Media Science

2013 年度　卒　業　論　文

音声入力による
効果音検索手法に関する研究

指導教員：渡辺 大地 講師
三上 浩司 准教授

メディア学部 ゲームサイエンスプロジェクト
学籍番号　M0110453
山田 龍明

2013 年度　　卒　業　論　文　概　要

論文題目

音声入力による
効果音検索手法に関する研究

メディア学部 学籍番号：M0110453	氏名	山田 龍明	指導教員	渡辺 大地 講師 三上 浩司 准教授

キーワード	音声、検索、高速フーリエ変換、 相関係数、自己相関関数

ゲームや映画のようなマルチメディアコンテンツにおいて、効果音は作品を演出していく上で必要不可欠なものである。しかしながら、近年のインターネット上には大量の効果音ファイルが存在し、想定した効果音を探すのには時間がかかる。既存の検索ツールには、予め付加しておいたキーワードなどのメタデータと照合して検索結果を出すものが多いが、効果音を追加する毎にキーワードを付加する手間がかかる。また、楽曲検索では入力した音声からリズムや歌詞などの特徴を抽出してマッチングするものがあるが、効果音のような明確なリズムや歌詞がなく再生時間が短いデータに適応させるのは難しい。

本研究では、より想定した音に近い効果音を効率良く見つけるために、音声入力を用いた効果音検索手法を提案する。本手法は、入力した音声と被検索対象の効果音の類似度を算出するために、発音時間、音高、音量といった物理量に着目した。このとき、音信号の前後に無音の部分があれば削除し、音高は自己相関関数を用いて求め、音量は波形の振幅値をもとにして求めた。

本手法を用いた検索ツールを制作し、効果音を種類別に分類したフォルダから手作業で探す方法との比較実験を行った。検証として 20 人の被験者に対し、指定した効果音を見つけ出すまでの時間を計測する実験と、どれくらい想定した音に近い効果音を検索できるかの実験を行った。実験の結果、本手法はユーザーの声域を考慮しておらず、音声入力にはある程度の慣れが必要という問題点も残ったが、どのフォルダに分類しているか想像しづらい音に関しては本手法の方が早く検索することができ、有効であると分かった。また、ある程度想定した音に近い効果音を見つけやすいという有効性も示した。

目 次

図 目 次

表 目 次

第 1 章

はじめに

1.1　研究の背景と目的

近年、ゲームや動画といったコンテンツの制作は個人規模でも盛んに行われている。これらのコンテンツにおいて、効果音は作品を演出していく上で必要不可欠なものである。制作者が欲しい効果音を得る方法としては、次の 2 つに大別できる。第一に、自ら録音を行う方法があり、第二に、インターネット上や CD（コンパクトディスク）などで提供されている効果音集を利用して探す方法がある。しかしながら、自ら録音を行う方法は適切な録音環境、録音機材、録音技術が必要であるため、専門の知識を持たないユーザーには困難な方法である。そこで、提供されている効果音集の中から欲しい効果音を探すことがある。このとき、効果音集に収められている効果音の数が多いほど、想定した音に近い効果音が存在する確率は高くなるが、効果音ファイルの題名や説明文などを手がかりとしても、一つ一つ音を聴きながら探さなければならないため非常に時間がかかる。それを踏まえ本研究は、想定した音に近い効果音を効率的に検索する手法構築を目的とする。

効果音の整理や検索を行うためのツール開発や研究が行われている。効果音を検索する際に用いる入力情報は、大きく分けて文字と音声である。まず、入力情報を文字とした場合の検索について特徴と問題点を述べる。文字で検索を行う既存の検索ツールは複数提供されている [1][2] 。その一つに、Crypton Future Media 社が開発した MUTANT [3] がある。MUTANT は音楽ファイルの効率的な一元管

理のためのソフトウェアであり、音楽ファイルをデータベース化し、整理、分類、検索が可能である。図 1.1 は、MUTANT において、予め付加しておいたキーワードと入力した文字を照合して検索結果を表示した例である。

図 1.1: キーワードと入力した文字を照合して検索結果を表示した例

MUTANT では標本化周波数、量子化ビット数などの音情報や波形を一覧できるため、聴覚的だけでなく視覚的にも音ファイルを比較することができる。また、一度探し出した音ファイルを見失ってしまわないように、後で使いたい音ファイルをブックマークしたりメモ書きを残したりすることができる。しかしながら、管

理する効果音を増やす度にキーワードを付加する手間がかかり、文字列では表現しにくい音の場合適切なキーワードを付加することが難しい。また例えば同じ爆発音でも、検索するユーザーによって「ドカーン」と表現したり「バゴーン」と表現したりするように、擬音語表現のニュアンスが変わることもあるため、キーワードに差異が生じるという問題点がある。

音声入力を用いた楽曲検索では、いくつかのサイトやアプリが提供されている [4][5]。その一つに、SoundHound 社の midomi [6] がある。midomi は楽曲検索サイトであり、楽曲の一部を歌った音声や鼻歌を入力することで該当する楽曲の候補を検索する。このとき、入力した音声から歌詞、リズム、メロディなどの複数の特徴を抽出し、各楽曲の中から似ている部分を探すためにマッチングを行っていく。そのため、音声入力を用いた楽曲検索では歌詞や明確なリズムを持っていない音には対応できない。また、10 秒程度の音声を入力しなければ高い精度の検索結果を得るだけの特徴を抽出することができない。そのため、効果音のような再生時間が極めて短い音信号に対しては特徴のマッチングが難しい。

音を必要とするコンテンツの制作において、音の発注は「明るい感じ」や「早いリズム」といったような抽象的な言葉ですることが多く、音の制作の大部分がクリエイターの感性やスキルに依存している面が強い。和氣ら [7] の研究では、人間は頭の中で想定した音をどのように表現するのかを明らかにする実験を行った。その結果、人間は頭の中で想定した音を、波形の説明、音源の説明、主観の説明によって表現するということを解明した。波形の説明とは、音の聞こえ方そのものを表現しようとするものであり、音の高さ（以下、音高）や音の長さといった物理量の説明と擬音語に分けられる。この実験結果をもとに和氣ら [8] はキーワード入力による効果音検索ツールを制作し、検索ツールが合成音のような音源を特定できない音などに関して有効であると証明した。ただし、全ての効果音にキーワードを付加するため、データベースの構築に多大な作業を要するなどの問題点もあった。効果音にキーワードを付加するとき、擬音語や音源の説明などはキーワードとして付加しやすいが、楽曲や効果音の音高は常に一定というわけではな

く時間経過とともに変化するのが一般的であるため、音高をキーワードとして付加することは難しい。

これまでに述べた既存の検索手法の問題点をまとめると次のようになる。

- キーワード付加には手間がかかり、人によってはキーワードに差異が生じる。
- 楽曲検索で用いるような特徴マッチングは、短い音信号で行うのは難しい。
- 常に変化する音高や音量をキーワードとして付加することは難しい。

これらの問題点から、誰でも手軽に、想定した音に近い効果音を検索できる手法が必要であると考えた。そこで、想定している音を直接的に検索結果として反映するために、音声入力に着目した。音声の入力には特別な機材や技術を必要としないため、ユーザーにとって容易な検索手段であると言える。また、キーワードなどを付加することなく音信号そのものから検索結果を算出することにより、ファイル管理の手間を省きつつ、微妙な音のニュアンスや音高なども検索結果に反映させることができると考えた。

以上を踏まえ、短い音信号でも有効な検索を行うことと、キーワードを付加する手間を省くことの両立を研究意義とする。本研究では、音声入力を用いて、発音時間、音高、音量といった物理量の観点から音信号を分析し、似ている効果音を検索する手法を提案する。

1.2　本論文の構成

本論文の構成は以下の通りである。第 2 章では、本研究が行う提案手法について述べる。第 3 章では、制作したツールの概要と検証について述べる。第 4 章では、本論文のまとめと今後の展望を述べる。

第 2 章

音声入力を用いた効果音の検索手法

本研究では、検索したい効果音を真似て発声したものをイメージ音声と呼ぶことにする。まず、ユーザーがイメージ音声を録音する。その音声ファイルを、本手法を用いて制作したツールで読み込み、音信号から被検索対象の各効果音との類似度を算出する。その後、類似度の高い効果音から順番に検索結果に表示する。

制作したツールは、録音したイメージ音声と被検索対象の各効果音を読み込み、時間、音高、音量の各観点から解析を行う。2.1 節では発音時間の差を算出する。これにより、例えば「バン」という短めの破裂音を探したいとき、「バーン」という長めの破裂音と区別することができる。2.2 節では高い音や低い音を区別するために音高を解析し、その差を算出する。2.3 節では音量の変化を解析し、類似度を求める。その結果、例えば音量が徐々に大きくなっていく効果音や小さくなっていく効果音を区別することができる。2.4 節では解析を行った発音時間、音高、音量から総合類似度を求める。

イメージ音声の録音はマイクを用いて行う。このとき、音声の音量が大きすぎると録音可能な信号の許容値をオーバーしてクリップノイズというノイズが発生し、正確な解析が行えなくなるため、音声の音量には注意が必要である。

本研究では PCM という方式の音ファイルを用いる。そのフォーマットを表 2.1 に示す。

表 2.1: PCM に関するフォーマット

標本化周波数	44.1kHz
量子化ビット数	16bit
チャンネル	モノラル
ビットレート	705.6kbpm

PCM とは、音声などのアナログ信号をデジタルデータに変換する方式の 1 つで、信号を一定時間毎に標本化し、定めたビット数の整数値に量子化して記録する。こうして記録されたデジタルデータの品質は標本化周波数と、量子化ビット数で決まる。標本化周波数は 1 秒間に何回数値化するかを表し、量子化ビット数はデータを何ビットの数値で表現するかを表す。

以降の節で述べる各解析では、イメージ音声と被検索対象の各効果音を上記の表 2.1 で示したフォーマットで標本化した数値データを用いる。量子化ビット数は 16bit であるため、PCM で表現する数値データは-32768 から 32767 の値をとるが、これを正規化し、-1 が最小、1 が最大となるようにした。また、イメージ音声の録音は一つのマイクで行うため、チャンネルはモノラルとした。なお、ビットレートは 1 秒間に処理するデータのビット数を表す。

2.1 時間解析

本節では、イメージ音声と被検索対象の各効果音との発音時間の類似度を求める。まず、音信号の前後に無音の部分が含まれていた場合、無音の部分を削除する。音を発している箇所では波形の振幅の絶対値が大きくなるため、振幅の絶対値に対して閾値を設け、その閾値を下回る部分を無音とし、削除した後の音信号の長さを発音時間とした。図 2.1 は閾値を 0.1 としたときの音信号の発音時間と、削除する部分を表したものである。なお、ここでは数値データの標本数は実際より少ない簡略化した図を載せており、標本点を繋いだ線が見かけの波形を表している。以降の解析では全て、前後の無音部分を削除したこの音信号を用いる。

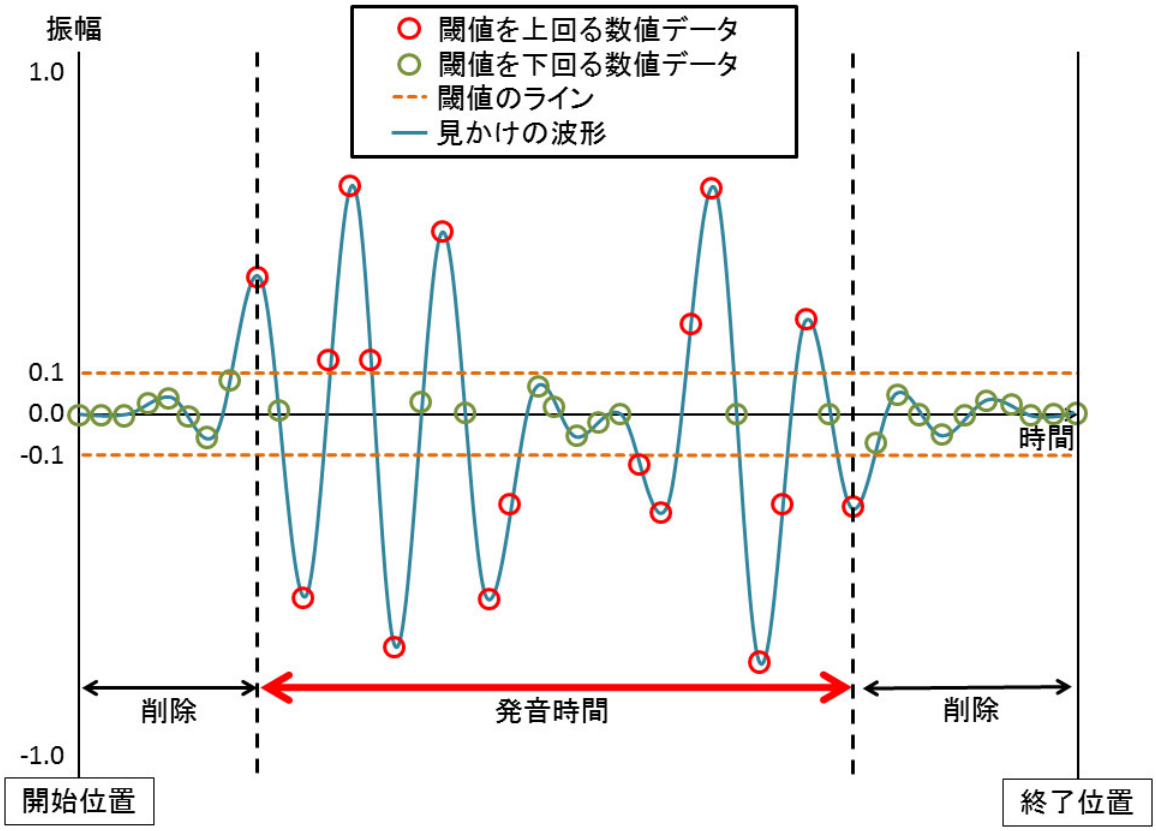


図 2.1: 音信号の発音時間

イメージ音声の標本化した数値データの個数を a、ある被検索対象の効果音の標本化した数値データの個数を b、標本化周波数を f_s としたとき、発音時間の差 ΔT を式 (2.1) に示す。

$$\Delta T = \left| \frac{a-b}{f_s} \right| \tag{2.1}$$

次に、発音時間の差の閾値を η としたとき、イメージ音声と、ある被検索対象の効果音との発音時間の類似度 α を式 (2.2) に示す。

$$\alpha = \begin{cases} 1 - \frac{\Delta T}{\eta} & (\Delta T < \eta) \\ 0 & (otherwise) \end{cases} \tag{2.2}$$

発音時間の差は、検索する音信号に応じて際限なく変化する。そのため、2.4 節で述べる音高と音量の類似度も含めた最終的な類似度を算出するとき、仮に音高

と音量の類似度が最大値だった場合でも発音時間の差が大きければ類似度を著しく下げてしまうことが考えられる。このことから、発音時間の差の閾値 η を設定し、本研究では 3 秒とした。

2.2 音高解析

本節では、イメージ音声と被検索対象の各効果音との音高の類似度を求める。音高は基本周波数によって決まる。音信号は一般的に、複数の周波数成分の合成から成り立っており、その中で最も低い周波数成分のことを基本周波数と言う。基本周波数を抽出する手法は様々な研究があり、その多くは時間領域での処理 [9][10][11][12]、周波数領域での処理 [13][14][15] 、またはその両方での処理 [16][17] に大別できる。本研究では、時間領域での高速な処理が可能でありながら、音声入力に伴う雑音や位相の変化に強い自己相関関数を用いて基本周波数を求める手法 [9] を採用した。自己相関関数とは、時間的に離れた 2 点の関係の強さを表した関数であり、ウィーナー＝ヒンチンの定理 [18] によると、音信号を高速フーリエ変換して得たパワースペクトルを、逆高速フーリエ変換することによって高速に得ることができる。パワースペクトルとは、信号が周波数毎に含んでいるエネルギーを表現したもので、各周波数毎の絶対値を 2 乗した値の分布のことある。また、フーリエ変換とは、信号の中にどの周波数成分がどれだけ含まれているかを抽出する処理であり、高速フーリエ変換は離散フーリエ変換を、逆高速フーリエ変換は逆離散フーリエ変換をそれぞれ高速に計算するアルゴリズムである。離散フーリエ変換を式 (2.3) に、逆離散フーリエ変換を式 (2.4) に示す。

$$H(p) = \sum_{q=0}^{N-1} h(q) e^{-i\frac{2\pi pq}{N}} \qquad (p = 0, 1, 2, ..., N-1) \tag{2.3}$$

$$h(q) = \frac{1}{N} \sum_{p=0}^{N-1} H(p) e^{i\frac{2\pi pq}{N}} \qquad (q = 0, 1, 2, ..., N-1) \tag{2.4}$$

ここで、$H(p)$ は周波数軸上の値、$h(q)$ は時間軸上の音信号、N は標本数を表す。本研究では、標本数 N を 2048 とした。

音高は時間経過とともに変化することが一般的であるため、一定時間区切りで音高を求める。この区切り時間 t を 式 (2.5) に示す。

$$t = \frac{N}{f_s} \tag{2.5}$$

なお、N は標本数、f_s は標本化周波数を表している。本研究では標本数 N を 2048、標本化周波数 f_s を 44.1kHz としたため、区切り時間 t は約 0.04644 秒となる。

フーリエ変換は周期性が成り立っている波形でなければ、本来含まれている周波数以外に余分な周波数を抽出するため、音高を決めるための基本周波数が正しく抽出できなくなってしまう。一般的に、一定時間毎に区切った波形は区間の両端の値が一致せず、不連続性が生じるため、周期性が成り立つ可能性は限りなく低い。そこで、この不連続性をできるだけ目立たなくするために窓関数 [19][20] を用いた。窓関数にはいくつかの種類があるが、本研究では、周波数の測定能力に優れているハミング窓関数を用いた。ハミング窓関数 $W(q)$ を式 (2.6) に示す。

$$W(q) = 0.54 - 0.46 \cos \frac{2\pi q}{N} \tag{2.6}$$

窓関数適応前の音信号 $h(q)$ にハミング窓関数を適応した音信号 $h'(q)$ を式 (2.7) に示す。

$$h'(q) = h(q)W(q) \tag{2.7}$$

ここで得た $h'(q)$ を式 (2.3) の $h(q)$ に当てはめることにより、フーリエ変換時に生じる不連続性を抑制する。図 2.2 は、窓関数適応前の波形にハミング窓関数を適応する例を表している。

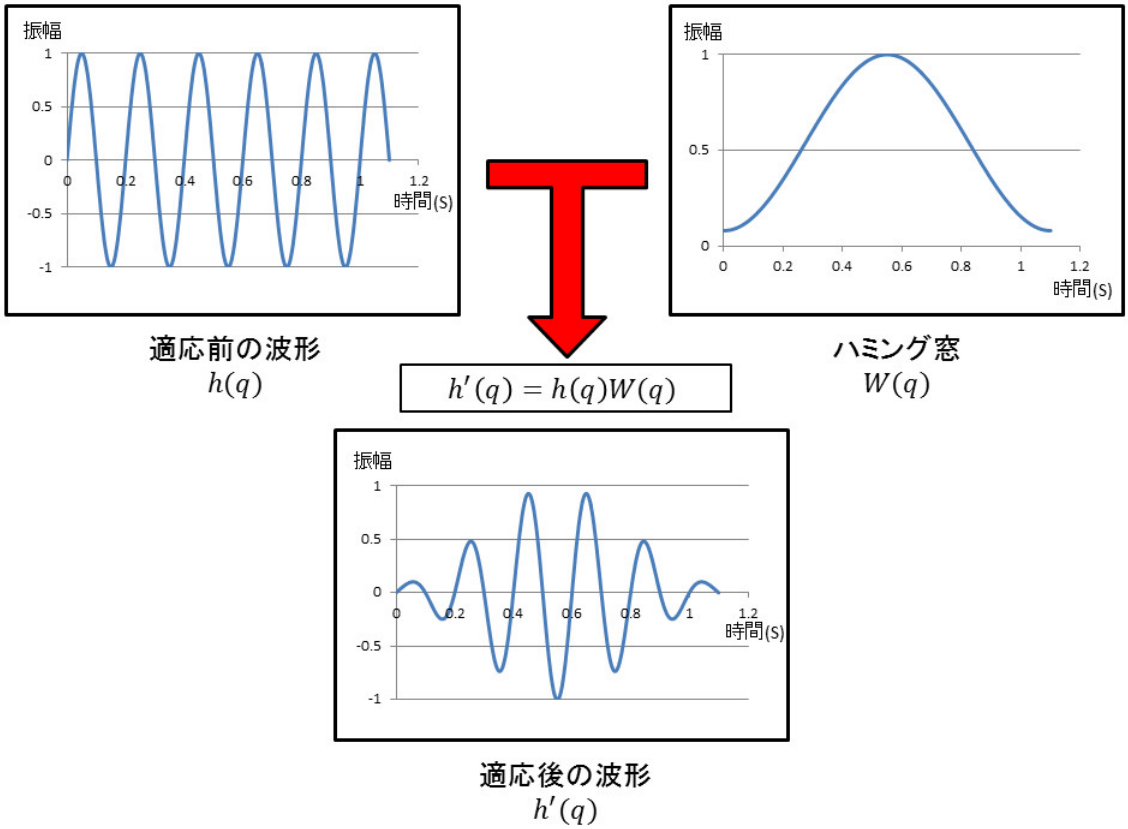


図 2.2: ハミング窓関数の適応例

次に、式 (2.4) の $h(q)$ において最大のピークとなる時の q の値を求める。自己相関関数の性質上、$q = 0$ の時に $h(q)$ は最大値となる。しかしながら、これは同じ音信号同士の相関を表す量であるため、ピークとしては扱わない。よって、q を 0 から 1 ずつ増やしていき、$h(q)$ が最初に 0 以下になったときの q 以降の最大値を最大のピークとする。

$h(q)$ が最大のピークを迎えた時の q を q' とし、標本化周波数を f_s としたとき、基本周波数 f_0 を式 (2.8) に示す。

$$f_0 = \frac{f_s}{q'} \tag{2.8}$$

図 2.3 は基本周波数の算出を表したものである。

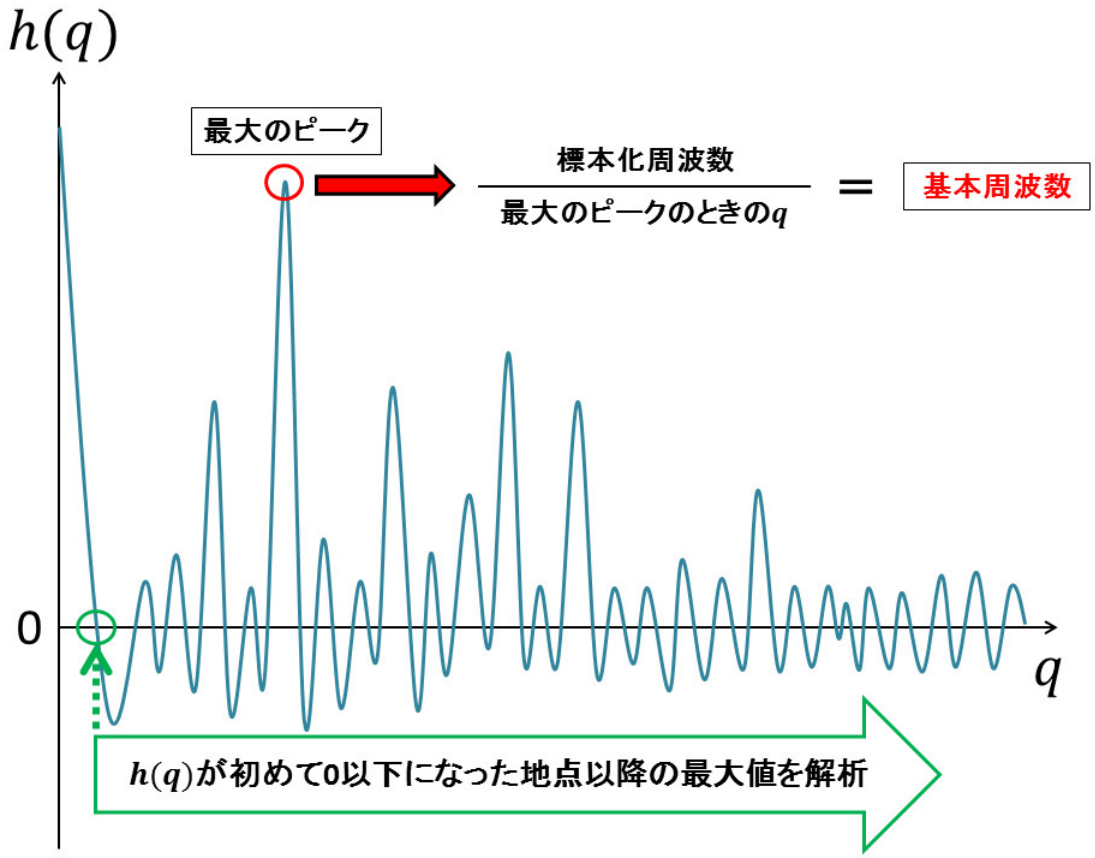


図 2.3: 基本周波数の算出

次に、音程を求めるために基本周波数から最も近い音名を推定する。音程とは、2 つの音高の間隔のことである。音名は周波数に応じて順番に並べることが出来るため、最終的には音名がいくつ離れているかを求めることで音程を算出する。音名は 440Hz(A4) を基準値として、1 オクターブを 12 等分する一般的な十二平均律を用いて求める。音名と十二平均律の対応を表 2.2 に示す。なお、推定範囲は A0 から G#10 までとし、最も周波数の低い A0 から順番に番号をつけた。また、周波数は小数点第三位を四捨五入して表記している。

表 2.2: 十二平均律を用いた音名と周波数

番号	音名	周波数 (Hz)
1	A0	27.50
2	A#0	29.14
3	B0	30.87
⋯	⋯	⋯
48	G#4	415.31
49	A4	440.00
50	A#4	466.16
⋯	⋯	⋯
118	F#10	23679.64
119	G10	25087.71
120	G#10	26579.50

イメージ音声と、ある被検索対象の効果音において、音信号を式 (2.5) で求めた区切り時間 t 毎に区切る。その区間数がより少ない方の区間数を m としたとき、一区間あたりの音程の相加平均 $\overline{A}$ を式 (2.9) に示す。

$$\overline{A} = \frac{1}{m}\sum_{j=1}^{m}(|u_j - v_j|) \tag{2.9}$$

なお、u_j はイメージ音声の j 番目の区間における音名の番号を表し、v_j は被検索対象の効果音の j 番目の区間における音名の番号を表している。

次に、音程の相加平均 $\overline{A}$ の閾値を τ としたとき、イメージ音声と、ある被検索対象の効果音との音高の類似度 β を式 (2.10) に示す。

$$\beta = \begin{cases} 1 - \frac{\overline{A}}{\tau} & (\overline{A} < \tau) \\ 0 & (otherwise) \end{cases} \tag{2.10}$$

式 (2.2) に示した発音時間の類似度の算出と同様に、音程が広すぎると最終的な類似度を著しく下げてしまうことが考えられるため、音程の相加平均 $\overline{A}$ の閾値 τ を設定し、本研究では 50 とした。

2.3 音量解析

本節では、イメージ音声と被検索対象の各効果音との音量の類似度を求める。まず、音量の相関係数を求める。相関係数とは、2 組の数値からなるデータ列の間の類似性の度合いを示す統計学的指標である。-1 から 1 の間の実数値をとり、1 に近いほど 2 組のデータ列には正の相関があるといい、-1 に近いほど負の相関があるという。したがって、求めた相関係数が 1 に近いほど類似性が高いと言える。

音信号を区切り時間 t で区切り、その区間の最大音量をデータ列に保存する。このとき、音量は波形の振幅の絶対値を取り算出した。イメージ音声と、ある被検索対象の効果音の音量データ列 $\{(x_k, y_k)\}(k = 1, 2, 3, ..., m)$ があるとき、相関係数 r を式 (2.11) に示す。

$$r = \frac{\sum_{k=1}^{m}(x_k - \overline{x})(y_k - \overline{y})}{\sqrt{\sum_{k=1}^{m}(x_k - \overline{x})^2}\sqrt{\sum_{k=1}^{m}(y_k - \overline{y})^2}} \tag{2.11}$$

なお、$\overline{x}, \overline{y}$ はそれぞれ $x = \{x_k\}, y = \{y_k\}$ の相加平均を表す。

次に、イメージ音声と、ある被検索対象の効果音との音量の類似度 γ を式 (2.12) に示す。

$$\gamma = \frac{r + 1}{2} \tag{2.12}$$

r が取り得る値は、-1 から 1 の間の実数値ということが明確であるため、r が-1 のとき γ が 0、r が 1 のとき γ が 1 となるように計算した。

2.4 解析結果

本節では、これまでに解析を行った発音時間、音高、音量の類似度から総合類似度を求める。イメージ音声と、ある被検索対象の効果音との総合類似度 Z を、式 (2.2) に示した発音時間の類似度 α と、式 (2.10) に示した音高の類似度 β と、式

(2.12) に示した音量の類似度 γ を用いて、式 (2.13) に示す。

$$Z = \frac{1}{3}(\alpha + \beta + \gamma) \tag{2.13}$$

検索結果としては、全ての被検索対象の効果音から総合類似度 Z を求め、その値が高い効果音から順番に上から表示する。

第 3 章

検証と考察

本章では、第 2 章で述べた手法を用いて制作した検索ツールで実験を行う。その後、実験結果をもとに考察を述べる。

3.1 実験目的と実験方法

実験の目的は次の 2 点を検証することである。

1. 検索にかかる時間を短縮できるかどうか。

2. ユーザーが想定した音に近い効果音を検索できるかどうか。

上記の 2 点をそれぞれ検証するために、被験者には次の 2 つのタスクを課す。

A. 指定の効果音を探す。

B. 想定した効果音を探す。

それぞれのタスクにおいて、効果音を種類別に分類したフォルダを用いて手作業で探す方法と、本研究で制作した検索ツールを用いて効果音を探す方法とで比較実験を行う。被検索対象の効果音は以下の 2 つのサイトで配布しているものを使用し、配布元で記載されている音の説明にしたがって種類別にフォルダの分類を行った。

- 魔王魂 (http://maoudamashii.jokersounds.com/)
- On-jin ～音人～(http://on-jin.com/)

使用したファイルの個数は合計 2062 個である。また、音声の録音は「♪超録 - パソコン長時間録音機」[21] を用いて行う。

タスク A では、被験者に指定した効果音を聴いてもらい、その効果音を見つけ出すまでの時間を計測し、検索にかかる時間を短縮できるかどうかを検証する。タスク B では、被験者に指定したシチュエーションで鳴る効果音を自由に想定してもらい、その想定した音に近い効果音を探してもらう。このとき、視聴する音ファイルの数を 30 個に制限し、その中で最も想定した音に近かった効果音に対してどれくらい想定した音に近かったかを 5 段階で評価する。この評価値を効果音の満足度とし、満足度を高めることができるかどうかを検証する。

図 3.1 は、制作した検索ツールで効果音の検索結果を表示した例である。

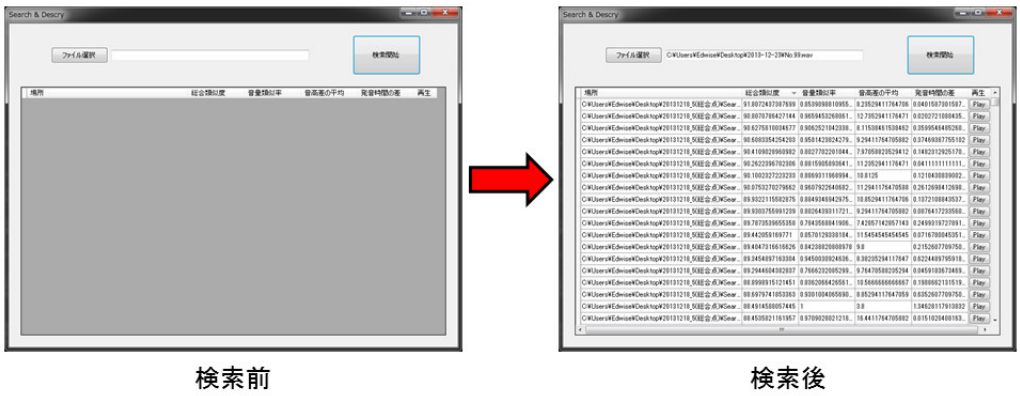


図 3.1: 制作した検索ツールで効果音の検索結果を表示した例

式 (2.4) で算出する総合類似度が高い効果音から順に上から表示し、発音時間、音高、音量での解析結果の値も表示している。また、再生ボタンをクリックすると選択した効果音を視聴することができる。

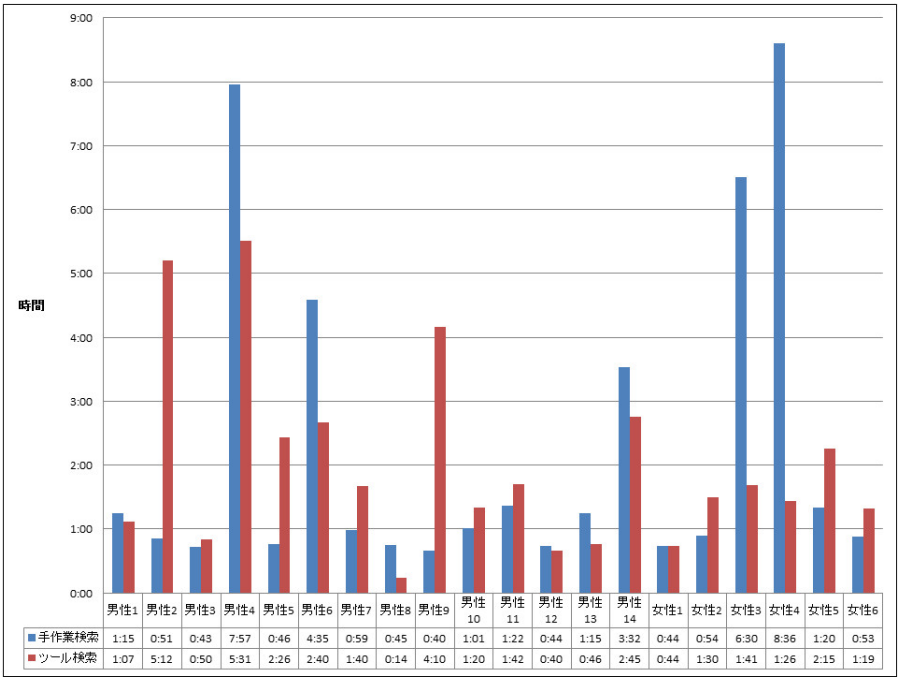
実験時の環境は表 3.1 のとおりである。

表 3.1: 実験環境

OS	Windows 7 Professional 64bit
CPU	Intel(R) Core(TM) i7 CPU M 640 @ 2.80GHz
メモリ	6.00GB
マイク	ECM-PCV80U

3.2 実験結果

本節では、今回行った実験の結果を述べる。実験は被験者 20 名（男性 14 人、女性 6 人）に対し行った。タスク A では手作業での検索とツールでの検索を効果音の種類を変えて 2 回ずつ行い、指定した効果音を見つけ出すまでの平均時間を求めた。効果音 I の実験結果を図 3.2 に、効果音 II の実験結果を図 3.3 に、それぞれの効果音を見つけ出すまでの平均時間を表 3.2 に示す。効果音 II のツール検索においては、全体的に男性の方が早く見つけ出す傾向が見られた。



	男性1	男性2	男性3	男性4	男性5	男性6	男性7	男性8	男性9	男性10	男性11	男性12	男性13	男性14	女性1	女性2	女性3	女性4	女性5	女性6
手作業検索	1:15	0:51	0:43	7:57	0:46	4:35	0:59	0:45	0:40	1:01	1:22	0:44	1:15	3:32	0:44	0:54	6:30	8:36	1:20	0:53
ツール検索	1:07	5:12	0:50	5:31	2:26	2:40	1:40	0:14	4:10	1:20	1:42	0:40	0:46	2:45	0:44	1:30	1:41	1:26	2:15	1:19

図 3.2: 効果音 I を見つけ出すまでの時間

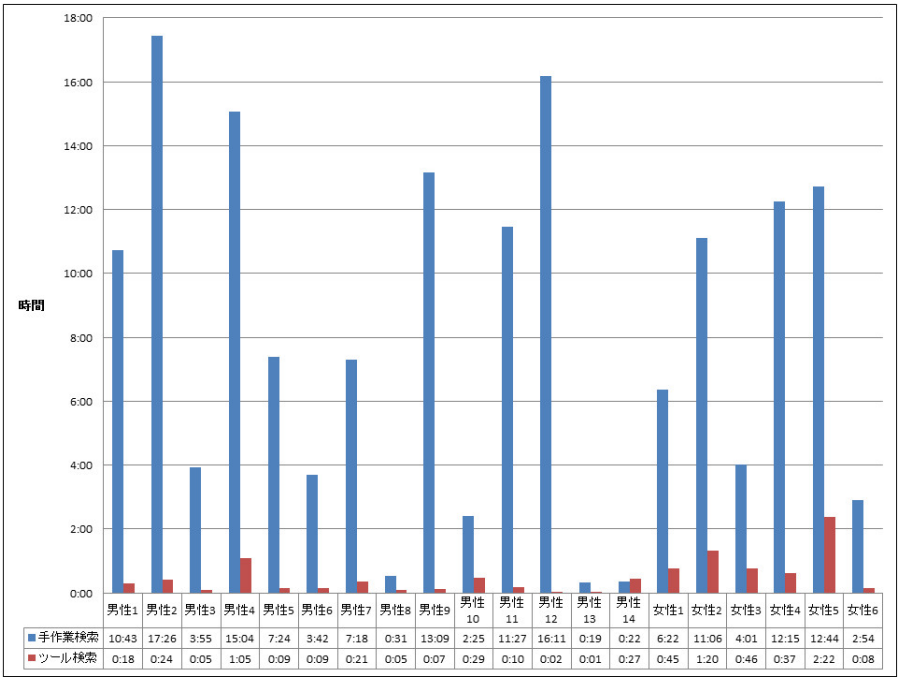


図 3.3: 効果音 II を見つけ出すまでの時間

表 3.2: 指定した効果音を見つけ出すまでの平均時間

	手作業検索	ツール検索
効果音 I	2 分 16 秒	1 分 59 秒
効果音 II	7 分 57 秒	0 分 29 秒
全体	5 分 7 秒	1 分 14 秒

タスク B において、本研究では「ゲームのタイトル画面でスタートボタンを押した時」というシチュエーションを想定して実験を行った。そして、選択した効果音に対しての平均満足度を求めた。実験結果を図 3.4 に、選択した効果音の満足度の平均値を表 3.3 に示す。

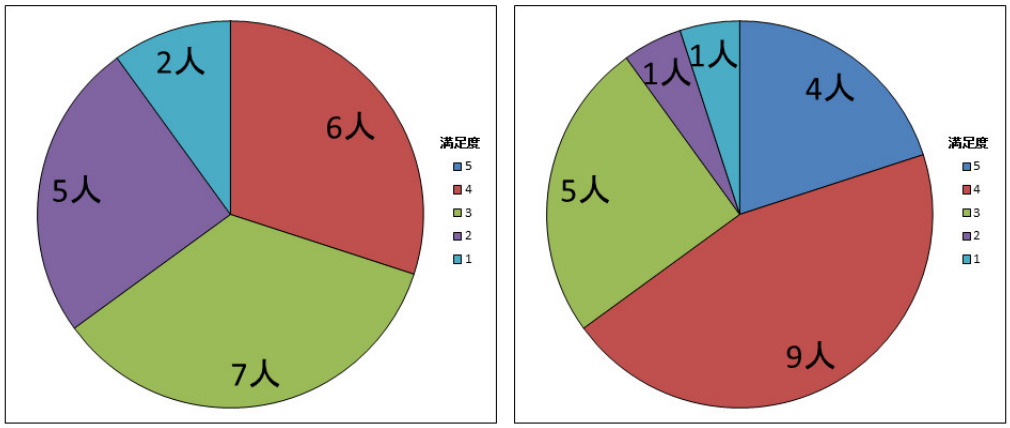


図 3.4: 選択した効果音の満足度

表 3.3: 選択した効果音の満足度の平均値

手作業検索	ツール検索
2.85	3.7

3.3 考察

本節では、実験結果をもとに考察を述べる。表 3.2 で示したように、ツール検索の方が指定した効果音を見つけ出すまでにかかった時間が短いという結果が得られたが、t 検定 [22] を行うことにより、両検索手法の指定した効果音を見つけ出すまでの平均時間に有意差があるかどうかを調べた。なお、有意水準は 5%と定め、有意水準信頼区間の外側に来る確率を p とする。計算の結果、t 検定の p 値は効果音 I において 0.63619（＞0.05）、効果音 II において 0.000006（＜0.05）、全体では 0.000075（＜0.05）となった。したがって、効果音 II と全体においては、両検索手法の指定した効果音を見つけ出すまでの平均時間に有意差があるため、本ツールで検索した方が手作業での検索よりも早く行えたと言える。また、被験者

が手作業での検索を行っているときにおいて、作業時間が 10 分を超過しても指定の効果音を見つけ出せなかった場合、指定の効果音が入っているフォルダの種類を伝えるといったヒントを提示したため、実際にはより有意差があったと推測できる。しかしながら、効果音 I においては有意差がなかったため、本ツールで検索した方が手作業での検索よりも早かったとは言い切れない。

同様に、選択した効果音の満足度の平均値に対しても t 検定を行うことにより、有意差があるかどうかを調べた。計算の結果、t 検定の p 値は 0.043286（＜ 0.05）となった。したがって、両検索手法の選択した効果音の満足度の平均値には有意差があるため、視聴する音ファイルの個数が同じ場合、本ツールで検索した方がより想定した音に近い効果音を得ることができたと言える。

タスク A では、指定した効果音によって大きな差が見られた。爆発音のように効果音の分類が明確な効果音に関しては、手作業で検索した方が早く見つけ出せるという場合が多数あった。一方で、どのフォルダに分類しているのかが想像しづらい効果音に関しては、複数のフォルダにわたり検索していく必要があるため、手作業での検索に時間がかかっていた。本ツールで音を視聴する場合はフォルダの階層を選択したり別の階層に戻ったりする必要がなく、再生ボタンを順番に押していくだけであることも検索時間の短縮に起因したと考えられる。なお、今回実験で用意した被検索対象の効果音は、例えば「戦闘音フォルダ」→「爆発音フォルダ」→「爆発音ファイル」というように 3 階層構成で管理し、合計 70 個のフォルダを用いて分類していたが、最下層のフォルダでも最大で 192 個の音ファイルが含まれているフォルダもあったため、さらに細かくフォルダを分類すれば手作業での検索時間はある程度短縮すると考えられる。

効果音 II のツール検索において、全体的に男性の方が早く見つけ出す傾向が見られたのは、音高が低めの効果音を指定していたためと考えられる。一般に、男性は低い声を出しやすく、女性は高い声を出しやすい。発声できる最も低い声から最も高い声の範囲のことを声域というが、声域によって差異が見られるのは、ユーザーの声域を考慮していない本手法の特徴でもあり問題点でもある。例えば女性

ユーザーが自らでは発声できないほどの低い効果音を検索したい場合、音高を合わせることは不可能である。

タスクBでは、両手法で検索して選んだ効果音の満足度を比較すると、本ツールで検索して選んだ効果音の方に高い満足度をつけた被験者は12人であり、逆に手作業で検索して選んだ効果音の方に高い満足度をつけた被験者は5人であった。なお、残りの3人は両方に同じ満足度をつけた。視聴する音ファイルの数を30個に制限したため、手作業で検索する手法では視聴のために選んだフォルダ内に期待通りの音ファイルがなく、想定した音とは程遠い音ファイルを視聴し続けて上限に達してしまい、満足度として1や2をつけた被験者が複数人いた。一方、ツールで検索する手法では全ての音ファイルから類似度を算出し、類似度が高い音ファイルから順番に表示するため、視聴の回数を制限してもその順番通りに視聴していくことで、ある程度想定した音に近い効果音を見つけやすかったと考えられる。

実験に対する被験者の意見としては、全体的に効果音に似せた声を発するのが難しいという意見が多かった。また、実際に録音した声を聴くと思っていたよりも音高がずれていたという感想もあり、発音時間、音高、音量を全て考慮しながら発声するのは難しく、ある程度の慣れが必要であると考えられる。その他には、本ツールで検索したとき、想定した音に近い効果音とともに想定していなかった音だが類似度の高い効果音も視聴するため、効果音を探す際の発想が広がるという意見もあった。

第 4 章

まとめ

本研究では、誰でも手軽に、想定した音に近い効果音を検索することができるように、音声入力を用いた効果音の検索手法を提案した。効果音を種類別に分類したフォルダから手作業で探す手法との比較実験の結果、特にどのフォルダに分類しているか想像しづらい音に関しては早く見つけることが可能になった。また、想定した音に近い効果音を見つけやすいことも分かった。しかしながら、現状の提案手法にはいくつかの問題点が存在する。

第一に、ユーザーの声域が考慮されていないということである。そのため、ユーザーが発声不可能な音高の効果音を探すときには適していない。この問題を解決するためには、事前にユーザーの声域を調べておき、解析の際に声域に合わせて音高を調整する方法がある。

第二に、音声入力にはある程度の慣れが必要ということである。発音時間、音高、音量を全て考慮しながら想定した効果音に似せて発声するのは難しく、慣れるまでは思い通りの検索結果が得られないことが多い。特に音高に関しては、実際に録音した声を聴くと思っていたよりも音高がずれており、音高の類似度が低くなった結果として総合類似度を大きく下げてしまうことがあった。この問題を解決するためには、録音した音声を簡単に加工、編集する機能を追加し、発音時間や音高をユーザーが調整できるようにする方法がある。

今後の展望として、以上に述べたような問題点の解決とともに、検索精度を高

めることがある。また、検索エンジンの画像検索機能のように、インターネット上の効果音を音声入力で検索する仕組みが整えば、より手軽に大量の効果音の中から検索することが可能になると考える。

謝辞

本研究を行うにあたり、多大なるご指導を頂きました本校メディア学部の渡辺大地講師、三上浩司准教授、石川知一助教、阿部雅樹先生に心より感謝致します。また、研究テーマに悩んでいた時期に相談に乗って頂いた本校メディア学部の相川清明教授と伊藤彰教先生、さらには院生の方々にも深く感謝致します。そして、苦楽を共にしながら研究を進めてきた研究室のメンバーに厚く御礼申し上げます。

最後に、実験に協力して頂いた方々と、効果音を実験に使用させて頂いた魔王魂 (http://maoudamashii.jokersounds.com/) 様と、On-jin ～音人～(http://on-jin.com/) 様に、この場を借りて感謝致します。

参考文献

[1] Apple Inc. iTunes. http://www.apple.com/jp/itunes/.

[2] Soundminer Inc. Soundminer. http://store.soundminer.com/.

[3] Crypton Future Media Inc. MUTANT. http://sonicwire.com/mutant.

[4] Shazam Entertainment Ltd. Shazam. http://www.shazam.com/.

[5] Sony Mobile Communications Inc. TrackID. http://appnavi.sonymobile.co.jp/pc/ag/index.php?page=cate&cid=26&id=925.

[6] SoundHound Inc. midomi. http://www.midomi.co.jp/.

[7] 和氣早苗, 旭敏之, 井関治. 効果音検索システム ～「音」の表現方法に関する実験と考察～. 情報処理学会, 1994.

[8] 和氣早苗, 旭敏之. 擬音語による効果音データの検索. 情報処理学会, 1996.

[9] Philip McLeod, Geoff Wyvill. A Smarter Way to Find Pitch. *Proc. International Computer Music Conference, Barcelona, Spain*, pp. 138–141, September 2005.

[10] Alain De Cheveigné, Hideki Kawahara. YIN, A Fundamental Frequency Estimator for Speech and Music. *The Journal of the Acoustical Society of America*, Vol. 111, p. 1917, April 2002.

[11] Lawrence R. Rabiner. On the Use of Autocorrelation Analysis for Pitch Detection. *IEEE Trans. Acoust., Speech & Signal Process.*, Vol. ASSP-25, pp. 24–33, February 1977.

[12] M.J. Ross, H.L. Shaffer, A. Cohen, R. Freudbereg, H.J Manley. Average Magnitude Diffrence Function Pitch Extractor. *IEEE Trans. Acoust., Speech & Signal Process.*, Vol. ASSP-22, No. 5, pp. 353–362, October 1974.

[13] Adriano Mitre, Marcelo Queiroz, Regis R. A. Faria. Accurate and Efficient Fundamental Frequency Determination from Precise Partial Estimates. *Proc. 4th AES Brazil Conference*, pp. 113–118, 2006.

[14] M.S. Andrew, J. Pincone, R.D. Degroat. Robust Pitch Determination via SVD Based Cepstral Methods. *IEEE Int. Conf. Acoust., Speech & Signal Process., Albuquerque, U.S.A.*, Vol. 1, pp. 253–256, April 1990.

[15] C. Nadeu, J. Pascual, J. Hernando. Pitch Determination using the Cepstrum of the One-sided Autocorrelation Sequence. *IEEE Int. Conf. Acoust., Speech & Signal Process., Toronto, Canada*, Vol. 5, pp. 3677–3680, April 1991.

[16] 石本祐一. 時間情報と周波数情報を用いた雑音環境における基本周波数推定に関する研究. PhD thesis, 北陸先端科学技術大学院大学, March 2004.

[17] Stephen A. Zahorian, Hongbing Hu. A SpectralOtemporal Method for Robust Fundamental Frequency Tracking. *The Journal of the Acoustical Society of America*, Vol. 123, pp. 4559–4571, April 2008.

[18] D. G. Lampard. Generalization of the WienerKhintchine Theorem to Nonstationary Processes. *Journal of Applied Physics*, Vol. 25, No. 6, p. 802, June 1954.

[19] 井澤裕司, 信州大学工学部. 窓関数 (Window Function). http://laputa.cs.shinshu-u.ac.jp/~yizawa/InfSys1/basic/chap9/index.htm.

[20] Andrew Greensted. FIR Filters by Windowing. http://www.labbookpages.co.uk/audio/firWindowing.html.

[21] PINO.TO. ♪超録 - パソコン長時間録音機. http://pino.to/choroku/index.htm.

[22] 首都大学東京 大学教育センター 情報教育担当. 2 グループの平均の t 検定. http://www.spc.tmu.ac.jp/lit/2013/1a/stat3/index.html.