

Note concernant les diapos

Dimension : 4/3

Faire tous les diapos avant d'appliquer le nombre de page (la pagination)

premiere page :

Titre : Étude d'un lecteur de code à barres

Diapo 2:

objectif : mettre en place un programme capable de lire un code à barres en prenant en comptes les deficultés associées

Simplification : Image en noir et blanc de bonne qualité (et bien contrastée) et réduite à la zone des codes barres. (le traitement de l'image sera détaillé plus loin)

Problemes pontentiel : (mettre en couleurs les problemes à traiter, ordonner par niveaux de difficulté)

- **contraste insuffisant entre les barres**
- **scalabilité** : le programme doit s'adapter à la variabilité des dimensions
- **largeur d'espacement inegale**
- **distorsion géométrique** (inclinaison ou rotation) : impacts sur le calcul des largeurs des barres
- **distorsion des barres** : de mm
- bruits des pixels

- **barres manquantes ou incomplete** : Si certaines barres sont tronquées ou partiellement visibles, la séquence de lecture sera faussée

Description des problemes pour l'explication orales:

Distorsion géométrique

- **Inclinaison ou rotation** : Si le code-barres est légèrement incliné ou tourné, cela peut rendre difficile la détection des barres verticales.
- **Perspective** : Si l'image est prise sous un angle (non orthogonal au code-barres), les barres peuvent apparaître déformées, compliquant leur lecture.

Distorsion des barres

- **Largeur irrégulière des barres** : Si les barres du code-barres sont déformées, soit parce qu'elles sont étirées ou compressées, il devient difficile de maintenir la bonne interprétation des séquences larges/étroites.

Contraste insuffisant entre les barres

- il pourrait y avoir des zones localement où le contraste est plus faible, rendant difficile la détection des transitions entre les barres noires et blanches.

Largeur d'espacement inégale

- Dans les cas où les espaces blancs entre les barres varient en taille de manière irrégulière (par exemple en raison de petites erreurs d'impression ou de numérisation), cela peut induire des erreurs dans l'interprétation du code.

Diapo 3: Explication breve de la norme EAN-13

- Contient 2 images centrales à editer avec gimp pour nommer les champs respectifs
- Explication de la structure d'un code barre de type EAN-13

- le code binaire en fonction de plage A, B, C
- delimitation et nomination des zones
- Puis utilisation de la page sur la Norme EAN-13 pour generer un texte bref explicatif

Diapo 4 : Plan et explication des modules

denoising : Le "bruit" dans une image, c'est comme des "parasites" visuels qui viennent troubler l'image originale. Ces parasites sont des pixels de couleur ou d'intensité aléatoire qui ne font pas partie de l'image qu'on veut observer. Le bruit peut rendre les images floues ou granuleuses, surtout dans des photos prises en basse lumière ou avec de l'équipement moins performant.

Diapo 5: Probleme 1 et solution

Autre exemple de diapo

1. Denoising (Dénombrement du Bruit)

- **Objectif** : Supprimer le bruit tout en préservant les détails importants, notamment les contours, pour une meilleure lecture de l'image.
- **Technique** :
 - Utilisation d'un **filtre bilatéral**, qui réduit le bruit tout en préservant les contours, suivi d'un **filtre médian** pour finaliser le débruitage.
- **Illustration** : Affiche une image avant et après le débruitage pour montrer comment les contours sont conservés.

2. Contrast Improvement (Amélioration du Contraste)

- **Objectif** : Améliorer le contraste de l'image pour rendre les zones sombres plus distinctes et faciliter la détection des barres.

- **Technique :**
 - Application de la **normalisation de l'histogramme** qui étale l'intensité des pixels sur toute la gamme, ce qui rehausse les contrastes.
- **Illustration :** Montre une image avant et après la normalisation d'histogramme pour illustrer l'amélioration visuelle du contraste.

3. Correction de la Rotation

- **Objectif :** Redresser les images inclinées pour que les codes-barres soient parfaitement horizontaux ou verticaux, facilitant la détection.
- **Technique :**
 - **Filtre Canny** pour détecter les bords.
 - **Transformée de Hough** pour détecter les lignes. On calcule l'angle moyen des lignes détectées pour estimer l'inclinaison de l'image, puis on applique une rotation pour redresser l'image.
- **Illustration :** Montre une image inclinée puis redressée, en expliquant brièvement comment l'angle de rotation est calculé.

4. Détection de la Texture du Code-Barres (Transformée de Fourier)

- **Objectif :** Identifier les zones de l'image contenant un code-barres en repérant les caractéristiques de texture spécifiques.
- **Technique :**
 - **Transformée de Fourier** pour analyser les fréquences dans l'image.
 - En définissant une zone d'intérêt autour des fréquences centrales, et en appliquant un filtre passe-haut pour accentuer les caractéristiques de texture des codes-barres, l'algorithme détecte les pics de fréquence associés aux motifs de barres.
- **Illustration :** Montre une image de la zone de fréquence analysée et explique brièvement la méthode de filtrage pour détecter les textures caractéristiques.

5. Segmentation

- **Objectif** : Diviser l'image en fenêtres verticales et repérer celles contenant des codes-barres.
- **Technique** :
 - Une **fenêtre mobile verticale** de hauteur fixe parcourt l'image. Chaque fenêtre est analysée pour vérifier la présence de texture de code-barres, en utilisant les mêmes critères que l'étape précédente.
 - Les fenêtres détectées sont sauvegardées pour une analyse plus poussée.
- **Illustration** : Montre un exemple de balayage et indique une ou deux fenêtres avec des codes-barres pour illustrer la méthode.

6. Binarisation

- **Objectif** : Convertir l'image en noir et blanc pour simplifier la structure de l'image et faciliter l'analyse des barres.
- **Technique** :
 - **Binarisation d'Otsu** : une technique de seuillage automatique qui convertit l'image en noir et blanc selon un seuil calculé en fonction de l'histogramme des niveaux de gris.
- **Illustration** : Montre une image avant et après binarisation, pour expliquer comment cette étape permet de mieux discerner les barres du fond.

Structure de la Présentation

1. **Introduction** : Expliquer brièvement pourquoi ces étapes sont nécessaires pour détecter des codes-barres (amélioration de l'image, détection des textures).
2. **Étapes du Prétraitement** : Chaque technique expliquée de façon visuelle, avec un avant-après pour chaque transformation.
3. **Résumé** : Rappeler les bénéfices de chaque étape et comment elles s'enchaînent pour permettre une détection efficace des codes-barres.