

Segmentation couleur pour le suivi d'objets en temps réel

Partie 1 : Segmentation couleur de l'image pour la reconnaissance d'objet

La segmentation couleur d'images est une opération qui vise à rassembler des pixels entre eux suivant un critère de couleur. Cette segmentation est fondée sur la classification ou le seuillage des pixels en fonction de leur intensité sur les trois canaux de couleurs R, V, B (Rouge, vert, bleu). Dans un premier temps, il est demandé d'écrire un programme qui permet la classification des régions de la scène selon une couleur prédéfinie qui est la couleur rouge. Toutes les régions de l'image ayant cette couleur sont segmentées. Afin de réaliser ce traitement, les opérations suivantes doivent être appliquées sur l'image :

1. Ouvrir le flux vidéo, vérifier si le flux est correctement ouvert et que l'image acquise n'est pas vide.
2. Modifier la taille de l'image à 340x480 en utilisant une interpolation linéaire et des facteurs d'échelle nuls.
3. Afficher le flux vidéo des images redimensionnées.
4. Convertir l'espace couleur de l'image du RVB en TSL (Teinte, Saturation, Luminosité).
5. Retrouver les intervalles de valeurs T, S, L correspondant à la couleur rouge.
6. Écrire une fonction qui permet de retourner une image résultat dont les pixels sont dans l'intervalle de la couleur rouge.
7. Afficher l'image segmentée avec la couleur rouge.

Partie 2 : Détection d'objet et reconstitution de la forme

Dans cette partie, les traitements permettent de trouver des objets ayant une forme géométrique caractéristique qui les distingue des autres régions de l'image. En effet, un autre type de segmentation doit être appliqué afin de déterminer des formes à partir de la segmentation couleur. Cette approche cherche à exploiter les transitions détectables entre deux régions connexes, c'est la détection de contours. Pour permettre une détection d'objet de manière pertinente, les opérations suivantes sur l'image TSL, segmentée précédemment, doivent être appliquées :

1. Pour permettre de diffuser le maximum de pixels des régions disjointes d'un même objet, une opération de morphologie mathématique qui est la fermeture doit être appliquée. La fermeture remplit toutes les parties du fond qui sont plus étroites que l'élément structurant, ce qui permet d'avoir des régions plus structurées. Appliquer un filtre morphologique de fermeture dont l'élément structurant est un carré de taille 7 et ayant le point d'ancrage (3,3).
2. Détecter les contours de l'image en employant le filtre de Canny avec deux seuils hystérésis 100 et 200 et un opérateur d'ouverture égal à 3. Afficher les contours obtenus.
3. Les contours détectés présentent des discontinuités et ne forment pas des objets fermés. Dilater l'image avec un élément structurant carré de taille 7 et dont l'ancrage est au point (3,3).
4. Trouver les contours externes en utilisant l'approximation de Teh-Chin. Afficher les contours obtenus.

Partie 3 : Affinage de la détection pour le suivi robuste

L'objectif de cette partie est l'amélioration de la détection pour permettre une approximation fiable de l'objet d'intérêt. En effet, la segmentation couleur a permis la reconnaissance d'objets par la composante couleur pour les classer parmi les autres régions de l'image. Ensuite, la segmentation par contour a déterminé les formes susceptibles de constituer des objets d'intérêt. Cette approche basée sur les deux segmentations, couleur et contours, diffère du schéma classique de l'identification où la détection est appliquée avant la reconnaissance. Toutefois, ces traitements ont permis de caractériser les objets par leurs couleurs afin d'effectuer leur tracking en temps réel.

Afin d'améliorer la détection et l'identification, un filtrage de formes doit être appliqué. Les formes dont la taille est inférieure à un certain seuil de surface doivent être filtrées et les contours finaux sont approximés pour une meilleure représentativité de l'objet. Les traitements suivants sont appliqués pour permettre l'affinage de la détection et le suivi robuste :

1. Écrire une fonction qui permet de filtrer les formes par leurs surfaces, cette surface doit être au minimum de 500 pour que le contour soit retenu. Afficher les contours trouvés.
2. Calculer l'enveloppe convexe des formes obtenues afin d'affiner l'aspect géométrique des bords d'objets. Afficher le résultat de cette opération.
3. Recaler un objet sur l'objet d'intérêt trouvé dans l'image. Cet objet, dont la forme est définie par l'enveloppe convexe calculée dans la question 2, est rempli et de couleur rouge.

Partie 4 : Interface utilisateur pour le contrôle du flux vidéo

L'objectif de cette partie est d'ingérer une interface utilisateur qui permet de contrôler les images de la vidéo. Ainsi, il est primordial d'ajouter un curseur sur l'interface graphique qui permet de se déplacer dans le flux sur toute la longueur de la vidéo. Le curseur doit être dynamique : se déplace en fonction du temps écoulé et affiche le numéro de l'image.

Un curseur de défilement (slider) est attaché à la fenêtre graphique pour permettre à l'utilisateur de parcourir la vidéo plus rapidement. Afin de d'ajouter ce curseur et lui associer les fonctionnalités de contrôle, il est demandé d'écrire les instructions pour réaliser les opérations suivantes :

1. Créer un curseur de défilement dynamique dont la position est liée à celle de l'image dans la vidéo.
2. Remettre le curseur à sa position initiale (0) lorsqu'il atteint la fin de la vidéo.
3. À la position initiale, repositionner le flux d'images au début de la vidéo et arrêter la vidéo.
4. Afficher un message permettant de démarrer la vidéo en appuyant sur une touche clavier quelconque.
5. La fonction callback du curseur de défilement ralentit fortement les traitements en invoquant à chaque instant l'image par sa position dans la vidéo. Ajouter la condition qui permet de déplacer le curseur et l'incrémenter avec la vidéo sans ralentir les traitements réalisés dans les parties précédentes (temps de traitement sans et avec le curseur doivent être similaires).