SEVA System Analysis

Description of the Multimodal System

SEVA (Speech-Enabled Visual Assistant) are AI-powered smart glasses for people with visual impairments. They provide multimodal interaction through voice, vision, and audio feedback to support daily tasks. SEVA's three key areas include:

- **Explore:** Real-time navigation/guidance and face recognition to identify family and friends.
- **Experience:** Converts documents or handwritten text to speech and offers AI support via ChatGPT
- **Empower:** SOS alerts, emergency contact features, and up to 8x magnification.

SEVA enhances accessibility, independence, and social connection for users navigating the world with sight loss.

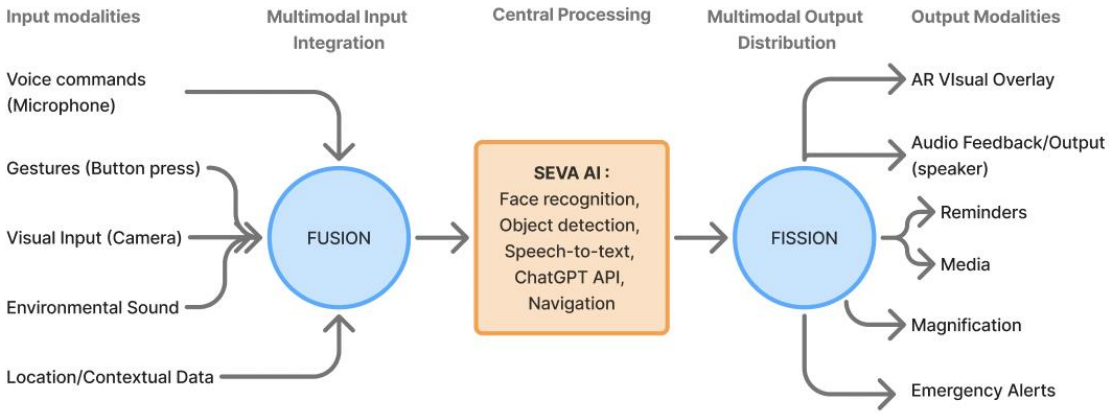

Figure 1 – Multimodal Interaction Diagram of SEVA

Analysis of Interactive System Design

Norman’s Design Principles (Norman, 1998):

Conceptual Model:

A conceptual model helps users understand and predict how a system works. SEVA adopts a voice/gesture-driven interaction model, aligning with user expectations for assistive tech like Siri or Alexa – reducing the learning curve for visually impaired users.

To improve the conceptual model, SEVA could add an onboarding tutorial through audio narration to guide first-time users.

Visibility:

Visibility includes affordances, signifiers, and constraints (Norman, 1999). It refers to how easily users can perceive what actions are possible and how to perform them. Affordances suggest what can be done, signifiers guide the user’s attention, and constraints prevent errors – all helping users understand the system at first interpretation. For example, when a user looks at artwork and presses a button, SEVA describes the art. The button signals the action is possible, and the audio confirms success.

To improve visibility adding subtle haptic feedback when buttons are touched would help confirm the action is complete to the user.

Natural Mapping:

Natural mapping means controls and responses correspond logically. When a user points SEVA at a document and gives it a command, it reads the text aloud – creating an intuitive interaction.

To improve the conceptual model, SEVA could use context prompts and head gestures (tilting to zoom) to make actions more intuitive without voice commands.

Feedback:

An effective system provides immediate, relevant feedback. For example, SEVA may say “Laptop detected in front of you” including direction, distance, and colour – helping users act confidently.

Feedback could be enriched by integrating spatial audio – useful for users in fast-moving or noisy environments.

Concept of Breakdown – Phenomenology

Stage 1: "Ready to hand"

SEVA functions transparently, blending into the user's activity. For example, the text-to-speech feature works smoothly without conscious thought – reflecting Heidegger's read-to-hand state (Dourish, 2001, pp. 106-110)

Stage 2: "Present-to-hand"

When a familiar voice command changes, the user must stop and re-learn the action. The system draws attention to itself, disrupting flow and becoming present-at-hand.

Stage 3: "Being-at-hand"

If SEVA misidentifies someone, trust breaks down. The tool becomes unreliable, blocking the user's task – a deeper form of disruption where breakdown reveals assumptions about the system (Dourish, 2001).

Design suggestion

To preserve flow (Csikszentmihalyi, 1990), SEVA could offer subtle help after repeated breakdowns – e.g., "Need help with this command?". This keeps users engaged and allows them to recover quickly without disrupting their task.

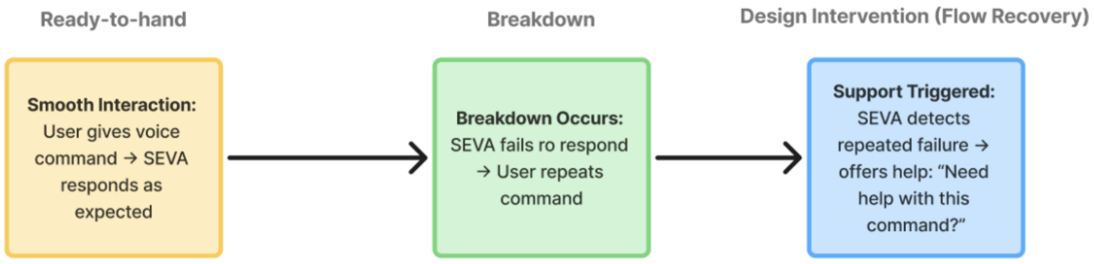
Ready-to-hand
Breakdown
Design Intervention (Flow Recovery)
Smooth Interaction: User gives voice command → SEVA responds as expected
Breakdown Occurs: SEVA fails ro respond → User repeats command
Support Triggered: SEVA detects repeated failure → offers help: "Need help with this command?"

Figure 2: Breakdown with intervention diagram

Figure 2 illustrates this intervention: when a command fails repeatedly, SEVA detects it and offers gentle support. This aligns with Heidegger's view of breakdown revealing the system (Dourish, 2001) and help maintain flow by keeping the user 'in the zone'.

Distributed Cognition Analysis

Traditional Human-Computer Interaction (HCI) models focus on individual interactions with technology without considering how cognition is distributed. In contrast, Distributed Cognition (DC) extends these models by incorporating more than individual thinking. Based in cognitive psychology, DC looks at the whole system — how information moves between people and tools, and how it's transformed to complete tasks (Perry, 2003). It's useful because it includes people, tools, and the environment in one comprehensive analysis. DC is ideal for analysing SEVA, which involves human decision-making, AI processing, and environmental interaction. SEVA offloads tasks to its AI and sensors. Therefore, analysing SEVA through DC shows how users and technology work together to reduce cognitive effort and support the visually impaired.

Unit of analysis

The unit of analysis in Distributed Cognition helps reveal how SEVA functions as a system rather than just a tool. By identifying the people, technologies, environments, and interactions involved, we can see how cognitive work is shared and offloaded. For example, rather than the user remembering everything, tasks are supported by SEVA's interface, AI processing, and environmental cues. This perspective shifts the focus from individual use to collaborative cognition, where success depends on how effectively information is distributed and coordinated across all elements shown in the table below.

People	Artifacts	Environment	Interactions
Primary user: Visually impaired individual **Support agents:** ChatGPT, Family, colleagues etc	**SEVA glasses: -** Camera, microphone, speaker, gesture control **Audio feedback system** **Waveguide display for AR overlays** **Interface components** (menu, alerts, etc) **Saved profiles and preferences** **Bluetooth and WIFI** **8MP Camera**	**Physical spaces:** offices, transit environments, unfamiliar locations **Task constraints:** safely navigating, identifying people or objects, reading **Supportive infrastructure:** Availability of Wi-Fi, GPS, Bluetooth connectivity	**User-Device:** Multimodal interactions (audio, voice, touch) **User-People:** Coordination with coworkers, family, friends **User-Environment:** Navigating dynamic surroundings with SEVA's feedback

Memory Representations

In DC analysis, identifying internal, external, and social memory shows how cognitive tasks are distributed across people, tools, and interactions. SEVA spreads memory across the user (internal), the device (external), and others (social). Internal memory involves personal recall; external memory is stored and displayed by SEVA; and social memory is supported through collaboration and shared knowledge. The table below provides examples of how these memory types operate in SEVA:

Internal	External	Social
Remembering voice commands	Stored face profiles and contact names	Team member reads aloud presentation slides during meeting
User's procedural habits	Visual overlays in waveguide display	Friend confirms that the object SEVA detected is correct
Recalling previous SEVA feedback	Saved preferences and user settings	Tech support helps user learn new feature through spoken steps
Understanding SEVA's feedback vocabulary	Navigation instructions and route guidance	Friend reminds user to activate SEVA before it's needed
Error recovery steps	Real-time object and obstacle detection cues	Chat GPT conversation
Personal knowledge of contacts, routines or surroundings	Reminder alerts	
Deciding which feature to use based on the situation.	SEVA confirmation messages	

Information Flow

The information flow in SEVA shows how user input is sensed, processed, and turned into feedback. As shown in Figure 3, data moves between the user, sensors, AI, and output channels in a continuous loop. This allows SEVA to adapt in real time to support dynamic interaction situations. It reflects how cognition is distributed across all the system's components, reducing user effort.

1. User input (Voice or Gesture)
    a. The user interacts through voice commands or gestures, such as pressing a button on the glasses. SEVA detects this input via its microphone or touchpad.
2. Sensors and Data Capture
    a. Visual and audio sensors collect environmental data. The camera captures scenes, while the microphone detects sounds and voice inputs.
3. AI Processing and Data Interpretation:
    a. SEVA's AI processes commands, object data, or text. Depending on the task, it applies speech-to-text, object recognition, or face recognition algorithms.
4. Data retrieval and Storage Access
    a. The system accesses stored profiles and preferences to recognise faces or adjust settings. For external requests like news or media, it connects to online sources.
5. Response generation
    a. Based on the interpreted input and retrieved data, SEVA creates an appropriate response such as audio feedback, a visual overlay, or both.
6. Output Delivery
    a. Responses are delivered through audio (via speaker) and visual cues (via waveguide display), supporting features like navigation and magnification.
7. Real-time adjustments
    a. SEVA continuously monitors the environment, updating feedback if new objects or people appear. User interactions can also prompt SEVA to reprocess data or repeat actions.

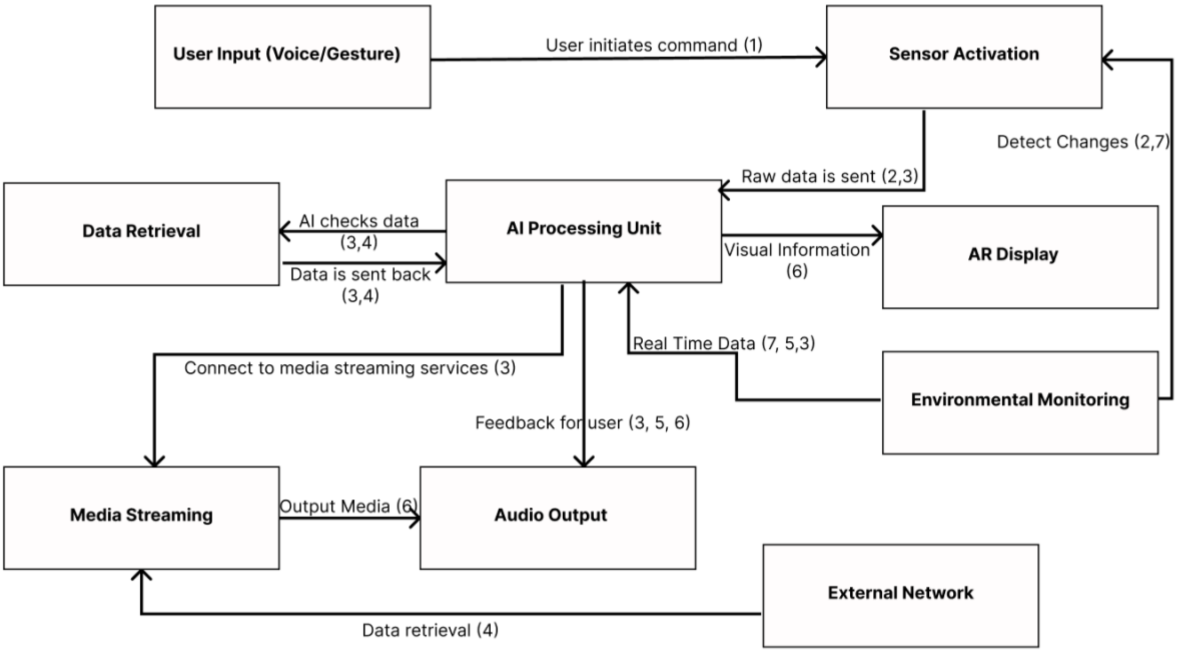

Figure 3: Information Flow Points Diagram

Computations

The table below applies Marr's 3-Level Hypothesis (Marr, 1982), which breaks down how a system processes information into three levels: computation (what the system does), representation (how information is displayed or structured), and implementation (the physical attributes involved). This framework is useful in DC analysis as it highlights how cognition is not just internal, but shaped by how tasks are performed and supported in context.

Mapping SEVA's features to Marr's framework shows how tasks such as navigation and face recognition are distributed across user input, system processing, and sensory output. It also demonstrates how SEVA reduces cognitive load by transforming complex tasks into simple, accessible representations (audio, AR overlay), allowing visually impaired users to interact with their surroundings, and complete desired tasks more efficiently.

Task	Computation	Representation	Implementation
Voice command interpretation	Decoding speech input into structured commands	Audio confirmation message	Microphone, speech recognition module
Face recognition	Matching detected face with stored profiles	AR display with name	Camera, AR display, database, face recognition AI
Navigation guidance	Calculating a safe and optimal route	Audio directions, AR display	GPS, pathfinding engine, AR display, speaker
Object detection	Identifying objects and their distance	Audio information message	Camera, object recognition algorithm, distance algorithm.
Text Reading	Converting printed text to audio	Spoken text	Camera, text to speech module
Magnification gesture	Recognising command to zoom in/out	Enlarged visual overlay on display	Touchpad, gesture detector, AR display
Emergency alert	Detecting fall or SOS trigger and initiating alert protocol	Audio confirmation message	Fall sensor, emergency API, speakers

Design Suitability

Based on the combined analysis, SEVA is highly suitable for visually impaired users. Its multimodal interaction (voice, gesture, and audio feedback) reduces cognitive load by distributing tasks across the system and environment. SEVA also adapts well to real-world situations by offering multimodal input and feedback. For example, if voice commands fail, users can rely on gesture control instead. This backup system keeps SEVA reliable and safe, even in unpredictable situations.

Phenomenological analysis shows SEVA supports seamless interaction during ready to hand use, while offering helpful responses when breakdowns occur. Through the suggested system improvements (e.g. haptic feedback), it would enable users to return to a state of flow (Csikszentmihalyi, 1990) and continue tasks without anxiety or confusion.

SEVA also aligns with Norman's design principles, offering clear feedback, intuitive mapping, good visibility, and an understandable conceptual model. Its design supports not just ease of use, but also user autonomy. By reducing reliance on others and offering reliable, real-time feedback, this allows users to make independent decisions with confidence. As a distributed cognitive system, SEVA balances human input and machine intelligence, creating an experience that feels both empowering and intuitive.

References

Perry, M. (2003). Distributed cognition. In J. M. Carroll (Ed.), HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science (pp. 193-223). Morgan Kaufmann. https://doi.org/10.1016/B978-155860808-5/50008-3

Rogers, Y., & Ellis, J. (1994). Distributed cognition: An alternative framework for analysing and explaining collaborative working. *Journal of Information Technology*, *9*(2), 119-128. https://www.dourish.com/classes/ics234bs03/14-RogersEllis-DistCog.pdf

Marr, D. (1982). Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. W.H. Freeman. Available at MIT Press: https://mitpress.mit.edu/9780262514620/vision/

Norman, D. A. (1999). Affordance, conventions, and design. *Interactions*, 6(3), 38–43. https://doi.org/10.1145/301153.301168

Norman, D. A. (1988). *The design of everyday things* [PDF]. Basic Books.

Dourish, P. (2001). *Where the action is: The foundations of embodied interaction*. MIT Press.

Csikszentmihalyi, M. (1990). *Flow: The psychology of optimal experience*. Harper & Row.

Word count (excluding references and tables and diagrams): 1364