無音動画に対する効果音貼付けシステムの試作

鈴木 喜也† 岡部 誠†* 尾内 理紀夫†

†電気通信大学 〒182-8585 東京都調布市調布ケ丘 1-5-1
*科学技術振興機構 さきがけ

E-mail: †suzuki@onailab.com,

あらまし 動画製作における効果音の貼付け作業は、効果音の試聴とタイミングの微調整の繰り返しで行われている。この繰り返し作業は多くの時間がかかるため、動画製作者の負担となっている。そこで動画に対する効果音合成を効率化することを目標として、既存の動画内の音を無音動画に貼り付けるシステムの試作を行った。システムはオプティカルフローを用いることにより、動画内のオブジェクトの動きを特徴量として取得する。このシステムを用いて幾つかの動画で効果音が適切に合成されるか実験を行った。

キーワード 動画，特徴量抽出，オプティカルフロー，効果音合成

Prototype System for Sound Effect Synthesis Using Video Example

Nobuya SUZUKI† Makoto OKABE†* and Rikio ONAI†

† University of Electro-Communications 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan
*JST PRESTO

E-mail: †suzuki@onailab.com,

Abstract In video production, the work of sound effect synthesis is listening to the sound effect and fine-tuning the timing of repeat. These work take much time and the efforts are large. Our aim is reducing effort in video production by automation of sound effect synthesis for soundless video. In this paper, we make a prototype of this system. In this system, we extract the feature values from existing video and soundless video using optical flow. We did the experiments of synthesize the sound effect properly for some videos.

Keyword Videos, Feature Extraction, Optical Flow, Sound Effect Synthesis

1. はじめに

動画に対する効果音の合成は動画製作においてほぼ必ず通らなければならない工程であり、同時に動画の質を左右する大切な工程でもある。近年、ニコニコ動画や YouTube といった動画コンテンツの普及によりアマチュアの動画製作者が日々増え続けている。それと同時に動画製作を支援するツールもフリーウェア、シェアウェア問わず増加している。しかし画像編集に重点をおいたソフトウェアは増加しているが、効果音の合成に重点をおいたソフトウェアは少なく、効果音の合成にかかる労力は未だ大きいままというのが現状である。現代の映画やアニメといった動画製作の場において、音の合成をそれ専門の企業に委託しているということが、音の合成にかかる労力の大きさと難しさを表していると言える。

動画製作における効果音合成の工程を説明する。製作者はまず効果音素材を動画内で使う形に伸縮する。次に効果音を鳴らし始めるタイミングと最も音が大きくなるタイミングの調整を行う。これらの作業は現在、何度も試聴と再編集を繰り返すことで行われている。この作業は単純作業でありながら画面を注視し続けなければならず、動画製作者に時間と疲労の両面で大きな負担をかけている。このような現状に対し、我々は効果音合成の工程を効率化することによる動画製作における負担の軽減を考えている。

本論文では、効果音貼付けシステムの試作を行った。図 1 はシステムのイメージ図である。試作したシステムは、類似した動きを行う物体は似た音を発するという観点に基づき音の貼付けを行う。既存の音付き動画と無音動画を用意し、それぞれの動画から特徴量を抽出する。無音動画の特徴量と類似した部分を音付き動画から探し、その音を切抜き、貼り付けることによって無音動画の音データを生成する。

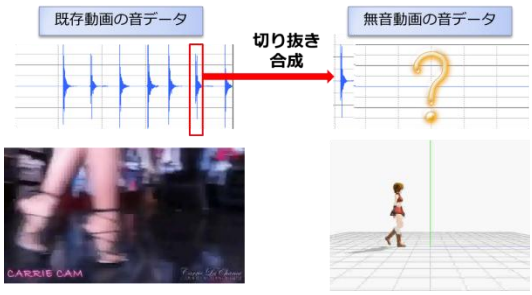

図 1：システムイメージ

以下、2 章では効果音の自動合成に関する先行研究を示し、3 章では本論文において試作したシステムの概要と内部で使用しているアルゴリズムについて述べる。4 章ではシステムを用いた実験結果を示し、5 章でまとめと今後の課題について述べる。

2. 先行研究

動画に対して効果音を自動合成する研究として、Sound-By-Numbers[1]があげられる。Sound-Bu-Numbers は Image Analogies[2]における Texture-by-Numbers の考え方に基づいて音の合成を行う。この研究で行われている効果音の自動合成は次のような手順で行われる。

① まず事前に音付き動画内における物体の軌跡データを用意し、同様に合成対象である無音動画内の物体の軌跡データを用意する。

② ①で用意した各軌跡データを、動きの変化する点ごとに断片化し、各断片においてマッチングを取る。無音動画の各断片に対し、マッチした音付き動画の断片部分の音を伸縮し、合成する。

この手法は事前にオブジェクトの軌跡データを用意する必要があるため、ユーザーの負担が大きいと考えられる。

また、教師データを用いた効果音合成の研究として Animating Fire with Sound[3]や Toward High-Quality Modal Contact Sound[4]があげられるが、これらは物理演算を用いて質の高い効果音を合成するものであるため、多大な時間的コストが掛かる。

3. 効果音貼付けシステム

ここでは試作した効果音貼付けシステムについて説明する。3.1 節ではシステムの概要について説明し、3.2、3.3 節ではシステムで使用している特徴量およびそのマッチング方法について説明する。

3.1. システム概要

システムにおける処理のおおまかな流れを説明する（図 2）。

① ユーザーが事前に作成した無音動画と教師データとなる既存の音付き動画をシステムに与える。

② システムは、入力された双方の動画内に存在するオブジェクトの動きを特徴量として抽出する。例えばボールがバウンドする動画を与えた場合、ボールの落下から静止までの連続した動きが抽出される。

③ 次に無音動画から得た画像特徴量と既存の音付き動画から得た画像特徴量のマッチングをとり、類似した部分の対応付けを行う。

④ ③によるマッチングの結果を基に、無音動画の各部分に対応した音の貼付けを行う。

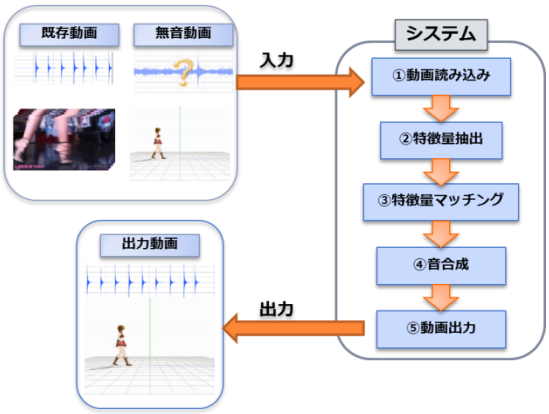

図 2：システム概要図

3.2. 動画からの特徴量抽出

システムは入力された双方の動画から特徴量を抽出し、動画内オブジェクトの動きを得る。ここではその具体的な処理について説明する(図 3)。

① 各動画の 1 フレーム分の画像に注目し、画像内の各ピクセルにおけるオプティカルフローを算出する。

② ①で算出した各オプティカルフローの値の平均値を取る。この平均値は、その画像のもつオプティカルフローとして扱う。

③ ①,②を各動画の全画像に対して適用し、時間軸に対するオプティカルフローの値の推移を得る。

④ ③で得たオプティカルフローのデータから物体のオプティカルフローの変化量の推移を算出する。ある時間 t におけるオプティカルフローの値を $F_v(t)$ とすると、時間 t における変化量を $F_a(t)$ は次式から求められる。

$$F_a(t) = F_v(t+1) - F_v(t) \quad (1)$$

以上の手順で得たオプティカルフローとその変化量の値の推移をその動画の持つ特徴量としてマ

ッチングに用いる。

オプティカルフローのデータ、およびその変化量は x,y 方向の値を持つ 2 次元ベクトルとして表される。

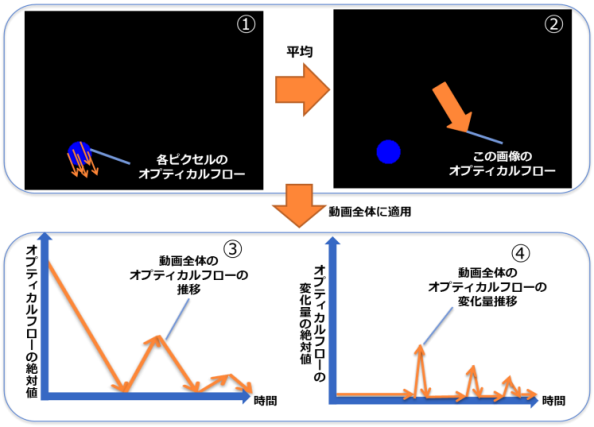

図 3：動画からの特徴量抽出

3.3. 視点移動の除去

3.2 節において抽出したオプティカルフローには、動画内におけるカメラの移動量が含まれる。図 4(a)に示すように、視点の移動によるオプティカルフローはフレーム画像全体に現れる。システムは RANSAC 法を用いてこのオプティカルフローの除去を行う。図 4 を例に具体的な処理手順を示す。

① フレーム画像内からランダムに T 個のピクセル(候補点)を選択する(図 4b)。

② 各候補点のオプティカルフローにおける X,Y 成分と候補点でない各ピクセルの差を取り、新しい 2 次元ベクトルを作る。

③ ②において作成した 2 次元ベクトルの L2 ノルムの値が ε 以下ならばそのベクトルの元となる候補点の得票値に 1 を加算する。

④ ②,③を候補点でない全ピクセルに対して適用し、最も得票値が高かったピクセルのもつオプティカルフローを視点のもつオプティカルフローとして扱う(図 4c)。

⑤ ④より得られた視点のオプティカルフローの X,Y 成分をフレーム画像内の全ピクセルのオプティカルフローから引く(図 4d)。

ただし、T, ε は任意である。現在のシステムでは暫定的に T=30, ε =0.4 としている。

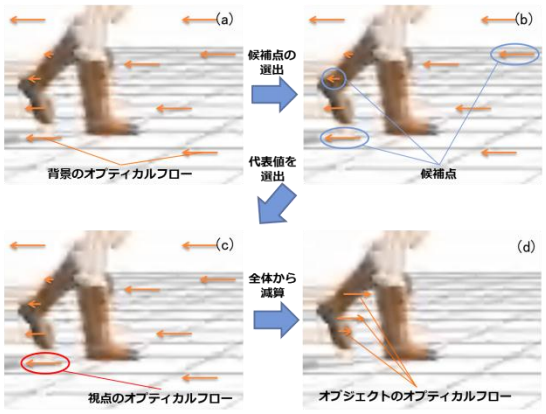

図 4：RANSAC 法による視点移動の除去

3.4. 特徴量マッチング

特徴量同士のマッチングについて説明する。ここでは、マッチングの特徴量として 3.2 節で得たオプティカルフローのデータを用いる場合について述べる。以下にその手順を示す（図 5)。

① マッチングの前準備として、各オプティカルフローの値（2 次元ベクトル）をデータ内のノルムの最大値が 1 になるように正規化する。

② 無音動画の特徴量データを時間軸に沿って細かい切り取り窓に分割する。切り取り窓内には I個のオプティカルフローのデータが含まれるものとする。また、時間軸上で各切り取り窓の前にあるデータK個を近傍点として抽出する。

③ ②で取得した各切り取り窓とその近傍点に対し、音付き動画の特徴量データから最も類似した部分の検索を行う。類似性は切り取り窓および近傍点内のデータと、マッチング対象のデータのユークリッド距離から判断する。

無音動画、音付き動画におけるk個目の近傍点のデータを

$(N_{Mx}(k), N_{My}(\mathrm{k})), (N_{Tx}(k), N_{Ty}(\mathrm{k}))$

切り取り窓内のi番目のデータを

$(W_{Mx}(i), W_{My}(\mathrm{i})), (W_{Tx}(i), W_{\mathrm{Ty}}(\mathrm{i}))$

とすると、類似度 S は次式で求められる。

$$S = \sum_{k=0}^{K} \sqrt{(N_{Mx}(k) - N_{Tx}(k))^2 + (N_{My}(k) - N_{Ty}(k))^2} + \sum_{i=0}^{I} \sqrt{(W_{Mx}(i) - W_{Tx}(i))^2 + (W_{My}(i) - W_{Ty}(i))^2} \quad (2)$$

Sが小さいデータほど類似しているものとする。

各切り取り窓だけでなく近傍の特徴量も考慮してマッチングを行うことにより、連続性のあるマッチング結果を得る。

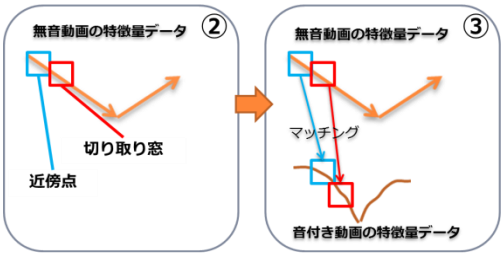

図 5：特徴量マッチング

3.5. 音の貼付け

音の貼付け処理は 3.3 節で行ったマッチングの結果に基づいて行う。

マッチングで得た無音動画の各切り取り窓に対応する音付き動画の音を切り抜き、無音動画の該当部分に合成する。この処理を無音動画の全切り取り窓に対して行うことで、無音動画に適した音を生成する（図 6)。切り取り窓単位で音の切り貼りを行うことにより、ある程度連続した音の合成を行うことができる。

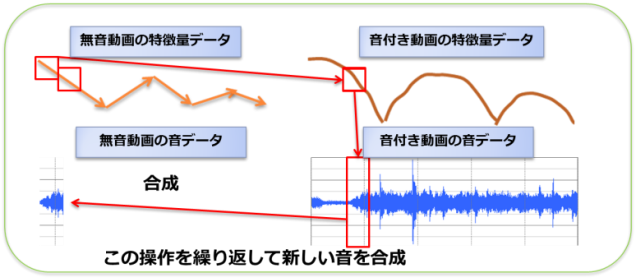

図 6：音の貼付け処理

3.6. BJ アルゴリズム

3.5 節のように単純に波形の切り貼りを行った場合、環境音や車の走行音のような連続した音の場合はつなぎ目に違和感が生じることがある。これの解消のために SoundTexture の合成手法である BJ アルゴリズム[5]を用いる。

図 7 に BJ アルゴリズムの概要を示す。アルゴリズムはまず入力された信号データをウェーブレット変換し、低周波成分を根とする MRA ツリーを作る。この MRA ツリーと同サイズの新しい MRA ツリーを低周波成分から再構築することにより、つなぎ目が自然な新しい信号データを得る。

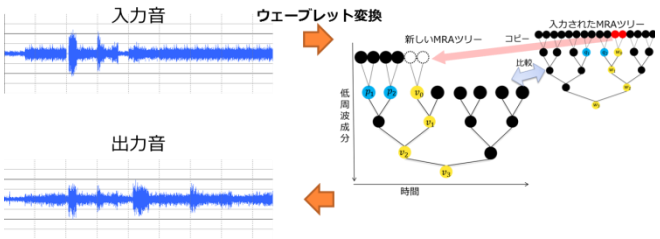

図 7：BJ アルゴリズムによる音の変化

4. 実験

システムの評価実験として、ボールがバウンド音、人の足音、車の走行音の 3 種類の音に関して音を貼り付ける実験を行った。その出力から、自動合成でどの程度適切な音が出力されるかを見た。

4.1. ボールのバウンド音

ボールがバウンドする動画を二種類作成した。その片方に手動でバウンド音を合成して音付き動画とし、もう一方の無音動画に対してシステムを用いて音を貼りつける実験を行った。このとき、切り取り窓に含まれるデータ数は 4 個、抽出する近傍点の個数は 3 個とした。特徴量にはオプティカルフローの変化量(加速度)を用いた。

図 8 に入力したそれぞれの動画から得られた特徴量と音データの波形を示す。特徴量は各フレーム画像における加速度の値をスプライン曲線で結んだものを表示している。出力された音データはボールが下に跳ね返るタイミングでバウンド音が鳴る、適切な形の音を得ることができた。

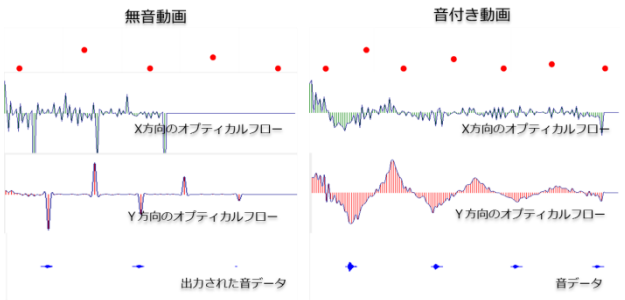

図 8：ボールの特徴量と出力された音

4.2. 人の足音

ニコニコ動画および Youtube から人が右方向に歩く無音動画と音付き動画を 1 種類ずつ用意した。それぞれの動画を前足のみが映るように加工し、足音を貼り付ける実験を行った。特徴量にはオプティカルフローの値(速度)を用い、切り取り窓に含まれるデータ数は 6 個、近傍点として抽出するデータ数は 4 個とした。

図 9 にそれぞれの動画から得られた特徴量と出力された音の波形を示す。無音動画の特徴量も足が前に出るタイミングで X 方向のオプティカルフローが強く現れる、直感に合うものが得られた。また、出力された音は足が地面につくタイミングと合っており適切な場所に音が貼り付けられたと言える。

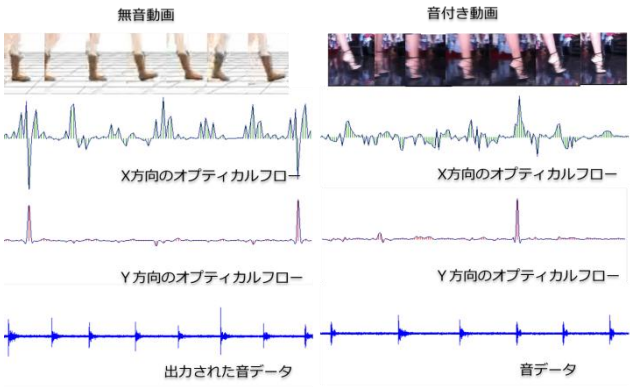

図 9：歩行動画の特徴量と出力音

4.3. 車の走行音

YouTube から一般車がカーブをドリフトしてくる無音動画と、F1 の車が観客席前を高速で通過する音付き動画を用意した。前者の動画内の車に対し、新たに F1 の走行音を貼り付ける実験を行った。

図 10 にそれぞれの動画から得られた特徴量と音データの波形を示す。特徴量にはオプティカルフローの値(速度)を用いた。また、切り取り窓に含まれるデータ数は 6 個、近傍点として抽出するデータ数は 4 個とした。

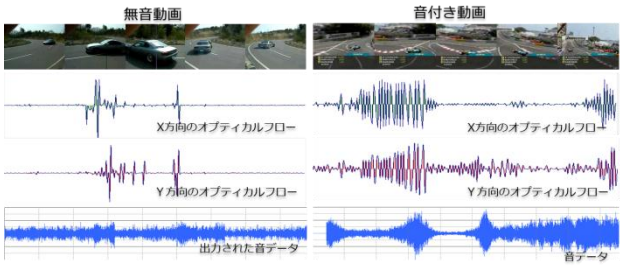

図 10：車の走行動画の特徴量と出力された音

出力される音の理想型は、車がカメラに近づくにつれて走行音が大きく鳴り、遠いところにいる場合は一定の走行音がフェードアウトしながら鳴るというものである。しかし出力された音は車の動きに反し、常にほぼ均質な走行音が鳴るものとなった。

この原因として動画内の物体が高速であるためオブジェクトの特徴量がうまく取れないことや物体の接近、離脱といった変化がオプティカルフローでは検出できないといったことが考えられる。

5. まとめと今後の課題

オプティカルフローを特徴量とし、無音動画の特徴量と類似した特徴量をもつ部分を音付き動画から検索し、その音を貼付けるシステムの試作を行った。ボールのバウンド、および人間の足音の例では音が適切な箇所に合成されることを確認した。

現在のシステムは 4.3 節で述べたような物体が接近する動画や、物体の爆発といった複雑な特徴量を持つ動画に対応することができない。また、複数のオブジェクトが存在する動画を扱うためには 4.2 節 図 9 のようにユーザーが手動で動画の必要部分を切り抜く必要がある。今後はトラッキングの併用やオプティカルフロー以外の特徴量の考慮、クラスタリングを用いた領域分割により、扱える動画の幅を広げていく。

謝辞

本研究の一部は、日本学術振興会 学術研究助成基金助成金(基盤研究(C)23500114)の交付を受けたものである。

参 考 文 献

[1] CARDLE, M., BROOKS, S., BAR-JOSEPH, Z., AND ROBINSON, P. 2003. : Sound-by-Numbers: Motion-Driven Sound Synthesis. In 2003 ACM SIGGRAPH / Eurographics Symp. on Computer Animatio

[2] Hertzman, A., Javobs, C., Oliver, N., Curless, B., and Salesin, D. H. 2001. Image Analogies. In *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, 327-340

[3] Jeffrey N. Chadwick Doug L. James. : Animating Fire with Sound. In 2011 ACM SIGGRAPH

[4] Changxi Zheng. Doug L. James : Toward high-quality modal contact sound. In 2011 ACM SIGGRAPH

[5] Shlomo Dubnov, Ziv Bar-Joseph, Ran El-Yaniv, Dani Lischinski, Michael Werman : Synthesizing Sound Textures through Wavelet Tree Learning . Published in Journal IEEE Computer Graphics and Applications Volume 22 Issue 4, July 2002