平 成 27 年 度　卒 業 研 究 論 文

題目　アニメ動画の音声とキャスト情報を用いた声優認識に関する研究

指導教員　服部 峻

提出者　室蘭工業大学 情報電子工学系学科

氏　名　榮田　基希

学籍番号　12024022

提出年月日　平成 28 年 2 月 12 日

目次

図目次

表目次

第 1 章

序論

近年日本には様々な娯楽メディアがあり，我々はそれらを普段の生活の中で目や耳にする機会が多くなっている．情報通信機器の普及で多くの人が，パソコンやモバイル端末などの機器で番組や動画の視聴，ゲームなどが今では手軽にできる．このような娯楽に触れる機会が多くなって来ると，どこかで聞いたことがある音声が流れて来ることがある．

その音声の発生源がアニメ動画の場合，誰の音声であるかを知るためには，エンディングのスタッフロールまで飛ばしたり，Web で作品のタイトル名やキャラクター名で検索したりするなどの余計な労力を掛ける必要が出て来る．例えば，あるユーザが適当なアニメを視聴していた際，そのアニメの中に出て来たキャラクター A の音声がユーザの聞いたことのある音声であったとする．そこで，そのユーザがキャラクター A の声優について調べようとするならば，エンディングまで飛ばしたり，アニメタイトルやキャラクター名で Web 検索して，そのアニメの公式サイトや Wikipedia などを探そうとするであろう．しかし，知りたいキャラクター A が作中の目立たない配役だった場合，Web で検索を掛けても中々出て来ないことも考えられる．また，主要なキャラクターではない場合，キャラクター名を記憶していない可能性もあり，エンディングのスタッフロールが流れてもわからないだろう．その上，脇役であった場合，スタッフロールには男の子 B，男の子 C というようにキャラクター名を不明瞭に表記していることもあり，どの場面に出て来たキャラクターかわからないことも考えられる．

そこで本研究では，アニメ視聴中に音声が流れたらリアルタイムに声優名を自動的にアプリケーション内の画面に表示するシステムを提案する．アニメのキャラクターと声優名を関連付けて映像として表示することができる，ユーザ側に「このキャラクターはこの声優だ」と強いイメージを植えつけやすいシステムになると考えた．このシステムを実現するにあたって，データベースにあらかじめ登録してある各声優の音声波形データと視聴中のアニメ動画から流れる音声波形データを使って類似度の計算を行い声優を判定する．また，本研究では YouTube やニコニコ動画など動画サイトで視聴中のアニメのタイトルが特定されて既にわかっている状態を想定する．アニメのタイトルが特定されていることで，そのタイトルに基づいて Web 検索されたキャスト情報で声優を絞り込み，声優認識の精度が上がると考える．まとめると，音声で声優認識するだけではなく，視聴中のアニメ動画が持つアニメタイトルを用いて，Web からキャスト情報を自動で取得してデータベースにある声優を絞り込むようにす

ることで精度が上がると考える．最終的には図 1.1 のように，認識した声優名を画面に表示するだけでなく，その声優のプロフィール情報や他の出演作品の情報などを余計な労力を掛けずに提供できるシステムを目指していく．

図 1.1　最終的なシステムイメージ図

第 2 章

関連研究

人，動画像の認識についての関連研究について述べる．人物認識の個人を特定する研究によく用いられる人物の特徴として声紋，指紋，掌紋，虹彩，DNA，顔認識などが挙げられる．これらの各人の固有の生体情報から個人を認識するのは重要な要素技術の一つであると考えられる．また，その中でも認識する対象が実際の人物であったり，動画像であったり音声だけであったり，それらからどのような情報が得られるかによって認識の方法が変わってくる．音声については，話者認識の手法として混合正規分布（GMM）により個人毎の音声の分布を表現する方法 [1] や音声パラメータ系列のモデル化手法として隠れマルコフモデル（HMM）を用いる方法がある [2]．顔画像については代表的なものに特徴量抽出に基づく手法などがある [3]．

人を対象にしたものではなく動画像を認識する研究についても紹介する．画像の認識において Histgrams of Oriented Gradient （HOG）の特徴量及び SVM を用いた判別器を利用したキャラクター位置の検出をする手法がある [4]．また，映像の認識をするために可視化情報に含まれる固有パターン（反射光強度，色相，色成分固有ベクトル）の 3 つの要素を用いて認識する手法がある [5]．

本研究において，アニメ動画を対象にして声優を認識することを試みる．前述で記載した様々な認識手法がある中，声優認識する際の個人特有の生体情報として音声を用いる．本研究では，多数の人の音声特徴を登録しておき誰がいるかを判定する不特定話者認識を採用する [6]．音声から Android 標準 API の Visualizer [7] を用いて音声波形の特徴を抽出して個人認識する．また，対象がアニメ動画であることから元々情報としてあるアニメタイトルテキスト情報を用いて Web 上からキャスト情報を取得してくる．キャスト情報を利用することで多数の声優の人の音声特徴を持つデータベースの人数の絞り込みが可能になり精度が向上することが考えられる．

第 3 章

提案システム

この項目では，提案手法について述べる．

3.1　システム概要

アニメ動画に流れる音声から声優名を認識するため，声優に限定しない一般の話者認識や声紋による個人認証などの従来研究 [8–10] を参考にして，それぞれの声優の声の特徴には音声波形の数値の軌跡が異なっているという仮説を立てた．本研究における声優認識システムは図 3.1 に示す処理を繰り返すことで声優名を認識する．提案システムでは，アニメ動画から流れる音声データを取得して波形表示するのに，Android 標準 API の Visualizer [7] を用いる．Visualizer とは音声の可視化のことであり，音声波形を表示するラインの頂点座標は，左上を基点とする Android 端末上の座標系で表されている．

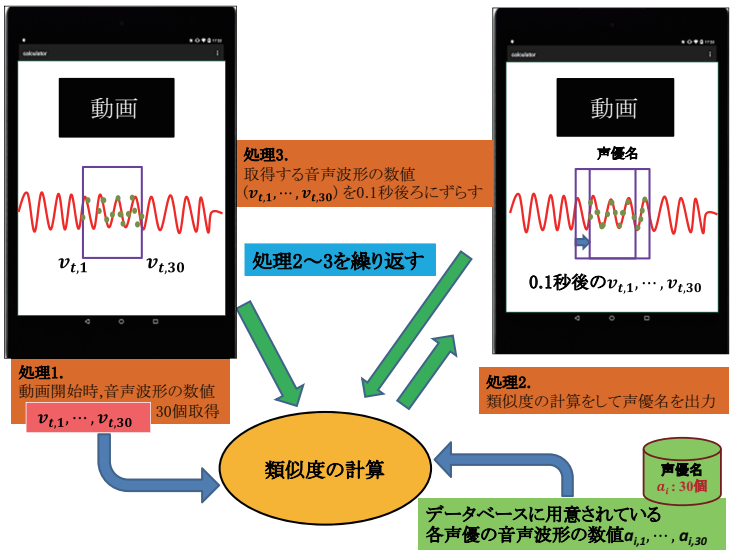

図 3.1　提案システムの処理の流れ

処理 1 では Android 端末で動画を流して音声波形を表示する．再生位置 t 秒において，新しく生成される音声波形の数値（以下，$\boldsymbol{v_t}$）を約 0.1 秒ごとに 30 個取得する．次の処理 2 で，

$\boldsymbol{v_t}$ とあらかじめデータベースに用意されている各声優 i の音声波形の数値（以下，$\boldsymbol{a_i}$）30 個を使って類似度の計算を行って，一番類似度の高かった声優名を画面に出力する．最後に処理 3 で $\boldsymbol{v_t}$ に格納されていた一番古い音声波形の数値を取り出し，約 0.1 秒後の次の再生位置で出て来る新しい数値を格納していく．以後，処理 2 と処理 3 を繰り返す．

3.2　声優データベース

本節では前述に記述しているデータベースの詳細について説明する．データベースに入っている要素を図 3.2 に示す．中身には，1 列目に声優名，2 列目以降には音声波形の数値 $\boldsymbol{a_i}$ がある．今回は 40 人分の声優データを用意した．つまり，40 人分の声優名と 40 人分の音声波形の数値 $\boldsymbol{a_i}$ が 30 個ある．この $\boldsymbol{a_i}$ は，微妙な誤差はあるが約 0.1 秒毎に記録したものである．よって 1 人につき約 3 秒分の数値が用意されている．データベースに入っている $\boldsymbol{a_i}$ は正規化されていない．

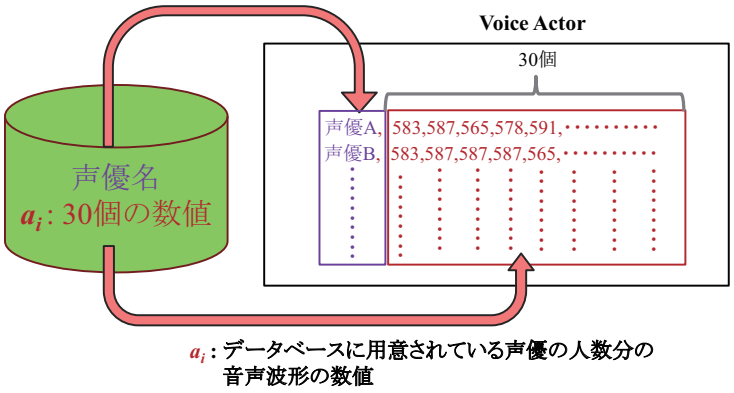

図 3.2　声優データベースの要素

3.3　音声波形の数値の正規化

類似度の計算をするにあたって，図 3.2 のデータベースに入っている音声波形の数値や視聴中のアニメ動画から得られた音声波形の数値は Android 端末上の座標であるため，0 を中心に振動する音声波形へと正規化する処理を行う．動画再生時に一番初めに生成される音声波形の数値（以下，startPoint）が基軸になると考えられ，この startPoint を用いて正規化を行った．

$$\boldsymbol{v_t} = (v_{t,1} - \mathrm{startPoint}, \cdots, v_{t,30} - \mathrm{startPoint})$$

$$\boldsymbol{a_i} = (a_{i,1} - \mathrm{startPoint}, \cdots, a_{i,30} - \mathrm{startPoint})$$

3.4　声優認識アルゴリズム

本節では，声優認識の為の類似度の計算方法，その類似度に基づく判定方法，及び，精度を上げるためのキャスト情報を用いた絞り込み方法について説明していく．

3.4.1　声優認識に用いる類似度の計算

図 3.1 の類似度の計算が行われる処理の詳細について説明する．まず，類似度の計算の為に $\boldsymbol{v_t}$ と $\boldsymbol{a_i}$ の要素を 30 個ずつ用意する．その詳細を図 3.3 に示す．本研究では類似度の定義として，ユークリッド距離とコサイン類似度，相関係数の 3 種類を用いる．$\boldsymbol{v_t}$ を取得して正規化した音声波形の数値を順番ごとに $v_{t,1}, v_{t,2}, \cdots, v_{t,30}$ と置き直すことにする．同様に，$\boldsymbol{a_i}$ を取得して正規化した音声波形の数値を順番ごとに $a_{i,1}, a_{i,2}, \cdots, a_{i,30}$ と置き直すことにする．以下の式で類似度を算出する．

1. ユークリッド距離に基づく類似度

$$\boldsymbol{v_t} = (v_{t,1}, \cdots, v_{t,30}), \quad \boldsymbol{a_i} = (a_{i,1}, \cdots, a_{i,30})$$

$$d(\boldsymbol{v_t}, \boldsymbol{a_i}) = \sqrt{(v_{t,1} - a_{i,1})^2 + \cdots + (v_{t,30} - a_{i,30})^2}$$

$$= \sqrt{\sum_{j=1}^{30} (v_{t,j} - a_{i,j})^2}$$

$$\mathrm{sim}(\boldsymbol{v_t}, \boldsymbol{a_i}) = \frac{1}{d(\boldsymbol{v_t}, \boldsymbol{a_i}) + 1} \tag{3.1}$$

2. コサイン類似度

$$\boldsymbol{v_t} = (v_{t,1}, \cdots, v_{t,30}), \quad \boldsymbol{a_i} = (a_{i,1}, \cdots, a_{i,30})$$

$$\mathrm{sim}(\boldsymbol{v_t}, \boldsymbol{a_i}) = \frac{\sum_{j=1}^{30} v_{t,j} \cdot a_{i,j}}{\sqrt{\sum_{j=1}^{30} v_{t,j}} \sqrt{\sum_{j=1}^{30} a_{i,j}}} \tag{3.2}$$

3. 相関係数

$$\boldsymbol{v_t} = (v_{t,1}, \cdots, v_{t,30}), \quad \boldsymbol{a_i} = (a_{i,1}, \cdots, a_{i,30})$$

$$\mathrm{sim}(\boldsymbol{v_t}, \boldsymbol{a_i}) = \frac{\sum_{j=1}^{30}(v_{t,j} - \overline{v_t})(a_{i,j} - \overline{a_i})}{\sqrt{\sum_{j=1}^{30}(v_{t,j} - \overline{v_t})^2}\sqrt{\sum_{j=1}^{30}(a_{i,j} - \overline{a_i})^2}} \tag{3.3}$$

式 (3.1) から (3.3) のいずれかを用いて，声優データベースに用意されている声優の人数分の類似度が求められる．算出された類似度をそれぞれ比較していき，一番類似度の高い声優が約 0.1 秒の区間毎の声優と判定される．この流れを図 3.3 に示す．しかし例外として，動画が開始された直後の約 3 秒間は $\boldsymbol{v_t}$ の値が 30 個たまりきっていないため類似度の計算はされない．

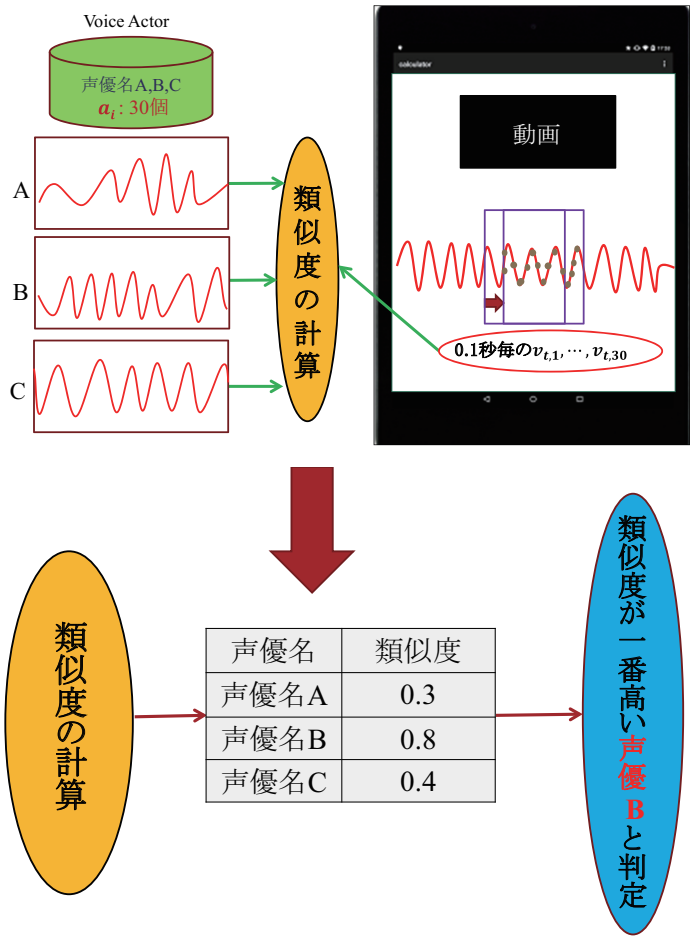

図 3.3　類似度計算と声優の判定処理の流れ

3.4.2　パラメトリック声優認識

前節の方法で声優認識をすると，約 0.1 秒の区間毎に声優名が判定されて出力される．そこで，約 0.1 秒毎に行う類似度計算及びランキングを連続 N 回分まとめてから声優認識し，その N 回の $P\%$ 以上をある声優が 1 位をどの声優よりも多く獲得したら，約 $0.1 \cdot N$ 秒の区間はその声優の音声であると判定されるように定義づける．また，どの声優も N 回中 $P\%$ 以上 1 位を獲れなかった場合には「なし」と判定する．

- N 回 ： 約 0.1 秒毎に行われる声優認識の回数
- $P\%$ ： N 回中何回 1 位を獲れば声優認識の解として採用されるかを定める割合

例として N が 10 回，P が 60% のパラメータの場合のシステムの処理を図 3.4 に示す．図 3.4 を見ると 0.1 秒区間毎の 1 位の回数が，声優 A が 6 回，声優 B が 2 回，声優 C が 2 回と声優認識されている．この例の場合，声優 A が 10 回中で 60% 以上 1 位を獲得していて，どの声優よりも一番多く 1 位を獲得しているので，この 1 秒区間は声優 A であると認識される．

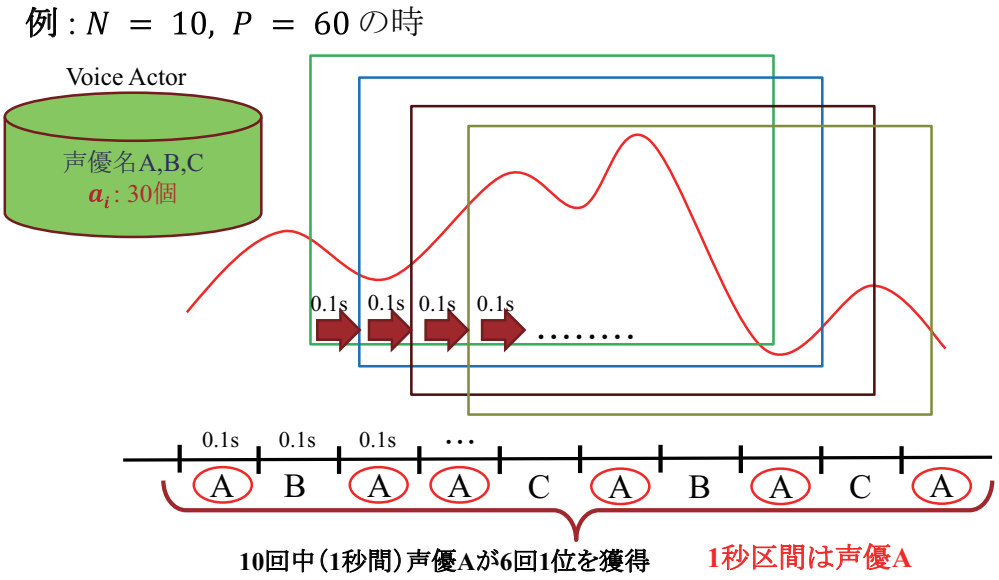

図 3.4　パラメトリック声優認識の処理の例

次に，同じパラメータ設定で 0.1 秒区間毎の 1 位の回数が，声優 A が 4 回，声優 B が 3 回，声優 C が 3 回の例を図 3.5 に示す．この場合，誰も 10 回中 6 割以上 1 位を獲得していないので，この 1 秒区間は誰でもないと判定されて「なし」となる．

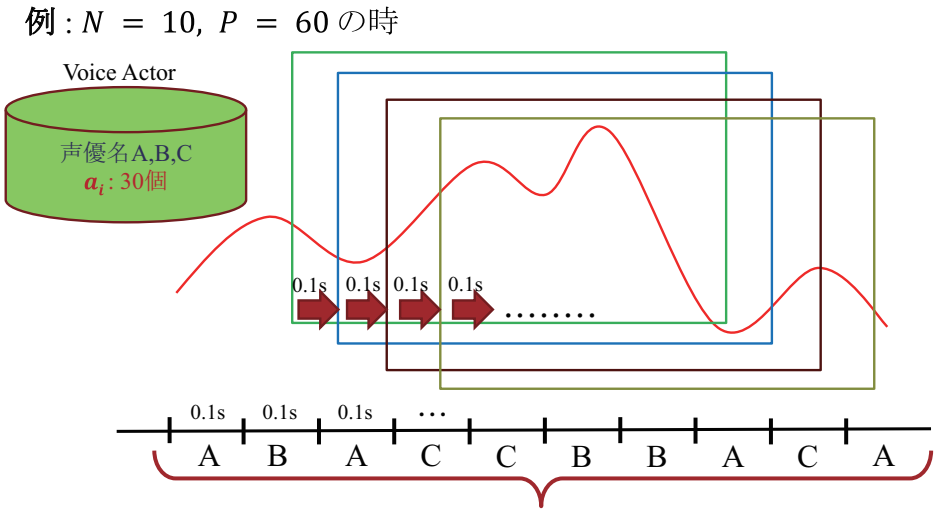

図 3.5　パラメトリック声優認識で判定「なし」となる場合

最後に N が 10 回，P が 40% のパラメータの場合に 0.1 秒区間毎の 1 位の回数が，声優 A が 4 回，声優 B が 4 回，声優 C が 2 回の例を図 3.6 に示す．声優 A と声優 B の両者とも 10 回中 4 割以上 1 位を獲得しており，その回数も同じであるため，優劣が決まらない．そこで，1 位を獲った回数が同じ声優が複数存在した場合，今までは 0.1 秒毎に算出していた類似度を，決定戦まで勝ち進んだ声優に対してのみ各々 10 回分足した合計で比較する．声優 A の場合 0.1 秒毎の類似度を 10 回足すと 2.5315 であり，声優 B の場合 0.1 秒毎の類似度を 10 回足すと 1.2521 である．よって，声優 A の方が声優 B よりも類似度の合計が大きいので，この 1 秒区間は声優 A であると認識される．

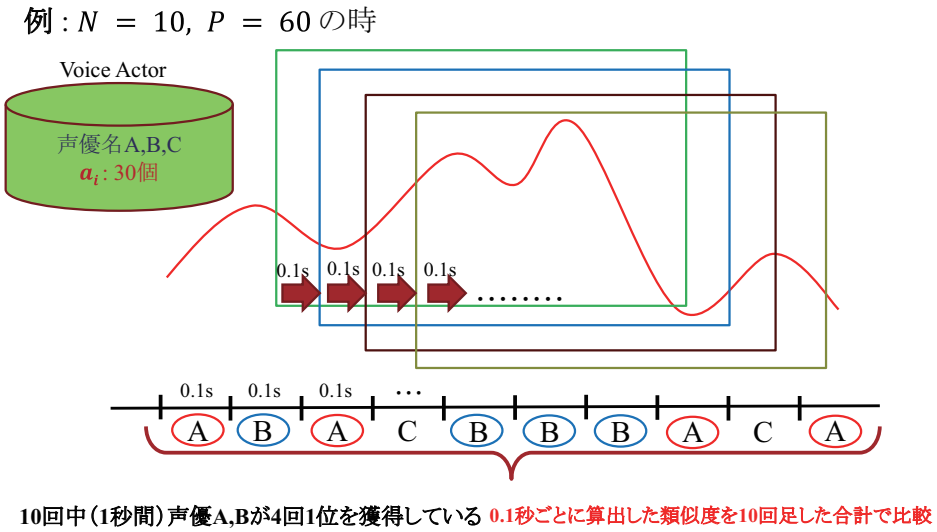

図 3.6　1 位を獲った回数が同じ声優が複数存在した場合

2 つのパラメータ（N 回と P%）を持つパラメトリック声優認識の処理の流れについて，以上の 3 種類の場合分けを含むフローチャートを図 3.7 に示す．

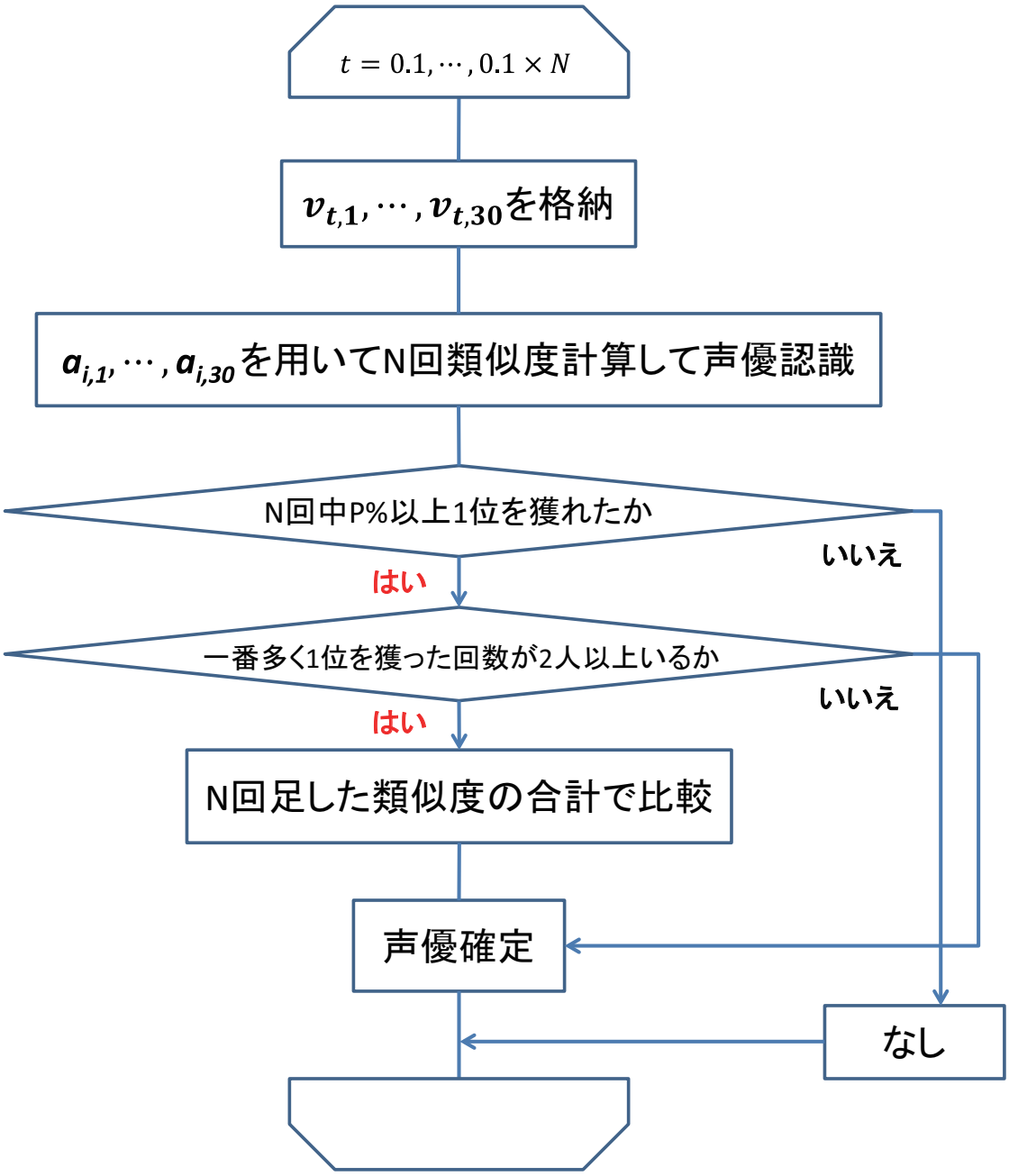

図 3.7　パラメトリック声優認識のフローチャート

3.4.3　キャスト情報による声優データベースの絞り込み

音声波形を用いて声優を認識するだけでなく，視聴中のアニメのタイトルを基に Web 検索して取得したキャスト情報（キャラクター名と声優名のペア）を活用することで，声優認識の精度を向上させる手法を提案する．キャスト情報は人手で作成することも考えられるが，Web 検索して自動取得する方法を採用し，Wikipedia やアニメの公式ページから取得することを想定している．図 3.8 のように，Wikipedia などのソースコードを見てみると定型文が見られるので，自然言語処理を使って Wikipedia のソースコードからキャスト情報を抽出し，声優データベースの絞り込みを行う．

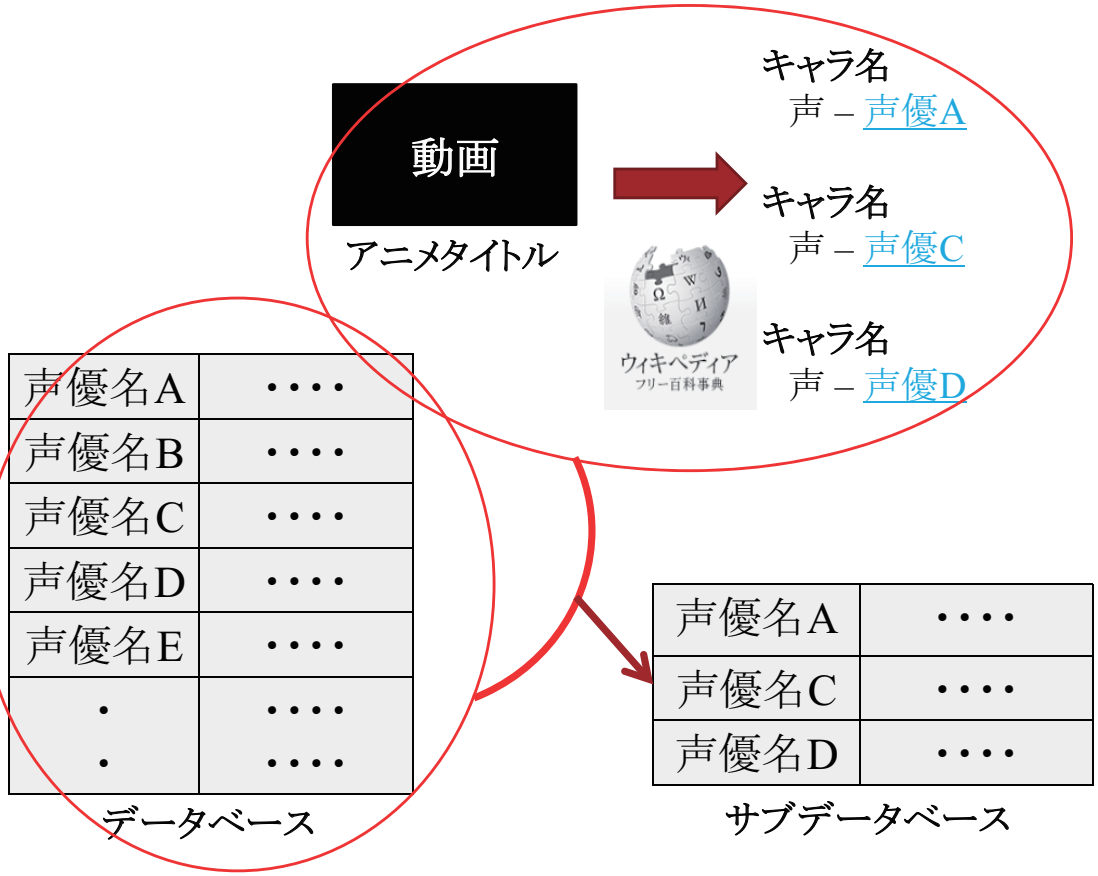

図 3.8　キャスト情報による声優データベースの絞り込み

第 4 章

評価実験

本章では，3 分の同じタイトルのアニメ動画 3 件を用いて，本システムの声優認識の精度に関して評価実験を行う．評価実験用のアニメ動画 3 件に出て来るキャストであるキャラクターと声優のペアは 2 組，5 組，5 組であるが，このアニメ作品シリーズには全部で 16 名の声優が出演している．また，声優データベースには男性 20 名，女性 20 名の声優 i の名前と音声データ $\boldsymbol{a_i}$ が入っており，評価実験用のアニメ動画 3 件に出て来る声優は確実に含まれている．このフルの声優データベースに加えて，アニメタイトルで検索した Wikipedia からテキスト抽出したキャスト情報で絞り込まれた声優 16 名が入っているサブデータベースの 2 種類を用いる．

3 種類の類似度計算と様々なパラメータ設定で声優認識した結果がテキストファイルとして出力される．このテキストファイルを PC 上に実装した評価システムに流すと，2 つのパラメータ（N 回と P%）に応じて認識精度を出力する．声優認識の精度を測る指標として，再現率と適合率の以下の式を用いる．

$$\text{再現率} = \frac{\text{システム認識した正解合計時間}}{\text{正解の声優名の時間}}$$

$$\text{適合率} = \frac{\text{システム認識した正解合計時間}}{\text{システム認識した声優名の時間}}$$

また，Web からどれくらいの精度でキャスト情報を取得することができるか実験を行う．本システムでアニメタイトルを実際に打ち込み,Wikipedia からキャスト情報を取得するようにしている．しかし，Wikipedia に載っている情報が必ずしも正しいとは限らない場合がある．アニメのエンディングのスッタフロールに流れてくるキャスト情報を正しい情報と定義づけると，Web 上に載っているキャスト情報では足りない場合がある．そこで，公式のアニメのキャスト情報と本システムで用いる Wikipedia から得られるキャスト情報の評価実験を行う．更に，本システムの精度を測るために Wikipedia に載っている声優の情報をどれくらい取得することが可能か評価実験してみる．自然言語処理を用いて Wikipedia からどれくらいキャスト情報の抽出ができているのか測る指標として，以下に再現率と適合率を求める式を示す．

$$\text{再現率} = \frac{\text{Wikipedia から本システムで正しく取得できた声優の数}}{\text{Wikipedia に載っている声優の数}}$$

$$\text{適合率} = \frac{\text{Wikipedia から本システムで正しく取得できた声優の数}}{\text{本システムで認識した声優の数}}$$

また，実際の公式のアニメのキャスト情報と Wikipedia から得ることができるキャスト情報の精度を測る指標として以下にその式を示す．

$$\text{再現率} = \frac{\text{公式のキャスト情報と本システムで取得したキャストの正解の声優の数}}{\text{公式のキャスト情報の声優の数}}$$

$$\text{適合率} = \frac{\text{公式のキャスト情報と本システムで取得したキャストの正解の声優の数}}{\text{Wikipedia から本システムで取得できたキャスト情報の声優の数}}$$

4.1　3 種類の類似度計算に依る声優認識精度の比較

本システムの評価実験では以下の項目について注目する．

- 3 種類の類似度計算の評価
- キャスト情報を取得した場合としない場合
- パラメータ N 回と $P\%$ の最適化

まず，3 種類の類似度計算のうち，本システムではどの類似度計算が最適なのかを比較する．比較する際，同じアニメの動画 3 件のキャスト情報を取得している状態で，パラメータを $N = 1$ 回の場合（パラメータ P は関与しない）に固定している．動画 3 件の 3 種類の類似度計算の再現率と適合率，F 値を表 4.1，4.2，4.3 と図 4.1，4.2，4.3 に示して比較する．また，これら動画 3 件の精度の平均を表 4.4，図 4.4 に示す．

表 4.1　類似度計算の種類に依る声優認識精度の比較（動画 1 件目）

類似度の計算	再現率	適合率	F 値
ユークリッド距離	0.029	0.020	0.024
コサイン類似度	0.057	0.039	0.046
相関係数	0.060	0.041	0.049

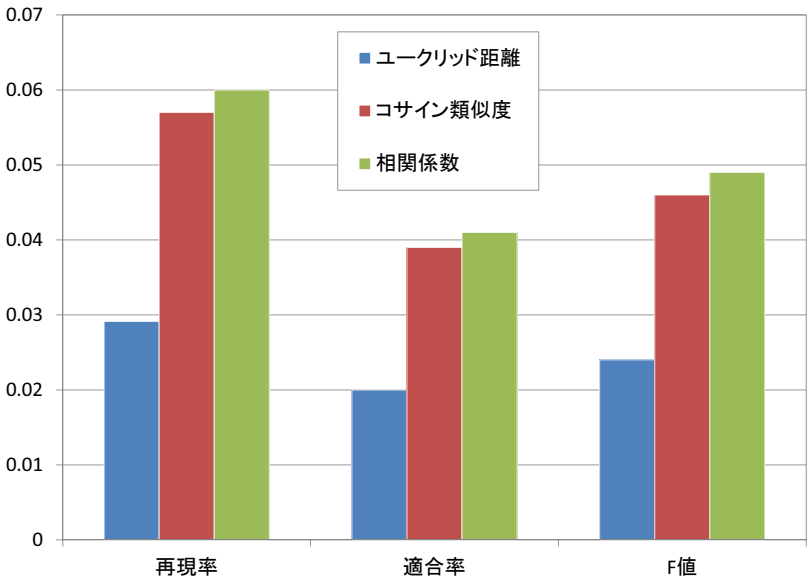

図 4.1　類似度計算の種類に依る声優認識精度の比較（動画 1 件目）

表 4.2　類似度計算の種類に依る声優認識精度の比較（動画 2 件目）

類似度の計算	再現率	適合率	F 値
ユークリッド距離	0.031	0.025	0.027
コサイン類似度	0.069	0.056	0.062
相関係数	0.070	0.057	0.063

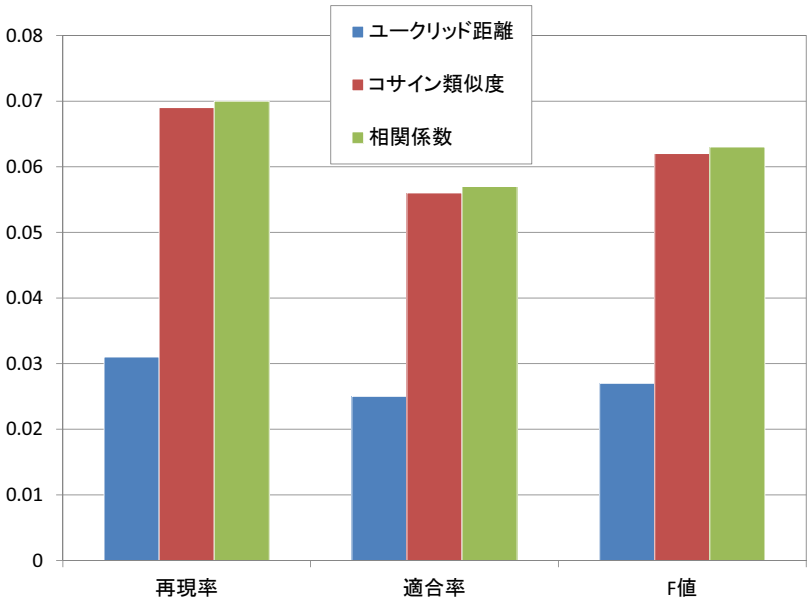

図 4.2　類似度計算の種類に依る声優認識精度の比較（動画 2 件目）

表 4.3　類似度計算の種類に依る声優認識精度の比較（動画 3 件目）

類似度の計算	再現率	適合率	F 値
ユークリッド距離	0.085	0.065	0.074
コサイン類似度	0.062	0.047	0.054
相関係数	0.071	0.054	0.061

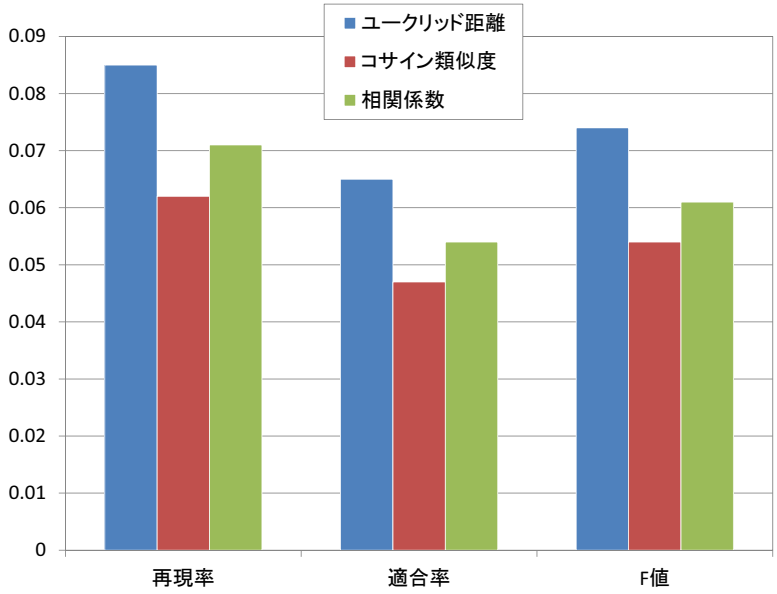

図 4.3　類似度計算の種類に依る声優認識精度の比較（動画 3 件目）

表 4.4　類似度計算の種類に依る声優認識精度の比較（動画 3 件の平均）

類似度の計算	再現率	適合率	F 値
ユークリッド距離	0.049	0.037	0.042
コサイン類似度	0.063	0.048	0.054
相関係数	0.067	0.051	0.058

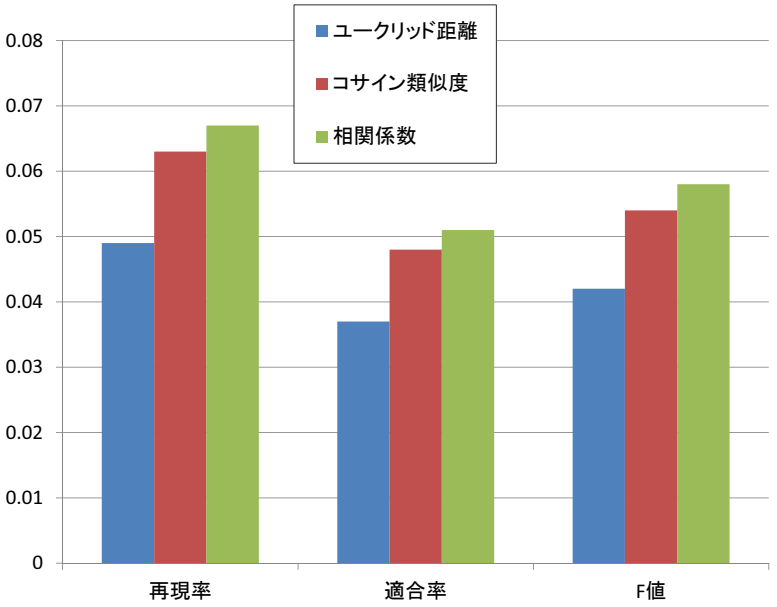

図 4.4　類似度計算の種類に依る声優認識精度の比較（動画 3 件の平均）

表 4.1，4.2 や図 4.1，4.2 の結果から読み取れるようにユークリッド距離を用いるよりも，コサイン類似度や相関係数を用いた方が全体的にシステムの声優認識の精度が高いことがわかる．このような結果が出たのは，音声波形データ間の類似度がユークリッド距離では適切に表現されていないからであり，本システムで用いる類似度の計算にユークリッド距離は好ましくないとわかる．しかし，表 4.3 と図 4.3 を見てみるとユークリッド距離の精度が一番大きくなっているのがわかる．この原因として考えられることは，3 件のアニメ動画のユークリッド距離を用いた場合の声優認識の結果を見てみると，認識結果に特定のキャストの出現頻度が偏っている．この 3 件目のアニメ動画には，その偏ったキャストが長い間登場していた．結果，3 件目の動画の場合ユークリッド距離を用いた時の精度が一番大きくなったと考えられる．しかし，複数の動画を用いた時にある特定のキャストに偏って認識されるということはシステムにふさわしくないと言える．表 4.4，図 4.4 の平均で求めた精度の値から本システムの類似度計算に適したものは，この 3 つの類似度計算の中では相関係数だと言える．よって，僅差ではあるがコサイン類似度よりも精度の高い相関係数が本システムの類似度計算にふさわしいと言える．

4.2　キャスト情報の有無に依る声優認識精度の比較

次に，キャスト情報を用いる場合と用いない場合とで比較評価を行う．使用するアニメ動画 3 件は同じアニメタイトルのものである．比較する際に類似度は相関係数を用いて，パラメータを $N = 1$ 回の場合に固定している．アニメ動画 3 件のキャスト情報の有無それぞれの再現率と適合率，F 値を表 4.5，4.6，4.7 と図 4.5，4.6，4.7 に示す．また，これら動画 3 件の精度の平均を表 4.8，図 4.8 に示す．

表 4.5　キャスト情報の有無に依る声優認識精度の比較（動画 1 件目）

キャスト情報	再現率	適合率	F 値
あり	0.060	0.041	0.049
なし	0.028	0.019	0.023

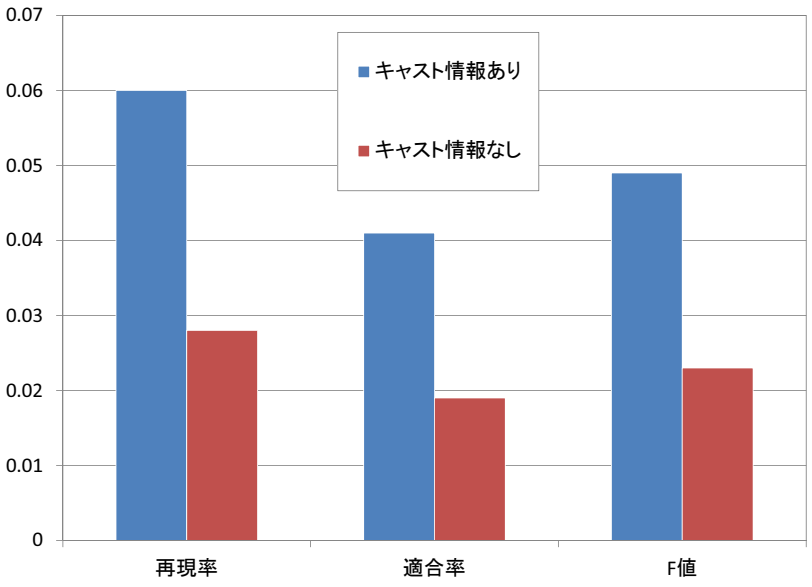

図 4.5　キャスト情報の有無に依る声優認識精度の比較（動画 1 件目）

表 4.6　キャスト情報の有無に依る声優認識精度の比較（動画 2 件目）

キャスト情報	再現率	適合率	F 値
あり	0.070	0.057	0.063
なし	0.040	0.033	0.036

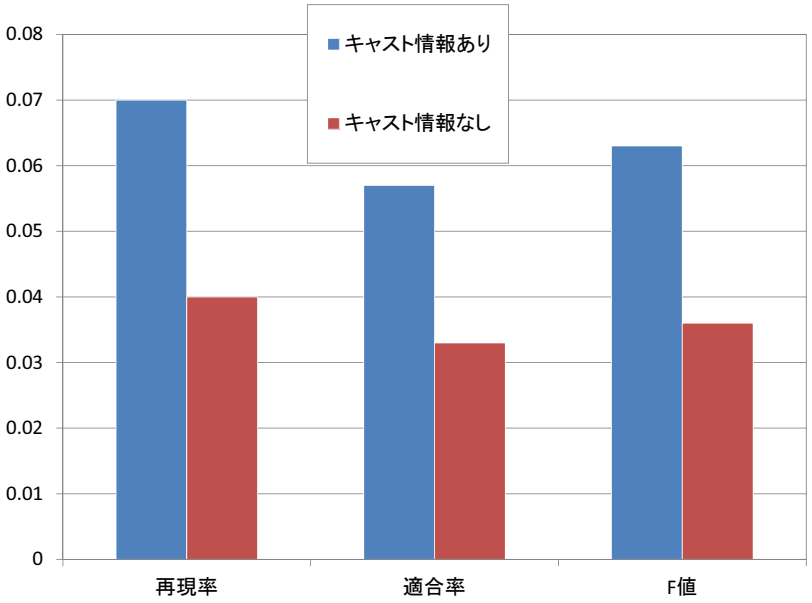

図 4.6　キャスト情報の有無に依る声優認識精度の比較（動画 2 件目）

表 4.7　キャスト情報の有無に依る声優認識精度の比較（動画 3 件目）

キャスト情報	再現率	適合率	F 値
あり	0.071	0.054	0.061
なし	0.026	0.020	0.022

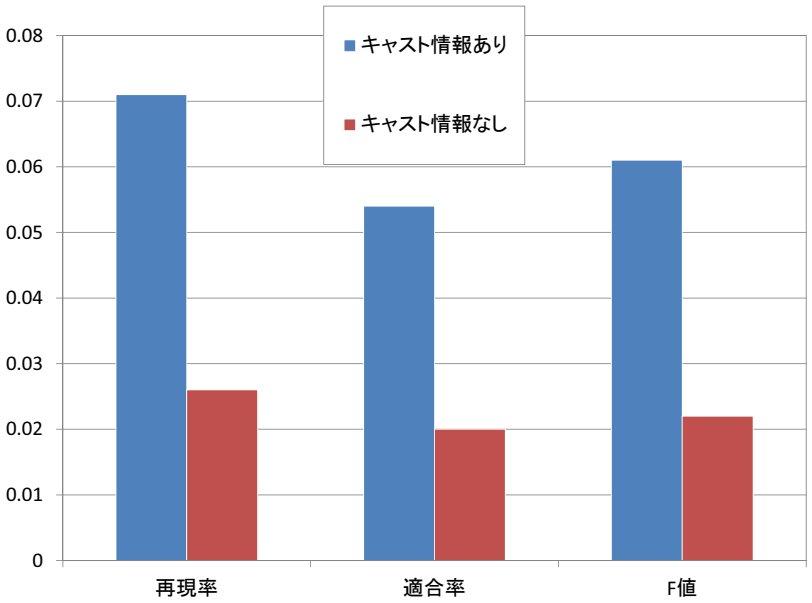

図 4.7　キャスト情報の有無に依る声優認識精度の比較（動画 3 件目）

表 4.8　キャスト情報の有無に依る声優認識精度の比較（動画 3 件の平均）

キャスト情報	再現率	適合率	F 値
あり	0.067	0.051	0.058
なし	0.030	0.023	0.026

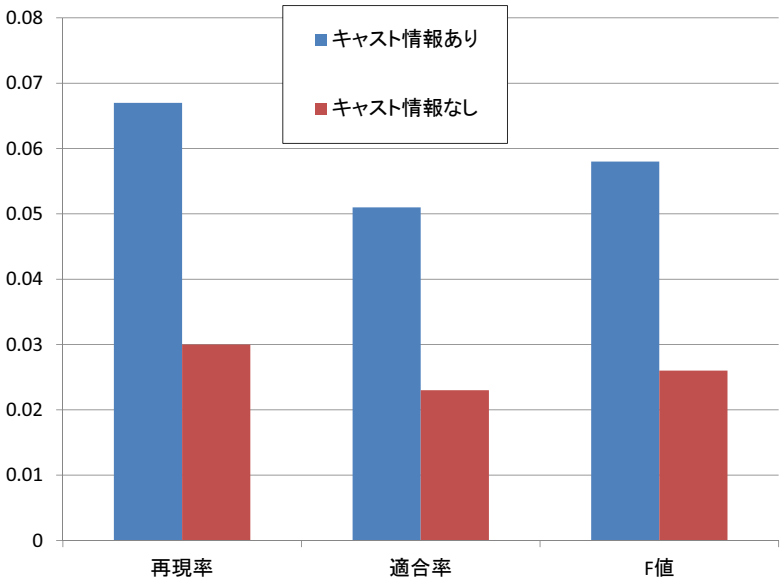

図 4.8　キャスト情報の有無に依る声優認識精度の比較（動画 3 件の平均）

表 4.5，4.6，4.7 と図 4.5，4.6，4.7 から，キャスト情報を用いて声優認識した方が良い精度を出していることがわかる．表 4.8，図 4.8 の結果から，声優データベースに入っている声優の候補を絞ることができれば，候補の数を減らすことができるので声優認識の精度の向上につながると考えられる．

4.3　パラメータの最適化

4.3.1　F 値に基づくパラメータ N 回と P% の最適化

N 回と P% のパラメータが再現率と適合率，F 値に影響を与えることが想定されるため検証する．前提条件として，類似度は相関係数を用いてキャスト情報を取得している場合に固定する．この条件下でのアニメ動画 3 件の F 値がパラメータの変動に依って，システムにどのような影響を及ぼしているのか検証する．アニメ動画 3 件の F 値の変動の 3 次元のグラフを以下の図 4.9，4.10，4.11 に示す．また，これら動画 3 件の精度の平均を図 4.12 に示す．

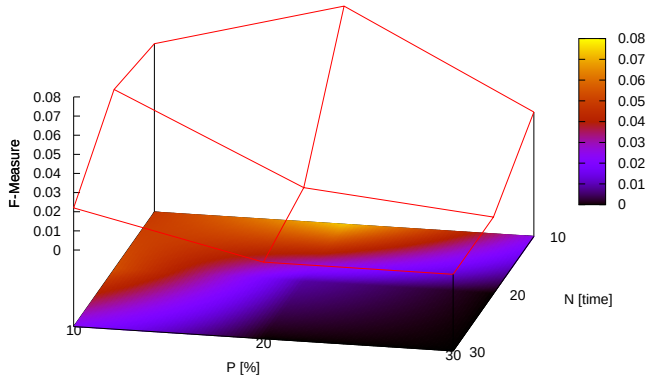

図 4.9　キャスト情報ありで相関係数を用いた時の F 値（動画 1 件目）

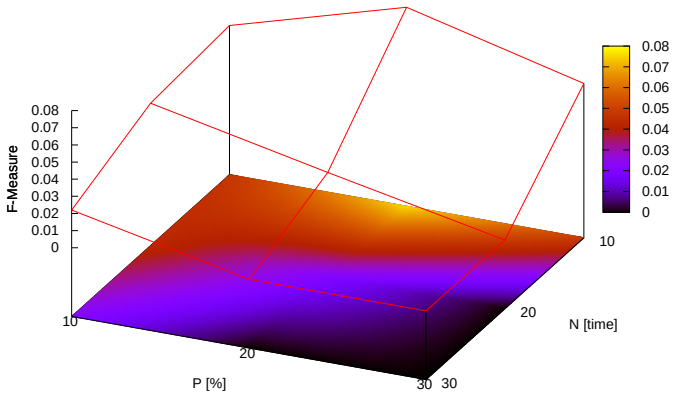

図 4.10　キャスト情報ありで相関係数を用いた時の F 値（動画 2 件目）

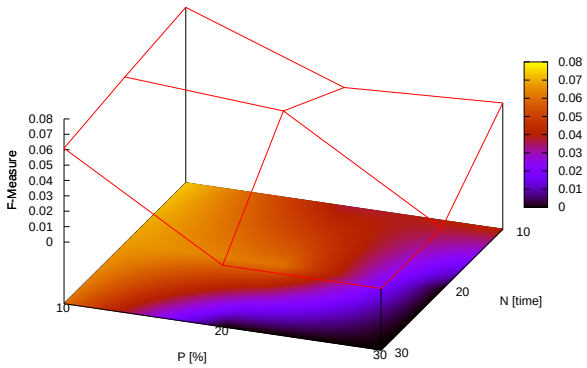

図 4.11　キャスト情報ありで相関係数を用いた時の F 値（動画 3 件目）

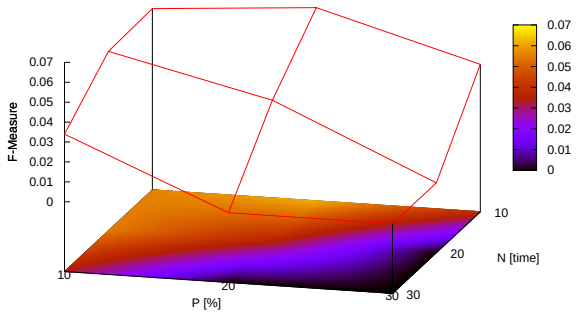

図 4.12　キャスト情報ありで相関係数を用いた時の F 値（動画 3 件の平均）

図 4.9 から図 4.11 の全体を比較してみると，どの条件下でも 2 つのパラメータが小さい場合に F 値が高いことがわかる．これは各パラメータの値が小さいと，判定「なし」となる危険性も低くなるからである．また，F 値が高くなるか低くなるかは，パラメータ $P\%$ の変動に依って大きく変わっている．これは N 回の声優認識が精確にされていないため，声優認識にばらつきが生じているのが原因ではないかと考えられる．0.1 秒毎の声優認識にばらつきがあると，N 回中 $P\%$ 以上の閾値には届かないため，必然的に判定「なし」が多くなる．また，パラメータ $P\%$ と同様に，パラメータ N 回の方も少なからず影響を及ぼしている．図全体から N が大きくなるにつれて F 値がほとんど下がっている．これは P のパラメータを固定していても，N の値が大きくなれば閾値を超える声優認識の 1 位の獲得回数が増えるためだと考えられる．0.1 秒毎の声優認識にばらつきがあると必然的に閾値を超えることは厳しくなる．図 4.12 から，2 つのパラメータのうちのどちらかがパラメータ 30 の時，精度は低くなっている．パラメータの値が 10 か 20 の時は比較的精度は安定している．その中でもパラメータ N が 10 の時，P が 20 の時に F 値が他のパラメータよりも一番大きくなっている．よって，本システムの最適なパラメータは N が 10，P が 20 であると言える．

4.3.2　平均順位に基づくパラメータ N 回の最適化

前節の 4.3.1 の評価方法では，N 回毎の声優認識で一番多く 1 位を獲った声優のみが評価されている．もしかすると，動画の音声の正解の声優が本システムの声優認識で惜しくも 1 位を獲れずに上位で彷徨っている可能性もある．そこで動画から流れる音声の正解の声優がシステムの声優認識でどの順位に位置しているかを確かめるため，本システムの N 回毎の声優認識の時の正解の声優の平均順位を求める．前提条件として，類似度は相関係数を用いてキャスト情報を取得している場合とキャスト情報を取得していない場合とする．また，正解の声優の順位に注目していくためパラメータ N の変動には依存するが，パラメータ P には依存しない．アニメ動画 3 件の正解の声優が位置する平均順位の結果を表 4.9，4.10，4.11 に示し, 表 4.12 に動画 3 件の平均順位の平均を示す．

- キャスト情報あり ： データベース 16 人
- キャスト情報なし ： データベース 40 人

表 4.9　動画 1 件目の正解の声優が位置する平均順位

N	キャスト情報あり	キャスト情報なし
1	8.658 位	20.665 位
10	7.768 位	19.826 位
20	8.279 位	18.077 位
30	7.976 位	19.582 位

表 4.10　動画 2 件目の正解の声優が位置する平均順位

N	キャスト情報あり	キャスト情報なし
1	8.492 位	19.801 位
10	8.982 位	17.589 位
20	8.402 位	20.126 位
30	8.032 位	18.151 位

表 4.11　動画 3 件目の正解の声優が位置する平均順位

N	キャスト情報あり	キャスト情報なし
1	8.231 位	19.807 位
10	8.111 位	20.704 位
20	8.570 位	18.437 位
30	7.576 位	21.591 位

表 4.12　動画 3 件の平均順位の平均

N	キャスト情報あり	キャスト情報なし
1	8.460 位	20.091 位
10	8.287 位	19.373 位
20	8.417 位	18.880 位
30	7.861 位	19.775 位

表 4.9 から表 4.12 を見てみると声優認識による正解の声優の順位は約半分の位置にいる．この結果から，このシステムによる音声の認識は無作為に声優を選出しているのに近い精度であると予測される．これらから，本システムで類似度の一番高い 1 位の声優だけをカウントするのではなく順位の高い方から半数をカウントするようにすると，また違った結果が得られたかもしれないことが考えられる．

4.4　キャスト情報の取得に関する精度評価

4.4.1　本システムで Wikipedia から取得できるキャスト情報の評価

初めに，どれくらいの精度で Wikipedia からキャスト情報を抽出できるかの評価実験を行う．まず Wikipedia から精確にキャスト情報を取得することができるかどうかを適当なアニメタイトル 30 件を用いて評価する．まず，評価の定義づけを説明する．キャスト情報の抽出の際に声優名に関係ない文字が取得されることがある．他にも人名の場合，実写版に出て来る俳優名などの関係ない情報が一緒に取得される場合がある．しかし，それらが一緒に取得されて来ても，データベースと照らし合わせて排除する後処理を行うことでシステム上問題は生じないので無視する．また，アニメタイトルで検索する処理をしてもそのアニメタイトルの Web ページに到達しないことがある．その場合，キャスト情報が取れなかったものとして Wikipedia から本システムで正しく取得できた声優の数は評価しない．

表 4.13　本システムでキャスト情報を Wikipedia から取れる精度（30 件）

アニメタイトル	再現率	適合率
アニメ A	0.966	1.000
アニメ B	0.034	1.000
アニメ C	0.895	1.000
アニメ D	0.964	1.000
アニメ (19 件)	1.000	1.000
アニメ (7 件)	評価なし	評価なし
アニメ（23 件の平均）	0.95	1.000

表 4.13 を見ると，適合率から本システムで取得できたキャスト情報には問題がないことがわかるが，再現率から Wikipedia に載っているキャスト情報の一部または大きく取得できていないことがわかる．これは普段のパターン（声 - 声優名）とは違う例外的なパターンで Wikipedia に書かれているからであると言える．例を挙げると 1 人のキャラクターに 2，3 人の声優名が載っていたり，声優名の前の記号のパターンが（声 - 声優名 → 声 ： 声優名）異なっていたりする．また，キャラクターの声優が途中で違う声優に交代していた場合「声 - 声優名→声優名」とパターンが違っていたり，キャラクターの説明文のところに違うキャラクターの名前とその違うキャラクターの声優名が載っていたりしていた．他にも声優名にリンクが張られていたり張られていなかったりすることでパターンが変わってしまうこともある．こういった問題の対応策としては問題に合わせて例外的な処理をしていくことが考えられる．次に表 4.13 からアニメ（7 件）がキャスト情報の取得に失敗したことがわかる．この 7 件は動画サイトから取得したアニメタイトルを Wikipedia の URL の一部に挿入しても，そのアニメの情報が載っている Wikipedia の Web ページに辿り着くことができなかった．これは，一

概に動画サイトから取得したアニメタイトルと言われても色々なパターンがあり，Wikipedia に載っているキャスト情報のページの Wikipedia のタイトルにも色々なパターンがあるからである．まず，Wikipedia のキャスト情報が載っている基本的なタイトルはアニメタイトルをそのまま用いたものである．しかし，視聴しているアニメ動画のタイトルが 2 期のものであったり，副題がついていたり，記号が半角ではなく全角であったりするとページが見つからない問題が出て来る．また，その逆も然りで視聴しているアニメ動画のアニメタイトルが基本のものであっても，Wikipedia のキャスト情報が載っている Wikipedia のタイトルが「○○○の登場人物」,「○○○のキャラクター」や漫画が原作のアニメ化だった場合「○○○_（漫画）」などのパターンが存在する場合もある．次にアニメ（23 件の平均）の結果から, 適合率に問題はないと言えるが，再現率から必ずしも Wikipedia に載っている情報全ては取得できないと言える．これから先，アニメが増えて Wikipedia の声優の情報の載せ方に新しいパターンが出て来る可能性がある．再現率の精度を 1.0 に近づけるために，Wikipedia のソースコードの色々なパターンに対応させる必要がある．アニメ（23 件の平均）には評価できなかったアニメ（7 件）は評価対象に入れていない．これは，本研究の評価実験が Wikipedia のソースコードから本システムでどのくらい精確にキャスト情報を取得できているのかを評価しているからである．

4.4.2　公式のアニメのキャスト情報に基づく Wikipedia から取得できたキャスト情報の比較

公式のアニメのキャスト情報と Wikipedia からのキャスト情報を比較するためにそれぞれ違うアニメタイトル 3 件を使った．公式のアニメのキャスト情報を調べる際にそのアニメ全話のエンディングのスッタフロールを見て確かめた．表 4.14 にそれぞれの違うアニメタイトル 3 件の再現率と適合率の結果と，アニメタイトル 3 件の再現率と適合率の平均を示す．

表 4.14　公式のアニメのキャスト情報に基づいて Wikipedia から取得できたキャスト情報の精度（3 件）

アニメタイトル	公式のキャスト	本システムで取れたキャスト	再現率	適合率
アニメ E	23 名	15(1 人はずれ) 名	0.609	0.933
アニメ F	21 名	16 名	0.762	1.000
アニメ G	38 名	31 名	0.816	1.000
アニメ（3 件平均）	82 名	62(1 人はずれ) 名	0.744	0.984

表 4.14 の結果から，適合率にあまり問題はないと言えるが再現率からキャスト情報の取りこぼしが見られる．アニメの出現頻度の高いキャスト情報は取得できていたが，名前の無いキャスト（客 A，B など）になると Wikipedia に確実に載っているかわからないためキャスト情報の取得の失敗につながったと考えられる．また，声優ではない人などの特別出演枠などが原因で公式のキャスト情報には載っていても Wikipedia のキャスト情報には載っていなかったことが再現率の低下につながっている．アニメ（3 件平均）の結果から，Wikipedia からは公式のアニメのキャスト情報全てを取得するのは厳しいことがわかる．

第 5 章

まとめと今後の課題

本研究ではアニメ動画から声優を認識するために，動画の音声データを Android 標準 API の Visualizer を用いて音声波形として出力させて，その音声波形から取得できる数値を用いた 3 種類の類似度計算に基づいて声優認識する手法を提案した．さらに声優認識の精度をより向上させるために，Web 上でキャスト情報を取得したり，2 種類のパラメータを設けたり，音声波形データの数値を正規化したり，様々な改善方法を検討した．その結果，キャスト情報を取得してデータベースに入っている声優の候補を出来る限り絞った方が声優認識の精度が向上する．また，類似度の計算において，ユークリッド距離を用いると特定の声優の認識に結果が偏ってしまい著しく精度が低くなる．パラメータに関しては，N 回毎にばらついた声優認識結果が出ているために高い閾値 $P\%$ を設けると途端に精度が低くなることを確認した．考察として，Visualizer で取得する音声波形データを使って声優認識するシステムの精度が低いと感じる．これは，Android 標準 API の Visualizer から取得できる音声波形データが合成波形であるからではないかと考えられる．Wikipedia からキャスト情報を取得することに関しては，精度を向上させるために色々なパターンの自然言語処理を必要とする．Wikipedia で検索をするために，アニメのタイトルを取得したらそのタイトルを自然言語処理で置換したり，付け加えたりして改変することでキャスト情報の取得の成功に近づけられる．公式のキャスト情報量ほど Wikipedia にキャスト情報量が無いが，主要なキャスト，出現頻度がある程度高いキャストなら確実に載っていることを確認した．

今後の課題として，音声の認識の精度の向上を目指していく．初めに，今後は Android の他の機能を使って音声データをフーリエ変換して周波数の情報も取り入れることが考えられる．また，本研究では Visualizer の音声波形データの数値の軌跡を用いた声優認識を行ったが，Android 搭載の dB を算出できる機能を使って dB の情報を使うことも考えている．次に，本研究の声優データベースには声優 1 名につき 1 種類の 1 つの音声波形データしか入っていなかったが，複数の種類の複数の音声データを入れておき，それらを組み合わせることで声優認識の精度向上を図る．また，本研究で用いた類似度計算だけでなく，他の類似度の定義を用いる方法なども検討する．キャスト情報の取得に関する今後の課題として，アニメタイトルの例外パターン，文から声優名を取得する時の例外パターンの処理を増やすだけではなく，Wikipedia 以外の Web サイトからのキャスト情報の取得も試す．

謝辞

本研究に際して，様々なご指導を頂きました服部峻助教を初めとして，服部研究室の皆様に感謝を致します．また，実験に使った Wikipedia の製作者の皆様にも感謝致します．そして，本研究で用いたフリーの API を提供している一社に感謝致します．

参考文献

[1] 杉江 嘉昭, 小林 哲則 “Dempster - Shafer 理論を用いた音声・画像情報の統合による個人認識システム,” 電子情報通信学会 MVE 研究会, 信学技報, Vol.101, No.425, pp.63–68 (2001).

[2] 徳田 恵一, “音声情報処理技術の最先端：1. 隠れマルコフモデルによる音声認識と音声合成,” 情報処理, Vol.45, No.10, pp.1005–1011 (2004).

[3] 宋 炳卓, 小沢 慎治, “時系列顔画像処理による個人の認識,” 電子情報通信学会, PRU, パターン認識・理解, Vol.93, No.268, pp.29–36 (1993).

[4] 石井 大祐, 渡辺 裕, “マンガからの自動キャラクター位置検出に関する検討,” 研究報告オーディオビジュアル複合情報処理（AVM）, Vol.2012-AVM-76, No.1, pp.1–5 (2012).

[5] 佐藤 隆紀, 早野 誠治, 齋藤 兆古, 堀井 清之, “知的可視化情報処理による動画像認識,” 可視化情報学会誌 = Journal of the Visualization Society of Japan 22, pp.243–246 (2002).

[6] 山口 順一, “人の認識, 展望,” 精密工学会誌, Vol.71, No.2, pp.159–162 (2005).

[7] Google Android – Visualizer, `http://developer.android.com/reference/android/media/audiofx/Visualizer.html`.

[8] 古井 貞熙, “話者認識の現状と展望,” 電子通信学会誌, Vol.67, No.5, pp.537–543 (1984).

[9] 小林 光, 田中 章浩, 木下 健太郎, 岸田 悟, “声紋による個人認証システムの構築,” 電子情報通信学会 ニューロコンピューティング研究会, 信学技報, Vol.108, No.480, pp.13–17 (2009).

[10] @y_benjo, “音声による既婚声優の判別問題,” 日本声優統計学会, 声優統計, Vol.2 (2013).