

2009年度 卒業論文

音声入力を用いた
爆発音合成手法に関する研究

指導教員：渡辺 大地 講師

メディア学部 ゲームサイエンスプロジェクト
学籍番号 M0106024
池田 祐太

2009年度 卒業論文概要

論文題目

音声入力を用いた
爆発音合成手法に関する研究

メディア学部

学籍番号：M0106024

氏名

池田 祐太

**指導
教員**

渡辺 大地 講師

キーワード

音声入力、効果音、合成、爆発音、自己相関関数

近年、コンピュータの処理能力の向上などにより、個人単位であってもマルチメディアコンテンツを制作できる時代になっている。この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。しかし効果音の制作には、制作初心者が困難と感じる要素が多く、誰でも簡単にイメージした音が作れるというわけではない。また、連続して効果音が鳴るような場面においては、それぞれの音量やタイミングを考えながら配置をしていくのには時間がかかる。そこで本研究では、音声入力を用いてイメージをより直接的に効果音制作に反映させることを提案し、コンテンツにおいて用いられる機会が多く、連続して鳴ることの多い爆発音に注目してその手法を実装した。本手法は、作成したい爆発音のニュアンスを発声、録音し、そのデータを読み込み、解析を行う。解析では、録音したデータから、爆発音を配置するタイミングや音量、基本周波数などの情報を取り出す。そして取り出した情報をもとに爆発音を選択、配置、合成する。なお、タイミングや音量に関しては、波形の振幅値を基準に分析し、基本周波数は自己相関関数を用いて求める。

実装したプログラムを用いて、既存の波形編集方法との比較実験を行った。本手法の検証として、13名の被験者に本手法と既存手法の両方の手法で制作を行ってもらい、その時間を記録、分析した。分析の結果、時間的な短縮にはつながることが分かった。しかし実験後に意見や感想を述べてもらったところ、確実に思い通りの波形を制作するには既存手法の方が良いという意見や、音声入力には慣れが必要であるという意見が多く集まった。

目次

第1章	はじめに	1
1.1	研究の背景と目的	1
1.2	本論文の構成	3
第2章	手法概要	4
2.1	音声の入力	5
2.2	爆発音の配置位置の推定	7
2.3	発音片の長さの推定	9
2.4	発音片の高さの推定	10
2.5	配置する爆発音の選択	11
2.6	爆発音の配置と合成	12
第3章	実装プログラムの検証	16
3.1	生成結果	16
3.2	実験	18
3.3	実験結果	19
第4章	まとめ	22
	謝辞	24
	参考文献	25

目 次

2.1	本手法の流れ	5
2.2	録音した波形の例	6
2.3	発音片	6
2.4	増幅前の波形	7
2.5	増幅後の波形	8
2.6	爆発音の配置位置	9
2.7	発音片の長さ	10
2.8	基本周波数の推定	11
2.9	発音片のグループ分け	12
2.10	音量表現のための処理	13
2.11	重ね合わせの原理	14
2.12	本来の波形	15
2.13	オーバーフローを起こした波形	15
2.14	クリッピングを行った波形	15
3.1	音量に関する比較	17
3.2	配置する波形素材	17
3.3	発音片の基本周波数による音の選定と合成	18
3.4	それぞれの手法での波形	21
3.5	目標波形と本手法での波形の比較	21

第 1 章

はじめに

1.1 研究の背景と目的

近年、コンピュータの処理能力が向上したことや、ネットワークの配備、高速化が進んだことにより、個人規模であってもマルチメディアコンテンツを制作できる時代になっている。マルチメディアコンテンツとは、さまざまなメディアを複合して生み出すコンテンツのことであり、ここでは主に映像と音からなる音楽、映画、ゲームなどのようなコンテンツのことをいう。この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ために必要不可欠なものである。よって音声制作初心者であっても効果音を編集する機会が増えている。効果音の制作方法として最も簡潔な方法が、実際にその音を録音する方法である。この方法を的確に行えば、制作者が作りたいとイメージした音に近いものを高い品質で得ることができる。しかし、ビルの倒壊する音がほしいと考えた際に、実際にビルを爆破して録音することは現実的ではないように、音によっては録音によっての準備が困難となる場合がある。またマイクなどの機材の質や性能によって成果物のクオリティを大きく左右してしまう問題がある。よって効果音制作においては、波形編集ソフト [1][2][3] を用いて何らかの波形素材を編集してイメージの音に近づけていく手法を用いることが多い。波形編集を巧みに行うことができれば、自分の思い通りに高品質の素材を作り出すことができる。しかし波形の編集が行えるソフトには専門的な用語が多く、ある程度の音声処理の知識を必要とするという問

題がある。さらに、イメージしている音声に近づけていくためには、音を聴いて編集するという流れを何回も繰り返す必要があり、制作に時間がかかってしまうといった問題も存在する。

また、マルチメディアコンテンツの制作において、効果音が鳴るタイミングや、その音量は極めて重要である。しかし、連続して効果音が鳴るような場面においては、使用する音声ひとつひとつのタイミングや音量を調整し、配置していくのは非常に時間がかかる。以上のような理由から効果音の制作は、音声制作、編集の初心者には敷居が高いものになっている。敷居の高さを解消するために、音声制作初心者にも簡単に制作が行えるようにした工夫を施したアプリケーションも存在する。例えば擬音の入力により効果音を制作することができるアプリケーション「kawawave」[4]や、声によるMIDI入力[5][6][7][8][9]がそうである。ちなみにMIDIというのは、Musical Instrument Digital Interfaceの略で、電子楽器の演奏データを機器間でデジタル転送するための世界共通規格のことである。しかしこれらのアプリケーションにも問題がある。kawawaveにおいては、電子的な音しか作ることができず、また連続した音を一気に配置することはできないため、リアルな音声を制作できない。声によるMIDI入力においては、MIDIの情報を音声入力によって操作することができるが、MIDIにしか対応していないことや、音量を考慮できない等の問題がある。よってこれらでは、マルチメディアコンテンツのための効果音制作としての使用には厳しい。

このような問題点があることから、誰でも簡単に、イメージに近い連続した効果音を制作できる手法が必要であると考えた。このとき作りたいイメージを直接的に成果物反映させるために、音声入力に着目した。頭の中で描いている効果音のイメージをを発声してもらい、その情報を元に効果音を配置することができれば、イメージを直接的に成果物に反映させることができると考えたためである。しかし、効果音の種類というのはさまざまであり、音声的な特徴もそれぞれ異なっているため、すべての効果音に対して対応していくのは難しい。よって本研究では対象となる効果音を爆発音に限定した。爆発音を対象としたのには3つの理由が

ある。まず、マルチメディアコンテンツで使用する機会が多いためである。次に、アクセントがはっきりしていて、擬音として表現しやすいという特徴があり、音声入力に適しているためである。最後に、「バン、ババン、ドン」や「ドゴゴゴゴ」というように連続して音が鳴る場面が多いためである。

以上を踏まえ本研究では、音声入力を用いて爆発音の配置、合成を行う手法を提案する。本手法では、発声した「ドン、ババン、バン」というような爆発音のニュアンスを、音量、音高の観点から分析し、その結果をもとに爆発音を生成した。爆発音を配置する位置を決定するために、発声した爆発音のニュアンスにおいて、どういったタイミングで音を発しているのかを調べる。音を発している箇所では音量が大きくなるため、音量に対して閾値を設け、超過の有無によりその位置を特定した。また、複数種類の爆発音を配置できるようにするために、発した音の高さをもとに配置する爆発音を選択するという手法をとった。発した音の高さは、時間的に離れた2点の関係の強さを表した関数である自己相関関数を用いることで求めた。最終的に、選択した爆発音素材を、求めた配置位置に合成することで完成形となる爆発音を生成した。また、実装した手法と、波形編集ソフトでの効果音制作手法を比較するために実験を行った。実験では目標となる爆発音を用意し、その爆発音を目指して両手法で制作を行い、それぞれの制作時間や成果物を比較した。その結果、制作時間の短縮につながるものが検証できた反面、精密さに欠けてしまうという問題が残るものとなった。

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、提案手法について述べる。第3章では、提案手法の評価について、生成結果、既存手法との比較を行うための実験、実験結果という順で述べる。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

手法概要

第 2 章では本手法について説明する。本手法は、作成したい爆発音のニュアンスを発声し、それを録音し、そのデータを読み込み、解析を行う。その結果をもとに爆発音を選択、配置、合成する。以後、連続した爆発音のニュアンスを声にしたものをイメージ音声と呼ぶ。図 2.1 は本手法の流れを示すフローチャートである。節番号は各処理を説明する節を表している。

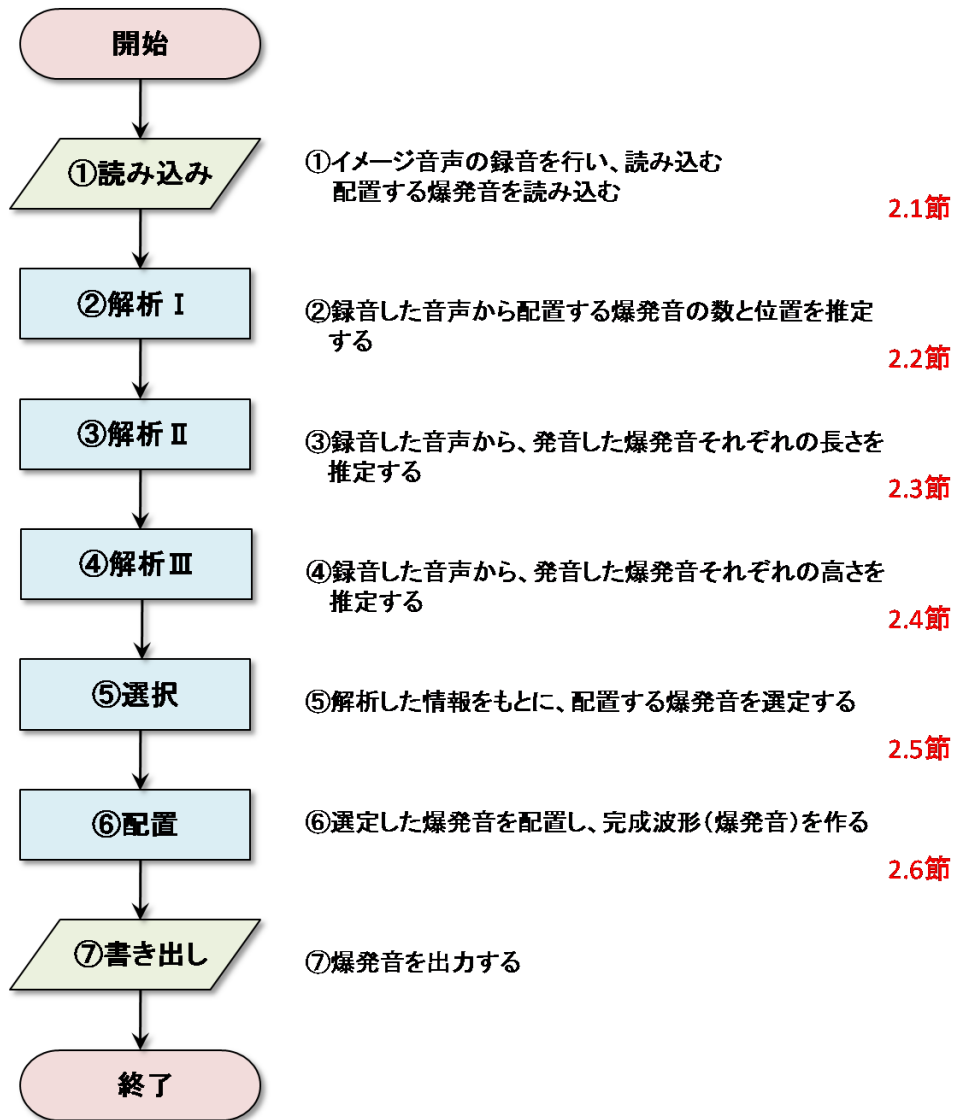


図 2.1: 本手法の流れ

2.1 音声の入力

マイクを用いてイメージ音声を録音する。このとき、イメージ音声内の発音した爆発音1つ1つを発音片と呼ぶ。図 2.2 は録音した波形の例、図 2.3 はその録音した波形における発音片を示している。

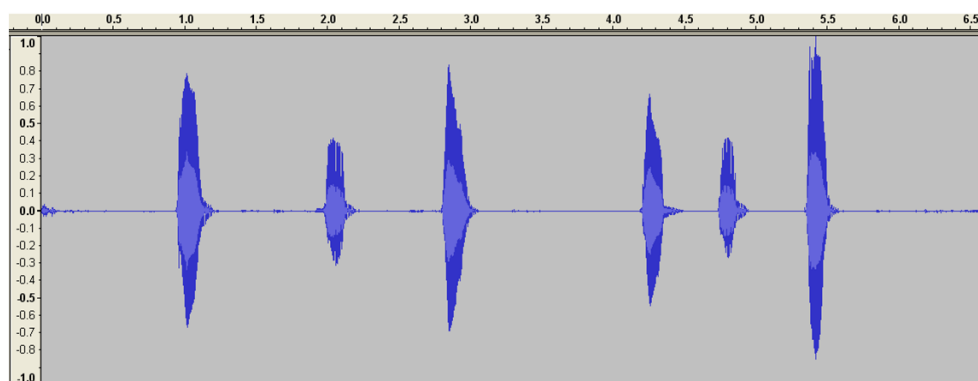


図 2.2: 録音した波形の例

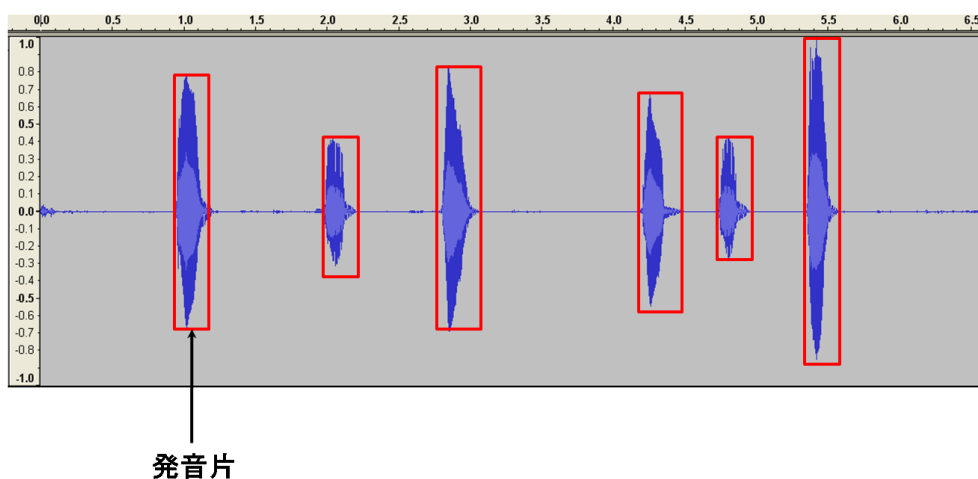


図 2.3: 発音片

本研究では PCM という方式の音声ファイルを用いる。そのフォーマットは以下の通りである。

表 2.1: PCM に関するフォーマット

標本化周波数	44.1kHz
チャンネル	モノラル
ビットレート	705kbpm
量子化ビット数	16bit

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、

信号を一定時間ごとに標本化し、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。

本研究では録音したファイルを分析し、その結果をもとに爆発音の配置を行う。また上記のフォーマットでは量子化ビット数が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。なお、ビットレートは1秒間におけるビット数を表す。またチャンネルはモノラルとする。

2.2 爆発音の配置位置の推定

本節では、それぞれの発音片の発音のタイミングを求める。まず、マイクの感度によって録音したイメージ音声の音量が異なってくるため、波形を増幅させる必要がある。録音したイメージ音声の最大振幅が1となる場合の増幅率を計算し、それをもとに波形全体を増幅させる。図2.4は増幅前の波形、図2.5は増幅後の波形である。

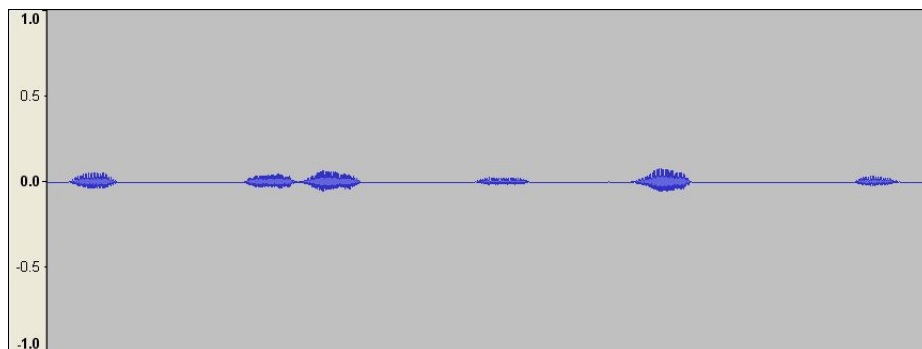


図 2.4: 増幅前の波形

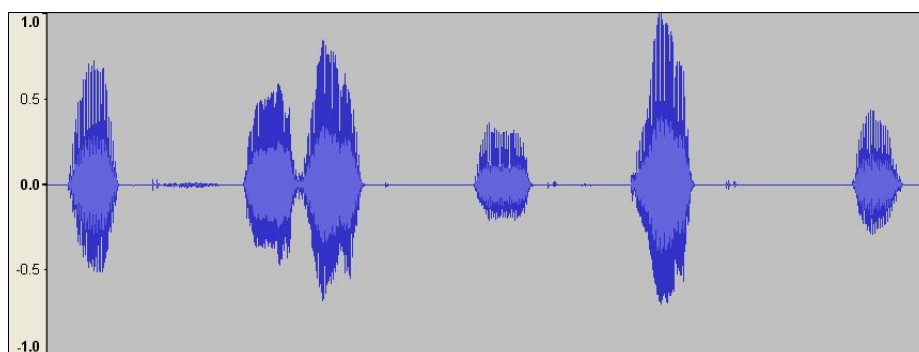


図 2.5: 増幅後の波形

次に発音のタイミングを推定する。推定は、数値データに閾値を設け、それを基準に分析することで行う。入力音声全体をタイムラインに沿って調査し、閾値を超えた箇所、超えない箇所に分類する。このとき、1つの爆発音として扱うべきであるにもかかわらず、複数の爆発音として認識するのを防ぐため、爆発音同士の最小間隔を設ける。本研究ではこれを 0.02 秒に設定した。閾値を超過した箇所から 0.02 秒前まですべての数値データに関して、閾値の超過の有無を調査する。閾値を超えた箇所がない場合は、そこを爆発音発声位置とし、閾値を超えた箇所がある場合は、1つの爆発音とみなし、爆発音発声位置にはならない。本研究では閾値を 0.25 とした。図 2.6 は爆発音の発声位置を表したものである。なお表現の都合上、数値データの個数は実際より少なくしている。

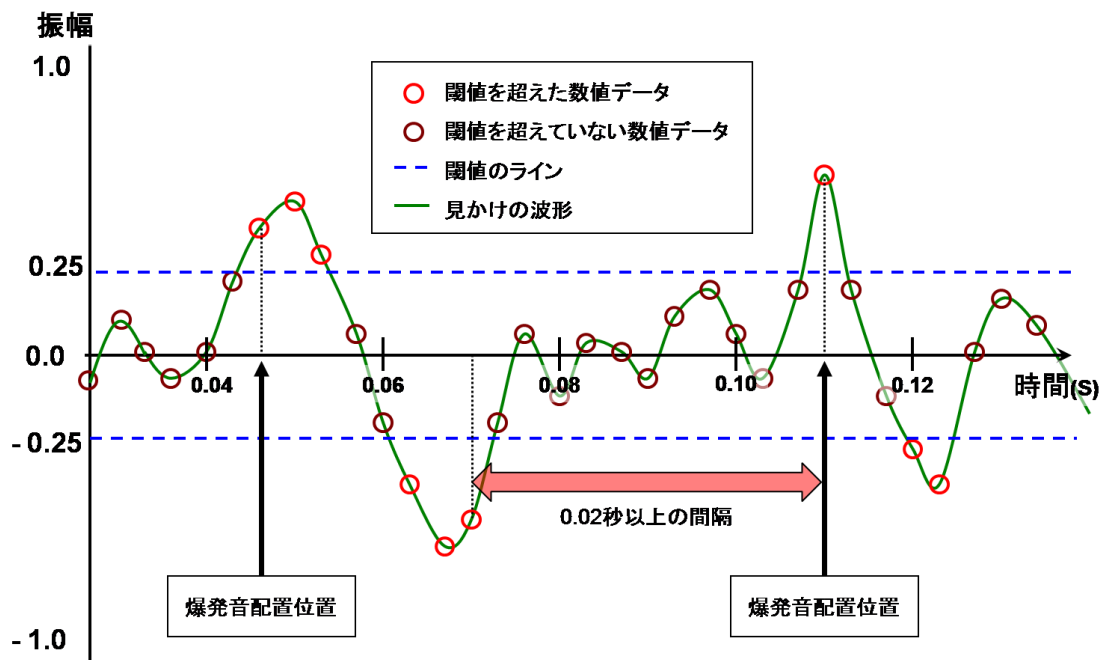


図 2.6: 爆発音の配置位置

2.3 発音片の長さの推定

発音片の長さは、2.2節で行った閾値の超過の有無における分類により分析する。爆発音配置位置から、次の爆発音配置位置より1つ前の閾値を超えた箇所までの長さを発音片の長さとする。図2.7は爆発音の長さを表したものである。図2.6と同様、数値データの個数は実際より少なくしている。

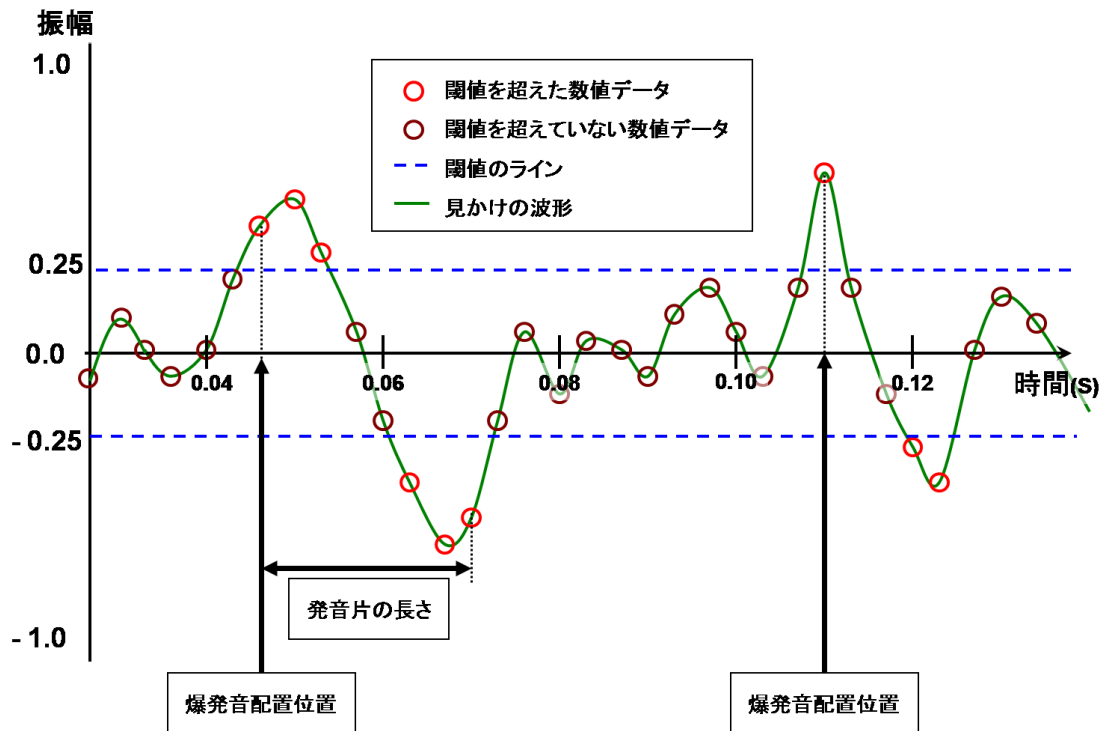


図 2.7: 発音片の長さ

2.4 発音片の高さの推定

音声の高さ（基本周波数）を抽出する手法には数多くの研究があり、相関関数による処理 [10][11][12]、スペクトル領域での処理 [13][14][15][16][17][18]、時間波形に対する処理 [19][20] に大別できる。それぞれの手法にメリットとデメリットが存在するが、厳密な抽出精度を必要とせず単純なアルゴリズムがのぞましいことや、単純なアルゴリズムでありながら雑音や位相の変化に強いことから、本研究においては自己相関関数を用いて基本周波数を求める一般的な手法を用いる。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、式 (2.1) のように表せる。

$$R(m) = \frac{1}{L} \sum_{t=0}^{L-m} x(t)x(t+m) \quad (m = 0, 1, 2, \dots, L) \quad (2.1)$$

$x(t)$ は音声信号、 t は時間、 L はサンプル数を表す。本研究では $x(t)$ を発音片の音声信号とし、 L の値を 11978 とする。 m がとり得る値すべてにおいて、式 (2.1) を

計算し、 $R(m)$ を得る。

次に、 $R(m)$ において最初のピークとなるときの m の値を求める。このとき、自己相関関数の性質上 $m = 0$ のときに $R(m)$ は最大値をとる。しかしこれは同じデータ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、 $R(m)$ が最初に 0 以下になった地点以降の最大値を最初のピークとする。

基本周波数は、 $R(m)$ が最初のピークをむかえたときの m の値から求める。標本化周波数をその m の値を割ったものが基本周波数となる。図 2.8 は基本周波数の推定を表したものである。

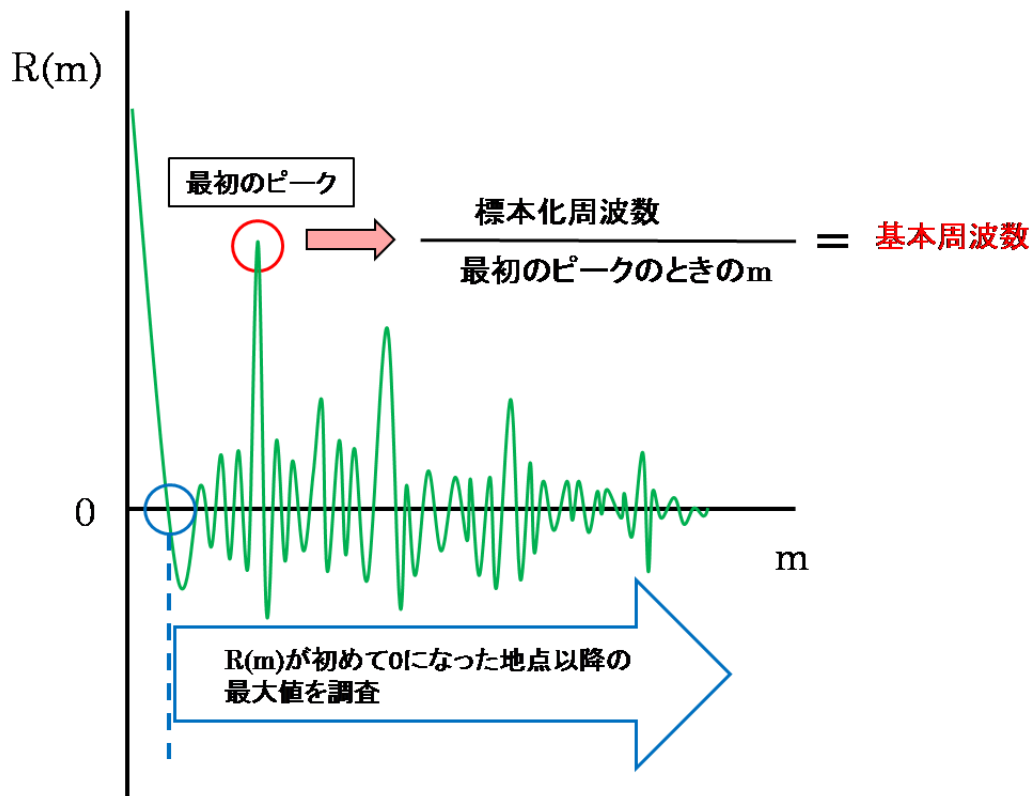


図 2.8: 基本周波数の推定

2.5 配置する爆発音の選択

2.4 節で求めた情報をもとに、発音片を高、中、低の 3 段階に分類する。全ての

発音片中最大の基本周波数を a として、 a から $a - 15$ までの範囲で基本周波数をもつ発音片を高音グループとし、最小の基本周波数を b として、 b から $b + 15$ までの範囲で基本周波数をもつ発音片を低音グループ、それ以外を中音グループと分類した。高音、中音、低音として配置する爆発音を用意しておき、それぞれのグループに対応する音を配置していく。なお実装プログラムにおいては、選定ミスを考慮して、選定後に配置予定波形を変更できるようにしている。図 2.9 は発音片のグループ分けについて表したものである。

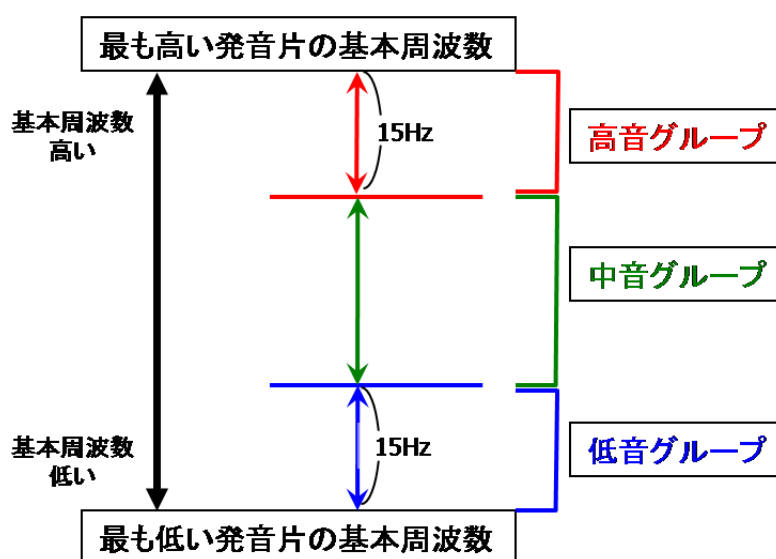


図 2.9: 発音片のグループ分け

2.6 爆発音の配置と合成

入力音声の分析が終了した後、その情報をもとに爆発音を合成する。合成を行う前に、発音片の音量の違いを考慮する必要がある。発音片の数値データの最大値と配置する予定の波形を乗算することで音量を表現する。図 2.10 はこの処理を示したものである。

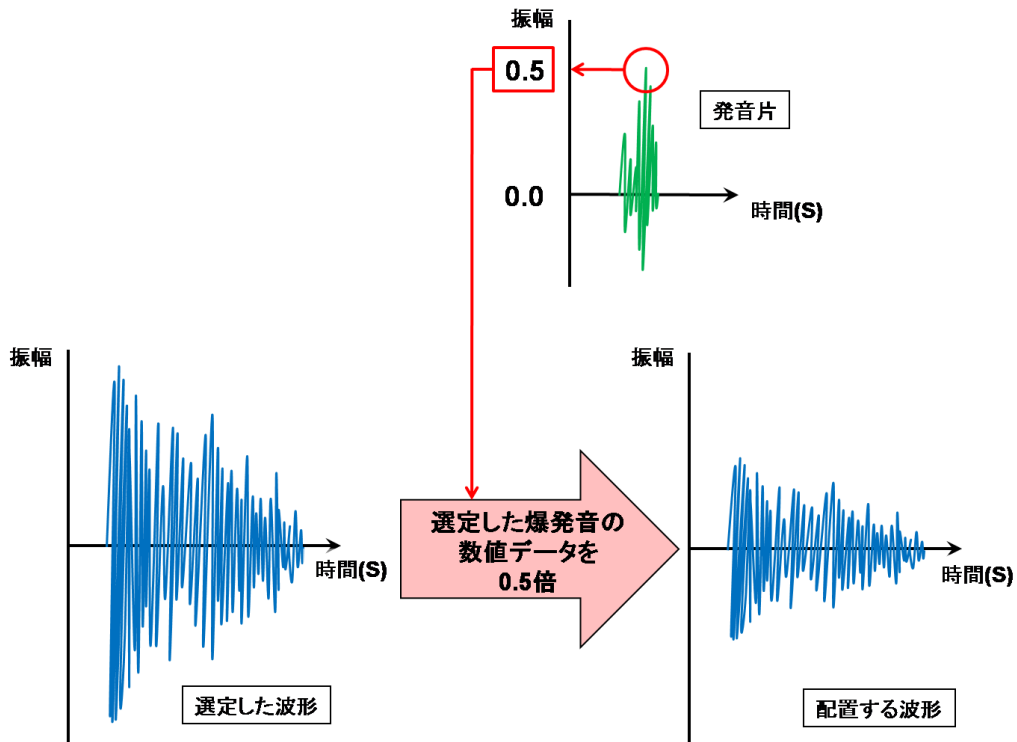


図 2.10: 音量表現のための処理

音声の合成には重ね合わせの原理を用いる。重ね合わせの原理とは2つの波が同時に存在するとき、実際に観測できる波はそれらの単純な和で表すことができるとするものである。式 (2.2) は重ね合わせの原理を表した式である。

$$y = y_1 + y_2 = f_1(x, t) + f_2(x, t) \quad (2.2)$$

波 $y_1 = f_1(x, t)$ と波 $y_2 = f_2(x, t)$ が同時に存在するとき、観測できる波は y_1 と波 y_2 の単純な和で求めることができる。図 2.11 は、重ね合わせの原理をグラフで表現したものである。

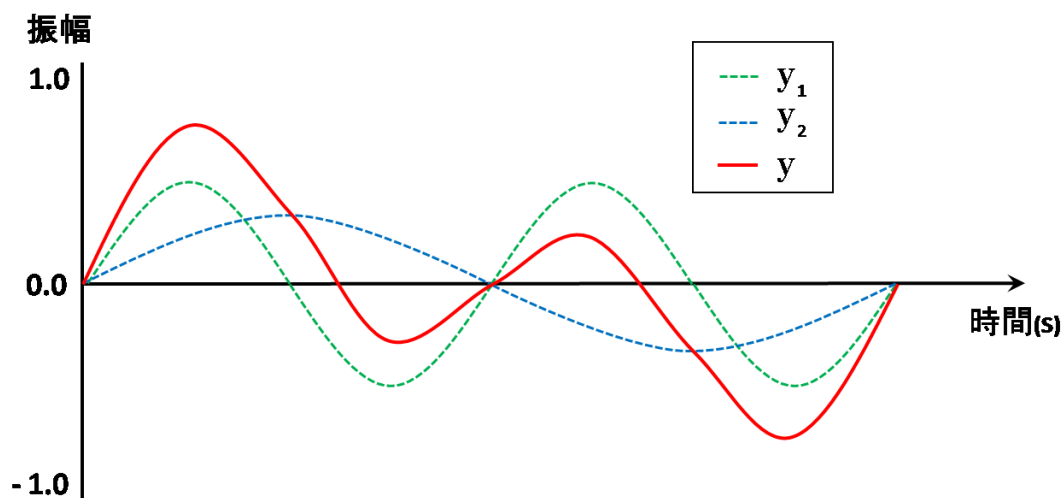


図 2.11: 重ね合わせの原理

数値の加算を行った際に、数値データが -1.0 以上 1.0 以下の範囲を超える場合、オーバーフローが起こり、数値データの正負が逆転する。これにより波形が大きく変化して、品質を損なうことになってしまう。そのため、やむ負えずオーバーフローが発生してしまう場合は、その範囲に収まるように数値データを打ち切るクリッピングという手法を用いる。本研究では数値データの加算を行った際に、数値データが 1.0 より大きくなった場合は 1.0 を代入し、 -1.0 より小さくなった場合は -1.0 を代入する。図 2.12 は本来の波形、図 2.13 はオーバーフローを起こした波形、図 2.14 はクリッピングを行った波形である。

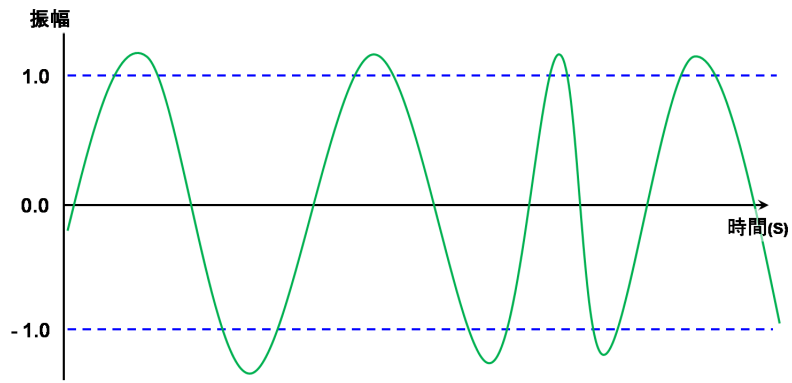


図 2.12: 本来の波形

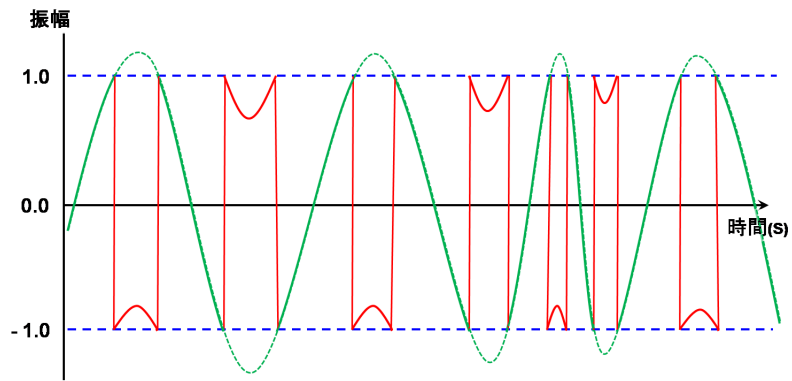


図 2.13: オーバーフローを起こした波形

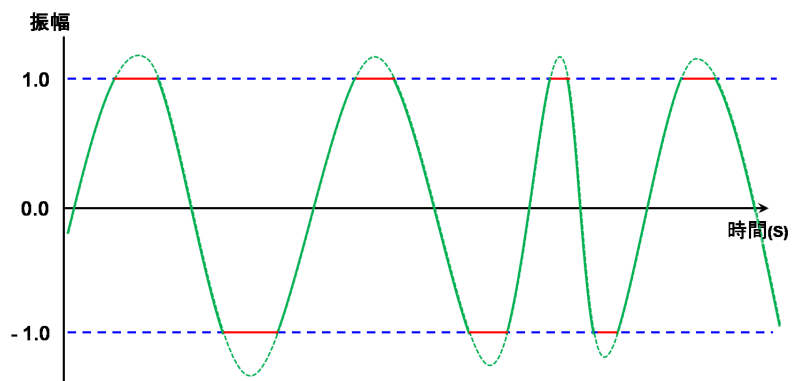


図 2.14: クリッピングを行った波形

第 3 章

実装プログラムの検証

本章では、制作したシステムを検証する。まず、実装したプログラムにより生成した波形について分析する。次に制作したシステム、波形編集ソフトのそれぞれで目的の効果音を制作してもらい、その時間を比較するという実験を行い、優位性を検証した。

3.1 生成結果

本節では、実装プログラムにより生成した波形について分析する。

まず、音量の変化に関して分析する。音量の変化が中、小、大となるように意識しながら、発音片が3つとなるようにイメージ音声を発声し、録音した。それを実装プログラムで読み込み、爆発音の生成を行った。このとき、音量の変化を視覚的にわかりやすく確認するために、配置する爆発音は高音素材、中音素材、低音素材にすべて同じものを使用した。図 3.1 は音量の変化を意識したイメージ音声の波形と、それをもとに生成した波形との比較である。3つある爆発音の音量がイメージ音声に合わせて変化しているのがわかる。

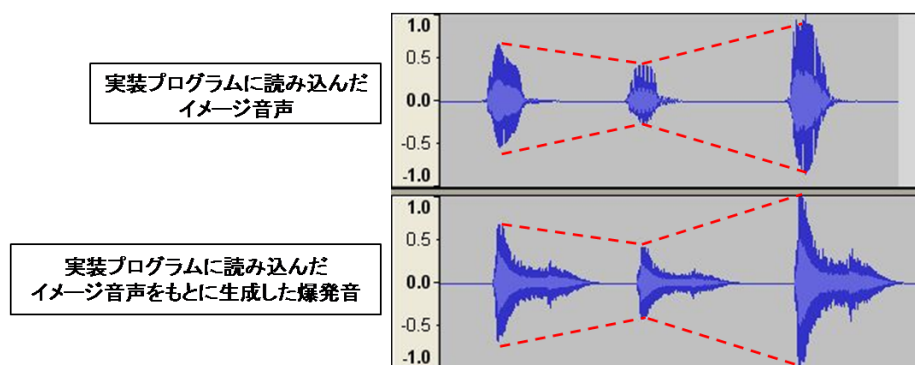


図 3.1: 音量に関する比較

次に、爆発音がきちんと選定され、上手く合成されているかどうかを分析する。
 使用した爆発音素材を図 3.2 に示す。

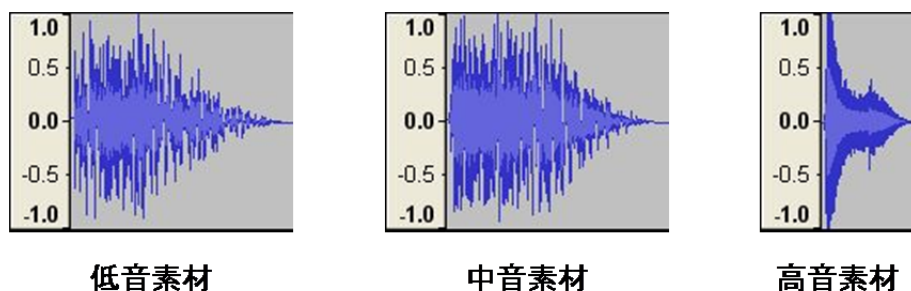


図 3.2: 配置する波形素材

読み込むイメージ音声は、以下の通りである。

発音片	開始位置	基本周波数	配置する爆発音
発音片 1	0.59 秒	137Hz	中音
発音片 2	1.33 秒	190Hz	高音
発音片 3	1.59 秒	184Hz	高音
発音片 4	2.34 秒	121Hz	低音
発音片 5	3.01 秒	166Hz	中音
発音片 6	3.94 秒	124Hz	低音

図 3.3 は、表 3.1 の波形をもとに爆発音を配置した様子を表したものである。生成した爆発音を聞いてみたところ、きちんと生成できていた。

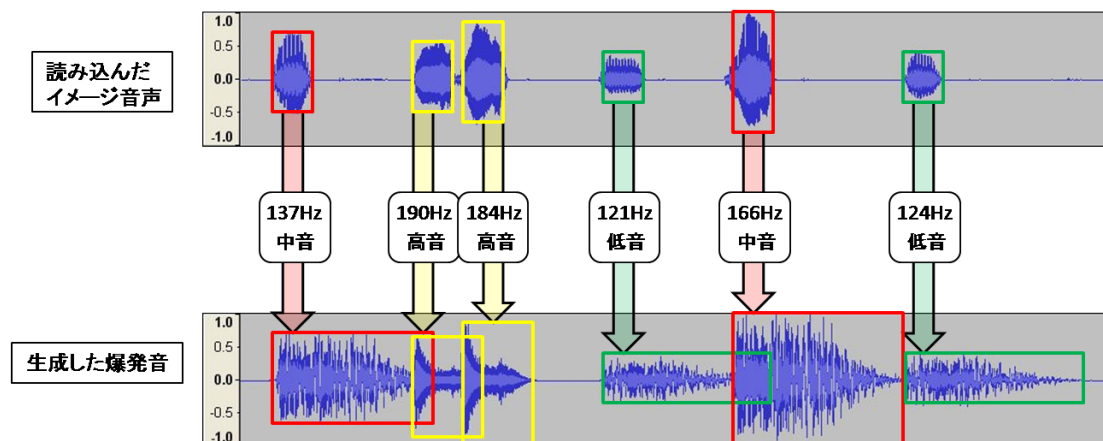


図 3.3: 発音片の基本周波数による音の選定と合成

3.2 実験

本手法と既存手法のそれぞれの手法を用いて指定の爆発音を制作してもらう。制作目標とする爆発音は3種類の爆発音で構成し、それぞれ音量を変えて数個配置したものとする。制作の際には以下の爆発音をあらかじめ被験者に与える。

1. 目標とする爆発音
2. 目標とする爆発音に用いている素材 A
3. 目標とする爆発音に用いている素材 B
4. 目標とする爆発音に用いている素材 C

これらの素材を用いて、2つの手法で制作を行ってもらい、制作にかかった時間を記録する。各手法で用いるソフトや機材は以下の通りである。

- 既存手法で用いるもの
 - － Audacity (波形編集に用いるソフト)
- 本手法で用いるもの
 - － 実装したプログラム

- SHURE SM58 (音声入力に用いるマイク)
- Audacity (録音のみに使用)

13名を対象に実験を行った。そのとき実験参加者をA、Bの2チームに分け、Aチームは本手法から、Bチームは既存手法から制作を行った。これにより、手法の実践順序による目標波形の記憶状況の差を考慮する。制作終了後には、それぞれの手法について思ったことを述べてもらった。

3.3 実験結果

本節では、今回の実験を行った結果を述べる。まずA、Bそれぞれのチームにおける各手法の制作時間を表3.3、表3.3に示す。

表3.2: Aチームの作業時間

Aチーム	本手法	既存手法
被験者 A1	145 秒	285 秒
被験者 A2	140 秒	225 秒
被験者 A3	173 秒	379 秒
被験者 A4	167 秒	269 秒
被験者 A5	224 秒	296 秒
被験者 A6	187 秒	244 秒
被験者 A7	198 秒	234 秒
平均時間	176.3 秒	276 秒

表 3.3: B チームの作業時間

B チーム	本手法	既存手法
被験者 B1	153 秒	417 秒
被験者 B2	82 秒	275 秒
被験者 B3	285 秒	532 秒
被験者 B4	96 秒	311 秒
被験者 B5	284 秒	446 秒
被験者 B6	187 秒	373 秒
平均時間	181.2 秒	392.3 秒

両チームの平均時間において、本手法の方が早く制作できたという結果が出ているが、t 検定 [21] を行うことで、両手法の平均作業時間に優位差があるかどうかを調べた。なお有意水準を 1% と定めた。計算により P 値を求めたところ、A チームでは $0.001547 (< 0.01)$ 、B チームでは $0.002476 (< 0.01)$ となったので、本手法と既存手法の作業時間には優位差があるといえる。よって本手法の方が既存手法よりも時間がかからなかったといえる。

次に、実験の被験者の意見を示す。全体として、本手法では手軽に楽しく制作できるが、確実に思い通りの波形を作るには既存手法の方が優れているという意見が多かった。本手法ではイメージしている音声を一気に録音するため、目標波形とイメージした目標波形との音量の違いや、タイミングの微妙なズレに気がつかない。よって 1 回の入力でも手軽にそれなりのものが作れるが、トライアンドエラーを繰り返す既存手法での生成波形より精密さに欠ける。図 3.4 は目標波形、本手法での生成波形例、既存手法での生成波形例を並べたものであり、図 3.5 は目標波形と本手法での生成波形例を比較したものである。

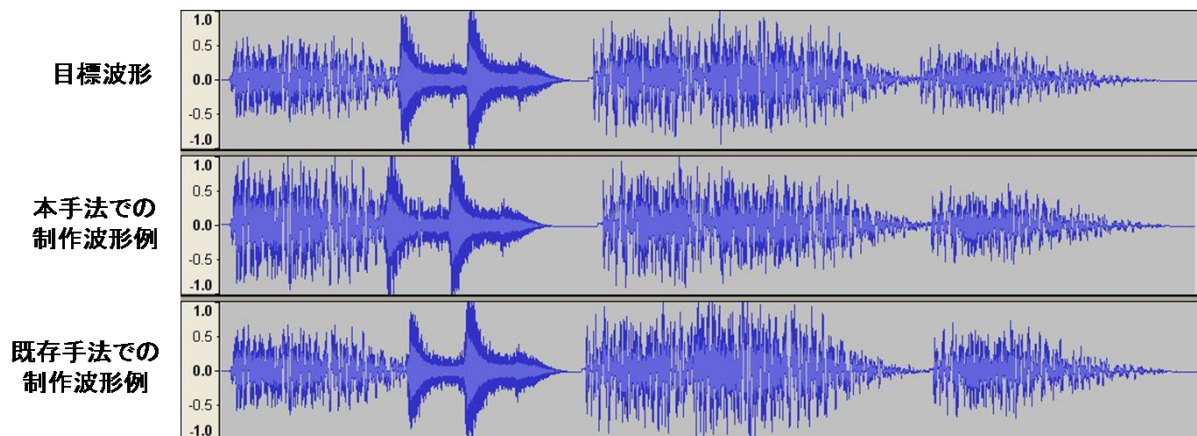


図 3.4: それぞれの手法での波形

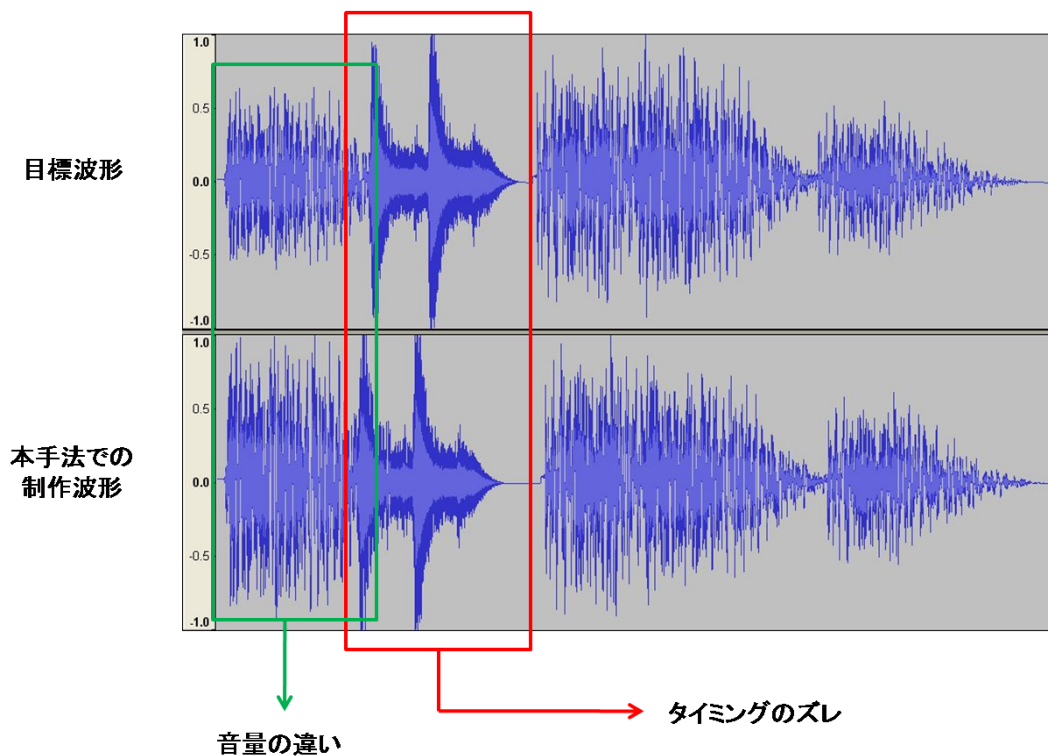


図 3.5: 目標波形と本手法での波形の比較

同様に多かった意見が、音声入力には慣れが必要ということである。発声するに当たり、タイミング、音の高さ、音の大きさをすべて考慮しながら発音するのは困難で、ある程度の訓練が必要である。また音声編集に声を用いるということに戸惑いを感じ、どのように発音すればいいか悩んでしまう例もあった。

第 4 章

まとめ

本章では論文の締めくくりとして、まとめと今後の展望について述べる。

本研究では初心者でも簡単に連続した効果音を制作することができるように、音声入力を用いることで効果音を配置・合成する手法を提案した。爆発音に焦点を当てて実装を行い、既存手法との制作時間の比較を行った。比較の結果、作業時間において既存の方法よりも短縮することが可能になった。しかし同時にいくつかの問題点も存在した。

1つは、本手法では手軽に制作できるが、確実に思い通りの波形を作るには既存手法の方が優れている点である。問題解決のためには、イメージ音声を取り込んだ後に、タイミングや音量などを微調整できるようにすることや、既存の方法との統合的な環境が必要であると言える。

もう1つは、入力にある程度の訓練やコツが必要となることである。イメージ音声を発声するに辺り、発音のタイミング、音の高さ、音の大きさをすべて考慮するというのは難しく、慣れが必要になってくる。特に音の高さに関しては、発声の際に上手く高さを変化させることができず、検出ミスが起こり、イメージしている音とは異なる音が選定されてしまうことがあった。この問題を解決するためには、音声分析に関するパラメータを制作者側が調節できるようにする必要がある。

今後の展望としては、より精密な波形を生成するために、音声入力をスタートとして考え、イメージ音声の読み込み後に、ある程度微調整が行えるような仕組み

を設けることがある。これは本研究のプログラムで実装していなかったインターフェイス面も含めて検討する必要がある。また本研究は読み込む波形次第で爆発音以外の効果音も生成が可能である。よってアイデア次第で様々な音声の制作に役立てることができるだろう。

謝辞

本研究を行うにあたり、温かいご指導ご教授を頂きました、渡辺大地講師、三上浩司講師、本校大学院の先輩方に心から感謝致します。また、実験に協力してくださった友人、サウンドジェネシスの皆さんに深く感謝申し上げます。

最後に、私がこの激動の日々を乗り越えることができたのは、ゲームサイエンスプロジェクトのメンバーがいたからです。皆さんと過ごした日々を糧とし、未来を歩んでいきたいと思えます。

参考文献

- [1] 株式会社 サイクル・オブ・フィフス, Sound Engine Free, <<http://www.cycleof5th.com/products/soundengine/?lang=en>>.
- [2] Audacity <<http://audacity.sourceforge.net/?lang=ja>>.
- [3] Steinberg 社, WaveLab6 <http://japan.steinberg.net/jp/products/audio_editing/wavelab_6.html>.
- [4] 河合 章悟, KanaWave <<http://www.vector.co.jp/soft/win95/art/se232653.html>>.
- [5] ヤマハ株式会社, ボイストウスコア <<http://www.yamaha.co.jp/>>.
- [6] 株式会社インターネット, Singer Song Writer <<http://www.ssw.co.jp/>>.
- [7] arakisoftware, 採譜の達人 <<http://www.pluto.dti.ne.jp/~araki/soft/st.html>>.
- [8] 平野 賢史, れっつ, みみ! <<http://www.geocities.co.jp/SiliconValley-SanJose/9004/>>.
- [9] Masahiro Fukui, EasyMusicAnalysis2 <<http://www.vector.co.jp/soft/winnt/art/se308052.html>>.

- [10] A. Moreno and J.A. Fonollosa, "Pitch determination of noisy speech using higher order statistics," IEEE Int. Conf. Acoust., Speech & Signal Process., San Francisco, U.S.A., vol.I, pp.133-136, March. 1992.
- [11] L.R. Rabiner, "On the use of autocorrelation analysis for pitch detection," IEEE Trans. Acoust., Speech & Signal Process., vol.ASSP-25, pp.24-33, Feb. 1977.
- [12] M.J. Ross, H.L. Shaffer, A. Cohen, R. Freudberg, and H.J Manley, "Average magnitude difference function pitch extractor," IEEE Trans. Acoust., Speech & Signal Process., vol.ASSP-22, no.5, pp.353-362, Oct. 1974.
- [13] M.S. Andrew, J. Pincone, and R.D. Degroat, "Robust pitch determination via SVD based cepstral methods," IEEE Int. Conf. Acoust., Speech, Signal Process., Albuquerque, U.S.A., no.S4b.10, pp.253-256, Aoruk 1990.
- [14] L. Hodgson, M.E. Jernigan, and B.L. Wills, "Nonlinear multiplicative cepstral analysis for pitch extraction in speech," IEEE Int. Conf. Acoust., Speech, Signal Process, no.S4b.11, pp.257-260, April 1990.
- [15] 加藤誠二, 三輪讓二, "移動平均と帯域制限を用いたケプストラム型基本周波数抽出とその応用" 信学技報, SP94-95, Feb. 1995.
- [16] A.M. Noll, "Cepstrum pitch determination," J.Acoust. Soc. Am., vol41, no.2, pp.442-448, Aug. 1969.
- [17] 島村徹也, 高木浩司, "帯域制限をかけた振幅スペクトルのべき乗に基づく基本周波数抽出法" 電子情報通信学会論文誌 A Vol.J86-A, No.11, pp.1097-1107, Nov. 2003.
- [18] 阿竹 義徳, 入野 俊夫, 河原 英紀, 陸 金林, 中村 哲, 鹿野 清宏, "調波成分の瞬時周波数を用いた基本周波数推定方法 (音声情報処理 : 現状と将来技術論

文特集) ” 電子情報通信学会論文誌. D-II, 情報・システム, II-パターン処理
J83-D-II(11), 2077-2086, 20001125.

[19] B. Gold and L. Rabiner, "Parallel processing techniques for estimating pitch periods of speech in the time domain," J. Acoust. Soc. Am., vol.46, no.2, pp.442-448, Aug. 1969.

[20] N.J. Miller, "Pitch detection by data reduction," IEEE Trans. Acoust. Speech & Signal Process., vol.ASSP-23, no.1, pp.72-79, Feb. 1975.

[21] 柳井久江, 「4Steps エクセル統計 (第2版)」, 2004.