解 説

# ロボット聴覚オープンソースソフトウエア HARK

Robot Audition Open-Sourced Software HARK

奥乃 博[*1] 中臺一博[*2] [*1]京都大学大学院情報学研究科 [*2]ホンダ・リサーチ・インスティチュート・ジャパン
Hiroshi G. Okuno[*1] *and* Kazuhiro Nakadai[*2] [*1]Kyoto University, Graduate School of Informatics [*2]Honda Research Institute Japan Co., Ltd.

## 1. ロボット聴覚ソフトウエアは総合システム

ロボット聴覚機能はロボットビジョンの機能と同様に一言で定義できない．例えば，オープンソース画像処理ソフトウエア OpenCV が膨大な処理モジュールの集合体であるように，ロボット聴覚ソフトウエアも同様な方向のフレームワークである必要がある．

ロボット聴覚ソフトウエア HARK（HRI-JP Audition for Robots with Kyoto Univ., hark は listen を意味する中世英語）は『聴覚の OpenCV』を目指したシステムである．HARK 第 1 版では，音情報を基に音環境を理解する音環境理解（Computational Auditory Scene Analysis）の三つの課題である音源定位（sound source localization），音源分離（sound source separation），および，分離音声の音声認識（automatic speech recognition）を最低限提供すべき機能として，開発してきた．現在，研究用にはオープンソースで無償公開[†]を行っている [1].

以下，第 2 章で HARK の設計思想について述べ，HARK が現在ミドルウェアと利用している FlowDesigner について概説する．第 3 章で HARK のモジュール群について概説する．第 4 章で今後の開発予定とまとめを述べる．

## 2. HARK の設計思想

HARK の設計思想を以下にまとめる．
(1) 様々なマイク配置への対応,
(2) 様々な A/D 装置への対応,
(3) 様々な音響処理モジュールの提供,
(4) 実時間処理.
このような思想の下に，2008 年 4 月に HARK 0.1.7 をオープンソースとして公開し，開発者自身での改良，ユーザからのフィードバックの反映，バグフィックス，ドキュメントの充実などを通じて HARK 1.0.0 にバージョンアップを行っている．主な新機能は以下の以下のとおりである:
(1) 音源分離の新規実装,
(2) 比較的複雑なロボット形状への対応,
(3) ロボットの定常雑音対応,
(4) 移動音源を見据えた対応,
(5) 音源分離のパラメータ詳細設定機能,
(6) 新音声特徴量の利用,
(7) 設定データ可視化・作成ツール提供,
(8) Flowdesigner の操作性向上.

原稿受付 2009 年 11 月 19 日
キーワード：Robot Audition, Sound Source Localization, Sound Source Separation, Automatic Speech Recognition, Missing Feature Theory, MUSIC, GSS, Multichannel Post-filter
[*1]〒 606–8501 京都市左京区吉田本町
[*2]〒 351–0188 和光市本町 8–1
[*1]Sakyo–ku, Kyoto–shi, Kyoto
[*2]Wako–shi, Saitama

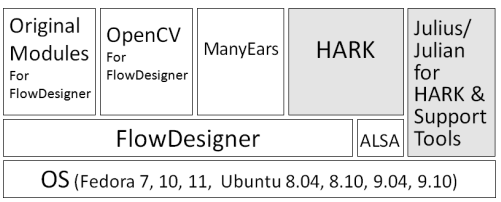


**図 1** HARK と FlowDesigner, OS との関係

HARK は，**図 1** に示すように，音声認識部（Julius）やサポートツールを除き，FlowDesigner をミドルウエアとして用いている．

### 2.1 ミドルウェア FlowDesigner

ロボット聴覚では，音源定位データを基に音源分離し，分離した音声に対して音声認識を行うことが多い．各処理は，アルゴリズムが部分的に置換できるよう複数モジュールで構成するほうが柔軟である．このため，効率のよいモジュール間統合が可能なミドルウェアの導入が不可欠である．しかし，統合するモジュール数が多くなると，モジュール間接続の総オーバヘッドが増大し，実時間性が損なわれる．モジュール間接続時にデータのシリアライズを必要とする CORBA （Common Object Request Broker Architecture）のような一般的な枠組みではこうした問題への対応は難しい．実際，HARK の各モジュールでは，同じ時間フレームであれば，同じ音響データを用いて処理を行う．この音響データを各モジュールがいちいちメモリコピーを行って

[†]http://winnie.kuis.kyoto-u.ac.jp/HARK/

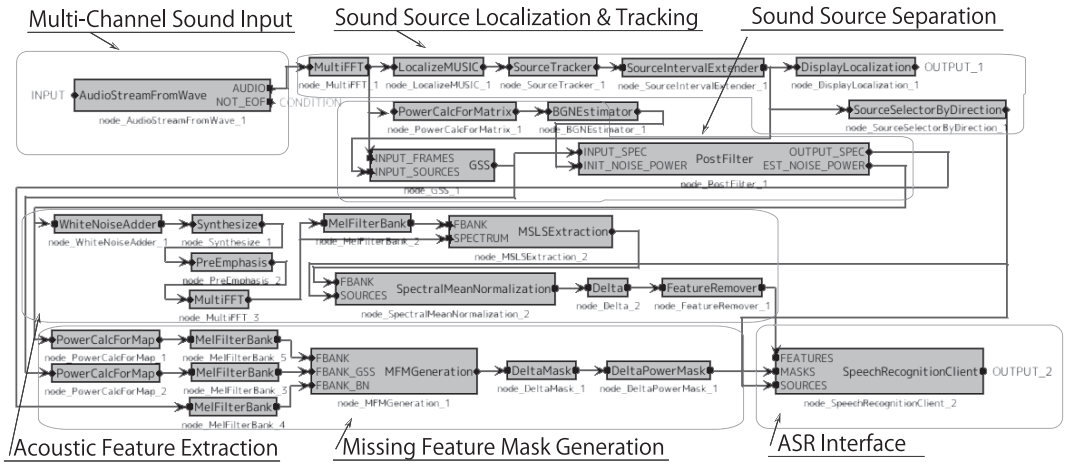


**図 2** HARK を用いたロボット聴覚の例

使っていたのでは，速度的にもメモリ効率的にも不利である．

FlowDesigner [2] は，単一コンピュータ内の利用を前提とすることで†，高速・軽量なモジュール統合を実現したデータフロー指向の GUI 開発環境を備えたフリー（LGPL/GPL）のミドルウェアである．FlowDesigner では，各モジュールは C++ のクラスとして実現される．これらのクラスは，共通のスーパークラスを継承するため，モジュール間のインタフェースは自然と共通化される．モジュール間接続は，各クラスの特定メソッドの呼び出し（関数コール）で実現されるため，オーバヘッドが小さい．データは，参照渡しやポインタで受け渡されるため，前述の音響データのような場合でも，高速にかつ少ないリソースで処理できる．つまり，FlowDesigner の利用によって，モジュール間のデータ通信速度とモジュール再利用性の両立が可能である．我々は，メモリリーク等のバグ対処，操作性向上（主に属性設定部）を図った FlowDesigner も公開している††．

HARK を用いた典型的なロボット聴覚に対する FlowDesigner のネットワークを**図 2** に示す．ファイル入力によりマルチチャネル音響信号を取得，音源定位・音源分離を行う．得られた分離音から音響特徴量抽出，ミッシングフィーチャマスク（MFM）生成を行い，これらを音声認識（ASR）に送る．各モジュールの属性は，属性設定画面で設定することができる（**図 3** は GSS の属性設定画面の例）．また，現在提供している HARK モジュールを**表 1** に示す．

### 2.2 入力装置

HARK では複数のマイク（マイクアレイ）をロボットの耳として搭載して処理を行う．ロボットの耳の設置例を**図 4** に示す．この例では，いずれも 8 チャネルのマイクアレイを搭載しているが，HARK では，任意のチャネル数のマイクアレイが利用可能である．HARK がサポートするマルチチャネル A/D 装置は，下記の 3 種類である．

- ALSA ベースの A/D 装置，例えば，RME 社製
- 東京エレクトロンデバイス社製 8 チャネル A/D ボード TD-BD-8CSUSB（USB インタフェース）†††
- 日本電子システムテクノロジー社製 16 チャネル A/D 付き信号処理プロセッサ RASP-2.

マイクは，安価なピンマイクで構わないが，ゲイン不足解消のため，プリアンプがあった方がよい．TD-BD-8CSUSB や RASP-2 は，プリアンプおよび，プラグインパワー対応の電源供給機能を有しているので，使い勝手がよい．

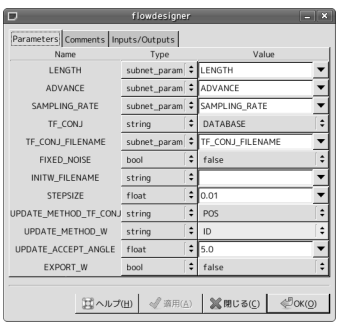


**図 3** GSS の属性設定

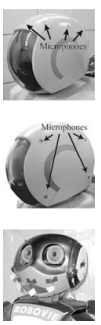

**図 4** ロボットの耳

## 3. HARK のモジュール群

### 3.1 音源定位

音源定位には MUltiple SIgnal Classification（MUSIC）法を用いる．MUSIC 法は，音源位置と各マイク間のインパルス応答（伝達関数）を用いて，音源定位を行う手法である．インパルス応答は，実測値もしくは tftool を使用して計算したものを用いることができる．

†コンピュータをまたいだ接続は，HARK における音声認識との接続部のようにネットワーク接続用のモジュールを作成することで実現可能である．

††FlowDesigner のオリジナルは，http://flowdesigner.sourceforge.net/から，FlowDesigner 0.9.0 の機能向上版は，http://winnie.kuis.kyoto-u.ac.jp/HARK/からそれぞれダウンロードできる．

†††TD-BD-8CSUSB 用のドライバは古く，最新の Linux カーネルではパッチが必要．同社の 16 チャネル A/D ボード TD-BD-16ADUSB に対応したサードパーティモジュールが公開予定．

**表 1** Modules provided by HARK 1.0.0

| 機能 | モジュール名 | 説明 |
|---|---|---|
| 音入出力 | AudioStreamFromMic | マイクから音を取得 |
| | AudioStreamFromWave | ファイルから音を取得 |
| | SaveRawPCM | 音をファイルに格納 |
| 音源定位・追跡 | LocalizeMUSIC | 音源定位 |
| | ConstantLocalization | 定位置を出力 |
| | SourceTracker | 音源追跡 |
| | DisplayLocalization | 定位結果の表示 |
| | SaveSourceLocation | 定位をファイルに格納 |
| | LoadSourceLocation | 定位をファイルから取得 |
| | SourceIntervalExtender | 追跡結果を前方に延長 |
| 音源分離 | DSBeamformer | 遅延和型ビームフォーマ |
| | GSS | GSS 音源分離 |
| | Postfilter | 音源分離後処理 |
| | BGNEstimator | 背景雑音推定 |
| 特徴量抽出 | MelFilterBank | メルフィルタバンク |
| | MFCCExtraction | MFCC 抽出 |
| | MSLSExtraction | MSLS 抽出 |
| | SpectralMeanNormalization | スペクトル平均正規化 |
| | Delta | Δ 項計算 |
| | FeatureRemover | パワー項削除 |
| | PreEmphasis | プリエンファシス |
| | SaveFeatures | 特徴量を格納 |
| | MFMGeneration | MFM 生成 |
| | DeltaMask | Δ マスク項計算 |
| | DeltaPowerMask | Δ パワーマスク項計算 |
| ASR との通信 | SpeechRecognitionClient | ASR に特徴量を送る |
| | SpeechRecognitionSMNClient | 同上，特徴量 SMN 付 |
| ASR | Multiband Julius/Julian | 音声認識システム |
| データ変換 | MultiFFT | マルチチャネル FFT |
| | Synthesize | 波形変換 |
| | WhiteNoiseAdder | 白色雑音追加 |
| | ChannelSelector | チャネル選択 |
| | SourceSelectorByDirection | 方向による音源選択 |
| | SourceSelectorByID | ID による音源選択 |
| | MatrixToMap | Matrix→Map 変換 |
| | PowerCalcForMap | Map 入力のパワー計算 |
| | PowerCalcForMatrix | 行列入力のパワー計算 |
| 外部ツール | tftool | 伝達関数作成ツール |
| | hark-tool | データ可視化ツール |

HARK 0.1.7 では，ManyEars [3] のビームフォーマが利用可能だった．このモジュールは，2D 極座標空間で，マイクアレイから 5 [m] 以内，かつ，音源間が 20° 以上離れていれば，定位誤差は約 1.4° であると報告されている．しかし，もともと 48 [kHz] サンプリング用に作られており，HARK で利用している 16 [kHz] サンプリングと合致しないこと，MUSIC のような適応ビームフォーマは一般的なビームフォーマよりも音源定位精度が高いことから HARK 1.0.0 では，MUSIC 法のみをサポートしている．

[†]http://www.furui.cs.titech.ac.jp/mband_julius/，MFT-ASR には，Sheffield 大学で開発された CTK (Casa Toolkit) http://www.dcs.shef.ac.uk/~jon/ctk.html もある．

### 3.2 音源分離

ビームフォーマ（適応型，遅延和型），独立成分分析（ICA）などの使用経験から，HARK では音源分離に Geometric Source Separation（GSS）と PostFilter の組み合わせを推奨する．GSS は，音源からマイクへの伝達関数を幾何制約として使用し，与えられた音源方向から到来する信号を分離する．HARK 1.0.0 では，実測の伝達関数を使用できる．一方，HARK 0.1.7 で利用可能だった ManyEars 版 GSS は，マイク位置と音源位置の関係から伝達関数を計算している．マイク配置が同じでもロボットの形状が変わると伝達関数が変わるので，性能劣化の原因となっていた．また，GSS は分離する音源の方向情報が必要であるが，方向性音源としての性質が比較的強いロボット定常雑音は，つねに定位されるとは限らず，定常雑音の分離性能が劣化する場合があった．HARK 1.0.0 の GSS は，特定方向につねに雑音源を指定する機能が追加され，定位されない音源でもつねに分離し続けることが可能となった．なお，GSS の属性設定変更により，遅延和型ビームフォーマが構成できる．遅延和型ビームフォーマ DSBeamformer や別途提供のパッチを利用して ManyEars 版 GSS の使用も可能である．

GSS だけでは，音源間の相関，定位誤差，拡散性雑音などの影響で，分離性能が十分でない場合がある．PostFilter は，「パワーが大きければ音声，さもなければ雑音」という確率モデルに基づき，非線形な雑音の推定・抑圧を行う（例，三話者同時発話での分離音の S/N 向上は 10.3 [dB]）．ただし，非線形処理でスペクトル歪が生ずる．

### 3.3 MFT-ASR: MFT に基づく音声認識

分離音のスペクトル歪に対応するために，HARK ではミッシングフィーチャ理論（Missing Feature Theory，MFT）に基づいた音声認識（MFT-ASR）を推奨する．MFT-ASR は，分離音の音響特徴量とこの音響特徴量に対応する信頼度マップ（ミッシングフィーチャマスク，MFM と呼ぶ）を入力とし，最尤の音素の列を出力する．一般的な音声認識と同様に隠れマルコフモデル（Hidden Markov Model, HMM）に基づいているが，MFM が利用できるよう HMM から計算する音響スコア（主に出力確率計算）に関する部分に変更を加えている．MFM はポストフィルタから得られる定常雑音とチャネル間リークのエネルギーから求めている [4]．

MFT-ASR として，東工大古井研究室より公開されているマルチバンド版 Julian[†]をベースにオンライン化パッチをあてたものを HARK 公式サイトで提供している．以降のリリースでは，Julius 4 系のプラグイン機能を積極的に利用し，MFT-ASR の主要部分は Julius プラグインとして提供する予定である．MFT-ASR は FlowDesigner から独立したサーバ／デーモンとして動き，HARK の音声認識クライアントからソケット通信で送信された音響特徴量とその MFM に対し，結果を出力する．

### 3.4 音響特徴量抽出と音響モデルの雑音適用

スペクトル歪を特定の音響特徴量だけに閉じ込めて，MFT の有効性を高めるために，音響特徴量には，メルスケール対数スペクトル特徴量（Mel Scale Log Spectrum, MSLS）[4] の利用を推奨する．HARK では，音声認識で一般的に使用されるメル周波数ケプストラム係数（Mel-Frequency Cepstrum Coefficient, MFCC）も提供しているが，MFCC では，歪がすべての特徴に拡散するので，MFT との相性が悪い．HARK 1.0.0 では，MSLS 特徴量で，新たに Δ パワー項を利用するためのモジュールを提供する [1]．Δ パワー項は，MFCC 特徴量でもその有効性が報告されている．各 13 次元の MSLS と ΔMSLS，及び，Δ パワーという 27 次元 MSLS 特徴量が HARK 0.1.7 で使用していた MSLS，ΔMSLS 各 24 次元の計 48 次元 MSLS 特徴量よりも性能がよいことを確認している．

HARK では，上述の非線形分離による歪の影響を，少量の白色雑音を付加することで緩和している．クリーン音声と白色雑音を付加した音声とを使ったマルチコンディション学習により音響モデルを構築するとともに，認識音声にも分離後に同量の白色雑音を付加してから 音声認識を行う．これにより，一話者発話では，S/N が −3 dB 程度でも，高精度な認識が可能である [1]．

### 3.5 HARK の応用

HARK の応用として，3 人の同時料理注文を聞き分けるロボット（HRP-2, Robovie-R2），口じゃんけん判定を行うロボット（ASIMO, Robovie-R2），さらには，HARK が定位・分離した音を実時間，あるいは，アーカイブされたデータを可視化するシステムを開発してきた．

3 人の実話者全員が話し終えてから認識終了までに従来のファイル経由ベースの処理では，約 7.9 秒を要していたが，HARK の使用により，応答が約 1.9 秒に短縮された†．応答が速いため，全員の注文終了後，ただちにロボットがそれぞれの注文を復唱し，合計金額を答えるように感じられる．なお，モジュールの設定にも依存するが，ベンチマークテストの結果では認識までの遅延時間は 0.4 秒程度である．

## 4. まとめと今後の課題

本稿では，HARK 1.0.0 の概要を報告した．ミドルウェア FlowDesigner を使って，音環境理解の基本機能である音源定位，音源分離，分離音認識をモジュールとして実現し，ロボットの耳への応用について概説した．


**図 5** 3 人が料理を同時に注文するのを聞き分ける Robovie-R2

HARK 1.0.0 は，ロボット聴覚研究をさらに展開するための機能を提供している．例えば，移動音源処理に向けた機能，音源分離の各種パラメータの詳細設定機能，設定データ可視化・作成ツールなどである．また，Windows サポート，OpenRTM，ROS へのインタフェースなども進行中である．

HARK は，ダウンロードし，インストールするだけでもある程度の認識は可能であるものの，個々のロボットの形状や使用環境に合わせたチューニングを行えば，さらに音源定位，音源分離，分離音認識の性能が向上する．このようなノウハウの顕在化には，HARK コミュニティの形成が重要である．本稿がロボット聴覚研究開発者のクリティカルマスを超えるきっかけとなれば幸いである．

**謝　辞**　ロボット聴覚の研究を共同で推進してきた奥乃・尾形研究室の皆さん，HRI-JP の皆さんに感謝します．

## 参　考　文　献

[1] K. Nakadai, H.G. Okuno, H. Nakajima, Y. Hasegawa and H. Tsujino: “Design and Implementation of Robot Audition System “HARK”,” Advanced Robotics, accepted, VSP and RSJ.

[2] C. Côté, et al.: “Code Reusability Tools for Programming Mobile Robots,” IEEE/RSJ IROS 2004, pp.1820–1825, 2004.

[3] J.-M. Valin, F. Michaud, B. Hadjou and J. Rouat: “Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach,” IEEE ICRA 2004, pp.1033–1038, 2004.

[4] S. Yamamoto, J.-M. Valin, K. Nakadai, T. Ogata and H.G. Okuno: “Enhanced robot speech recognition based on microphone array source separation and missing feature theory,” IEEE ICRA 2005, pp.1427–1482, 2005.

†デモは http://winnie.kuis.kyoto-u.ac.jp/SIG/

**奥乃　博**（Hiroshi G. Okuno）
京都大学大学院情報学研究科教授，博士（工学）．1972 年東京大学教養学部基礎科学科卒業．NTT 研究所，JST，東京理科大学を経て，現職．ロボット聴覚，音環境理解，音楽情報処理，人工知能研究に従事．六十而耳順（論語・為政）．　（日本ロボット学会正会員）

**中臺一博**（Kazuhiro Nakadai）
（株）ホンダ・リサーチ・インスティチュート・ジャパン，シニア・リサーチャ．東京工業大学大学院情報理工学研究科連携准教授兼務．博士（工学）．1993 年東京大学工学部電気工学科卒業．1995 年同大学大学院情報工学専攻修了．NTT，NTT コムウェア，JST を経て，現職．ロボット聴覚，実時間情報統合，音環境理解の研究に従事．
（日本ロボット学会正会員）