

仮想世界における効果音検索

高橋 克郎[†] 神門 典子^{††} 大島 裕明^{††}

[†] 兵庫県立大学応用情報科学研究科 〒650-0047 兵庫県神戸市中央区港島南町 7-1-28

^{††} 国立情報学研究所 〒100-0003 東京都千代田区一ツ橋 2-1-2

E-mail: [†]katsurou.tkhs@gmail.com, ^{††}kando@nii.ac.jp, ^{†††}ohshima@ai.u-hyogo.ac.jp

あらまし 我々は、VR デバイスを用いた効果音検索インタフェースを提案する。検索対象である効果音は仮想世界内のオブジェクトとして表現し、ユーザは仮想世界を探索して求める効果音を発見する。効果音は、メル周波数ケプストラムを用いて特徴ベクトルとして表現する。特徴ベクトルの類似度を考慮したバネモデルによって類似する効果音を表すオブジェクトを近くに配置されるようにする。YouTube Audio Library の効果音 1,448 個を用いて、従来のリスト表示による検索インタフェースとの比較を行い、提案インタフェースの優位性を示す。

キーワード 仮想世界, VR, 効果音検索

1. はじめに

1.1 効果音検索の重要性

近年、映像や楽曲制作の際には、いくつもの効果音を制作者が制作の度に採集して制作する作品に使用するのではなく、すでに Web 上あるいは使用するアプリケーション内部に存在する効果音を利用するケースが主流になってきている。YouTube やニコニコ動画などの動画配信サイトで配信される動画は、映像制作会社や音楽制作会社が制作した効果音を、一般ユーザが上記のように二次的に利用して制作された動画が配信の大きな割合を占めている。したがって、このような一般ユーザの映像や楽曲制作のために、効果音のインタフェースは重要である。

1.2 効果音検索インタフェースの一例

前節のように、近年 YouTube などの動画投稿サイトで一般ユーザが自分で映像を制作し、投稿する例が多く見られる。こうした効果音検索の際に、一般ユーザは検索したい効果音をその項目名や説明文、印象に当たる語に言語化して検索クエリとし、通常の Web 検索と同様に検索結果の効果音のリストから検索するのが一般的である。YouTube はこのようなユーザのために YouTube Audio Library^(注1) という Web 上のデータベースサイトを開設した (図 2)。このインタフェースの検索例は、YouTube Audio Library 中の検索意図の「Falling Whistle」というホイッスルの音に対して「Whistle」、「Falling」などをクエリを入力とするなどである。ここで、効果音とは映像や楽曲の制作に使用される断片的な環境音をさす。

1.3 YouTube Audio Library の検索機能の問題点

このデータベースには効果音ごとに 20 種類 (後述、表 1) のカテゴリと項目名が、それぞれのファイルにメタデータとしてあらかじめ与えられている。YouTube などの動画サイトに投稿するための動画を制作する一般ユーザ (以後、ユーザと呼ぶ) の効果音検索は、YouTube Audio Library などのインタフェースを利用して大規模な効果音データベースから適切な効果音を

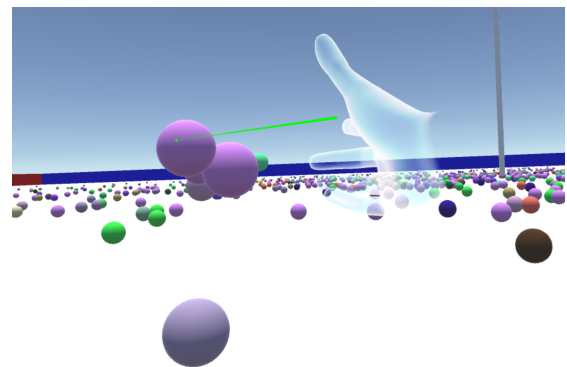


図 1 提案システムの概観。

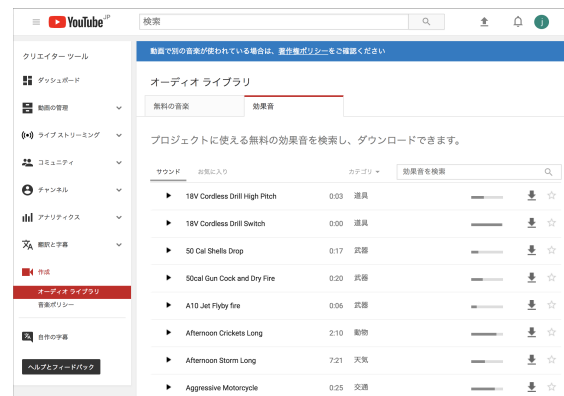


図 2 YouTube Audio Library (従来システム)。

発見する作業が必要となる。

しかし、項目名や説明文などのテキストから想起される効果音の印象と、実際に聞いた印象が一致しない場合が少なくない。そのため、ユーザが意図した効果音にたどり着くためには試行錯誤を繰り返す必要があるため負荷が大きい。

そればかりでなく、効果音の項目名や説明文によりユーザが意図した効果音に対して誤った印象を与えてしまい、検索意図

(注1) : YouTube Audio Library: <https://www.youtube.com/audiolibrary/>

の効果音を検索対象から除外してしまう場合もある。具体例として、YouTube Audio Library のファイルの中に「Falling Whistle」というホイッスルの音と「Microwave In Use」という電子レンジの音が存在する。両者は同一の検索クエリで該当することはないが、ホイッスルの音と電子レンジのファンの音とが似ている。また、クエリで検索した場合該当する効果音が多すぎて検索が困難な場合もある。以上から、効果音検索では、言語化されたクエリで検索することは効率的とは言えない。

効果音検索の際には、探したい音はユーザのイメージの中にあり、データベース内に探したい音が存在しているか不確かなまま検索を開始し、より多くの効果音に探索的にアクセスを繰り返すのが一般的である。これらの理由により、実際の効果音検索の際に、候補となり得る数多くの効果音にアクセスして、それらをひとつずつ聞きながら取捨選択することがユーザに求められる。この効果音検索は多大な時間を要し、負担を強いる作業である。

我々は、YouTube Audio Library より効率的な効果音検索システムとして仮想現実による効果音検索インタフェース（図1）を提案する。YouTube Audio Library は効果音検索のインタフェースとして平面で与えられた効果音のリストを一つ一つ再生する。このインタフェースは効果音の検索インタフェースとして一般的であるため、YouTube Audio Library を従来手法とする。提案システムでは、仮想世界に没入して効果音を検索する。各効果音は仮想世界内に形状をもつオブジェクトとして表現される（以後、効果音オブジェクトと呼ぶ）。ユーザは効果音オブジェクトが配置された平面上を移動し、効果音を探索する。仮想現実により、ユーザの視野の外にある効果音オブジェクトを探そうとするときに、振り返るなどの自然な動作で視野の中に入れることができる。仮想空間内でいくつかの効果音が常に再生され、効果音オブジェクトは再生されている間色が変わる。ユーザから遠い音は小さく、近い音は大きく聞こえる。上記の視覚情報と聴覚情報を利用して検索の効率化を図る。

2. 関連研究

2.1 MFCC

データベース内にある音を再生するときに、検索空間のなかで人間が類似していると感じる音同士が近い位置に配置されていると検索効率は上がると予想される。Davis らはメル周波数に音声を変換した後で、そのケプストラムであるメル周波数ケプストラム（以下、MFCC）を提示した[4]。MFCC は現在、人間の知覚で類似度が高いと感じる音の指標として広く使われている。

2.2 効果音とオノマトペ検索

効果音あるいはオノマトペの検索に関する先行研究は数多くなされてきている（[17], [5], [15], [19], [18], [9]）。Zhang らは MFCC として表現された音声ファイルを隠れマルコフモデル（以下、HMM）と混合ガウス分布（以下、GMM）による効果音の分類をしている[17]。Gygi らは MFCC を類似度の算出に用いた環境音を擬音語で表現する試みをしている[5]。Turnbull らはテキストと効果音を GMM により学習させ、テストデータ

として効果音を入力すると Text をクエリとして出力できるシステムを提示した[15]。岡本らは、テキストをクエリとして入力し、可視化された効果音オブジェクトを出力する効果音検索のインタフェースを開発した（[19], [18], [9]）。清水らは MFCC を使用して、オノマトペを可視化した検索インタフェースを開発した[20]。岡本らと清水らの共通点は効果音可視化している点である。これらの共通点は MFCC を音の類似度の指標として用い、効果音あるいはオノマトペを機械学習により分類している点である。

2.3 楽曲検索

楽曲検索に関しての従来研究は以下の通り行われて来た（[14], [16], [12], [13], [7], [11], [3]）。Turnbull らは Pop, Jazz などのジャンル名をクエリとした楽曲検索システムを提示した[14]。これは GMM によりジャンル名と楽曲ファイルを分類している。Zhang らは テキストと音と個人の最適化タグをクエリとする、複合的な音楽検索システムを提示した[16]。Schedl らは、効果音検索に関しての使用されるクエリを環境の文脈、個人の文脈、タスクの文脈、社会的な文脈と時系列文脈の5つに種類分けし、従来研究を分類した[12]。Skov らは一般的なメタデータに抽象的なメディアと科学的なメタデータの両方を付与することが音楽検索に有効だと示している[13]。Jarrett は楽曲の Web での説明文テキスト、タグと歌詞、の楽曲の文脈情報のクエリに対して、どの類似度が有効か議論している[7]。Piczak は楽曲を SVM, k-NN, ランダムフォレストによる分類のうち、ランダムフォレストが音楽検索に有効であることを示した[11]。Cunningham らは Adam algorithm を利用して音楽データとテキストデータを入力した、テキストベースの楽曲検索システムを提示している[3]。

2.4 VR による検索インタフェース

仮想現実を用いた検索インタフェースに関しての先行研究は以下の通り行われてきた（[10], [8], [6], [1]）。仮想世界を検索空間とした文字の検索速度に関して、Pausch らは仮想世界内で文字列の検索の研究を行っている[10]。Johnson らは科学論文の VR インタフェースを提示している[8]。Jacobsen らは Johnson らのシステムを Web ベースのデータベースに改良した VRML を提示した[6]。これらのインタフェースは基本的には仮想世界内のポイントのみを使用したものであり、音声やジェスチャーなどの入力とは考慮されていない。Bolt はより自然なジェスチャーや発話などのインタラクティブインタフェースが仮想世界のインタフェースとして有効であると主張している[1]。現在は Oculus Rift^(注2) などより自然な仮想現実のデバイスが普及している。

3. タスクの設定

本節で、効果音検索のタスクを定義する。本タスクは、端的にはユーザが想定する効果音を効果音検索システムを利用してなるべく早く発見するというものである。

効果音検索タスクとして、「既知情報検索 (Known-item search)

(注2) : Oculus Rift: <https://www.oculus.com/rift/>



図3 VR デバイス装着時のユーザ.



図4 VR コントローラ.

現実世界では、かつて聞いたことがあるものをさがす。」場合を考える。

このタスクにおいては、ユーザに検索意図として効果音を一つ聞いてもらう。そして、その効果音そのものを発見してもらう。たとえば、ユーザに「Falling Whistle」という名前のホイッスルの音を聞いてもらったとする。この場合、「Falling Whistle」そのものを発見する。

それに対してユーザが検索するのは、「Falling Whistle」というホイッスルの音である。 S を YouTube Audio Library に含まれる効果音の集合とする。 S_{img} をユーザのイメージの中にある効果音の集合とする。タスクの入力は $e \in S$ である。タスクの出力は $e'' \in S$ である。上記の出力はユーザの検索によって与えられる。これらの出力を評価するためには、ユーザが検索した $e'' \in S$ が正解の $e' \in S$ と一致するかを判定し、開始から入力と出力が一致するまでのユーザの検索時間が得られれば十分である。ユーザの検索時間が短いほど、インタフェースとして優れていることを示している。

4. 仮想現実効果音検索システム

4.1 提案システムの概要

提案システムは VR デバイスを使用して仮想世界に没入し効果音を検索するシステムである。ユーザは効果音オブジェクトが配置された平面（以後、移動平面と呼ぶ）上を移動し、効果音を探索する。提案システムの概観を図1に示す。検索対象の効果音が仮想世界においてボールとして表現されている。この



図5 仮想空間内の VR コントローラ (手).

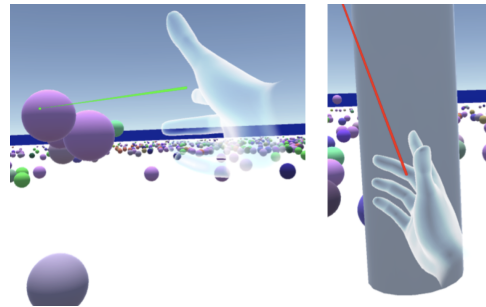


図6 レーザーポインタ.

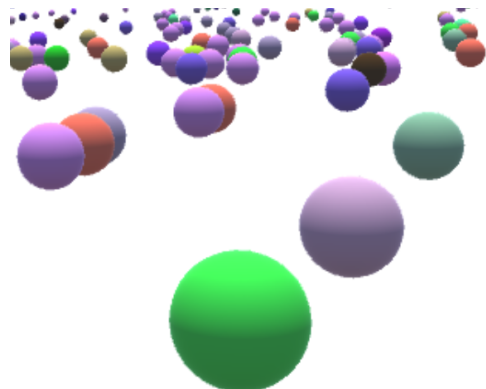


図7 効果音オブジェクトのボール表現.

ボールを効果音オブジェクトと呼ぶこととする。ユーザは任意の効果音オブジェクトの効果音を再生させることができる。図3は VR デバイス装着時のユーザの様子である。VR デバイスのヘッドマウントディスプレイを装着し、VR コントローラを両手に持っている様子である。

図4は VR コントローラを装着した様子である。VR コントローラには 360 度方向を指示できるレバーといくつかのボタンがあり、それらを用いて仮想世界における移動や効果音オブジェクトの再生を行うことができる。

前述の Oculus Rift をデバイスとして使用した。実装は Unity^(注3) により行なった。

(注3) : Unity: <https://unity3d.com/jp>

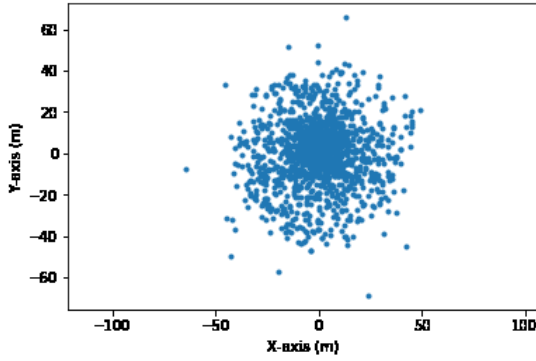


図 8 効果音オブジェクトの移動平面上の配置例.

4.2 ユーザインタラクション

3 節のタスクを行うために以下の機能が必要となる.

- 移動する
- ボールを再生する
- ボールを視野に入れる

上記の機能を以下のインタラクションで実現する.

- VR コントローラのレバーによる移動平面上の移動
- VR コントローラのトリガーによる効果音の再生
- VR ヘッドマウントディスプレイ装着時に振り返る

ユーザが装着している VR ヘッドマウントが向いている方向に対して, VR コントローラでレバーを倒した方向にユーザは仮想世界内で移動する. VR コントローラのトリガーに指が接触することにより仮想世界内の VR コントローラの表示である手 (図 5) からレーザーポインタ (図 6) が発せられる. レーザーポインタは効果音オブジェクト, 壁または移動平面に当たると緑色 (図 6 右) になり, それ以外の場合は赤色 (図 6 左) となる. 効果音オブジェクトにレーザー接触した時にトリガーが押されると効果音が一度だけ再生される. VR ヘッドマウントディスプレイ (図 3 のユーザ頭部) 装着時に振り返ることにより, 自然に任意の効果音オブジェクトを視野に入れることができる. ユーザから遠い音は小さく, 近い音は大きく聞こえる. 音の大きさの減衰は距離に比例する.

4.3 効果音オブジェクトの配置

検索開始前に仮想世界内で効果音オブジェクトを配置するための処理を行う. 前節の通り, 仮想世界内に効果音を検索するための移動平面が用意されている. まず, この移動平面上に効果音オブジェクトが自動配置される. 次に, 効果音オブジェクトの位置が最適化される. 今回は, 効果音オブジェクトの位置の最適化のためにバネモデルを使用した. バネモデルにより, バネの引力と斥力, 速度抵抗が効果音オブジェクト間に完全グラフとして働く. バネの斥力は効果音間の類似度に基づいて重み付けされる. 効果音間の類似度として MFCC (4.2.2 節に後述) による効果音の特徴ベクトルを使用する. これらの音の特徴ベクトルの任意の 2 つの組 (t_i, t_j) のコサインを効果音の類似度とした. バネの引力は一律に作用する. 効果音オブジェクトにかかる力はすべて水平成分のみである. 効果音オブジェクト同士の衝突は考慮せず, 各オブジェクトは互いに透過する.

上記の力を平衡状態になるまで働かせた後, 効果音オブジェクトの位置を固定する. 以下の節でバネモデルと効果音の特徴ベクトル表現, 効果音間の類似度についてそれぞれ説明する.

4.3.1 バネモデル

各効果音オブジェクト間にバネモデルとしての引力と斥力が働いている. 各効果音オブジェクトには以下の力が働いている. 各オブジェクトに対して働く力は全て水平成分のみである.

- (1) バネモデルによるオブジェクト間の斥力 $\mathbf{R}$
- (2) バネモデルによるオブジェクト間の引力 $\mathbf{A}$
- (3) オブジェクトの移動にかかる速度抵抗 $\mathbf{K}$

各効果音オブジェクトに働く力 $\mathbf{F}$ は $\mathbf{R}$, $\mathbf{A}$, $\mathbf{K}$ の水平成分をそれぞれ $\mathbf{R}_{hrz}$, $\mathbf{A}_{hrz}$, $\mathbf{K}_{hrz}$ として以下で表される.

$$\mathbf{F} = \mathbf{R}_{hrz} + \mathbf{A}_{hrz} + \mathbf{K}_{hrz} \quad (1)$$

バネモデルによる各ノード間の斥力は以下の式で表現される. I を仮想世界に配置される効果音が付加されたオブジェクトの基数の集合とする. 仮想世界に配置される効果音が付加されたオブジェクト $a_i (i \in I)$ の集合を A とおく.

$$\mathbf{R} = \sum_{a_j \in A (j \neq i)} \frac{\alpha \cdot \max(0, |\mathbf{a}_i \mathbf{a}_j| - k)}{\text{sim}(\mathbf{t}_i, \mathbf{t}_j)} \cdot \mathbf{a}_i \mathbf{a}_j \quad (2)$$

ここで, $\mathbf{a}_i \mathbf{a}_j$ はオブジェクト a_j の位置ベクトル $\mathbf{a}_j$ のオブジェクト a_i の位置ベクトル $\mathbf{a}_i$ からの相対的な位置ベクトルをさす. α はバネ荷重, k はバネ定数であり, 実数値として入力される. 今回は $\alpha = 1.0$, $k = 1.0$ とした. また, $\text{sim}(\mathbf{s}_i, \mathbf{t}_j)$ は i 番目と j 番目の効果音の特徴ベクトル $\mathbf{t}_i$ と $\mathbf{t}_j$ の間の類似度である. 各ノード間の引力は以下の式で表現される. 類似度が大きいほど斥力は小さくなる.

$$\mathbf{A} = -\frac{g}{|\mathbf{a}_i \mathbf{a}_j|^2} \cdot \mathbf{a}_i \mathbf{a}_j \quad (3)$$

ここで, g は引力定数である. 今回は $g = 1.0$ とした. 効果音オブジェクトの移動にかかる速度に関する抵抗は以下の式で表現される.

$$\mathbf{K} = -r_i \cdot \mathbf{v}_{a_i} \quad (4)$$

ここで, $\mathbf{v}_{a_i}$ はオブジェクト a_i の速度ベクトルを示す. r_i は速度抵抗の定数である.

まず, 効果音オブジェクトをランダムに配置する. 次に, 効果音オブジェクトの位置が平衡状態になるまで動かした後, 効果音オブジェクトに働くすべての力を消し位置を固定する.

4.3.2 効果音の特徴ベクトルでの表現

効果音間の特徴ベクトル $\mathbf{t}_i$ のために MFCC を加工して使用した. MFCC とは, メル周波数での音のケプストラムである. メル周波数とはメル周波数の差が同じであれば, 人間の知覚では音高に差がないという意図で設計された音高の変換則である. まず, 効果音の周波数を入力とする. 以下の式を用いて効果音の周波数 f をメル周波数 m に変換する.

$$m = m_0 \ln \left(\frac{f}{f_0} + 1 \right) \quad (5)$$

ここで f_0 は自由パラメータの周波数パラメータで, m_0 は

「1000 Hz は 1000 メル」という制約から導かれる式

$$m_0 = \frac{1000}{\ln\left(\frac{1000 \text{ Hz}}{f_0} + 1\right)} \quad (6)$$

から得られる。メル周波数に変換されたの効果音のスペクトラムを離散コサイン変換したのち、逆フーリエ変換（ケプストラム）して出力する。これが効果音の MFCC である。MFCC は通常、実数を成分にもつ（再生時間に関する窓関数の窓数）×（周波数のサンプリング数の 2 倍）次元の行列として表現される。以下が今回使用した MFCC の仕様である。MFCC を算出するにあたり、効果音はサンプリングレート 0Hz から 44,100Hz の WAV 形式に変換したファイルを入力とした。周波数のサンプルをとるファイルの再生時間の分割数はデフォルトの 20 とした。MFCC の実装は marcyas^(注4) の bextract 関数を使用した。1 つの効果音に関して、時間に関する窓関数ごとに、サンプルとして取られる 13 個の周波数の平均値と分散の計 26 次元のベクトルとして算出される。全窓関数の個数だけ、このベクトルは出力されるため、(全窓関数の個数)×26 次元の行列として出力される。窓関数全体の平均値と分散をそれぞれ求め、計 52 次元のベクトル $\mathbf{p}_i$ が得られる。通常 MFCC の第 1, 第 14, 第 27, 第 40 成分は他の成分の値に比べて 10 倍前後の大きな値をとる。これにより、他の 48 成分よりもこの 4 つの成分の影響がはるかに大きくなってしまふ。このため、第 1, 第 14, 第 27, 第 40 成分を省いた 48 次元のベクトル $\mathbf{t}_i$ を特徴ベクトルとした。

4.3.3 効果音の類似度

以下効果音間の類似度について説明する。効果音間の類似度として、一般的には使用される効果音の特徴ベクトル間のコサインを使用した。各効果音の特徴ベクトルを要素 t_i の集合を S とする。 $\mathbf{t}_i$ と $\mathbf{t}_j$ のコサインは以下の通りである。

$$\text{sim}(\mathbf{t}_i, \mathbf{t}_j) = \frac{\mathbf{t}_i \cdot \mathbf{t}_j}{|\mathbf{t}_i| \cdot |\mathbf{t}_j|} \quad (7)$$

MFCC による特徴ベクトルのコサインにより算出された効果音間の類似度 $\text{sim}(\mathbf{t}_i, \mathbf{t}_j)$ の値域は $0 < \text{sim}(\mathbf{t}_i, \mathbf{t}_j) \leq 1$ ($i, j \in I$) である。

上記の手法により配置される効果音の配置例を図 8 に示す。この座標の単位は 1 メートルである。

4.4 効果音オブジェクトの配色

仮想空間内にユーザが平行移動し、効果音オブジェクトを検索するための移動平面を配置する。移動平面上に効果音オブジェクトは色のついた効果音オブジェクトとして配置される(図 7)。各効果音オブジェクトはカテゴリーごとに色分けして表示される。効果音の類似性が高いカテゴリーが似た色であるように以下のように配色する。各カテゴリーごとに MFCC による特徴ベクトルの平均をとる。この特徴ベクトルを各行とする行列に対して主成分分析を行い、3次元に圧縮した。この 3次元ベクトルの各成分ごとに値が小さい順に 1 位から 20 位までの順位づけを行う。カテゴリーの数 20 で各順位を割って正規化す

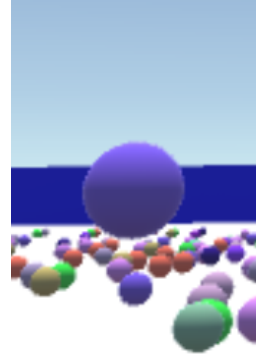


図 9 効果音再生中の効果音オブジェクト。

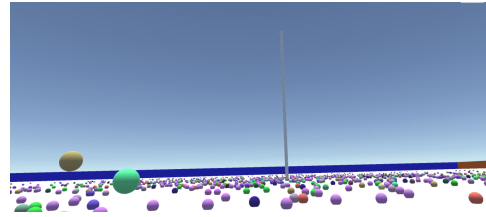


図 10 ランダム再生モード。

る。この正規化された 0.05 から 1.0 までの値にそれぞれ 255 をかけた値を RGB カラーの (R, G, B) の各成分の値とする。

4.5 効果音オブジェクトの再生時の挙動と再生モード

効果音再生中のみ効果音オブジェクトはその効果音の再生時のみに垂直方向に往復運動し、その大きさの伸縮を繰り返す(図 9)。効果音再生中にレーザー接触時にトリガーが押されるとその再生が停止する。

常時ランダムに選択された 5 つの効果音が再生されている。ユーザは、再生された効果音の方向や音の大小により検索意図の効果音の大きな位置を把握する。再生している効果音オブジェクトの状態は、レーザーポイントによる効果音の再生の状態と同じである。この状態をランダム再生モードと呼ぶことにする(図 10)。

レーザーポイントにより他の効果音オブジェクトが再生されている場合は、ランダムに再生されている効果音オブジェクトの再生は全て止まる。ユーザは同時に複数の効果音を再生可能である。このモードをポイント再生モードと呼ぶ。レーザーポイントにより再生された音が全て再生を停止すると、ランダム再生モードに移行する。

上記の視覚情報と聴覚情報を利用して検索の効率化を図る。

4.6 ユーザの位置把握のためのオブジェクト

まず、前述の移動平面を仮想空間内に配置する。効果音オブジェクト全てをその領域内に含む正方形の境界上に壁を配置する。壁は互いに異なる色をもつ。今回は（青、赤、緑、橙）とした。壁の色により、ユーザは自分がどちらの方向を向いているかを把握する。ユーザはこの正方形領域の中心から検索を開始する。正方形の中心、ユーザの検索開始位置のすぐ横、には

(注4) : marcyas: <http://marsyas.info/doc/manual/marsyas-user/index.html>

カテゴリー	#mp3 ファイル
Alarms	16
Ambience	36
Animals	45
Arms	62
Caricature Cartoon	42
Crowd	18
Door	33
Emergency Vehicle	14
Foley	497
Horror	50
Human Voice	41
Impulsive Sound	104
In House	122
Office	13
SF Fantasy	17
Sports	102
Tools	56
Traffic	120
Water	27
Weather	33
合計	1,448

表 1 YouTube Audio Library の効果音におけるカテゴリ別の諸元

白い円柱が配置されている。ユーザはこの円柱との距離感から自分の位置を大まかに把握する。

5. 実 験

5.1 実験手順

検索対象の効果音を YouTube Audio Library(前述, 2017 年 11 月 11 日時点) から 1,448 ファイルをダウンロードした。諸元を表 1 に示す。

従来システムのインタフェースを YouTube Audio Library のインタフェース (図 2) とする。提案システムのインタフェースを 4 節の仮想世界における効果音検索インタフェース (図 1) とする。

3 節で与えられたタスクを行うために、YouTube Audio Library からカテゴリをランダムに 5 つ選択する。選択されたカテゴリからランダムに 5 つ選択し、各カテゴリからそれぞれ 1 音を選択し検索意図の音とする。被験者は検索意図の音の一つずつ順に検索する。検索の直前に、被験者は検索意図の音の一つ聞き、検索意図の音を提案システムと従来システムのインタフェースそれぞれで検索する。各インタフェース内に検索意図の効果音はインタフェース内に含まれている。被験者が検索意図の効果音を探し当てた時点で検索を終了する。

5.2 実験結果

効果音検索を開始した時間と効果音検索を終了した時間の差分を検索に要した時間とする。5.2 節の実験手順にしたがい 5 音検索し、検索に要した時間をストップウォッチで計測した。被験者は三人で、被験者が検索する 5 音は被験者によらず同じである。提案システムと YouTubeAudioLibrary の検索性能

を、5 つの各音に対する被験者の検索に要した時間の平均とした。1 音あたりの検索する制限時間は 5 分とした。結果は、従来システムを使用した場合の検索に要した時間の平均は 3 分 3 秒であり、提案システムを使用した場合は制限時間内に検索意図の音を発見できなかった。

6. 考 察

6.1 提案システムと従来システムの検索の手がかり

タスクを行った際、制限時間内に検索意図の音を検索できなかった場合、被験者が検索意図に一番近いと思う音を提案システムと従来システムそれぞれに対して記録した。検索意図と記録した音との MFCC による特徴ベクトルの類似度を算出した。また、検索終了後に被験者に対して、インタフェース内の何を手がかりにして検索したかを聞いた。従来システムでは、音の印象から想起されるクエリとカテゴリだった。提案システムでは以下の手がかりが挙げられた。

- ポインタにより再生した効果音の印象とボールの色
- ランダムモードで再生された効果音の印象とボールの色
- 音の印象
- 複数のボールの色と位置関係のパターン

6.2 提案システムの利点と改善点

従来システムのインタフェースは通常一度に再生できる音の数は一つである。それに対し、提案システムの利点は、一度に 10 個以上の音を検索可能な点である。また、検索インタフェースが従来システムより楽しんで検索できるとのコメントが複数見られた。検索性能は、従来システムに及ばなかったが、その要因としては以下の要素が考えられる。

- 効果音オブジェクトの配置
- ランダム再生する効果音の個数
- 検索候補の位置の把握

今回の手法では、MFCC による効果音の類似度を元にしたバネモデルを使用した。被験者から「同じ色のボールが一箇所にクラスターのように分布するような場所がなく、似た音が近くにあると言えない」とのコメントがあった。今回実験で使った効果音の配置がより人間の直感に近い配置であれば、検索性能は向上すると考えられる。

ランダム再生する効果音の個数は今回 5 つだった。しかし、ポインタ再生モードでのポインタによる効果音の再生音数が 10 音を超えても検索に支障をきたすことがないとわかった。また、被験者から「ランダム再生での再生音を頼りにせず、主にポインタでの再生音を頼りにした。」というコメントが複数あった。したがって、ランダム再生モードでの音の再生する音の数を増やすと良いと考えられる。

検索意図の音を検索時間内に見つけられなかった際、一番近いと思った音を実験終了後に指摘してもらったが、いくつかの音は被験者が記憶していた音の位置を思い出すことができず、探し出せなかった。被験者は、「自分と壁や中央の柱との距離感が感覚的に記憶できなかった」ことを要因として挙げていた。この問題を改善するために、ユーザが検索意図の音の候補を検索の際、随時記録できるブックマーク機能が必要であるとわ

かった。他方、「複数のボールの色位置関係を記憶して検索時に記憶していたが、検索終了後に似たパターンが複数あったり、配置のパターンの記憶が検索終了時に曖昧になった。」ことも挙げていた。この問題を解決するために、ボールが配置されている平面にグリッドを設けるなどして、ボール間の位置関係の把握を容易にする必要があるとわかった。

7. 結 論

我々は、VR デバイスを用いた効果音検索インタフェースを提案する。検索対象である効果音は仮想世界内のオブジェクトとして表現し、ユーザは仮想世界を探索して求める効果音を発見する。効果音は、MFCC を用いて特徴ベクトルとして表現する。特徴ベクトルの類似度を考慮したバネモデルによって類似する効果音を表すオブジェクトを近くに配置されるようにする。YouTube Audio Library の効果音 1,448 個を用いて、従来のリスト表示による検索インタフェースとの比較を行い、提案インタフェースの優位性を示す。今後はさらに負荷の少ない、より効果的な効果音検索インタフェースの構築に取り組む。

謝 辞

本研究の一部は JSPS 科学研究費助成事業 JP15H01718, JP16H02906, JP16H01756 による助成を受けたものです。ここに記して謝意を表します。

文 献

- [1] Richard. A Bolt. Put-that-there: Voice and gesture at the graphics interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques (ACM 1980)*, page 262270, 1980.
- [2] Nick Craswell. *Mean Reciprocal Rank*, pages 1703–1703. Springer US, Boston, MA, 2009.
- [3] Sally Jo Cunningham, Zhiyao Duan, Xiao Hu, and Douglas Turnbull. Proceedings of the 18th international society for music information retrieval conference, ISMIR 2017, suzhou, china, october 23–27. 2017.
- [4] Steven B. Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366, 1980.
- [5] Brian Gygi and Charles S. Kidd, Gary R. and Watson. Similarity and categorization of environmental sounds. *Perception & Psychophysics*, 69(6):839–855, 2007.
- [6] Colette Jacobsen, Susan Kaugher, Farshad Fotouhi, Aashima Narula, and Nitán Jadhav. A distributed web-based virtual reality interface to database systems. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI 1998)*, pages 267–269, 1998.
- [7] Daniel Jarrett. Spherical microphone array processing for acoustic parameter estimation and signal enhancement. 2013.
- [8] Andrew Johnson and Farshad Fotouhi. Sandbox: Scientists assessing necessary data based on experimentation. *Interactions*, 2(3):34–45, 1995.
- [9] Kahori Okamoto, Ryosuke Yamanishi, and Mitsunori Matsushita. Sound-effects exploratory retrieval system based on various aspects. *IEEJ Transactions on Electronics, Information and Systems*, 136(12):1712–1720, 2016.
- [10] Randy Pausch, Dennis Proffitt, and George Williams. Quantifying immersion in virtual reality. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 1997)*, pages 13–18, 1997.
- [11] Karol J. Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia (MM 2015)*, pages 1015–1018, 2015.
- [12] Markus Schedl and Peter Knees. Personalization in multimodal music retrieval. In *Proceedings of the 9th International Conference on Adaptive Multimedia Retrieval: Large-scale Multimedia Retrieval and Evaluation (AMR 2011)*, pages 58–71, 2011.
- [13] Mette Skov and Marianne Lykke. Unlocking radio broadcasts: User needs in sound retrieval. In *Proceedings of the 4th Information Interaction in Context Symposium (IIiX 2012)*, pages 298–301, 2012.
- [14] Douglas Turnbull, Luke Barrington, David Torres, and Gert Lanckriet. Towards musical query-by-semantic-description using the cal500 data set. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007)*, pages 439–446, 2007.
- [15] Douglas Turnbull, Luke Barrington, David A. Torres, and Gert R. G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE Trans. Audio, Speech & Language Processing*, 16(2):467–476, 2008.
- [16] Bingjun Zhang, Qiaoliang Xiang, Ye Wang, and Jialie Shen. Compositemap: A novel music similarity measure for personalized multimodal music search. In *Proceedings of the 17th ACM International Conference on Multimedia (MM 2009)*, pages 973–974, 2009.
- [17] Tong Zhang and C-CJ Kuo. Classification and retrieval of sound effects in audiovisual data management. In *Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, pages 730–734, 1999.
- [18] 山西良典 岡本香帆里, 松下光範. 大量の効果音に対する効率的検索に向けての一検討. In *Web インテリジェンスとインタラクション研究会 (WI2 2016)*, 2016.
- [19] 松下光範 岡本香帆里, 山西良典. 複数観点に基づく探索的效果音検索システム: Serva の開発とユーザ観察. In *データ工学と情報マネジメントに関するフォーラム (DEIM 2016)*, 2016.
- [20] 駒谷和範 尾形哲也 奥乃博 清水敬太, 北原鉄朗. Onomatree: 擬音語と木構造を併用した環境音検索インターフェース. In *情報処理学会全国大会講演論文集 (IPJSJ 2007)*, 2007.