

# 澤見研究室

非可逆音声圧縮について

I07I011 岩澤 哲央

I07I026 川田 洸輔

I07I048 中津 弘迪

Multi Media Player制作

I07I024 川木 進吾

I07I033 河口 英寛

I07I057 坂野 裕一

# 非可逆音声圧縮について

## 澤見研究室

1071011 岩澤哲央 1071026 川田洸輔 1071048 中津弘迪

### 1 はじめに

インターネットによる音楽配信サービスやポータブルオーディオなどの普及により、デジタル音声データを扱うことが多くなっている。

現在ではソフトウェアによる圧縮処理をし、汎用メディアへの記録という方法が一般的で、よく利用されている方法として、非可逆圧縮と呼ばれる音声データの圧縮方法が多くある。

本研究では、数多くの音声圧縮形式の中から、非可逆圧縮形式のものを選び、その中からいくつかを取り上げ、比較・評価を行う。

### 2 データの圧縮形式

非圧縮形式には WAVE や AIFF があげられ、CD と同等の音質とされている。可逆圧縮形式は元のデータに戻すことのできる圧縮形式で FLAC や Monkey'sAudio などがある。非可逆圧縮は圧縮したデータを元に戻すことのできない形式であり、MP3 や AAC などがある。

### 3 非可逆音声ファイルフォーマット

本研究で比較・評価に用いる非可逆音声ファイルフォーマット MP3, OggVorbis, AAC, WMA について簡単に紹介する。

#### 3.1 MP3

MP3 は MPEG Audio layer3 の略で、非可逆音声圧縮ファイルフォーマットの中でも普及率は高く有名である。特許を持つ企業が存在しており、MP3 を使用したソフトを作る場合、メーカーにライセンス料が請求されるため、ゲーム音楽などに使用されることは少ない。拡張子は「.mp3」である。

#### 3.2 OggVorbis

OggVorbis は特許により自由に使用することのできない MP3 にかわって開発された。よってライセンスフリーなフォーマットで誰でも自由に利用することができる。拡張子は「.ogg」である。

#### 3.3 AAC

AAC は AdvancedAudioCoding の略であり、MPEG2 または MPEG4 で使用される音声圧縮方式であり、iTunesStore や着うたで採用されている。拡張子には「.m4a」「.3gp」「.mp4」などがある。

#### 3.4 WMA

WMA は WindowsMediaAudio の略であり、マイクロソフトが開発した形式である。マイクロソフト独自の形式のため、MP3 や OggVorbis などの形式と比べると汎用性に劣る。拡張子は「.wma」である。

### 4 非可逆音声圧縮の仕組み

非可逆音声圧縮では知覚符号化による処理を行っており、人間には聞き取れない小さな音や、判別できない高い周波数の音を取り除いてデータ圧縮をしている。

#### 4.1 最小可聴限界

最小可聴限界は、人間が聞き取ることのできる周波数ごとの音の大きさである。一般に人間が聞き取ることのできる周波数は 20Hz から

20kHz とされている。周波数により聞き取れる音の大きさも異なる。

## 4.2 マスキング

マスキングとは、大きな音が出ている時に、その音の周波数に近くて小さい音がマスクされ人の耳では聞き取れないことを意味している。時間的に近い場合には小さい音も人の耳では聞き取れなくなる。この人の耳では聞き取れない音の部分をマスキングと言います。

この聞き取れない部分を省きデータの容量を少なくしデータ圧縮している。

## 5 ビットレート

ビットレートとは、1秒あたりどれだけのデータを送受信できるかというデータの単位時間当たりの通信速度のことで、動画や音声の圧縮時に、1秒分の動画や音声をどれだけのデータ量で表現するかという意味でも使われる。単位は「bps」である。

## 6 周波数成分

WAVE の音声データと圧縮した音声データをスペクトラムに表し比較を行う。スペクトラムで表す事によりどの音域の成分が加工されているかを見る (WaveSpectra V1.40 を利用)。

### 6.1 ビットレートによる比較

44.1kHz, 16bit, 2チャンネルのWAVE形式音声データを64kbps, 96kbps, 128kbps, 160kbps, 192kbps, 320kbpsに圧縮し、それぞれの周波数成分のピーク値を出してデータの音質のバランスをAAC形式に関し比較・評価する。青色のスペクトラムは元のWAVEデータで、赤色は圧縮後のスペクトラムである。

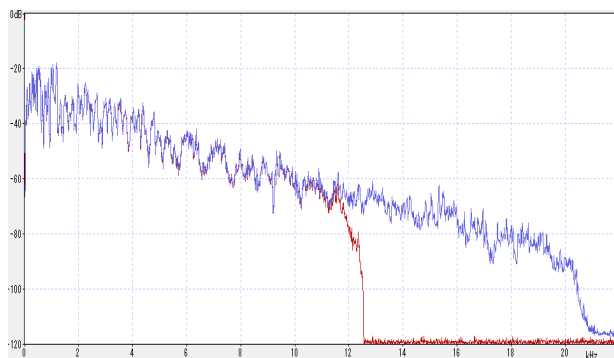


図 1 :64kbps のスペクトラム

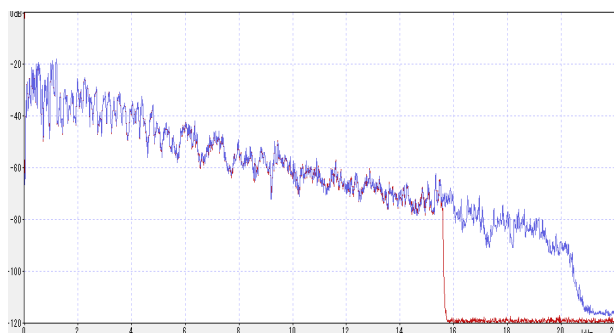


図 2 :96kbps のスペクトラム

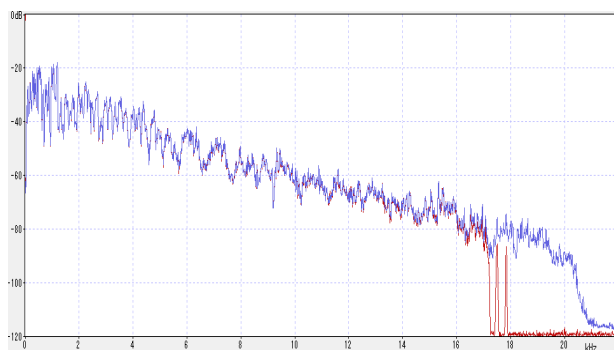


図 3 :128kbps のスペクトラム

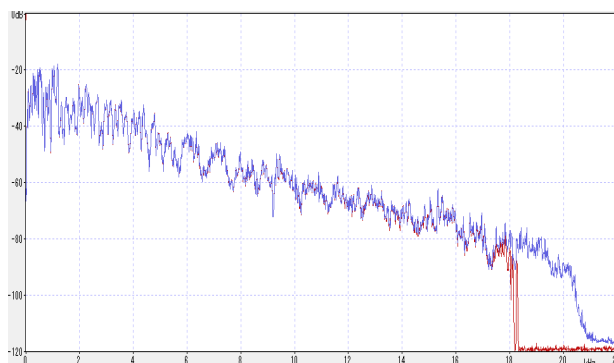


図 4 :160kbps のスペクトラム

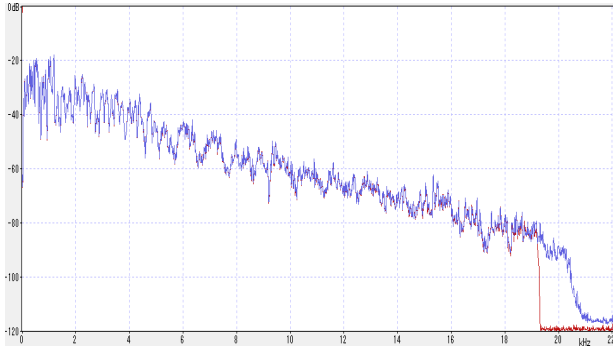


図 5:192kbps のスペクトラム

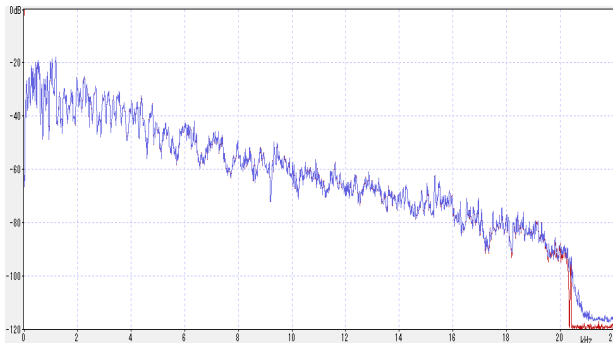


図 6:320kbps のスペクトラム

周波数成分が現れているが、64kbps では13kHz程しかでておらず、96kbps では一気に16kHz付近まで上がっている。さらに128kbps では18kHzまで一気に上がっており、320kbps では聞き取れる限界とされている20kHz以上の周波数の音が出ている。また、実際に耳で聞いたところ、64kbps と96kbps は音がこもっているように聞こえたが、それ以上のビットレートでは音質の差は感じられなかった。

## 6.2 フォーマットによる比較

MP3, OggVorbis, AAC, WMA に関しそれぞれ96kbps, 128kbps, 160kbps, 192kbps で圧縮し、それぞれのフォーマットでどれが1番高周波数まで音がでてくるかを各ビットレートで比較した。

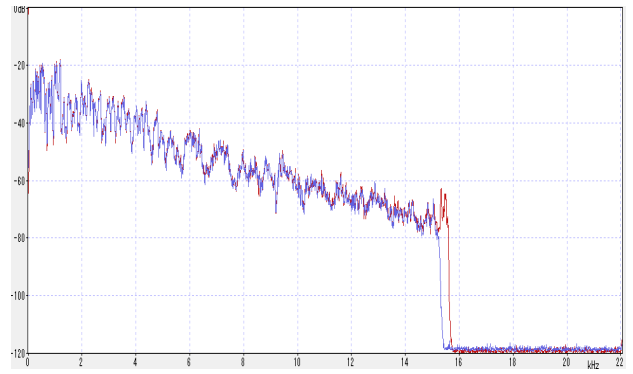


図 7:96kbps の AAC (赤) と MP3 (青)

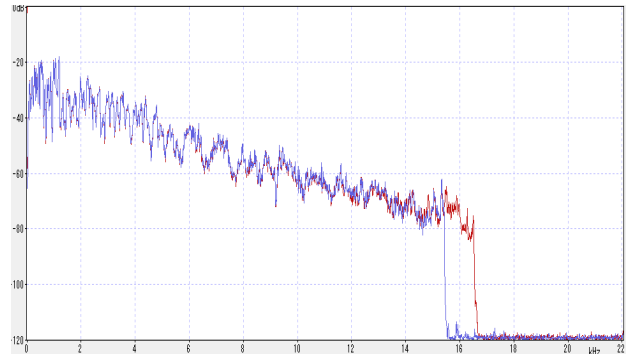


図 8:96kbps の OggVorbis (赤) と WMA (青)

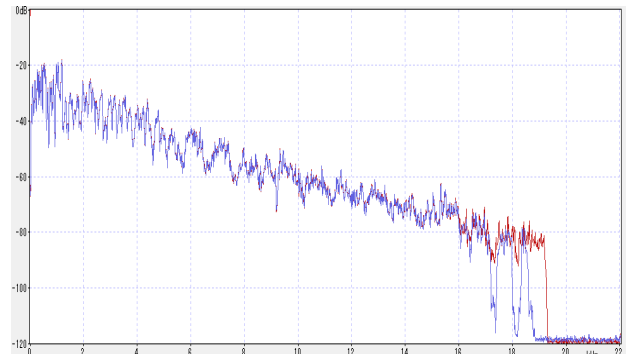


図 9:192kbps の AAC (赤) と MP3 (青)

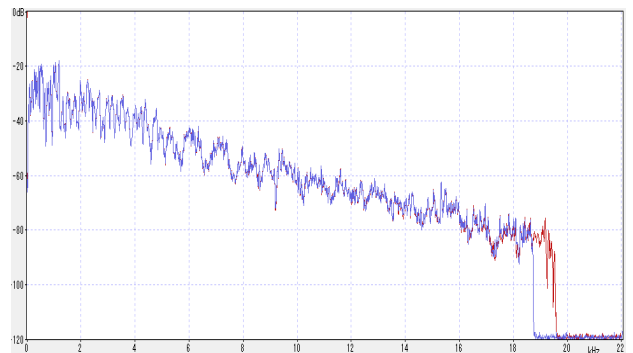


図 10:192kbps の OggVorbis (赤) と WMA (青)

1つの図に最大2本の成分しか表示していない

ため、この図からは読み取りにくいですが、96kbpsではAACとMP3、WMAでは1番高い周波数が15kHzなのに対し、OggVorbisは17kHzの周波数まで出ていた。192kbpsではWMA、MP3は18kHz後半までだったのに対し、AAC、OggVorbisは19kHz以上の音が出ていた。どのビットレートでも一番高い周波数の音が出ていたのはOggVorbisとなり、MP3とWMAはいずれも近い周波数となったが、OggVorbis、AACと比べると低い周波数となった。

## 7 1/f ゆらぎ

1/f ゆらぎとは、信号エネルギー（スペクトル密度）が周波数  $f$  に反比例することであり自然現象でも見るができる。例として小川のせせらぐ音、電車の揺、1部の人間の声にも現れるとされている。ヒーリングミュージックの効能の説明にも使われる言葉であり、1/f ゆらぎは快適感やヒーリング効果があるとされているが、科学的には証明されていないことが多く、疑似科学の一つとみなされている。各ビットレート、各フォーマットで圧縮した音声ファイルを用意し、ビットレート、フォーマットによってゆらぎに違いがあるのかを調べた（ゆらぎアナライザー V1.14を使用）。周波数の分布からゆらぎの程度を表す直線の傾きを求めその傾きを数値化する。ここでは数値が1に近づくほど1/f ゆらぎをもっているということになる。

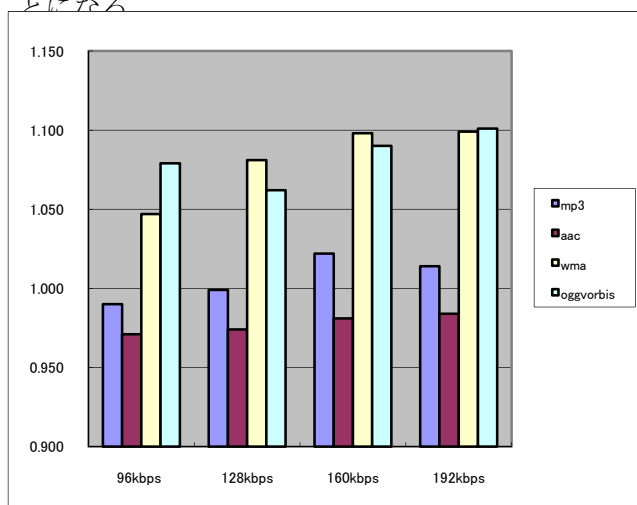


図 11:ゆらぎ解析結果

フォーマット別に見て、1/f ゆらぎを最も示しているのはMP3となり、つづいてAAC、WMAとOggVorbisは1/f ゆらぎが少ないという結果となった。

ビットレート別では、ビットレートがあがるにつれてゆらぎを示す数値が上昇する傾向にあり、MP3とOggVorbisは128kbps、AACでは192kbps、WMAは96kbpsがそれぞれ1/f ゆらぎに近いという結果になった。

## 8 まとめ

周波数成分では、低ビットレートほどビットレートによる成分の差が大きく、高ビットレートになればなるほど差が小さくなった。フォーマット別では、いずれのビットレートでもOggVorbisが1番高い周波数まで出ていた。ゆらぎについては、全体的にMP3が1番強い値を示すことがわかった。また、元のwaveファイルの数値が1.107なので、ビットレートは高いほうが1/f ゆらぎに近いということではなかった。

## 参考文献

- 1 よくわかるデジタルポータブルオーディオ  
加賀章喜 霧島煌一 電波新聞社 (2005)
- 2 サウンド圧縮テクニカルガイド  
藤本健 株式会社BNN新社 (2004)
- 3 ゆらぎアナライザー  
<http://mahoroba.logical-arts.jp/>
- 4 WaveSpectra  
<http://www.ne.jp/asahi/fa/efu/>
- 5 CDex  
<http://www.surf.to/cdex/>
- 6 xrecode II  
<http://xrecode.com>

# Multi Media Player 制作

## 澤見研究室

1071024 川木進吾 1071033 河口英寛 1071057 坂野裕一

### 1 はじめに

本研究では、マルチメディアプレイヤーの仕組みについて詳しく調べ、それらを参考にして仕様を決め、これに基づいてマルチメディアプレイヤーを作成し、簡単な比較評価を行った。

### 2 Multi Media Player とは

パソコンで音楽を聴いたり、動画などを再生したりするために使用するアプリケーションをここではマルチメディアプレイヤーと呼ぶ。

マルチメディアプレイヤーを使うことで音楽ファイルや CD, DVD ビデオ, パソコンに保存されている各種フォーマットのムービーファイルなどを再生利用することができる。

### 3 コーデック

コーデックとは、音声や映像の符号化 (encode) や復号化 (decode) のアルゴリズム、それを行うためのプログラムのことである。コーデックは、もともとデータをデジタル通信回路で送受信するための装置を意味している。

動画ファイルは、データを圧縮されていない動画ファイルをコーデックによって圧縮してからコンテナへと格納するようになっている。コンテナにどのようなものを入れることができるかは、コンテナの種類によって決まる。例えば、avi 形式であれば映像は、WMV, DV, H. 264 などがあり、音声では WMA, MP3 など様々なコーデックに対応して格納することができる。

### 4 再生方法

動画を再生するには様々な方法があるが、ここでは DirectShow を使用することにした。DirectShow とは、マルチメディア拡張 API 群である DirectX に含まれる API の一つである。

DirectShow を使うための手順は、まずフィルタグラフを作成し、次にフィルタグラフをレンダリングしてから最後にビデオサイズなどを調整し、動画像を出力するようになっている。

#### 4.1 レンダリング

レンダリングとは、コンピュータにより各種フォーマットのデータを処理し、画像、音声、映像を作り出すことである。これらを行うソフトウェアなどをレンダラと言う。

#### 4.2 フィルタグラフ

DirectShow を用いて音楽や動画像を再生するにあたって、まずフィルタグラフと呼ばれるものを作成する。

フィルタは入力されたデータを処理し、出力する。DirectShow ではこのフィルタを細かく分けている。フィルタ同士はピンと呼ばれるインタフェースでつながれ、つながり合わせてできたものをフィルタグラフと呼ぶ。

DirectShow のフィルタは、主にソースフィルタ、変換フィルタ、レンダラの三つに分類される。ソースフィルタはデータを作成し、次のフィルタに送り込む。変換フィルタは、データを受け取り転送する。レンダラは、データの受け取りをして表示を行う。

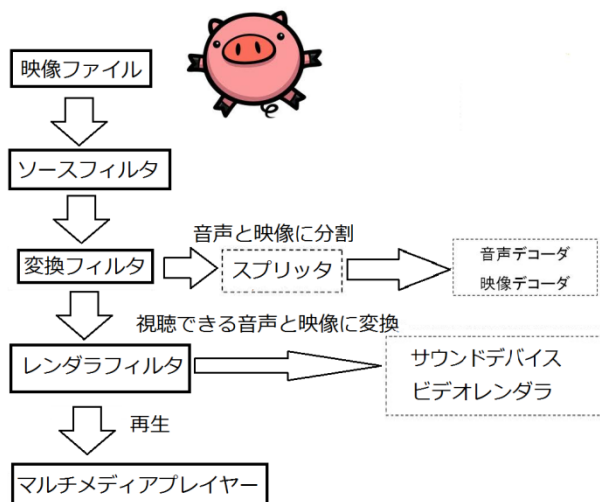


図 1 フィルタグラフ内の構造

### 4.3 フィルタグラフ内の処理

実際に動画を再生する場合、まずソースフィルタで HDD などの記憶装置からデータを読み込み、変換フィルタへ流す。ソースフィルタは、グラフの最初に位置することになる。次に変換フィルタにより、映像・音楽がそれぞれ別にデコード処理され、レンダリングファイルへと送られる。最後にレンダリングファイルによりデータを視聴可能な実際の映像、音声へと変換する。

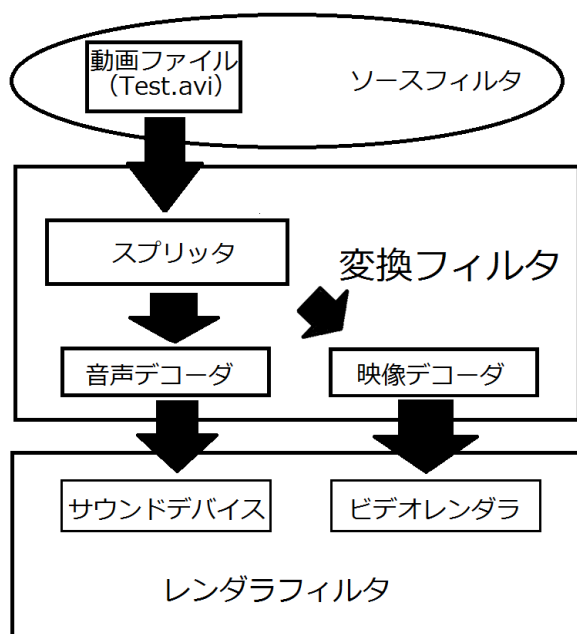


図 2 フィルタグラフ内の処理

## 5 サポートするフォーマット

ファイルフォーマット	映像コーデック	音声コーデック
avi	<ul style="list-style-type: none"> <li>・MPEG-4</li> <li>・WMV7/WMV8/WMV9</li> <li>・DV(DVSD/DVIS)</li> <li>・Motion JPEG (MJPG)</li> <li>・LossLess JPEG (LJPG)</li> <li>・H.264 (DAVC/H264)</li> <li>・H.263 (H263/S263)</li> <li>・H.261 (H261)</li> </ul>	<ul style="list-style-type: none"> <li>・LPCM</li> <li>・WMA</li> <li>・ADPCM</li> <li>・MP3</li> <li>・AC-3</li> <li>・AAC</li> <li>・HE-AAC</li> </ul>
asf, asx, .wmv .wvx, .wma, .wax	<ul style="list-style-type: none"> <li>・MS-MPEG4</li> <li>・ISO MPEG-4</li> <li>・WMV7</li> <li>・WMV8</li> <li>・WMV9</li> </ul>	<ul style="list-style-type: none"> <li>・WMA9</li> <li>・G.726</li> <li>・LPCM</li> <li>・MP3</li> </ul>
.mp4	<ul style="list-style-type: none"> <li>・MPEG-4 Video</li> <li>・MPEG-4 AVC</li> </ul>	<ul style="list-style-type: none"> <li>・AAC</li> <li>・HE-AAC</li> <li>・TwinVQ</li> </ul>

図 3 代表的なフォーマットの一覧

メディアプレイヤーでサポートされているフォーマットの映像コーデック、音声コーデックの一覧を表している。

## 6 制作したプレイヤーの説明

制作には VisualC++を使用し、動作確認には Windows XP, Windows Vista, Windows 7 を使用した。

今回制作した Multi Media Player の機能を以下に列記する。

### (1) メディアファイルの操作

実行ファイル直下に再生ファイルを置くか、オープンファイルダイアログ、ドラッグ&ドロップ、プレイリストに追加した音楽・動画の再生を行う。

各操作はキーボード、マウスによるウィンドウのクリック、ソフト上に表示されているアイコンの操作によって行う。

## (2) 音量調整

設定されている上限と下限の範囲内で音量を調整することができる。

制作したソフトウェアでは再生画面右下に表示される。

## (3) ドラッグ&ドロップ

再生したいファイルを再生画面内にドラッグ&ドロップをして再生させる。

## (4) シークバー

動画の再生時間と現在の経過時間を取得し、表示する。再生しているファイル再生時間のうちどこまで再生しているかをスライダーの位置によって視覚的に把握することができる。

## (5) 動画サイズの変更

動画のサイズをファイル本来のサイズで表示、ウインドウサイズに合わせる、アスペクト比を無視してウインドウサイズに合わせるなどの変更ができる。

## (6) フルスクリーン機能

再生させるファイルをソフトのメニューバー、ソフト上に表示されているアイコンやタスクバーを隠ぺいして画面全体に拡大して表示する。

## (7) プレイリスト機能

通常の再生手順を踏むと自動的にプレイリストに登録される。また、ドラッグ&ドロップを行うこと追加することもできる。

プレイリストに追加されたファイルは、そ

のファイルの再生が終わると次に登録されているファイルの再生を行う。さらに再生中にプレイリスト内の再生したいファイルを選択することによって現在のファイルを停止し、指定ファイルを再生する。

## (8) プレイヤーの透過機能

プレイヤーを透過して下の画面を見るときなどに使うのを想定して実装した機能である。全画面で再生中に他のソフトの進捗を確認するときには有効である。以下の図が透過機能実行時のプレイヤーである。プレイヤーが透過され下のマイコンピュータを見ることができる。



図 4 透過機能

## (9) 貫通色指定機能

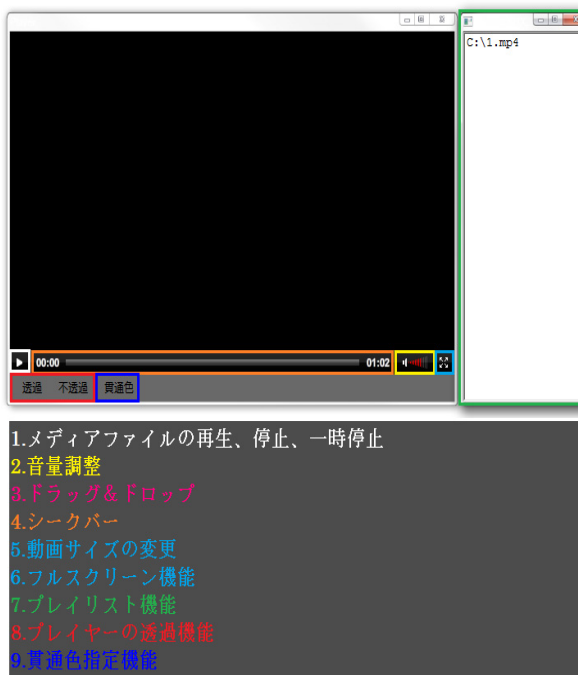
指定色を完全に消滅させて動画、画像を表示させる機能。

以下の図が貫通機能を実行中のプレイヤーである。この図では、#000000 (黒) を貫通した下のアイコンをクリックすることも可能である。





図 5 貫通機能



- 1.メディアファイルの再生、停止、一時停止
- 2.音量調整
- 3.ドラッグ&ドロップ
- 4.シークバー
- 5.動画サイズの変更
- 6.フルスクリーン機能
- 7.プレイリスト機能
- 8.プレイヤーの透過機能
- 9.貫通色指定機能

図 6 機能の説明

## 7 映像の比較

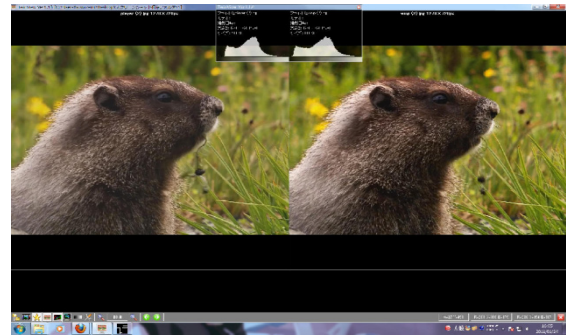


図 7 映像の比較

静止画の再生では,差はあまり感じられなかったが,動画においては,作成したプレイヤーは画像が暗い感じがあった.

## 8 まとめ

作成したプレイヤーと一般的なプレイヤーを比較した結果,再生などの機能は満足いく出来だったが,プレイリストなどのユーザーの補助的な機能が使いにくかった.今後の課題として,操作性を向上などがあげられる.

## 参考文献

- 1, MSDN ホームページ  
<http://msdn.microsoft.com/ja-jp/default.aspx>
- 2, finalstream  
<http://codec.finalstream.net/index.htm>
- 3, 2009年卒業研究論文  
 マルチメディアプレイヤーについて

**[空白]**