

Projet 1. Analyse et manipulation d'histogrammes

Description

On souhaite étudier l'histogramme d'une image et comprendre son rôle dans l'amélioration du contraste. Vous allez :

1. calculer l'histogramme manuellement puis avec une fonction,
2. tracer l'histogramme cumulé,
3. appliquer l'égalisation,
4. comparer avec la méthode CLAHE.

Questions / Travaux demandés

1. Charger une image en niveaux de gris.
2. Calculer l'histogramme manuellement (boucles).
3. Vérifier avec `cv2.calcHist` ou `imhist`.
4. Tracer l'histogramme cumulé et l'interpréter.
5. Appliquer l'égalisation globale : `equalizeHist`.
6. Appliquer CLAHE : expliquer son intérêt.
7. Comparer l'image originale, l'image égalisée et l'image CLAHE.
8. Calculer la dynamique (min/max) avant et après l'égalisation.

À rendre

- Figure : image originale + versions améliorées
- Histogrammes
- Analyse de 8–10 lignes
- Rapport

Projet 2. Filtrage linéaire : lissage**■ Description**

On étudie les effets de plusieurs filtres linéaires sur une image bruitée :

- Moyennneur
- Gaussien
- Nagao (filtrage avancé)

? Questions

1. Ajouter du bruit gaussien à une image.
2. Appliquer le filtrage moyennneur pour plusieurs tailles (3×3 , 5×5 , 9×9).
3. Appliquer un filtre gaussien ($\sigma = 1, 3, 5$).
4. Appliquer le filtre Nagao et expliquer pourquoi il préserve les contours.
5. Comparer les résultats visuellement et avec le PSNR.
6. Identifier le filtre "optimal" selon la nature du bruit.

📄 À rendre

- Tableau PSNR
- Figures comparatives
- Courte conclusion : quel filtre et pourquoi ?

Projet 3. Filtrage non linéaire : filtre médian**■ Description**

Le filtre médian est particulièrement efficace pour supprimer le bruit “sel & poivre”. L’objectif est de comparer le filtre médian à un filtre linéaire.

? Questions

1. Injecter du bruit impulsionnel (sel & poivre) pour différentes densités : 5%, 10%, 20%.
2. Appliquer un filtre linéaire (moyenneur).
3. Appliquer un filtre médian pour les mêmes tailles de fenêtre.
4. Comparer les résultats visuellement.
5. Calculer le PSNR et interpréter.
6. Expliquer pourquoi le filtre médian est meilleur sur ce type de bruit.

📄 À rendre

- Résultats filtrés
- Tableaux PSNR
- Explication de 5 lignes
- Rapport

Projet 4. Filtrage dérivateur : extraction de contours (sans segmentation)

■ Description

On veut extraire les contours d'une image à l'aide de filtres dérivateurs : Sobel, Prewitt, Laplacien, Canny.

L'objectif est de comprendre le comportement des détecteurs de contours.

? Questions

1. Convertir l'image en niveaux de gris.
2. Calculer les gradients Sobel (G_x , G_y) et magnitude.
3. Faire la même chose avec Prewitt et commenter la différence.
4. Appliquer le Laplacien : observer la sensibilité au bruit.
5. Laisser un léger flou gaussien puis refaire le Laplacien : analyser.
6. Appliquer Canny : tester seuils bas/haut différents.
7. Comparer les cartes de contours obtenues.

📄 À rendre

- Figures des différents détecteurs
- Tableau "sensibilité au bruit"
- Conclusion sur le détecteur le plus stable
- Rapport

■ Description

Une image a été dégradée par du bruit ou un flou.

Vous devez tester plusieurs méthodes de restauration et comparer les résultats.

? Questions

1. Simuler un flou de mouvement (kernel $1 \times N$).
2. Appliquer le filtre inverse (Inverse Filter).
3. Appliquer le filtre de Wiener : expliquer son principe.
4. Ajouter du bruit et recommencer les restaurations.
5. Comparer les résultats visuellement.
6. Calculer PSNR et SSIM pour évaluer la qualité.
7. Discuter pourquoi l'inverse filtering devient instable en présence de bruit.

📄 À rendre

- Images restaurées
- PSNR + SSIM
- Analyse critique
- Rapport

■ Description

Vous allez implémenter une version simplifiée du JPEG :
conversion YCbCr \rightarrow DCT \rightarrow quantification \rightarrow déquantification \rightarrow IDCT.

? Questions

1. Convertir l'image RGB en YCbCr.
2. Extraire le canal Y et le découper en blocs 8×8 .
3. Appliquer la DCT 2D bloc par bloc.
4. Appliquer une matrice de quantification (fournie ou simple).
5. Déquantifier puis appliquer l'IDCT.
6. Recoller les blocs et reconstruire l'image.
7. Comparer l'image originale et la reconstituée.
8. Mesurer :
 - Ratio de compression
 - PSNR
 - SSIM

📄 À rendre

- Explications des étapes
- Codes utilisés
- Images comparatives
- Mesures quantitatives
- Rapport

■ Description

Ce projet vise à étudier le comportement d'une image lorsqu'on change d'espace colorimétrique.

On va aussi modifier certains paramètres (saturation, luminosité...).

? Questions

1. Convertir une image RGB en HSV et LAB.
2. Visualiser séparément les canaux (H, S, V) et (L^* , a^* , b^*).
3. Modifier la saturation ($\times 0.5$, $\times 1.5$).
4. Modifier la luminosité (canal V ou L^*).
5. Equaliser uniquement le canal L^* et observer le résultat.
6. Reconvertir en RGB et interpréter les modifications.

📄 À rendre

- Images des canaux
- Images modifiées
- Commentaires sur l'effet de chaque espace coloré
- Rapport

■ Description

L'objectif est de détecter les zones de peau dans une image en utilisant des **seuils fixes** dans les espaces RGB, HSV et YCbCr. Aucun apprentissage : uniquement des règles basées sur les valeurs de couleur.

? Questions

1. Charger une image contenant un ou plusieurs visages.
2. Implémenter une détection naïve de peau en **RGB** (intervalles simples).
3. Convertir l'image en **HSV** : définir les intervalles usuels de peau (H, S, V).
4. Refaire en **YCbCr** avec seuils (Cb, Cr).
5. Comparer les trois masques obtenus.
6. Expliquer pourquoi HSV et YCbCr sont plus robustes que RGB.
7. Calculer le pourcentage de pixels détectés comme peau.

📄 À rendre

- Masques de peau RGB, HSV, YCbCr
- Comparaison + discussion (10 lignes)
- Rapport

Projet 10. Étude approfondie de la convolution (implémentation manuelle + OpenCV)

■ Description

On analyse la convolution 2D en l'implémentant à la main, puis en la comparant à `cv2.filter2D` ou `imfilter`.

? Questions

1. Écrire une fonction de convolution 2D manuelle (sans utiliser une fonction existante).
2. Tester sur un noyau de lissage 3×3 .
3. Tester avec un noyau Laplacien.
4. Comparer le résultat avec `filter2D`.
5. Étudier ≥ 3 modes de padding : zéro, miroir, répété.
6. Mesurer le temps d'exécution manuel vs OpenCV.

📄 À rendre

- Code de la convolution manuelle
- Comparaison manuelle / OpenCV
- Discussion sur le padding
- Rapport

Projet 11. Comparaison de noyaux de lissage : Mean, Gaussian, Box, Weighted Average

Description

Créer un “banc d’essai” de plusieurs types de noyaux pour le lissage.

Questions

1. Construire manuellement plusieurs noyaux : Moyenneur, Box, Gaussien, Moyenne pondérée.
2. Appliquer chaque filtre à une image bruitée.
3. Observer l’effet de l’augmentation du kernel (3×3 , 5×5 , 7×7).
4. Calculer PSNR pour chaque filtre.
5. Analyser lequel conserve le mieux les détails.

À rendre

- Tableaux PSNR
- Figures de comparaison
- Conclusions (≈ 8 lignes)
- Rapport

■ Description

Dans ce projet, vous étudiez et comparez plusieurs filtres dérivatifs.

? Questions

1. Implémenter Sobel en x et y manuellement (avec convolution).
2. Appliquer : Sobel, Scharr, Prewitt, Roberts, Laplacien.
3. Tester l'effet du bruit (bruit gaussien et sel/poivre).
4. Pré-lisser l'image avant les dérivatifs.
5. Déduire quel filtre est le plus robuste.

📄 À rendre

- Figures des dérivatifs
- Analyse de bruit
- Commentaire final
- Rapport

■ Description

Étude des opérateurs morphologiques fondamentaux (érosion, dilatation, ouverture, fermeture) appliqués à des **masques synthétiques**.

? Questions

1. Créer des images binaires simples : carré, cercle, lignes.
2. Appliquer érosion et dilatation avec différents SE (3×3 , 5×5 , disque).
3. Observer les effets de l'ouverture (suppression détails) et fermeture (remplissage).
4. Appliquer les mêmes filtres sur une image réelle binarisée.
5. Comparer l'effet selon la taille et la forme du structurant.

📄 À rendre

- Images avant/après
- Analyse des effets du SE
- Discussion sur les cas d'usage
- Rapport

■ Description

L'œil humain a une réponse logarithmique, le capteur une réponse linéaire.
On étudie l'effet du gamma sur une même image.

? Questions

1. Appliquer un gamma : 0.5, 1, 2, 2.2, 3.
2. Tracer l'histogramme pour chaque gamma.
3. Comparer le contraste perçu.
4. Expliquer pourquoi les écrans appliquent un gamma ≈ 2.2 .
5. Dédire l'importance de la correction Gamma.

📄 À rendre

- Figures gamma
- Histogrammes
- Une mini-analyse perceptuelle
- Rapport

■ Description

Comprendre comment chaque bruit affecte une image.

? Questions

1. Générer du bruit :
 - Gaussien
 - Poisson
 - Speckle
 - Sel & poivre
2. Visualiser les histogrammes.
3. Analyser l'impact visuel sur les détails.
4. Proposer le meilleur filtre pour chaque bruit.
5. Justifier les choix.

📄 À rendre

- Histogrammes par type de bruit
- Analyse comparative
- Rapport

■ Description

Étudier comment les contours varient selon les canaux R, G, B, H, S, V.

? Questions

1. Séparer l'image en canaux RGB et HSV.
2. Appliquer Sobel sur chaque canal.
3. Comparer : quel canal contient le plus d'information de contours ?
4. Reconstituer une carte de contours fusionnée (max ou moyenne).
5. Interpréter les résultats.

📄 À rendre

- Figures des contours par canal
- Fusion finale
- Analyse
- Rapport