改善を念頭に置いた。まず、前者を解決するため映像の速度感に着目し、音声の変化に合わせて描画される物体の移動速度を極端に変化させることにより、両者の関係性を明確なものとした。また、後者に関しては、前作では行わなかった視体積の定義の使い分けと視点の移動によって立体的で大きな動きのある映像を作ろうと考えた。加えて、演奏表現の豊かさを増し、演奏行為との一体感を得るため、システムに新たに演奏者を映すビデオカメラを加え、取り込まれた演奏者の姿をリアルタイムに処理し、映像の一部として使用するという手法を取った(図3)。

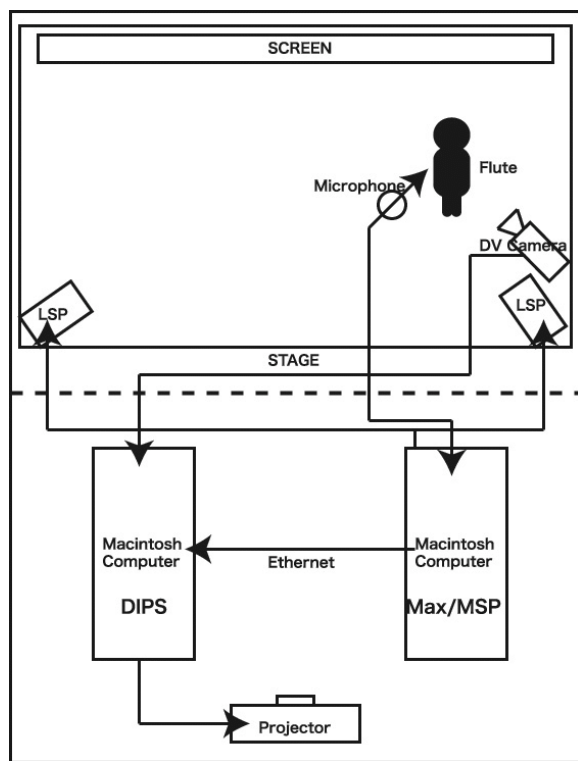


図3 システムダイアグラム

さらに、前作ではMax/MSPとDIPSのcue情報は独立していたため、演奏中には2台のコンピュータを操作しなければならなかったが、本作品ではこれらを連動さ

せ、音量のデータに加えてDIPSのcue情報もMax/MSP側からEthernetを通じて送られるようにプログラミングを施した。これにより演奏中のDIPSの操作は不要となった。

この作品のDIPS映像は8つのシーンで構成され、そのうち4つのシーンではビデオカメラからの映像を用い、それ以外のシーンでは音声とのインタラクションを重視したため、より明白に音との対応関係を表現できるDGLUTオブジェクトにより描画されるプリミティブや、前作と同様Mayaにより制作された3次元モデルを多く用いている。同時にDGLUPerspectiveオブジェクトを用いて視界を錐台形にすることにより映像に奥行きを与え、また視点を移動・回転させるためにDGLULookAtオブジェクトを使用し、その速度は演奏の音量に比例するようにプログラミングされている(図4)。

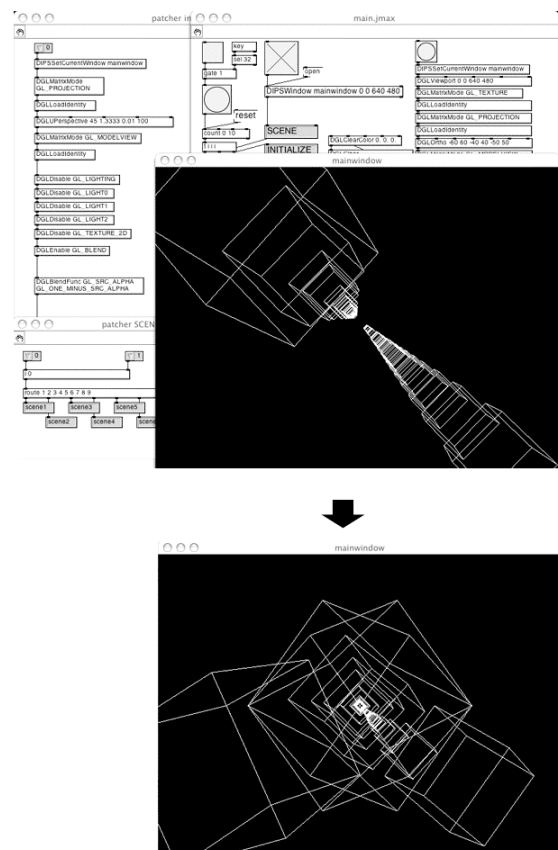


図4 「Monochrome」の1シーンでの映像の変化

ビデオカメラからの映像を用いたシーンでは、射影変換の設定に遠近感を伴わないDGLOrthoオブジェクトを

用いており、視点の移動も全く行われない。取り込まれた演奏者の姿はテクスチャとして矩形に適応されるようプログラミングされており映像の背景として扱われるが、この矩形の変形は行われない。これらのシーンは、フルートの音程やリズムの変化が乏しい楽曲中の静かなセクションとの一体感を目的としているため、映像の変化は背景の色の変化とその前面に描画される線や面などの2次元図形の僅かな動きに留められる。

前節で述べたように一度 DIPSWindow に描画された映像を DIPSPixTable に読み込むことが可能になったことから、本作品のパッチ上にはメイン、サブの2つのウィンドウを作成し、サブウィンドウに描画された映像を2次元画像として DIPSPixtable に保存した後、プロジェクトに映し出されるメインウィンドウで適宜テクスチャとしてその映像を利用した。また、サブウィンドウには演奏の音量に応じて D3DOBJHandler オブジェクトによって変形する OBJ 形式の3次元モデルを描画するようにプログラミングを施した。

3. まとめと今後の可能性

前節では筆者が DIPS を用いて制作した2つの作品について、主に用いたプログラミングの手法を述べたが、DIPS プログラミングにはそれら以外にも数多くの可能性がある。さらには1.3項で触れたような OpenGL 関数にとどまらない様々な機能も追加されている。ここでは今後の作品制作における DIPS の使用方法に関して考察をしていきたい。

まず、新たに追加された機能のうち、画像の解析機能によって映像から音声へのインタラクションを可能とする DPXA オブジェクト群を用いて、例えばライブ作品中で演奏者の何らかのジェスチャーを解析し、その結果を電子音響のパラメータに適応するというような方法も可能である。これにより演奏者の即興性を楽器演奏のみならず電子音響にも活かすことができると考えられる。

また、この DPXA オブジェクト群や、DIPS2 で強化された DIPSWindow によるマウスイベント・キーイベントの取得機能を利用した参加型インストールの制作も可能である。鑑賞者の動作と音声とのインタラクション、映像とのインタラクションの他、センサの使用など、インストール作品においても、様々な DIPS の活用法が考えられる。

今後はより完成度の高い作品制作を目指すとともに、論じてきたような新たな手法・機能も実際に作品の中で用い、またそれによって問題点や必要とされる機能等を浮き彫りにし、それらの解決やソフトウェアのさらなる充実を計っていきたい。

参考文献

- [1]OpenGL Architecture Review Board, Maison Woo, Jackie Neider, Tom Davis:The Official Guide to learning OpenGL
- [2]Dechelle, F., De Cecco, M., Maggi, E.,Schnell, N., "jMax: An Environment for RealTime Musical Applications", in Computer Music Journal, 1999, Beijing, China.
- [3]Apple Computer, Inc., Inside QuickTime: API Reference for QuickTime
[<http://developer.apple.com/techpubs/quicktime>]
- [4]Apple Computer, Inc., What's New in QuickTime6
- [5]Apple Computer, Inc., Preliminary Documentation; Broadcasting with QuickTime6
- [6]Matsuda, S., Rai, T., "DIPS : the real-time digital image processing objects for Max environment", in Proceedings of the International Computer Music Conference 2000.
- [7]Matsuda, S., Rai, T., Miyama, C., Ando, D., "DIPS for Linux and Mac OS X", in Proceedings of the International Computer Music Conference 2002.
- [8]松田 周:「DIPS: Max のためのリアルタイム映像処理オブジェクト群」,00-MUS-36
- [9]橋田 光代,美山 千香土,安藤 大地:「DIPS エクスターナルオブジェクト開発と作品制作への応用」,00-MUS-36
- [10]橋田 光代,美山 千香土,安藤 大地,松田 周:「DIPS プログラミングの実際」,00-MUS-38
- [11]松田 周,美山 千香土,安藤 大地:「マルチメディアプログラミング環境 DIPS : Linux と Mac OS X への移植」,02-MUS-48
- [12]松田 周,安藤 大地,美山 千香土,酒井 由:「マルチメディアビジュアルプログラミング環境 DIPS2」,03-MUS-51
- [13]美山 千香土:「マルチメディア・インタラクティブ作品の制作について」,03-MUS-53