

Module : Robotique éducatives et applications

Projet n°7 : « Reading Eye » Assistant de lecture pour malvoyants avec IA embarquée

I. Objectif :

Concevoir un dispositif d'assistance portable capable d'aider les personnes malvoyantes à lire des textes (imprimés ou manuscrits). Le système capture une image du texte via une caméra, analyse le contenu grâce à l'intelligence artificielle (OCR et Deep Learning), et restitue le texte sous forme de synthèse vocale (Text-to-Speech) via des écouteurs.

II. Architecture du système :

Entrée : Module Caméra (Capture d'image) + Bouton poussoir (Déclencheur).

Traitement : NVIDIA Jetson Nano (ou Raspberry Pi 4 avec modifications).

Sortie : Signal Audio (Synthèse vocale via Jack 3.5mm).

III. Matériel nécessaire

Composant	Description
NVIDIA Jetson Nano	Unité centrale (GPU requis pour l'IA en local)
Caméra (Pi Cam V2 ou USB)	Capteur 8MP (Grand angle recommandé)
Carte microSD (32Go ou plus)	Système d'exploitation et modèles d'IA
Adaptateur Audio USB	Pour connecter des écouteurs ou un haut-parleur
Bouton poussoir	Déclencheur physique pour la capture
Batterie (Power Bank)	5V, 2.5A min pour la portabilité

IV. Logiciels / Bibliothèques requis :

Logiciel / Bibliothèque	Rôle
Ubuntu (JetPack SDK)	Système d'exploitation Linux pour Jetson
OpenCV	Prétraitement d'image (correction fisheye, seuillage)
Tesseract / Google Vision API	Moteurs OCR pour la reconnaissance de texte
TensorFlow / Keras	Exécution du modèle d'IA (manuscrit)
pyttsx3 / gTTS	Bibliothèques de synthèse vocale (Text-to-Speech)
Python	Langage de programmation principal

V. Fonctionnalités principales

1. Capture et Prétraitement

- L'utilisateur place le document sous la caméra et appuie sur le bouton.
- Le système capture l'image et applique des corrections (redressement, suppression du bruit, passage en niveaux de gris) via OpenCV.

2. Reconnaissance de Texte (OCR)

- **Mode Texte Imprimé :** Utilisation de Tesseract pour extraire les caractères standardisés.
- **Mode Manuscrit :** Utilisation d'un modèle de Deep Learning (CRNN - Convolutional Recurrent Neural Network) entraîné sur la base de données IAM pour déchiffrer l'écriture manuelle.
- **Mode Hybride (Connecté) :** Si le Wi-Fi est disponible, appel aux API Google Cloud Vision pour une précision accrue.

3. Synthèse Vocale

- Le texte extrait est converti en chaîne de caractères.
- Le moteur TTS (Text-to-Speech) lit le texte à haute voix via la sortie audio.
- Gestion de la langue (Anglais/Français selon la configuration).

4. Gestion de l'état (Feedback)

- Signaux sonores simples pour indiquer le début du traitement et la fin de la lecture.
- Gestion des erreurs (ex: "Aucun texte détecté").

VI. Exigences non fonctionnelles :

- **Performance :** Le délai entre la capture et le début de la lecture doit être inférieur à 5 secondes (en mode local).
- **Portabilité :** Le dispositif doit être autonome énergétiquement (batterie) et contenu dans un boîtier compact.
- **Ergonomie :** Interface simplifiée à un seul bouton pour une utilisation "sans les yeux".

VII. Plan de test

- Tester la reconnaissance sur différents types de polices (Arial, Times New Roman) et tailles.
- Évaluer la performance sur du texte manuscrit (lisible vs illisible).
- Comparer la précision en mode "Hors-ligne" (Edge AI) vs "En ligne" (Cloud API).
- Tester l'autonomie de la batterie en utilisation continue.

VIII. Améliorations possibles

- Traduction instantanée du texte lu dans une autre langue.
- Détection d'objets (ex: reconnaître un billet de banque ou une bouteille) en plus du texte.
- Ajout d'un capteur de distance pour aider l'utilisateur à bien positionner le livre.

Ressources :

[1] Exemple de projet : <https://www.hackster.io/bandofpv/reading-eye-for-the-blind-with-nvidia-jetson-nano-8657ed>

Pour plus d'informations ou en cas de besoin d'aide supplémentaire, n'hésitez pas à me contacter à l'adresse suivante : ah.reragui@edu.umi.ac.ma