BrailleSense — Haptic Glove Reader (Braille & Morse)

One-line summary

Braille Sense is a low-cost glove that turns printed or on-screen text into fingertip vibrations—Braille patterns or two-finger Morse—so people can feel information in real time. A small Raspberry Pi + camera reads the text; an ESP32 in the glove drives tiny vibration motors at the fingertips.

The problem we're tackling

- Millions of people who are blind/low-vision still struggle to access printed handouts, menus, transit signs, and forms—especially in unfamiliar places.
- Learning Braille can be intimidating without engaging, immediate feedback tools.
- In noisy or sensitive environments, silent communication is hard: audio gives you away; phones are bright; hand signals require line-of-sight.
- Existing solutions are often expensive, bulky, or tied to a single ecosystem.

Our solution (what we built)

BrailleSense Glove: a soft glove with six fingertip vibration motors laid out like the six dots of a Braille cell. Two of those fingertips double as dot and dash channels for Morse Code as well. An ESP32-DevKitC controls the motors through L298N motor drivers.

BrailleSense Reader: a Raspberry Pi 4 with a NoIR camera mounted above the page. The Pi captures an image, enhances contrast (OpenCV), performs OCR (Tesseract), and sends the recognized text to the glove over USB as simple commands. You choose Braille mode or Morse mode per message.

Only hackathon-provided hardware used: Raspberry Pi 4, Raspberry Pi NoIR camera, ESP32-DevKitC, L298N driver(s), vibration motors, breadboard, jumpers, electrolytic capacitors (for power smoothing), official Pi power supply (and a separate 5 V for motors). No outside gadgets required.

How it works (end-to-end)

1. Capture – The Pi camera looks at a printed page (or screen). Good lighting = better OCR.
2. Enhance – OpenCV converts to grayscale and adaptive-thresholds the text for crisp letter shapes.

3. Read – Tesseract OCR extracts characters. (Optional language packs enable multi-language support.)
4. Send – The Pi streams plain-text lines over USB serial:
    - `MODE:B` + `SEND: …` → Braille
    - `MODE:M` + `SEND: …` → Morse
5. Feel – On the ESP32, our firmware maps each character to either:
    - a 6-bit Braille mask (six motors pulse in the exact pattern), or
    - Morse timing on two fingertips (left = dot, right = dash) with correct unit spacing.
6. Comfort & safety – Each pulse is capped (≈250–300 ms), with cool-down gaps and PWM “soft-start” so the motors are distinct but not harsh. A large capacitor on the motor rail keeps power stable.

Why this is different

- Dual-mode haptics: Braille and Morse in the same glove—switchable on the fly. That makes the device useful to both Braille learners/users and anyone who needs silent signaling.
- Camera-to-touch in one minute: No cloud; no phone dependency. The Pi handles vision locally, so it works offline.
- Composable, low-cost hardware: Every part is standard from the hackathon list. It’s repairable and easy to extend (extra sensors, language packs, or a larger OCR field).
- Human-centered: We tuned vibration length, strength, and spacing for comfort and reliable discrimination.

Key use cases

1. Reading physical documents
    - Receipts, restaurant menus, classroom handouts, forms at a clinic or airport. Place under the camera → feel the text on your fingertips.
2. Accessible travel / multi-language
    - A traveler who is blind can scan foreign text, translate on the Pi (optional local language packs), and output Braille in their preferred language. This lowers the “I can’t read this sign/menu” barrier abroad.
3. Braille learning & practice

    - Teachers can type words and students feel the patterns instantly. The dual mode lets novices start with Morse (timing is simpler) and graduate to Braille (spatial patterns).

4. Silent communication
    - Discreet Morse prompts during noisy events, backstage production, search-and-rescue, or other professional contexts where quiet, eyes-free cues are safer than whispers or screens. (We emphasize respectful, lawful, and ethical use.)

5. Hands-busy environments
    - Technicians, cyclists, or factory workers can receive short Morse/Braille alerts without looking away or uncovering ears.

What's inside (tech snapshot)

Electronics

- ESP32-DevKitC (motor control + serial)
- L298N motor drivers (2 channels per board) → six coin/cylindrical vibration motors
- Raspberry Pi 4 + NoIR camera (OCR)
- Breadboard, jumpers, 220–560 µF cap across motor +5 V↔GND for surges
- Separate 5 V rails for Pi and motors; common ground between Pi, ESP32, and driver

Firmware & software

- ESP32 firmware (Arduino IDE):
    - Parses `MODE:` and `SEND:` commands
    - Braille: 6-bit mask → simultaneous pulses on the correct dots
    - Morse: left fingertip = dot (short), right = dash (long), with ITU timing
    - PWM "kick + hold" profile for crisp yet gentle feel
- Raspberry Pi (Python):
    - libcamera / Picamera2 for capture
    - OpenCV (contrast, adaptive threshold)
    - Tesseract OCR (English by default; add language packs as needed)
    - Streams text to ESP32 over `/dev/ttyUSB*` or `/dev/ttyACM*`

Impact

- Access: Converts almost any printed surface into tactile information—no special paper, embossers, or proprietary readers required.
- Affordability: Built from common, replaceable parts already in many makerspaces.
- Privacy & resilience: Runs offline; no cloud dependency; works in loud, dark, or crowded places where audio and screens fail.

Responsible use & constraints

- We cap vibration duration/duty cycle to avoid skin fatigue.
- The device is not a medical aid; it complements, not replaces, existing tools.
- Silent communication uses are framed for safety, accessibility, education, and lawful professional contexts.
- OCR accuracy depends on lighting, font size, and camera angle; we mitigate with preprocessing and clear UX guidance.

Roadmap (post-hackumass)

- Auto language detection & translation packs on-device
- Continuous scan line (sweep-to-speech-to-Braille) for longer pages
- Haptic “cursor” to track position on the page
- Batteryized Pi carrier and compact camera mount
- Teacher dashboard for Braille lessons and progress tracking

Why this stands out

We didn’t just build another demo—we built a complete, feel-first reading pipeline that’s:

- Practical (reads real documents),
- Inclusive (Braille learners/users and non-visual silent comms),
- Portable & offline (no internet required).