

小嶋 明 (Akira Kojima)

AIアプリケーションエンジニア

複数のLLMを一つの画面で扱うFlutterアプリ「Clage Cook」、Android ChromiumへのAI機能統合「ChromiumforA」、PythonとFFmpegによる動画編集DSL「ScriptVEdit」を開発しています。実装にはAIエージェントを利用し、私は要件・仕様・設計上の判断、生成物のレビュー、実機確認、採否の判断を担っています。具体的な分担は各作品に記載しています。

東京大学 工学部 精密工学科卒・同大学院 修士課程修了（博士課程でEIT触覚センサを研究、2026年3月退学）。基本情報技術者。

- 公開OSS: Clage Cook（CI・Windows/Android実機）・ScriptVEdit（コード・テスト・出力動画・性能計測）
- Android実機開発: ChromiumforA（比較画像・操作動画、非公開）／受託開発: LINE応答AI（開発中・匿名掲載）
- 査読論文: Frontiers in Robotics and AI（筆頭著者）

代表作

Clage Cook — 4つのAIを、1つの会議へ（OSS）

[MIT License](#) [Flutter](#) [Dart](#) [Direct BYOK](#) [Secure Storage](#) [Python](#) [FastAPI](#)

Claude・Gemini・ChatGPT・Grokへ同じ質問を並列に送り、回答の比較・相互批評・統合までを1画面で行うアプリです。APIキーは利用者自身のもの（BYOK）を使います。



Windows版の実画面。「DIRECT・LOCAL」は中継サーバーを使わず、端末から各社APIへ直接接続して履歴を端末内に保存するモード

難所: 4社で異なるAPI契約（認証ヘッダ・エンドポイント・応答形式）の吸収と、一部のAIが失敗しても完了済み回答を保持する統合フロー
役割: 仕様・設計判断・生成物のレビュー・実機検証（Flutter/DartのUI実装はAIエージェントへ委任）
確認先: 公開リポジトリ（CI）・Windows / Android実機画面
公開状況: 公開ベータ。本番運用向けサポート・後方互換性は未保証

設計と検証の詳細・Android実機画面

設計判断: プロバイダごとに異なるAPI契約を共通のインターフェースに吸収し、一部のAIが失敗しても完了済みの回答を保持して統合まで進める流れにしました。進行状況はSSEでUIへ統一配信します。APIキーはOSのsecure storageへ保存し、会話履歴を端末内だけに置くDirect接続と、開発用のreference serverを分離しています。
実装・検証: 生成されたアプリをレビューして画面仕様・挙動・エラー処理の修正方針を指示し、WindowsとAndroid実機で動作を確認しました。



一覧：各AIの回答と統合回答 詳細：相互批評後も初回答を保持

開発背景 — 非公開版を毎日運用し、公式APIで再実装

[Kotlin / Jetpack Compose](#) [Python](#) [FastAPI](#) [Glance widget](#)

各社AIの回答比較・統合をAndroidから毎日使える常駐システムとして非公開で運用してきました。生成の継続、再接続後の復帰、使用量ウィジェットを設計し、この会議フローを公式APIベースで再実装したものが公開版です。

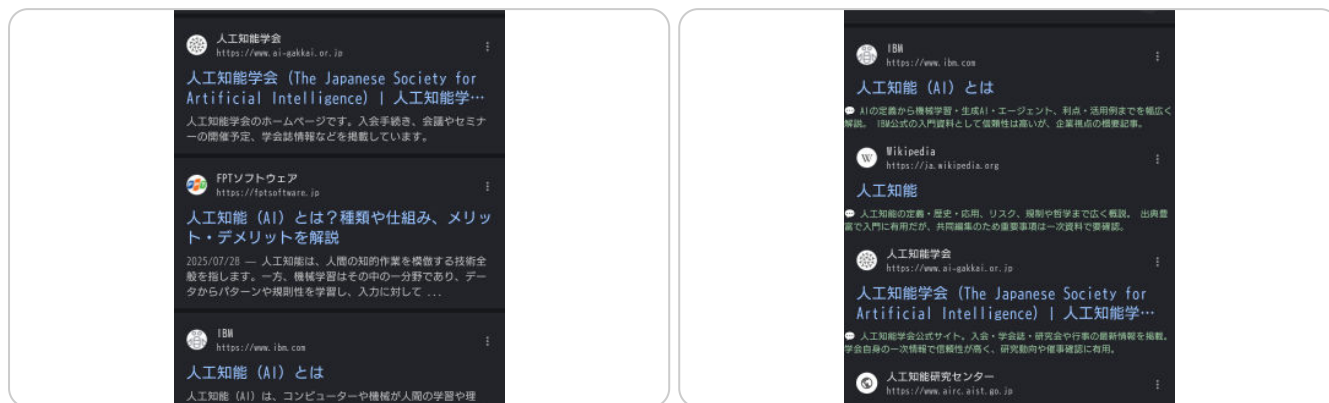
!コマンドによる会議操作、失敗時も完了済み回答を残す流れ、JSON / ZIPで他のAIへ渡せる連携を実装しています。



回答カード（実機） 使用量ウィジェット

Android Chromium改造 検索結果の再評価 ページ要約

Android版Chromiumを改造し、Google検索の結果ページへAIの評価コメントを挿入して、評価後の並び順を提示します。ページ要約と選択文へのAI処理も統合しました。



通常表示 — 「人工知能」の検索結果

AI評価後 — 根拠コメントの挿入と順位変更

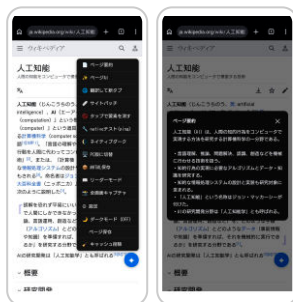
担当: Chromium本体への組み込み、操作設計、認証情報の扱い、安全性検証。

技術的な難所: 認証情報の扱いに関する問題を再現テストで検出し、構造を変更したうえで同条件で再検証しました。

Wikipedia要約・専用検索・関連項目

ページ要約・選択文AI

サイト別UI・動画再生補助



FABメニュー

ページ要約

実機画面をもう1枚（AI評価・並び替え完了）



AI評価・並び替え完了

確認できるもの: 同一検索語の通常表示／AI評価後の比較、Android実機画面、操作動画。個人検証用のため非公開で、画面・動画・設計のみ公開しています。

ScriptVEdit — Pythonで動画を記述する編集DSL（OSS）

[GitHub](#)[▶ デモ動画](#)[MIT License](#) [Python](#) [FFmpeg](#) [pytest](#)

動画のタイムラインをPythonコードで記述し、FFmpegコマンドへ変換して映像を生成するDSLです。コードだけで動画を組み立てられるため、AIエージェントによる動画制作にも使えます。解説動画自体もScriptVEditで制作しました。



左に使用したPythonコード、右にその実行結果を示す解説動画の一場面

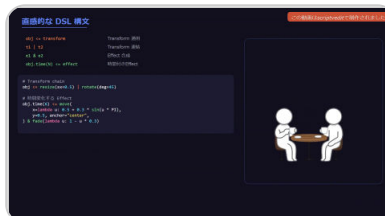
考案したもの: DSLの記法、素材キャッシュ、要素配置のアンカー解決、区間ごとの並列レンダリング。

AI利用の範囲: 上記の仕組みを含む実装全般と、テストケースの生成、スクリーンショットによる出力確認はAIエージェントへ任せました。DSLの仕様と各仕組みのアイデアはAIと相談しながらほぼ自分で決め、エフェクトの品質と、テーマだけ与えてAIに作らせた動画の出来は自分の目で評価しています。

計測条件: 2分56秒・87オブジェクトの実プロジェクトを20コアPCで計測 — 逐次1012秒、並列8で106秒。公開リポジトリ、テスト、コードと出力動画で確認できます。

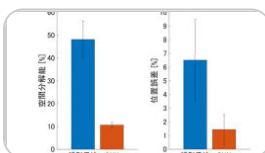


この動画自身のレンダリング時間を計測

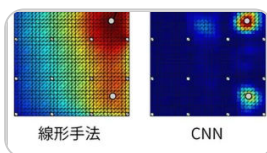


演算子で記述するDSL構文

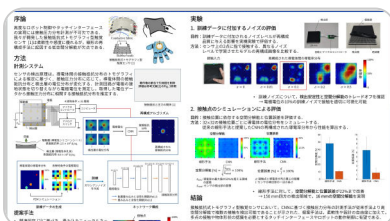
研究 — 機械学習×触覚計測



空間分解能・位置誤差（線形手法 vs CNN）



2点接触の再構成（線形手法 vs CNN）



ポスター全体（SI2022）

EIT（電気インピーダンストモグラフィ）触覚センサの逆問題にCNNを適用し、1点接触のシミュレーションで空間分解能指標と位置誤差を従来法の約22%まで低減しました（SI2022・IROS 2022でポスター発表）。

電極配置の最適化では、最悪領域の空間分解能指標（HWHM、小さいほど良い）を等間隔配置に対して最大19.2%低減し、筆頭著者としてFrontiers in Robotics and AIに掲載されました。

[CNN \(Keras\)](#) [シミュレーション訓練データ](#) [電極配置最適化](#)

[査読論文](#)

受託案件

美容クリニック向けLINE応答AI

[受託開発・開発中・匿名掲載](#)[Dify](#) [LINE Webhook](#) [RAG](#) [状態管理](#)

LINEでの問い合わせ一次対応を支援するAIです。FAQ回答、予約フォームへの案内、必要なときのスタッフへの引き継ぎを扱います。

Dify応答フロー、LINE Webhook、RAGナレッジ、状態管理を実装済みで、回答品質と安全性を検証中です。施設を特定しない範囲で掲載許可を得ています。顧客名・画面・利用者/運用データ・KPIは非公開です。

経歴・資格・主要技術

2018–22 東京大学 工学部 精密工学科
2022–24 東京大学大学院 修士課程 修了
2024–26 同博士課程でEIT触覚センサを研究（2026年3月退学）
2026– フリーランスのAIアプリケーションエンジニア

資格: 基本情報技術者試験 合格（2026年8月）
プログラミング実績: AtCoder 水色（最高レーティング1440）/
paizaスキルチェック Sランク

Python — EIT触覚センサ研究、過去作品の画像処理・CGの実装
DSL・ツール設計 — ScriptVEditの記法、素材キャッシュ、アンカー解決、並列化
Android / Chromium改造 — ChromiumforA
FFmpeg — ScriptVEditの映像生成
Dify / RAG — 受託のLINE応答AI
CNN (Keras) — EIT触覚センサ研究の再構成モデル
C++ / GLSL — 過去作品の制御・CG（倒立振り子、レイトレーシング）

Flutter/Dart（Clage CookのUI）は実装をAIエージェントへ委任し、UI仕様・生成物のレビュー・修正方針の決定とWindows/Android実機検証を担当。FastAPI（reference server）も実装は委任し、API設計・レビュー・結合動作の確認を担当しています。

過去作品

現在の代表作品より前に作ったものです。AIクライアント、制御、CG・画像処理の実装を実画面と動画で確認できます。

過去作品一覧（10件）



ChatGPT-Windows

Python・PySide6 (Qt6)・MIT

翻訳・補完・要約をホットキーとクリップボードから呼び出せるWindows常駐クライアント。ストリーミング表示と、Windows資格情報マネージャーへのAPIキー保存を実装。

GitHub

▶ デモ動画



ChatGPT-Web

JavaScript・単一HTML・MIT

入力の続きをゴーストテキストで提案するプロトタイプ。Tabと矢印キーで全体・1文字単位を確定。ローカル検証用。

GitHub

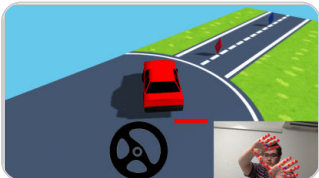
▶ デモ動画



Calcpy

Android・Kotlin/Compose・オンデバイスCPython 3.13/SymPy・非公開プロトタイプ

電卓の操作感にSymPyの厳密計算とPython REPLを統合したAndroid試作。4方向フリック入力と、Python実行の別プロセス隔離を設計。

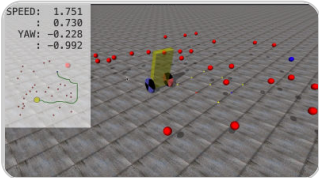


ハンドドライブ

JavaScript・Three.js

Webカメラで捉えた手の傾きと開閉を、ステアリング・アクセル・ブレーキへ変換する3D運転デモ。

▶ デモ動画

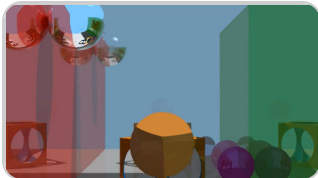


倒立振り子

C++・ODE

PID制御で倒立姿勢を保ちながら、障害物を避けてゴールへ移動するシミュレーション。

▶ デモ動画

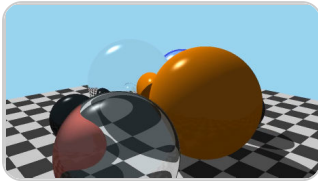


レイトレーシング (GLSL + CSG)

C++ · OpenGL/GLSL

CSGの和・差・積で形状を組み替え、反射・屈折・透過までリアルタイム描画。

[▶ デモ動画 ↗](#)



レイトレーシング (Python)

Python

レイと球の交差、材質、反射・屈折の再帰計算をフルスクラッチ実装。

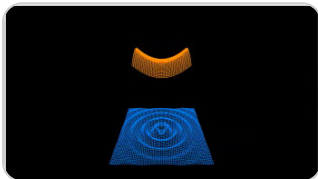
[▶ デモ動画 ↗](#)



シャドウマッピング

Python

デプスマップで影を生成し、バイアス調整でシャドウアークネとピーターバン現象を抑制。



波と放物面のアニメーション

Python

既存の描画基盤を再利用し、伝播する波と放物面の変形をワイヤーフレームで可視化。

[▶ デモ動画 ↗](#)



Canny法 (エッジ抽出)

Python

ライブラリに頼らず、Canny法のエッジ抽出処理を一から実装。左が入力、右が出力。

連絡先

[Mail: a.kojima8924@gmail.com](mailto:a.kojima8924@gmail.com)

[GitHub: kojima8924 ↗](https://github.com/kojima8924)

主な領域：AIを活用・統合したアプリ開発 / Python / C++

© 2026 Akira Kojima. All rights reserved.