

研究ノート | Research Notes

音響と画像の信号処理に関するクロスオーバー学習教材の制作

Development of Software Tools for Cross-over Learning on Acoustic and  
Image Signal Processing

茂出木 敏雄  
MODEGI Toshio

尚美学園大学  
情報表現学科 講師  
Shobi University

2019 年 10 月

Oct.2019

## 音響と画像の信号処理に関するクロスオーバー学習教材の制作

### Development of Software Tools for Cross-over Learning on Acoustic and Image Signal Processing

茂出木 敏雄  
MODEGI Toshio

#### [抄 録]

これまで映画・放送事業者でないと制作・編集や配信に携わることが困難であった映像メディアに対し、誰もがスマートフォンだけで手軽に扱える時代になった。このように身近になった映像メディアを理解するため、基盤となる音響と画像の信号処理を学習することは重要である。音響は1次元の時間次元に定義され、画像は2次元の空間次元に定義された媒体で、ヒトが認識する感覚系も聴覚と視覚で異なっており、音響と画像は互いに交わる要素が無い。しかし、信号処理という観点では殆ど共通の技術を適用でき、音響または画像に固有な技術を見出す方が寧ろ難しい。即ち、音響または画像の信号処理のいずれか一方を学習すれば、他方は学習しなくても類推できてしまう。あるいは、音響または画像の信号処理のいずれかの学習で躓いた項目があれば、他方の対応する項目の学習で補習できる、いわばクロスオーバー学習を実現できる。本稿では、筆者が制作してきたWindowsパソコン版の音響と画像の信号処理アプリケーションソフトについて述べる。

#### [ Abstract ]

In these days, anybody can deal in production, edition and distribution of video media, which had been monopoly of cinema or broadcasting professionals. For understanding of video media technology being easy to use, it is important to learn both acoustic and image signal processing which are fundamentals of video media. Acoustic media are defined in a one-dimensional temporal dimension whereas image media are defined in two-dimensional spatial dimensions. In human beings, acoustic media are sensed by ears whereas image media are sensed by eyes, and there no common features between them. However, in signal processing fields, almost common technologies can be applied to these two media, and it is difficult to find unique technologies dependent on either acoustic or image media. In other words, if we learn one field on either acoustic or image media, we can also understand the other field without learning it. Moreover, if we find a difficult point to understand in one field, we can understand the corresponding point in the other field by cross-over learning. In this report, we present our developed Windows application software tools for learning of both acoustic and image processing technologies.

#### キーワード

映像メディア、音響技術、画像技術、信号処理、クロスオーバー学習、Windowsアプリケーション

## Keywords

video media, acoustic technology, image technology, signal processing, cross-over learning, Windows application

### 1. はじめに

これまで一般生活者には身近な媒体でありながら、映画・放送事業者でないと手が出せなかった映像・放送メディアに対し、誰しものがパソコンやスマートフォンだけで手軽に扱える時代になった。その背景として、インターネットによる動画配信や動画投稿が活発になり、地上波・BS放送が完全にデジタル方式に移行し、ハイブリッドキャストに代表されるように、放送と通信の垣根が無くなってきたことが挙げられる。更に来年に東京オリンピックが開催されることもあり、既にBS波で4k/8kのスーパーハイビジョン放送も始まり、放送技術の進化は止まらない状態である。このように技術進化が目覚しく、より身近になった映像メディアを理解するため、動画技術の基盤となる音響信号処理と画像信号処理を学習することが益々重要になっている。

音響は1次元の時間次元に定義された媒体であるのに対し、画像は2次元の空間次元に定義された媒体で、更に動画は複数の2次元の画像が時間次元に定義された3次元の媒体である。ヒトが認識する感覚系も、音響が聴覚で、画像と動画が視覚となっており、音響と画像は互いに交わる要素が無く完全に異質な分野である。しかし、信号処理という観点では共通している部分が多い。まず、信号処理の代表的な技術として、周波数次元に変換するフーリエ変換<sup>1)</sup>が挙げられるが、変換対象は、時間次元か空間次元か、1次元か2次元か、即ち音響信号であるか画像信号であるかを問わない。

また、映像メディアの技術進化のコアは高精細化（スーパーハイビジョン、ハイレゾオーディオ）と多チャンネル化（マルチアングル再生、サラウンド音響）であり、いかなる時代においても最も重要視される信号処理技術はデータ圧縮<sup>2) 3)</sup>である。データ圧縮技術は可逆の方式と非可逆の方式があり、前者はロスレスとも呼ばれ圧縮による品質劣化を伴わず、文献4) 5) のようなプログラミング演習教材もある。前者は圧縮率が低いため、映像メディアでは主として後者の非可逆の方式が使用されるが、1次元と2次元という相違はあるものの、音響と画像のアプローチは驚くほど類似している。

更に、配信される映像メディアの品質向上に伴い、違法コピー対策などコンテンツの保護も重要になってきている。代表的なコンテンツ保護技術として、暗号化や電子透かし<sup>6) 7)</sup>があり、暗号化はいうまでも無いが、電子透かし技術においても、音響と画像に対して類似したアルゴリズムを適用できる。このように、音響信号処理と画像信号処理とでは、信号処理という観点では9割以上は共通の技術を適用でき、音響に固有な信号処理または画像に固有な信号処理を見出すことの方が寧ろ難しい。

これに対して、文献8) 9) は、動画処理技術についてC言語によるプログラミング事例を通して学習できるようになっているが、画像信号処理が主体であり、音響処理については音響データトラックの動画ファイルに対する読み書き方法に留まり、音響信号処理に対する記述が殆ど無い。ただし、音響信号処理に特化したものは文献10) -14) のように多数存在し、文献10) 11) ではWindowsPC上でC言語によるプログラミング事例を通して音響信号処理を学習できるようになっている。同様に、静止画像の信号処理に特化した

ものも、文献15) -17) のように多数存在し、いずれも静止画処理技術についてC言語によるプログラミング事例を通して学習できるようになっている。

このように、音響信号処理と画像信号処理の各々を個別に学習できる教本やプログラミング演習教材は多数存在するが、双方を対比させながら一緒に学べる教材が殆ど見当たらないため、本稿の趣旨である双方の信号処理上の共通性を把握することが困難であった。

筆者は、これまで担当してきた尚美学園大学・情報表現学科の授業「旧マルチフィールド体験演習」に活用することを主眼に、音響と画像の信号処理の基本を各々Windowsパソコンで学習できるように、表1に示すC言語で記述したアプリケーションソフトを各々15本程度開発してきた。

表1 制作した音響と画像の信号処理アプリケーションソフト一覧

(A) WAV形式音響信号処理ツール群	(B) BMP形式画像信号処理ツール群
(1A) 音響ファイルを波形で表示。“WAV_disp”	(1B) 画像ファイルを表示。“BMP_disp”
(2A) 音響ファイルを1次元スペクトルで表示。“WAV_fft”	(2B) 画像ファイルを2次元スペクトルで表示。“BMP_fft”
(3A) ファイルのハフマン符号化(ロスレス圧縮)“Huffman”	(3B) ファイルのハフマン符号化(ロスレス圧縮)“Huffman”
(4A) 波形のサンプル間引き・振幅圧縮“WAV_compress”	(4B) 画像の間引き・階調圧縮(ロッキー圧縮)“BMP_compress”
(5A) 線形予測誤差形式へ符号化(ロスレス圧縮)“WAV_diff” (5dA) 波形振幅の拡大縮小(振幅変調)“WAV_ampl”	(5B) 線形予測誤差形式へ符号化(ロスレス圧縮)“BMP_diff”
(6A) CD音源をハイレゾ音源(96kHz)に変換“WAV_highrs”	(6B) 画像の超解像処理(拡大と補間)“BMP_highrs”
(7A) 可聴電子透かし(波形合成、雑音制御)“WAV_add”	(7B) 可視電子透かし(画像合成)“BMP_synth”
(8A) 不可聴電子透かし(下位ビット置換法) 音楽にテキスト文字列を非可聴に埋め込む“WAV_text”	(8B) 不可視電子透かし(下位ビット置換法) 画像に文字パターン二値画像を埋め込む“BMP_embed” (8dB) 不可視埋込み画像の透かしを表示。“BMP_decode”
(9A) 音楽に別の音声のハイディング (エイリアシングを用いた埋め込み)“WAV_hide” (9dA) ハイディング音楽の間引き再生“WAV_samp”	(9B) 画像に別の画像のハイディング (エイリアシングを用いた埋め込み)“BMP_hide” (9dB) ハイディング画像の間引き表示“BMP_samp”
(10A) 音響信号の合成再生(聴覚マスキング演習)“WAV_mix”	(10B) 画像の合成表示(画像のマスキング、モアレ)“BMP_mix” (10dB) 画像の色分解処理(昔の写真、マスキング)“BMP_color”
(11A) ピンク雑音波形の発生“WAV_pink”	(11B) ピンク雑音を用いたテクスチャの生成“BMP_pink”
(12A) 白色雑音を用いた音声の補間実験“WAV_noise”	(12B) 白色雑音を用いた画像の補間実験“BMP_noise”
(13A) 音声の話速変換の実験“WAV_speed”	
(14A) 正弦波信号の生成(うなり・ビート演習)“WAV_sine”	(14B) 網点パターンの生成(画像のモアレ)“BMP_screen”
(15A) テキストよりモールス信号の生成“WAV_morse”	(15B) テキストよりQRコード画像の生成“BMP_QRcode”
	(16B) カラー画像をステレオ音声信号に変換。“BMP_WAVconv”

Windows環境では音響ファイルはWAV形式<sup>11)</sup>で、画像ファイルはBMP形式<sup>17)</sup>であり、入力系のプログラムは全く異なる。また、音響ファイルはサウンドカードを通してスピーカで再生し(関連ツール:Windows Media Player)、画像ファイルはビデオカードを通してディスプレイに表示するため(関連ツール:表1-(1B))、出力系のプログラムも全く異なる。しかし、途中の変換や加工処理の過程で、殆どの信号処理系のアルゴリズムを共用できることに気が付いた。即ち、音響信号処理または画像信号処理のいずれか一方を学習すれば、他方は学習しなくても類推できてしまう。あるいは、音響信号処理または画像信号処理のいずれかの学習で躓いた項目があれば、他方の対応する項目で学習を進めれば、先に躓いた項目を補習することができる、いわばクロスオーバー学習を実現できる。本稿では、表1に示す筆者が制作してきたWindowsパソコンで音響信号処理と画像信号処理のクロスオーバー学習を実現できる、音響と画像の信号処理アプリケーションソフトについて述べる。

## 2. 音響信号と画像信号の基本特性

### 2.1. 映像メディアのデジタル化と本稿で対象とするデータ形式

図1は映像メディアを構成する各種アナログ形式の信号をデジタル化した形態を示す。図1- (A) は音響信号を時間軸にデジタル化した音響データを示す。音響信号は時間軸上の正負の振幅値で定義された波形で表現されるように（関連ツール:表1- (1A) ）、時系列な音圧（マイクروفオンにより電圧に変換される）変化で、時間軸に沿って、例えば44,100分の1秒間隔（サンプリング周波数: 44.1kHz）にサンプリングした際の振幅値を16ビットでデジタル化（量子化ビット数: 16bits）したものがCD-DA（Compact Disc for Digital Audio）である。CD-DAは2チャンネルのステレオ形式をサポートするため、実際には図1- (A) の形態のデータが2系列収納されている。更に5.1チャンネルのサラウンドオーディオの場合には、図1- (A) の形態のデータが6系列収納される。

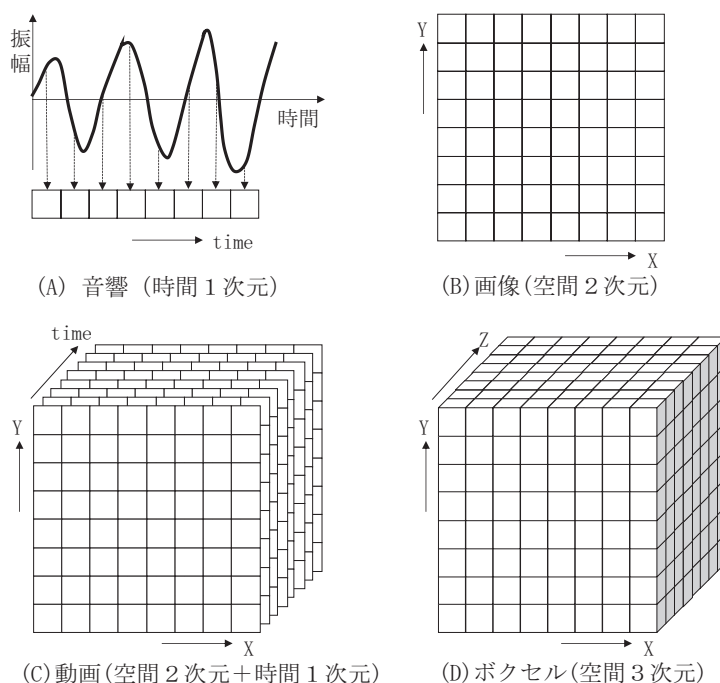


図1 映像メディアのデジタル化と本稿で対象とするデータ形式

図1- (B) は画像信号を空間的にデジタル化した画像データを示す。画像データは被写体空間を2次的に画素に分割し、RGBの3原色のフィルタを通して各画素の3原色の輝度値を各々8ビットでデジタル化したものが一般的な映像メディアに使用される。映像メディアでは画面比率が16:9であることを前提に、横方向の画素数で規格化され、1920画素で構成されるのがハイビジョン(2K)で、スーパーハイビジョンでは4K, 8K, 16K画素で構成される。前述の音響データと同様に立体視が行えるステレオ形式もあり、図1- (B) の形態のデータが2セット収納される。

映像メディアでは図1- (B) の画像データをフレームと称し、図1- (C) のように時系列に複数の画像データが配置された動画の形式で使用される。また、図1- (C) の時間軸に同期して図1- (A) の音響データもステレオ形式で2系列付加された構成が一般的である。

図1- (C) の動画は空間2次元+時間1次元の3次元構成であるが、図1- (D) のように空間3次元で定義されるボクセル媒体もある。これは3次元空間をある方向 (Z軸) に沿って断裁し、その方向に複数の2次元の断層画像が配置された形態で、CT (Computed Tomography) スキャナなどで断層撮像されたデータが代表的である。いわば3次元の空間を3次元的に画素 (3次元空間ではボクセルと呼ぶ) に分割したもので、CTスキャナでは各ボクセルの放射線 (X線) の透過強度を16ビットで表現したものが使用される。最近では医療用のCTスキャナやMRI (Magnetic Resonance Imaging) スキャナの撮像速度が高速化され、図1- (D) の一連のボクセルデータが単一のフレームとして、図1- (C) のように時系列に配置された動画形態で収集することも可能になった。この場合は図1- (D) を時間次元に拡張した空間3次元+時間1次元の4次元構成になる。

これと混同されやすいのに、3次元コンピュータグラフィックス映像がある。3次元コンピュータグラフィックスでは、制作過程では3次元的にデータが定義されるが、最終的にはレンダリングを行って2次元の投影画像を生成して映像メディアに使用される。従って、図1- (B) または図1- (C) の形態になる。

本稿では図1- (A) の音響と図1- (B) の画像のデータ形式に限定して以下述べる。

## 2.2. 音響と画像データの品質を決定する要素と品質変換の基本

図2- (1A) は図1- (A) の音響に対して横方向の時間軸のサンプリング間隔を小さくしたもので、これにより図1- (A) に比べ時間解像度および品質が向上する。また、図示していないが各サンプリング位置で振幅値をデジタル化する際のビットの深さを長くしても品質は向上する。近年登場したハイレゾ音源はCD-DA規格であるサンプリング周波数44.1kHzまたは量子化ビット数16bitsより大きな値に設定してデジタル化したものである (関連ツール:表1- (6A)) 。

図2- (1B) は図2- (1A) とは逆に図1- (A) の音響に対して横方向の時間軸のサンプリング間隔を粗くしたもので、これにより図1- (A) に比べ時間解像度および品質が低下する (関連ツール:表1- (4A)) 。

また、フィルタ処理により品質を変化させることができ、図2- (1B) の矢印で示されているように、両サイドに隣接するサンプルの振幅値に適切な重みを掛けて中央のサンプルの振幅値を改変すれば、元の音響信号に混入しているノイズをカットしたりすることができる。

図2- (1B) は画像データに対して図2- (1A) と同様な操作を行ったもので、2次元的に横方向および縦方向のサンプリング間隔を小さくしている。これにより図1- (B) に比べ空間解像度および品質が向上する (関連ツール:表1- (6B)) 。

同様に、輝度値をデジタル化する際のビットの深さを長くしても品質は向上する。前述のようにスーパーハイビジョンでは4K, 8K, 16K画素のように、横方向および縦方向のサンプリング間隔を各々ハイビジョンの1/2, 1/4, 1/8に設定している。

図2- (2B) は画像データに対して図2- (2A) と同様な操作を行ったもので、図2- (1A) とは逆に図1- (B) の画像に対して横方向および縦方向のサンプリング間隔を粗くしたもので、これにより図1- (B) に比べ空間解像度および品質が低下する (関連ツール:表1- (4B)) 。

同様に、フィルタ処理により品質を変化させることができ、図2- (2B) の実線の矢印で示されている4近傍の画素、更に破線の矢印で示されている斜め方向の画

素を加えた8近傍の画素の輝度値に適当な重みを掛けて中央の画素の輝度値を改変すれば、画像を先鋭にしたり逆にぼかしたりすることができる。

以上のように音響と画像の品質を決定する要素および品質変換の操作は基本的に同じで、1次元か2次元かの相違に過ぎない。ただし、画像の場合、2次元であるという特徴を活かして、横方向と縦方向のサンプリング間隔を変える変倍処理が行え、これについては音響信号処理では実現できない画像信号処理に固有な操作といえる。

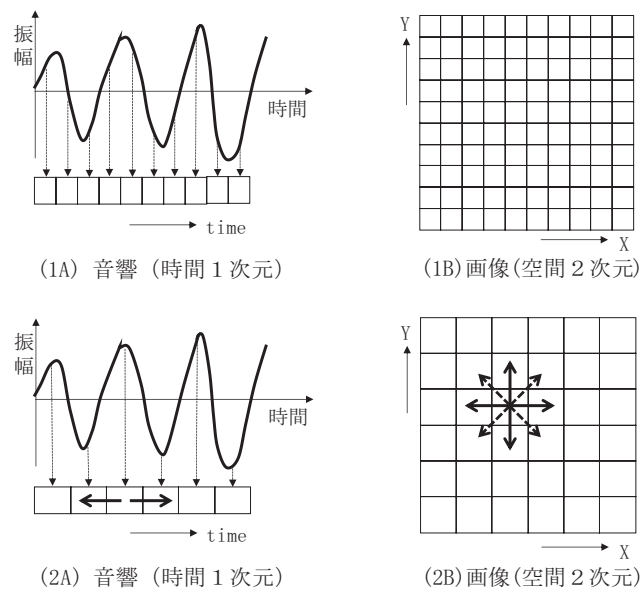


図2 音響と画像データの品質を決定する要素と品質変換の基本

### 2.3. 信号処理の基本:周波数次元への変換

前述の通り、音響は1次元の時間次元に定義された媒体であるのに対し、画像は2次元の空間次元に定義された媒体であるため、オリジナルの次元のまま処理を行おうとすると、アプローチが異なり問題が難しくなることがある。そこで、周波数次元という第3の統一的な次元に変換して処理を行う方法も用いられる。代表的な手法がフーリエ変換<sup>1)</sup>で、対象がデジタル化されたデータの場合は、離散フーリエ変換が用いられ、処理対象のサンプル数を2の累乗個に正規化できれば、高速フーリエ変換アルゴリズム (FFT: Fast Fourier Transform) を適用できる。

図3はデジタル化された音響信号に対して1次元離散フーリエ変換を用いて信号処理を行う基本構成を示す。ただし、音響信号に対してそのまま離散フーリエ変換を施すと、膨大な処理時間がかかり効率的な信号処理が行えないため、与えられた音響信号を100msec程度の区間 (フレームと呼ばれる) に等分割して、区間ごとに信号処理を行う。この時、分断された区間に対してそのままフーリエ変換を施すと疑似周波数成分が生成されるため、区間の両端の信号を減衰させる窓関数を乗算した上で、信号処理を行う。その際、各区間の両端の信号成分があまり反映されなくなるため、区間分割にあたり隣接

区間とオーバーラップさせる。図3- (1) に示すソース音響信号は、このように窓関数で重みづけされて、分割された1区間である。与えられた音響信号がM個の区間に等分割された場合は、図3に示す信号処理がM回実行される。

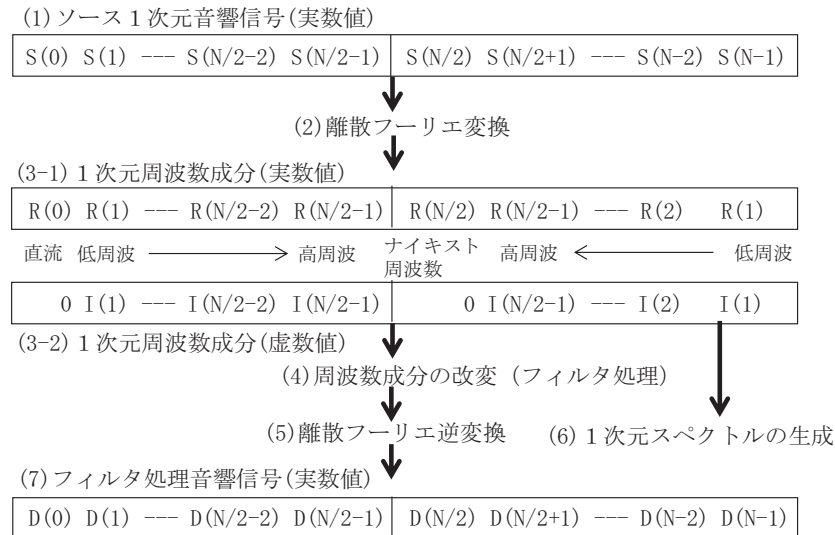


図3 1次元離散フーリエ変換を用いた音響信号処理の基本構成

離散フーリエ変換ではソース信号も変換信号も複素数列になるが、図3- (1) のようにソース信号として音響信号を与える場合は、N個の実数列のみ与えられ、虚数列の値は全て0として処理される。離散フーリエ変換 (図3- (2) ) を実行すると、図3- (3-1) および図3- (3-2) で構成される複素数の変換信号が得られる。ソース信号のサンプリング周波数をFとすると、図3- (3-1) では、R (0) が直流成分で、R (1) ,...,R (N/2) が周波数 F/N, ...,F/2 (最高の周波数、ナイキスト周波数と呼ばれる) に対応する実数の周波数成分である。図3- (3-1) の右側に配置されているR (N/2-1) ,...,R (1) は左側の成分が対称的にコピーされたものであるため、変換される実数成分はR (0) ,...,R (N/2) の N/2+1 個になる。図3- (3-2) では、直流成分とナイキスト周波数成分は無いため、I (1) ,...,I (N/2-1) が周波数 F/N, ...,F・(N/2-1)/N に対応する虚数の周波数成分である。図3- (3-2) の右側に配置されているI (N/2-1) ,...,I (1) は同様に左側の成分が対称的にコピーされたものであるため、変換される虚数成分はI (1) ,...,I (N/2-1) のN/2-1個になり、実数成分と合わせてN個になる。

図3- (3-1) および図3- (3-2) で構成されるN個の実数と虚数の周波数成分に対して種々の改変を行うことができる (図3- (4) ) 。例えば、低域フィルタを実現する場合は、R (1) ,...,R (N/2) および I (1) ,...,I (N/2-1) の要素の中で、カットオフ周波数より高い周波数成分の値を0にする。そして、離散フーリエ逆変換 (図3- (5) ) を実行して逆変換されたN個の実数成分を抽出すれば、低域フィルタ処理された音響信号 (図3- (7) ) として得ることができる。



また、図3- (3-1) および図3- (3-2) で構成されるN個の実数と虚数の周波数成分を正規化して1次元のグラフにプロットすれば1次元スペクトルを生成することができる(図3- (6))。ただし、生成されるスペクトルは前述のようにM個の区間に等分割された音響信号の1区間に対応するため、音響信号全体ではM種の1次元スペクトルが生成される。そのため、1次元グラフにプロットする際は、M種の1次元スペクトルの平均をプロットするか(関連ツール:表1- (2A))、1次元スペクトルを縦軸にプロットして横方向にM種の1次元スペクトルを時系列に配置した2次元グラフ(スペクトログラム、ソノグラムと呼ばれる)で表現する方法が用いられる。

#### 2.4. 周波数変換の2次元拡張

離散フーリエ変換の対象は音響信号のように時間次元に定義されたものである必要はなく、画像信号のように空間次元に定義されたものでも良い。ただし、画像の場合は2次元であるため、離散フーリエ変換を2次元的に行う必要があり、そうすると膨大な処理時間がかかってしまう。そこで、図3で用いた1次元離散フーリエ変換をX軸方向とY軸方向の2度に分けて実行する方法がとられる。

図4はデジタル化された画像信号に対して1次元離散フーリエ変換および1次元離散フーリエ逆変換を各々2度実行して信号処理を行う基本構成を示す。図4- (1) に示すソース2次元画像が $N \times N$ 画素で構成されるとすると、はじめに図4- (1) に示すように水平方向にN画素で構成されるN本の走査線の各々に対して1次元離散フーリエ変換をN回実行する(図4- (2))。続いて、水平方向フーリエ変換像に対して、図4- (3) に示すように垂直方向に1次元離散フーリエ変換をN回実行する(図4- (4))。この時、図4- (4) の1次元離散フーリエ変換に入力される値は図4- (2) の1次元離散フーリエ変換により変換された複素数成分になる。

図4- (4) までの離散フーリエ変換により2次元離散フーリエ変換と等価な処理が行われ、図4- (5) に示す2次元周波数成分が得られる。この像では4隅が直流成分で中央に最高のナイキスト周波数の成分が存在する。信号処理をしやすいように、習慣的に右上の第1象限と左下の第3象限の成分を入れ替え、左上の第2象限と右下の第4象限の成分を入れ替える処理が行われる(図4- (6))。これにより図4- (7) に示すように、2次元周波数成分の中央が直流成分で4隅に最高の周波数の成分が位置するようになる。

図4- (7) に示す $N \times N$ 個の複素数の周波数成分に対して改変を行うことができる(図4- (8))。例えば、低域フィルタを実現する場合は、中央よりカットオフ周波数を半径とする円形領域の外側の高い周波数成分の値を0にする。そして、改変された周波数成分に対して図4- (10) に示すように垂直方向に1次元離散フーリエ逆変換をN回実行する(図4- (11))。続いて、垂直方向フーリエ逆変換像に対して、図4- (12) に示すように水平方向に1次元離散フーリエ逆変換をN回実行する(図4- (13))。これにより、2次元離散フーリエ逆変換と等価な処理が行われ、逆変換像の実数成分だけを抽出すれば図4- (14) に示すフィルタ処理2次元画像が得られる。

また、図4- (7) に示す $N \times N$ 個の複素数の周波数成分を正規化して2次元のグラフにプロットすれば2次元スペクトルを生成することができる(図4- (9))、関連ツール:表1- (2B))。

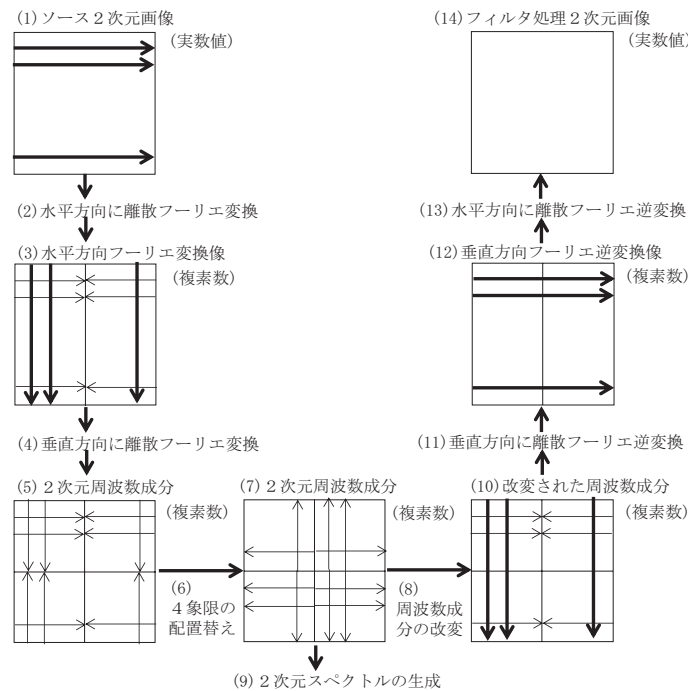


図4 2次元離散フーリエ変換を用いた画像信号処理の基本構成

## 2.5. 1次元・2次元ピンクノイズの生成とスペクトル表示

雑音の中で耳にする機会が多く計測にも使用されるものに、白色雑音 (white noise) とピンク雑音 (pink noise) がある。白色雑音は滝のように水が流れる音や放送中止中にテレビやラジオ放送で再生される音で、テレビの画面に映る映像 (砂嵐) も2次元の白色雑音である。これらを前節で述べた1次元スペクトルや2次元スペクトルに変換すると、全ての周波数成分が均一の強度をもつ特性を示す。電磁波の可視光線でこのようなスペクトル特性をもつと白色になるので白色雑音と呼ばれる。

一方、ピンク雑音は周波数に反比例して強度が下がるスペクトル特性をもつ雑音で、 $1/f$  雑音とも呼ばれ、白色雑音に比べ心地良さを与える。電磁波の可視光線でこのようなスペクトル特性をもつと赤系が強いピンク色になるのでピンク雑音と呼ばれる。エアコンやファンの騒音や、画像では木目やテキスタイルにこのようなスペクトル特性が見られる。

表1- (11A) および (11B) のツールで、白色雑音またはピンク雑音を生成することができる。白色雑音は擬似乱数を1次元または2次元に配置すれば良い。ピンク雑音は、白色雑音に対して時間周波数または空間周波数に対応するサンプル周期の区間をその平均値で置換することにより生成し、置換する平均値にサンプル周期の値で乗算 (周波数の逆数を乗算) すれば  $1/f$  特性をもたせることができる。ナイキスト周波数の範囲で周波数を連続的に変化させ、各周波数に対応するサンプル周期の区間に平均値を一様に加算して白色雑音の分布を改変する操作を行う。2次元のピンク雑音を生成する場合は、前述の2次元離散フーリエ変換と同様に、水平方向と垂直方向の2回に分けて白色雑音の分布を改変する操作を行う。

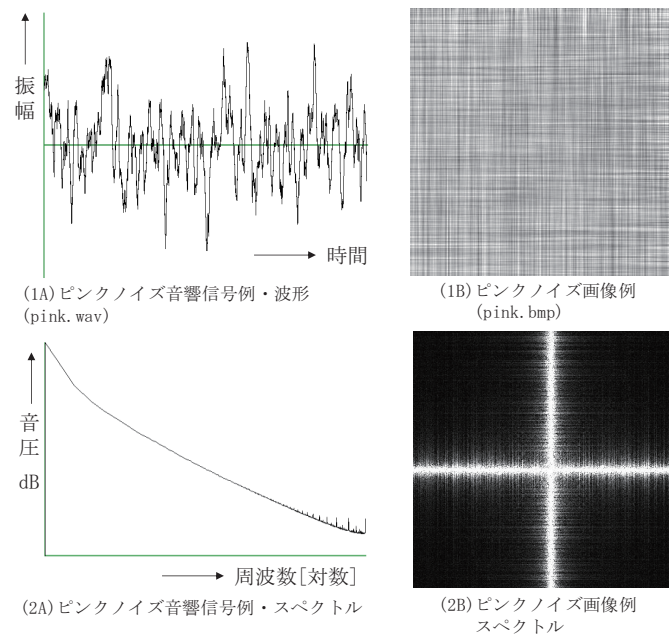


図5 1次元・2次元ピンクノイズの生成例とスペクトル表示

表1- (11A) のツールにより1次元のピンク雑音を生成し、表1- (1A) のツールにより表示した波形を図5- (1A) に、表1- (2A) のツールにより表示したスペクトルを図5- (2A) に示す。縦軸の音圧はデシベル (対数) で横軸の周波数も対数でプロットしているため、 $1/f$  特性は右下がりの直線になる。同様に、表1- (11B) のツールにより2次元のピンク雑音を生成し、表1- (1B) のツールにより表示した画像を図5- (1B) に、表1- (2B) のツールにより表示した2次元スペクトルを図5- (2B) に示す。図4- (6) の4象限の配置替えを行っているため、中央が直流成分で、4隅が最高の周波数成分になる。周波数のスケールは線形であるため、中央から4隅に向かって $1/f$ 特性で減衰する分布になる。

### 3. 音響信号と画像信号の圧縮と符号化

2.2節で述べたようにデジタル化された音響データと画像データの品質はサンプリングの間隔 (解像度) と各サンプルを量子化するビット長で決まり、原理的には品質に比例してデータ量が増大する。しかし、データ量が増大するにつれ、品質に大きな影響を与えずにデータ量を削減できる冗長性も増大する特性があるため、効率的に符号化するデータ圧縮技術が重要になる。特に映像メディアの中核をなす動画においては、図1- (C) のように3次元的にデータ量が増大するため、データ圧縮技術なしに伝送を実現することは困難である。来年からサービスが開始される 5G のように、通信速度も向上しているが、通信速度の向上と伝送コンテンツ品質の向上はいたちごっこの関係にあり、いつの時代においてもデータ圧縮技術が要求される<sup>2) 3)</sup>。

データ圧縮には原データを完全に復元でき品質劣化の無い可逆圧縮 (ロスレス圧縮) と品質の劣化を伴う非可逆圧縮 (ロッキー圧縮) の方式がある。可逆圧縮は ZIP, LHA のようなファイル統合ツールに使用され<sup>4) 5)</sup>、対象が音響や画像データのようにサンプリングされたデータ配列の場合、一般に圧縮されにくい。そのため、音響や画像データに対して

実用的な圧縮率を実現するため非可逆圧縮ツールが開発された。非可逆圧縮ツールでは圧縮率に依存して品質が劣化し、原データを復元できない。

### 3.1. 可逆圧縮と音響と画像の非可逆圧縮

図6は制作した音響と画像の可逆圧縮および非可逆圧縮ツールの処理構成を示す。左側は音響データで右側は画像データに対する圧縮と伸張処理の流れを示し、各々左側は非可逆圧縮、右側は可逆圧縮の処理である。いずれの場合においても、図6- (6) に示す代表的なハフマン方式による可逆圧縮をコアにして圧縮を行うが、前述の通り図6- (1A) の音響データや図6- (1B) の画像データをそのまま図6- (6) の可逆圧縮処理に渡しても殆ど圧縮されない。その理由は、サンプリングされたデータ配列の数値が多様で出現頻度分布が比較的均一であるためである。そこで、図6- (2A) (3A) (2B) (3B) に示す前処理により、データ配列の数値範囲やデータ配列の出現頻度分布に偏りをもたせることが重要になる。

音響データに対する代表的な非可逆圧縮方式であるMP3 (MPEG-1/Layer3)、画像データに対する代表的な非可逆圧縮方式であるJPEGでは、第2章で述べた離散フーリエ変換を簡素化したDCT (Discrete Cosine Transform) を用いて周波数次元に変換し、聴覚や視覚の感度が低い高周波成分のデータを粗くすることにより、可逆圧縮をかかりやすくしている。図6では、音響に対して図6- (2A) (関連ツール:表1- (4A)) と画像に対して図6- (2B) (関連ツール:表1- (4B)) により、図2- (2A) (2B) のようにサンプリング間隔を粗くし品質を低下させることによりデータ配列の数値範囲が偏り、可逆圧縮をかかりやすくしている。この方法では、サンプリング間隔や各サンプルを量子化するビット長を増減させることにより圧縮率を調整することができる。

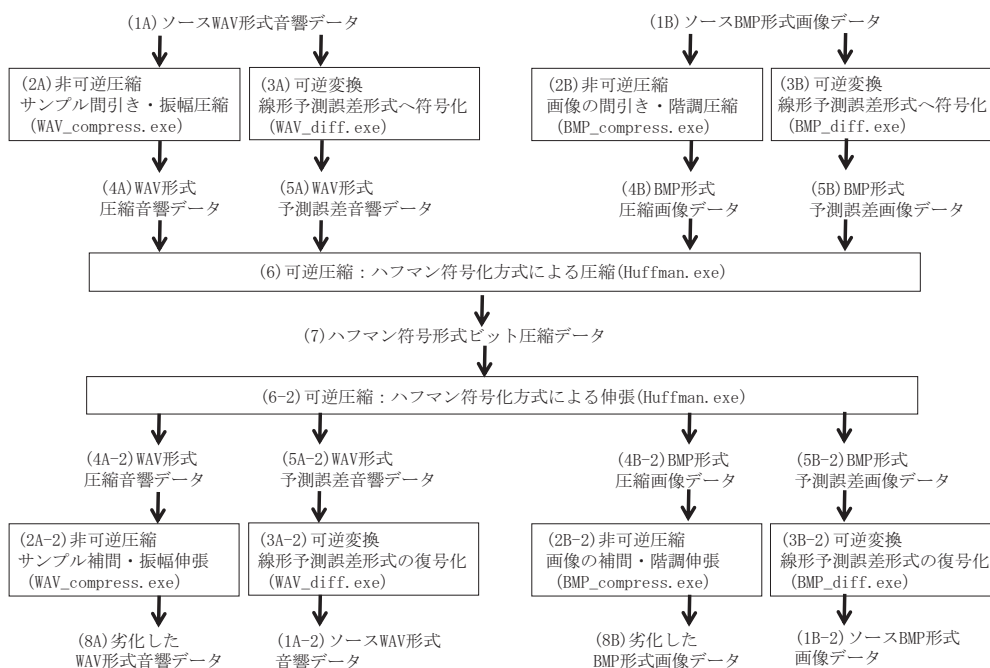


図6 制作した音響と画像の可逆圧縮および非可逆圧縮ツール群

### 3.2. 音響と画像の可逆圧縮を実現するための線形予測符号化

前節の非可逆圧縮では少なからず品質劣化を伴うが、音響と画像のコンテンツの保存や制作の過程でオリジナル品質の維持が求められる場合がある。特にハイレゾ音響では配信過程においても原品質の維持が求められる。そこで、音響や画像データに対して原品質を維持しながら可逆圧縮をかかりやすくする前処理として線形予測符号化が使用される<sup>4)</sup>。これは各サンプルの振幅値や輝度値をそのまま符号化せず、近傍の既知のサンプルを基に予測した値と実際の数値との差分（予測誤差）を符号化する方式である。

例えば、音響データの場合、各サンプルの振幅値を直前のサンプルの振幅値との差分値で符号化する。これは、既知の直前のサンプルの振幅値をそのまま次のサンプルの予測値とし、次のサンプルの実際の振幅値と予測値との予測誤差を符号化しているものと捉えることができる。この方式では、第1サンプルの振幅値を基に全てのサンプルの振幅値を復元できる。誤差データの形式で符号化すると、誤差データの数値の出現頻度分布が0近辺に偏り可逆圧縮がかかりやすくなる。

これを拡張したのが図7- (A) で、音響データに対して左から3番目の破線で示すサンプルを符号化する際、直前サンプル1と直前サンプル2の2つの既知のサンプルを基に、より精度の高い予測を行うようにしている。これにより、予測誤差データがより0に近付き可逆圧縮による圧縮効果が大きくなる。予測を算出する計算式が直前サンプルに係数を乗算するだけの形態なため、線形予測と呼ばれる。図8- (1A) の音響信号に対して図7- (A) の方法により線形予測符号化を行った予測誤差信号を図8- (5A) に示す。図8- (5A) では縦軸を20倍に拡大しており、誤差値は0近辺に集中している。

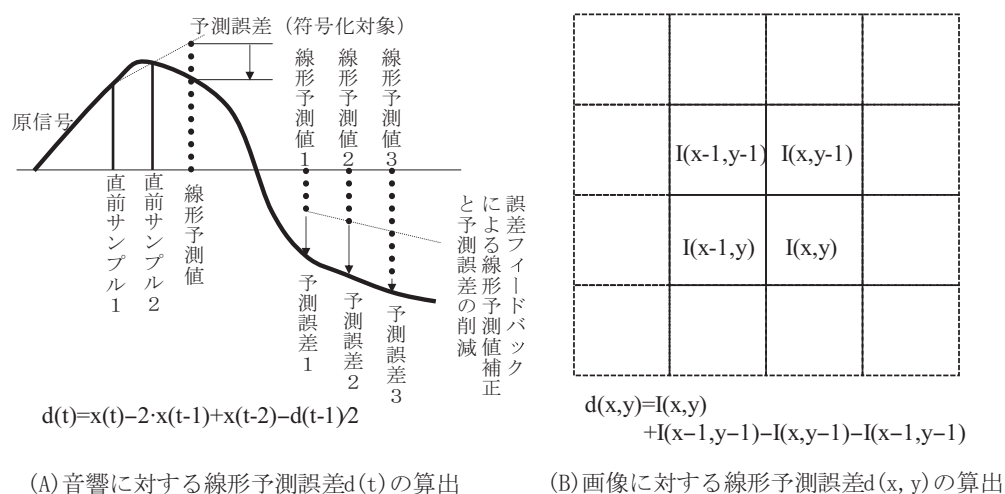


図7 音響と画像に対する線形予測符号化

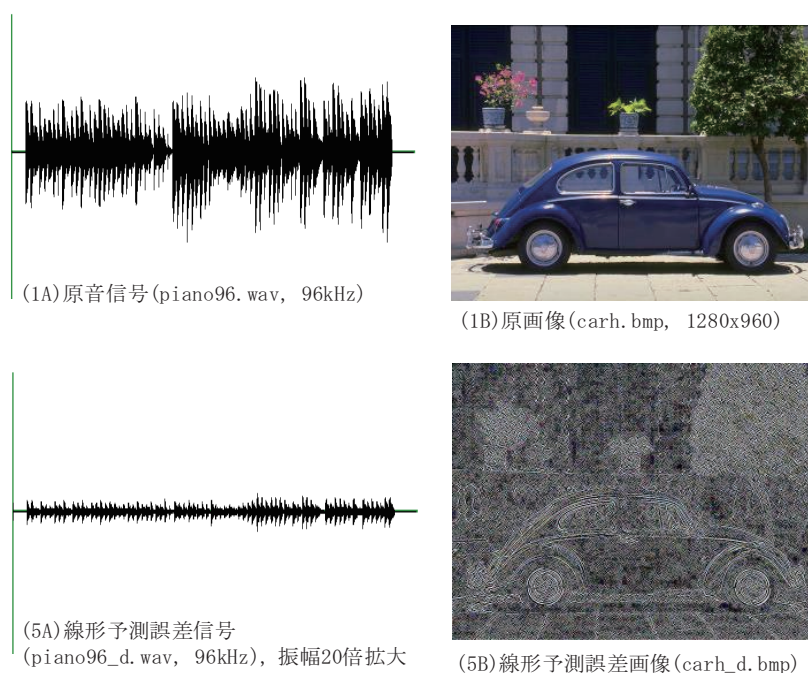


図8 音響と画像に対する線形予測符号化の具体例

図7- (B) は画像に対して2次元的に線形予測符号化を行う方法を示し、中央の $2 \times 2$ 画素の右下  $I(x,y)$  の輝度値を近傍の3つの画素の輝度値で予測し、予測誤差を符号化する方法を示している。図8- (1B) の画像データに対して図7- (B) の方法により線形予測符号化を行った予測誤差画像を図8- (5B) に示す。図8- (1B) の原画像はフルカラー画像であるが、図8- (5B) の予測誤差画像では RGB 3つの分色の誤差値がいずれも0近辺に集中しモノクロ画像のように変化している。

線形予測符号化方式の場合、前節で述べた非可逆圧縮方式のように圧縮率を増減させることが難しく、圧縮率も音響の場合は50%前後、画像の場合は30%前後で、非可逆圧縮方式に比べ圧縮率に限界がある。

### 3.3. 制作した音響と画像の可逆圧縮および非可逆圧縮ツールを用いた圧縮実験

図8- (1A) に示す音響データと図8- (1B) に示す画像データに対して、図6に示す一連の音響と画像の可逆圧縮および非可逆圧縮ツールを用いて圧縮実験を行い、各処理により変換されたファイルの容量を表2に示す。表2- (1) は図8- (1A) に示す音響ファイルと図8- (1B) に示す画像ファイルの容量で、各々に対して図6- (6) の可逆圧縮を施した結果が表2- (2) である。音響と画像の可逆圧縮後のファイル容量は表2- (1) と殆ど変わらないことがわかる。図8- (1A) に示す音響ファイルに対してサンプリングを1/4に間引きし、量子化ビット長を16bits から8bits に落とし、図8- (1B) に示す画像ファイルに対して縦横の画素数を各々1/4に間引きし、量子化ビット長を8bits から7bits に落とした結果が表2- (3) である。音響と画像の可逆圧縮後のファイル容量は表2- (1) に比べ各々6%程度に圧縮され、この段階で図6- (6) の可逆圧縮を施すと更に圧縮がかかることがわかる。

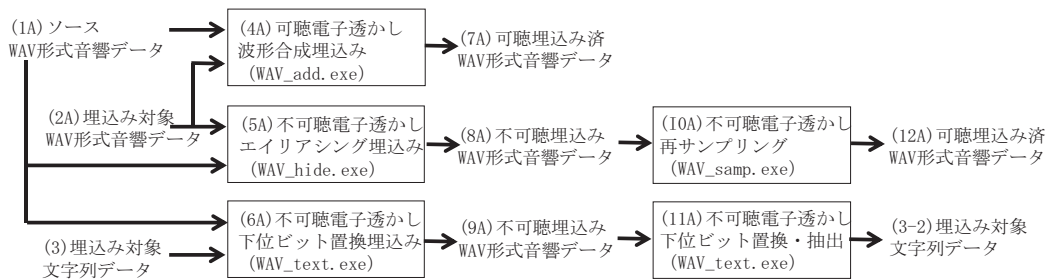
表2 制作した音響と画像の可逆圧縮および非可逆圧縮ツールを用いた実験結果

	音響情報	サイズ [kBytes]	画像情報	サイズ [kBytes]
(1) ソースファイル	piano96.WAV	5,695 (100%)	carh.BMP	3,686 (100%)
(2) ハフマン符号化 (可逆圧縮)	Huffman piano96.CMP	5,279 (92%)	Huffman carh.CMP	3,507 (95%)
(3) 品質劣化 (非可逆圧縮)	WAV_compress piano96_c.WAV サンプル1/4	356 (6.2%)	BMP_compress carh_c.BMP 縦横1/4縮小	230 (6.2%)
+				
(4) ハフマン符号化 (可逆圧縮)	Huffman piano96_c.CMP	262 (4.6%)	Huffman carh_c.CMP	189 (5.1%)
(5) 線形予測誤差 (可逆圧縮)	WAV_diff piano96_d.WAV	5,695 (100%)	BMP_diff carh_d.BMP	3,686 (100%)
+				
(6) ハフマン符号化 (可逆圧縮)	Huffman piano96_d.CMP	2,091 (36%)	Huffman carh_d.CMP	1,646 (44%)

図8- (1A) に示す音響データと図8- (1B) に示す画像データに対して、前節で述べた線形予測符号化を行った、図8- (5A) (5B) に示される音響と画像の予測誤差データのファイル容量が表2- (5) で、表2- (1) と同一である。しかし、これらに図6- (6) の可逆圧縮を施すと表2- (6) に示されるように40%前後に圧縮がかかっている。左欄の音響情報と右欄の画像情報を比較すると、ファイル容量の絶対値は異なるものの圧縮比率は非常に類似した傾向を示すことがわかる。

#### 4. 音響信号と画像信号への電子透かしの埋め込み

前章で述べたように、通信速度が向上するにつれ映像メディアのコンテンツ品質も向上しており、それに伴い違法コピー品の品質も向上し違法コピーによる損害も大きくなる。例えば、Blu-Ray Disc からデジタル的に違法コピーを行うことはコンテンツの暗号化により規制できても、Blu-Ray Disc を再生中のアナログの映像信号や音声信号を DVD などにアナログ的にコピーすることはできてしまう。いわゆるアナログホール<sup>13)</sup> と呼ばれ、デジタルシネマのスクリーンをカメラで盗撮する方法もある。アナログコピーでは品質劣化を伴うが、ソースの映像メディアの品質が高くなると、違法コピー品の Blu-ray や DVD は正規品と同等レベルの商用品質になってしまう。この対策として、違法コピー品を識別できるように音響や画像・動画コンテンツ自体にアナログ耐性をもつ電子透かしを埋め込む、種々の電子透かし技術<sup>6) 7)</sup> が開発されている。



(A) 音響信号に対する各種電子透かしの埋め込み

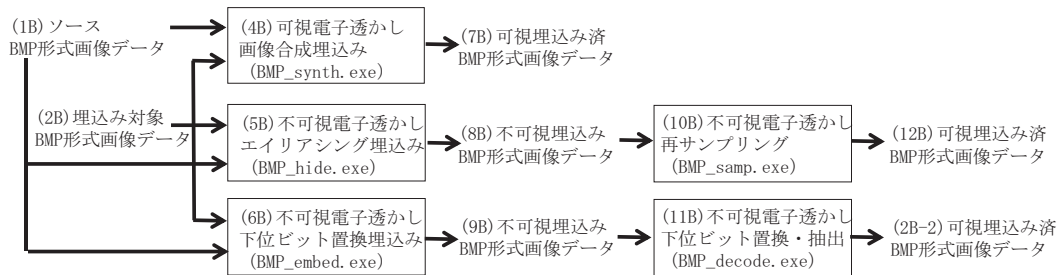


図9 制作した音響と画像データに対する電子透かしツール群

#### 4.1. 音響と画像データに対する電子透かし

図9は制作した音響と画像データに対する各々3種類の電子透かしツールの処理構成を示す。図9- (1A) に示す音響データに対して、図9- (4A) (5A) (6A) の3種のツール（関連ツール:表1- (7A) (9A) (8A) ）により、図9- (2A) に示す別の音響データや図9- (3) に示すテキストデータを埋め込むことができる。同様に、図9- (1B) に示す画像データに対して、図9- (4B) (5B) (6B) の3種のツール（関連ツール:表1- (7B) (9B) (8B) ）により、図9- (2B) に示す別の画像データを二値で埋め込むことができ、現状は実装していないが、図9- (3) に示すテキストデータも埋め込むことができる。

埋め込み方法としては、紙幣の透かしのよう、見えるように、または聞こえるように埋め込む可聴または可視の電子透かしの方法と、気づかれないように埋め込む不可聴または不可視の電子透かしの方法がある。後者は情報ハイディングとも呼ばれ、技術的に高度で実用的な意義が大きい。

#### 4.2. 音響と画像データに対する可聴または可視の電子透かし埋め込み手法

図10- (1A) は図9- (1A) に対応する音響データで、図9- (4A) の埋め込みツール（関連ツール:表1- (7A) ）により図9- (2A) に対応する別の音響データ（図10- (2A) ）を聞こえるように埋め込み、図9- (7A) に対応する可聴の電子透かし埋め込み音響データ（図10- (3A) ）を生成する具体例である。図10 (2A) として著作権者情報を読み上げた音声メッセージなどを使用すれば、電子透かし埋め込み音響データ（図10- (3A) ）を商用使用することが困難になる。種々の音響編集ツールを駆使しても、図10- (3A) の波形より邪魔な図10- (2A) のメッセージを削除することは困難である。しかし、埋め込んだ図10- (2A) の音声メッセージを持っている著作権者は以下のように削除可能である。



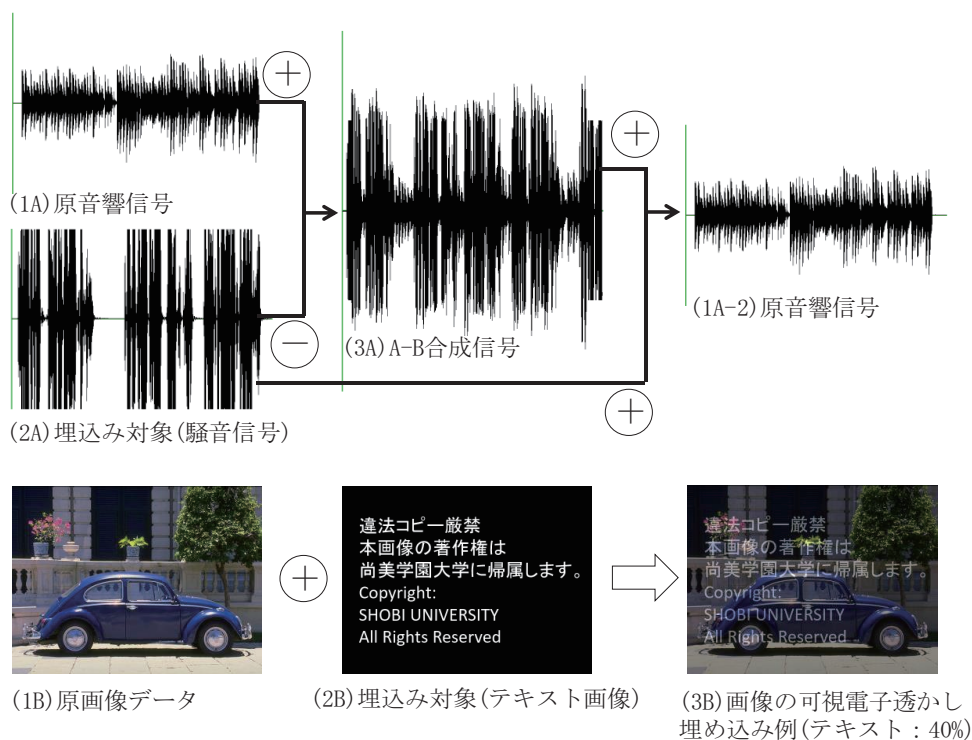


図10 音響と画像データに対する可聴または可視の電子透かし埋め込み

図9- (4A) の埋め込みツールは2つの入力波形を単純に加算しているだけであるが、図10- (2A) を減算させることもできる (減算とは図10- (2A) の波形振幅を正負反転させて加算する処理である)。そうすると、可聴の電子透かし埋め込み音響データ (図10- (3A) ) に対して、図10- (2A) の別の音響データを図9- (4A) の埋め込みツールを用いて再度加算する処理を行えば、図10- (1A-2) のように図10- (1A) と同一の原音響データを復元できる (可逆な電子埋め込みとも呼ばれる)。

この手法はアクティブノイズコントロール (ANC) やノイズキャンセリング・ヘッドフォンで活用されている。図10- (2A) を邪魔な騒音信号とすると、周囲の騒音をマイクロフォンで取り込み、波形を正負反転して図10- (1A) の原音響データに加算し、図10- (3A) の騒音が合成されたA-B合成信号をヘッドフォンに流す。ヒトはヘッドフォンから流れる図10- (3A) のA-B合成信号と周囲から聞こえる図10- (2A) の騒音信号を合成して聴取するので、騒音信号が削除された図10- (1A-2) の原音響データのみが耳に入ってくる。この機能は正負符号をもつ振幅値の特性を活用しているため、下記の負の輝度値をもたない画像データでは実現できない音響に固有の特徴である。

図10- (1B) は図9- (1B) に対応する画像データで、図9- (4B) の埋め込みツール (関連ツール:表1- (7B) ) により図9- (2B) に対応する別の画像データ (図10- (2B) ) を見えるように埋め込み、図9- (7B) に対応する可視の電子透かし埋め込み画像データ (図10- (3B) ) を生成する具体例である。図10- (2B) として著作権者情報を表示したテキスト画面などを使用すれば、電子透かし埋め込み画像データ (図10- (3B) ) を商用使用することが困難になる。種々の画像編集ツールを駆使しても、図10- (3B) の画像より邪魔な図10- (2B) のメッセージを削除することは困難である。

### 4.3. 音響と画像データに対する下位ビット置換法による情報ハイディング手法

図11- (A) は図9- (1A) に対応する音響データ（カバーデータと呼ばれる）に対して、図9- (6A) の埋め込みツール（関連ツール:表1- (8A) ）により図9- (3) に対応するテキストデータ（ペイロードと呼ばれる）を聞こえないように埋め込み、図9- (9A) の不可聴の電子透かし埋め込み音響データ（ステゴデータと呼ばれる）を生成する原理図である。CD-DA 形式の音響データの場合、各サンプルごとにステレオのLチャンネルとRチャンネルの振幅値が16bits でデジタル化されたデータ配列になっており、最下位の1bit を意図的に反転させても音響データの波形や聞こえは殆ど変化しない。そこで、各サンプルの最下位のビットを図9- (3) のテキストデータの文字コードのビット列に置換する。図9- (9A) の不可聴の電子透かし埋め込み音響データから、図9- (11A) の透かし抽出ツール（関連ツール:表1- (8A) ）により、図9- (3) と同一の図9- (3-2) のテキストデータを抽出することができる。

この方法は、処理が簡便で埋め込みによる品質劣化が殆ど無いという利点があるが、第3章で述べた非可逆圧縮やアナログ変換を行うことにより埋め込まれた電子透かしが破壊されるという弱点がある。現状ではアナログ耐性をもたせた電子透かし埋め込みツールについては実装していないが、文献13) のように多くの開発事例がある。

図11- (B) は図9- (1B) に対応する画像データ（カバーデータ）に対して、図9- (6B) の埋め込みツール（関連ツール:表1- (8B) ）により図9- (2B) に対応する図10- (2B) のような二値のテキスト画像（ペイロード）を見えないように埋め込み、図9- (9B) の不可視の電子透かし埋め込み画像データ（ステゴデータ）を生成する原理図である。BMP 形式のフルカラー画像データは各画素ごとに RGB の3分色の輝度値が8bits でデジタル化されたデータ配列になっている。音響データに比べビット長が短いため、最下位の1bit を意図的に反転させると画像データの変化が識別できる場合があるが、同様に各画素の最下位のビットを図10- (2B) の二値画像に対応する画素のビットに置換する。図9- (9B) の不可視の電子透かし埋め込み画像データから、図9- (11B) の透かし抽出ツール（関連ツール:表1- (8dB) ）により、図10- (2B) の二値画像を抽出したり、図9- (2B-2) に対応する図10- (3B) のテキスト画面が合成された画像データを生成することができる。

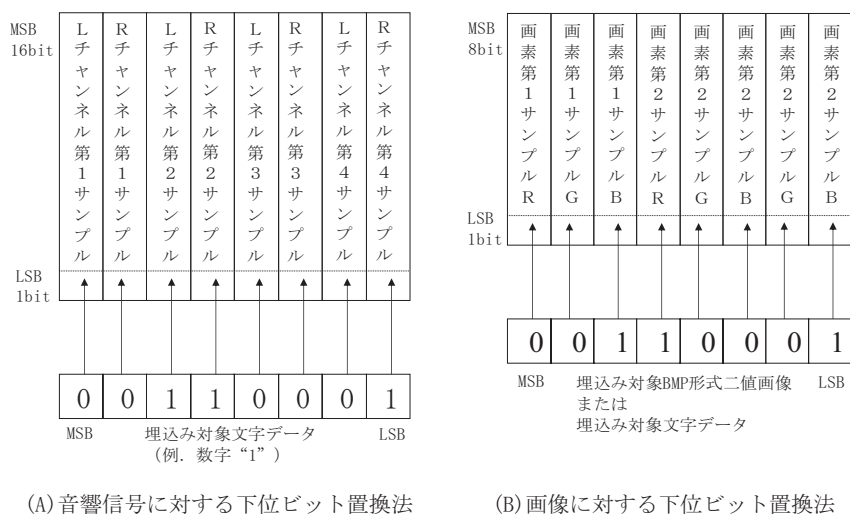


図11 音響と画像データに対する下位ビット置換による情報ハイディング手法

#### 4.4. 音響と画像データに対するエイリアシングを用いた情報ハイディング手法

4.3節で述べた情報ハイディング手法では、不可聴または不可視の状態では情報が埋め込まれているため、そのまま複製して使用してもコンテンツの視聴には支障をきたさないという問題がある。また、埋め込まれている情報を抽出するためには、情報の埋め込み方式に対応した、図9- (11A) (11B) に対応する専用の抽出ツールを必要とし、違法コピー品であることを識別することは容易では無い。そこで、情報ハイディングの別の手法として、図9- (5A) (5B) に対応するエイリアシングを用いたツールも実装している。

図12は文献13) に記載されている音響信号に対するエイリアシングを用いた情報ハイディング手法の原理図で、各音響信号を2.3節で述べた周波数次元に変換した形態で表現している。エイリアシング (折り返し雑音) とは、音響信号または画像信号に対してサンプリングを行うことによりデジタル化されるデータに重畳される雑音である。例えば、図12- (1) に示すサンプリング周波数 $F$ でデジタル化された音響信号に対して、 $1/2$ のサンプリング周波数  $F/2$  で再サンプリングを行うと、図12- (1) の右半分の高域1成分が左右反転して左半分の周波数が0から  $F/4$  の低域1成分に重畳される。そのため、アナログ信号に対してデジタル化する際は、エイリアシングを避けるため、サンプリング周波数の $1/2$ のナイキスト周波数 $F/2$ より高域成分をカットするフィルタ処理を事前に行っている。図9- (5A) (5B) に対応するツールはこの現象を逆手にとって応用したものである。

図12- (1) は図9- (1A) に対応するソース音響信号で、図12- (2) は図9- (2A) に対応する埋め込み対象の音響信号で、各々右半分の $F/4$ から $F/2$ の高域1成分および高域2成分をカットする。そして、図12- (2) の左半分の低域2成分を $F/8$ を中心に周波数軸方向に左右反転させて、図12- (1) のカットした右半分の高域1成分の領域と置換することにより、図12- (2) の音響信号の埋め込み処理を行う。その結果が図12- (3) であり、右半分の高域1成分には図12- (2) の音響信号の低域2成分が加わっているが、聴覚感度の弱い高域にずれており、更に周波数成分が左右反転して配置されているため、図12- (2) の信号とは異なる無意味なノイズとして聴取される。あらかじめ図12- (2) の音響信号のレベルを小さめに設定しておけば、図12- (1) のソース音響信号の低域1成分しか聴取されない。

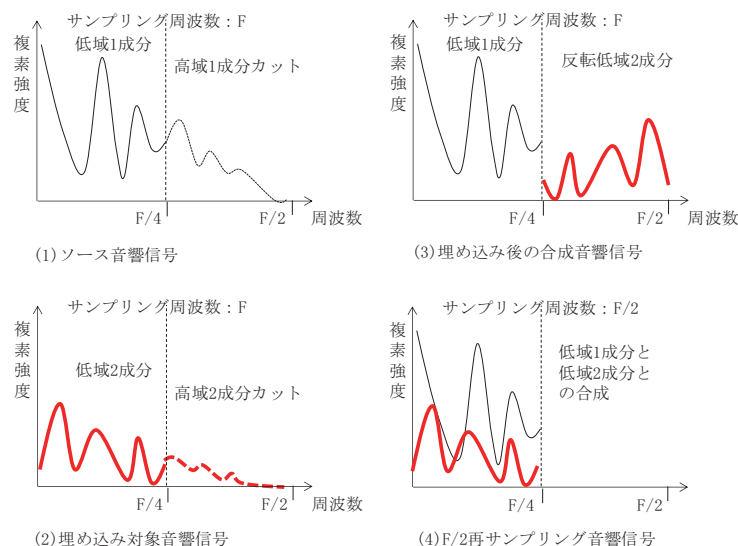


図12 音響と画像データに対するエイリアシングを用いた情報ハイディング手法

埋め込まれた音響信号を聴取可能な状態にするには、図9-（10A）の抽出ツール（関連ツール:表1-（9dA））により、図12-（3）の音響信号に対して1/2のサンプリング周波数F/2で再サンプリングを行えばよい。そうすると、図12-（4）に示されるように右半分の高域1成分の領域にある図12-（2）の反転された低域2成分が再度反転して元の状態に戻り、左半分の低域1成分の領域に重畳される。この状態は図10-（3A）のように図12-（1）のソース音響信号と図12-（2）の埋め込み対象音響信号が合成された状態になる。

画像データの場合は、エイリアシングを用いた埋め込み処理を簡便に実現できる。図9-（5B）の埋め込みツール（関連ツール:表1-（9B））では、図9-（1B）のソース画像データの縦方向および横方向の偶数番目に位置する画素に対して、図9-（2B）の埋め込み対象の画像データの対応する画素を指定の割合で合成する処理を行っている。合成する際、図9-（2B）の画像データの画素の輝度値を小さめにすれば、図9-（8B）の埋め込み後の画像データでは図9-（1B）のソース画像しか認識できない。この状態で図9-（10B）の抽出ツール（関連ツール:表1-（9dB））により、縦方向および横方向を各々1/2に間引けば、2つの画像が合成された偶数番目の画素が残り、図10-（3B）のような図9-（12B）に対応する抽出画像が得られる。

## 5. その他、実装した音響および画像の信号処理ツール

最後に、表1にリストアップされている実装済の音響および画像の信号処理ツールの中で、前章までに言及しなかったものを以下補足説明する。

### 5.1. 聴覚および視覚マスキング

隠したい秘密のメッセージを含む音響データや画像データに対して、別の音響データや画像データをマスキングデータとして同時に提示すると、秘密のメッセージが不鮮明になる現象をマスキングと呼び、視覚と聴覚の双方で発生する。この現象はヒトの感覚系に起因するもので、秘密のメッセージが物理的に消失する訳ではなく、マスキングデータに邪魔されて知覚されない、あるいは知覚されにくくなることである。画像の場合は、図13-（1）のような赤色で記載された秘密のメッセージが書かれた紙面の上に、図13-（2）のマゼンタ色のフィルムをかぶせると、図13-（3）のようにメッセージ箇所が見えにくくなる現象である。学生が参考書の赤字で書かれた重要なキーワードを覚えるのに活用されている。この時、図13-（2）のマスキングデータの色（波長）が重要で、緑色のフィルムをかぶせると全ての文字が読めてしまい、黒色のフィルムをかぶせると全ての文字が読めなくなる（関連ツール:表1-（10B）（10dB））。



図13 画像データに対するマスキング現象

同様な現象は音響でも日常的に体験することが多く、大音量の騒音により音声がかき消される現象である。図13- (1) を音声信号とすると、図13- (2) として2.5節で述べた白色雑音やピンク雑音を図13- (1) の音声信号より大きな音量で与えると、音声は聴きとりにくくなる（関連ツール:表1- (10A) ）。この現象を逆手にとって、調剤薬局のカウンターなどオープンスペースで交わされる会話音に含まれる個人情報や機密情報を聴き取りにくくする、スピーチプライバシーに活用されている<sup>18)</sup>。同様に、図13- (2) のマスクーデータの周波数（波長）が重要で、白色雑音のように全ての周波数成分をもつ音響信号が最も有効で、音楽や音声など特定の周波数成分しかもたない音響信号の場合は、音量を上げてもマスキング効果が弱い。

## 5.2. 白色雑音を用いた音響と画像データの補完

前節の聴覚と視覚の現象において、全ての周波数や波長成分をもつ白色雑音を音響データまたは画像データにかぶせると、コンテンツを認識しづらくなるが、元の音響データや画像データが消失している訳ではない。そのため、ヒトは単独で白色雑音を提示されると、背後に音響データや画像データのコンテンツが隠れているのではないかと錯覚する（関連ツール:表1- (12A) (12B) ）。音響の場合は、音脈分凝（おんみやくぶんぎょう）またはカクテルパーティ効果として知られている現象である。

図14- (1A) のスピーチ音声に対して、図14- (2A) のように等間隔に音声信号の区間が無音区間に変更された断続音声にすると、内容が聴き取りにくくなる。これに対して、図14- (3A) のように無音区間を白色雑音に置換すると、雑音の背後に削除された音声は補完されているように聴取され、内容が聴き取りやすくなる。

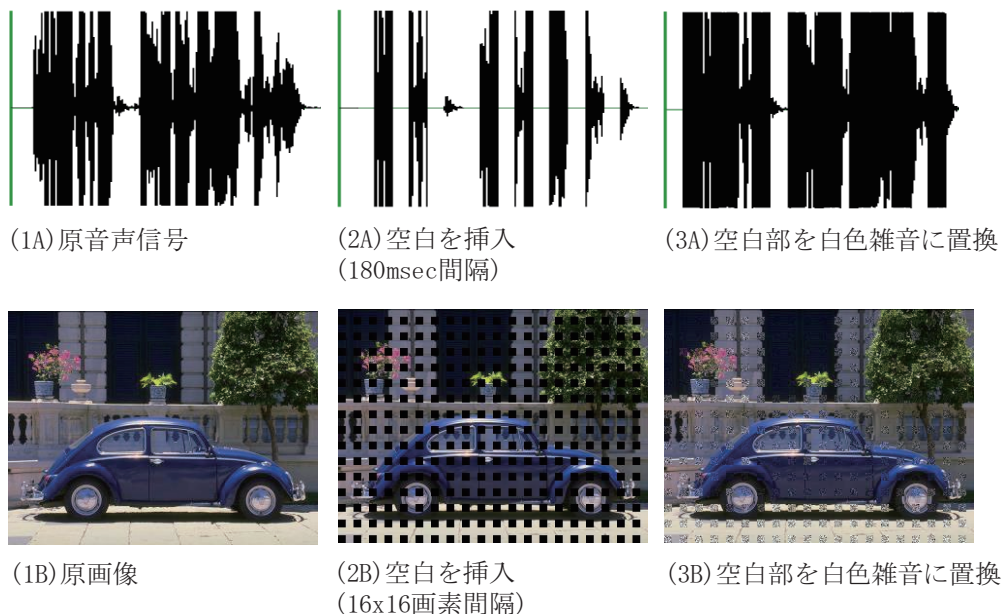


図14 白色雑音を用いた音響と画像データの補完

この現象は画像データでも発生し、図14- (1B) の画像に対して、図14- (2B) のように等間隔に画素ブロックを削除して空白を挿入すると、ジグソーパズルのように段差が目立つ不連続な絵になる。これに対して、図14- (3B) のように削除した画素ブロックを白色雑音のパターンに置換すると、白色雑音ブロックの背後に削除された画像ブロックが補完されているように見えて段差が和らぎ、自然な絵に近づく。

また、音響データに限定して、音声信号の補完ツールを応用した、スピーチ音声の話速変換ツールについても実装した（関連ツール:表1- (13A) ）。音響信号の再生速度を変換する方法として波形を伸縮させる方法が簡便だが、再生速度を上下させると音のピッチも同時に上下してしまい不自然になる。そこで、図14- (2A) に類似した方法で等間隔に音声信号を分割し（ただし、図14- (2A) のように音声区間の削除はしない）、再生速度を遅くする場合は分割した区間の間に白色雑音を挿入して波形を伸ばす。逆に再生速度を速くする場合は、分割した区間どうしをオーバーラップさせて波形を縮小する。これにより、再生速度だけを変化させることができる。

上記分断した区間は動画のフレームに対応し、本稿で扱っている静止画データに対しては適用できないが、動画であれば同様な手法を適用できる。

### 5.3. 音響のうなりと画像のモアレ

2.3節で音響データを時間1次元の周波数次元に、2.4節で画像データを空間2次元の周波数次元に変換できることを述べた。この時、時間周波数が僅かに異なる2つの音響データを重ねるとうなり（ビート）という現象が発生し、空間周波数が僅かに異なる2つの画像データを重ねるとモアレが発生する。うなり（ビート）とモアレは時間次元か、空間次元かの違いで、物理的には同一の波形現象である。

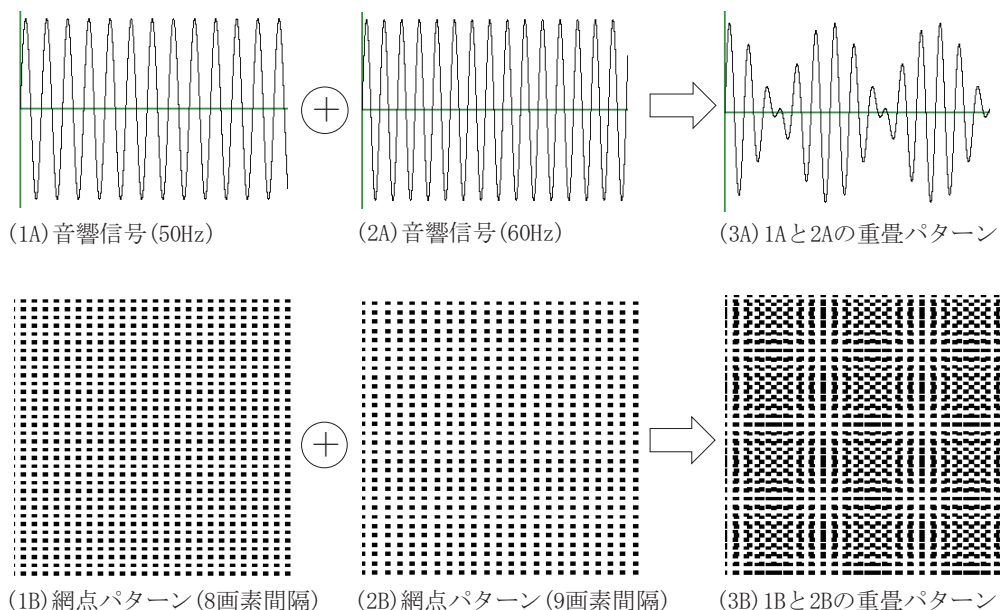


図15 音響のうなりと画像のモアレ

図15-（1A）と図15-（2A）は日本国内の2種類の商用交流電力の周波数の正弦波を生成した事例を示す（関連ツール:表1-（14A））。これらを表1-（7A）のツールを用いて合成すると、図15-（3A）のような、うなり波形が生成される。うなりの周波数は50Hzと60Hzの2種類の周波数の差分値である10Hzになり、周波数が低いため、うなり音として聴取は困難であるが、表1-（14A）のツールを用いて、例えば1桁高い2種の周波数（500Hzと600Hz）で正弦波を生成すれば、周波数100Hzのうなり音が聴取可能になる。

図15-（1B）と図15-（2B）は、表1-（14B）のツールを用いて網点周期が互いに1画素間隔だけ異なる2種類の網点パターンを生成した事例を示す。これらを表1-（7B）または表1-（10B）のツールを用いて合成すると、図15-（3B）のような、モアレパターンが生成される。

#### 5.4. 音響と画像および文字データの相互変換

音響データはCD-DAの場合、16bitsの値をもつサンプリングデータの集合で、画像データはデジタルカメラの場合、8bitsの値をもつサンプリングデータの集合であるため、いずれも2バイトのJISコードや1バイトのASCIIコードで文字データに変換することは容易である。その逆変換として、表1-（15A）のツールでは文字データをモールス符号に変換し、聴取可能なモールス音響信号に変換する機能を、表1-（15B）のツールでは文字データをQRコードの画像に変換する機能を実装している。

音響データを画像データに変換して可視化する機能としては、表1-（1A）の波形表示ツールと表1-（2A）の1次元スペクトルに変換する機能を実装している。スペクトログラムについては表1には実装していないが、文献19) 20)のように音響データをMIDI符号に変換し、ピアノロールで可視化する機能は既に開発されている。

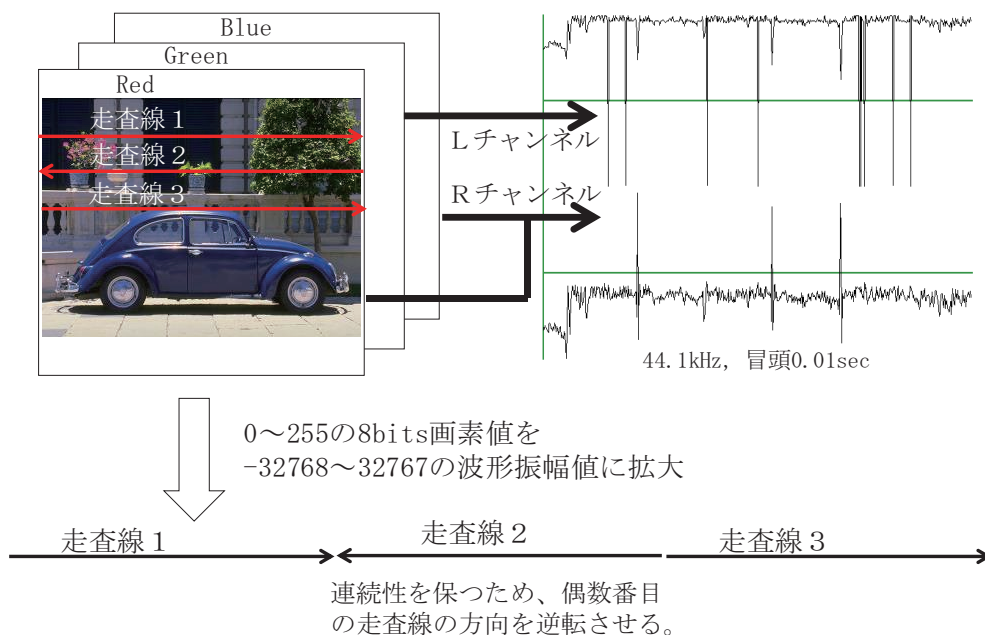


図16 画像データから音響データへの変換

逆に、画像データを音響データに変換して可聴化（ソニフィケーション）する方法も考えられ、文献21) 22) では画像データを解析して感性パラメータを抽出し、類似した感性パラメータをもつ楽曲をデータベースから検索して画像に付加する方法が提案されている。

本稿ではより直接的で簡便な方法として、図16のように、画像データの各走査線のサンプルデータをそのまま時系列に配置し、画像データの8bitsのサンプルデータを、正負符号をもつ16bitsの音響データに変換する方法を提案する（関連ツール:表1-(16B)）。図16のようにカラー画像の場合は、輝度成分に富むG成分をLチャンネルに割り当て、残りのR成分とB成分を平均化してRチャンネルに割り当てればステレオ音響信号に変換できる。また時系列に走査線を配置する際、連続性をもたせるため、図16のように偶数番目の走査線を逆に配置させれば良い。本方法は簡便だが、ソース画像が変わっても音響データに顕著な相違が生じないという問題がある。

## 6. おわりに

本稿では、これまで尚美学園大学・情報表現学科の授業として担当してきた「旧マルチフィールド体験演習」に活用しながら逐次開発と改良を進めてきた、音響と画像の信号処理のアプリケーションソフトについて紹介した。音響と画像の信号処理の基本と、C言語を用いた音響と画像のプログラミングをWindowsパソコンでクロスオーバー演習できることを目指していたが、偶然にも担当している授業の名称が本年度より「クロスオーバー学習」に改称され、妙な因縁を感じている。

表1において、Windowsパソコンで稼働するC言語で記述された音響と画像の各々15本のアプリケーションソフトを紹介したが、現段階で、音響に固有な機能は4.2節で述べたアクティブノイズコントロール（サンプル値が正負符号をもつため）しかなく、一方画像に固有な機能は2.2節で述べた横方向と縦方向のサイズを変える変倍処理（2次元配列のため）しか見当たらず、音響と画像のほぼクロスオーバー関係を実現できている。ただし現状では、実装途上のソフトがあり、表1-(16B)の画像データの可聴化ツールのように改良が必要なソフトが存在するので、今後実装が進展した段階で改めて報告したい。

これまで長期に渡り、本稿で紹介した信号処理ツールの開発と改良を進めることができたのも、本学の学生や教職員の皆様とのインタラクションによるものと考えており、改めて本学関係者の皆様に謝意を示す。本稿で紹介した音響と画像の信号処理ツールは開発途上であるが、教育・研究・その他にご活用されたい場合はC言語ソースコードを含め提供可能ですので、ご連絡ください（E-mail: modegi@bekkoame.ne.jp）。

## 引用文献

- 1) 永野宏治『信号処理とフーリエ変換』、朝倉書店、Jan.2014.
- 2) 藤原洋編『マルチメディア情報圧縮』、インターネット時代の数学シリーズ5、共立出版、Mar.2000.
- 3) 奥村晴彦「データ圧縮の基礎から応用まで」『Cマガジン』、Vol.154, No.7,ソフトバンクSep. 2002, pp.13-35.



- 4) Khalid Sayood, *Lossless Compression Handbook*, Academic Press, Elsevier Science USA, 2003.
- 5) 山崎敏、奥村晴彦監修『LHAとZIP 圧縮アルゴリズム×プログラミング入門』, ソフトバンクパブリッシング、Dec.2003.
- 6) 松井甲子・岩切宗利『情報ハイディングの基礎-ユビキタス社会の情報セキュリティ技術-』, 森北出版、Jun.2004.
- 7) Kazuhiro Kondo (Editor) , *Multimedia Information Hiding Technologies and Methodologies for Controlling Data*, IGI Global, Aug.2012.
- 8) 土井滋貴「動画処理とWindowsプログラミング」『TRY!PC』,2003 Spring, CQ出版社、Apr.2003.
- 9) 土井滋貴『はじめての動画処理プログラミング』CQ出版社、Aug.2007.
- 10) 田辺義和『Windowsサウンドプログラミング』、翔泳社、Apr.2001.
- 11) 北山洋幸『WAVプログラミング C言語で学ぶ音響処理』、カッパシステム、Sep.2008.
- 12) 茂出木敏雄「聴覚芸術への情報学的アプローチと音楽情報処理ツールの開発事例」『尚美学園大学紀要「芸術情報研究」』, Vol.18, Nov.2010, pp.15-35.
- 13) 茂出木敏雄「音楽に著作権保護機能を埋め込む音響信号に対するハイディング技術に関する研究」『尚美学園大学紀要「芸術情報研究」』, Vol.22, Mar.2013, pp.61-82.
- 14) 林伸二「非理科系学生のための音響信号処理」『尚美学園大学紀要「芸術情報研究」』, Vol.26, Mar.2017, pp.39-50.
- 15) 谷口慶治・長谷博行編『画像処理工学 応用事例編』、共立出版、May.2005.
- 16) 杉山賢二『実践 映像信号処理 - C言語を使って体感する-』、コロナ社、Mar.2008.
- 17) 井上誠喜・八木伸行・林 正樹・中須英輔・三谷公二・奥井誠人『C言語で学ぶ実践画像処理 Windows/X-Window対応』オーム社、Oct.2008.
- 18) 茂出木敏雄「スピーチプライバシー保護機能ポストラサウンドパネル」『騒音制御』、Vol.37, No.5、騒音制御工学会、Oct.2013, pp.264-269.
- 19) 茂出木敏雄「オーディオ-MIDI符号化ツール「オート符」における表情付け解析機能の実装」『尚美学園大学紀要「芸術情報研究」』, Vol.20, Nov.2011, pp.17-34.
- 20) 茂出木敏雄「音響信号のMIDI 符号化ツール「オート符」のWindows10対応に伴う改修」『尚美学園大学紀要「芸術情報研究」』, Vol.26, Mar.2017, pp.85-104.
- 21) 茂出木敏雄「映像コンテンツ解析によるBGMサウンドトラックの自動生成」『電気学会 電子情報システム部門誌』、Vol.125-C, No.7、オーム社、Sep.2005, pp.1004-1010.
- 22) 岩宮眞一郎編、日本音響学会編『視聴覚融合の科学』, 音響サイエンスシリーズ 11, コロナ社, Nov.2011.