

Rapport de Projet :

MANIPULATION D'UN LECTEUR VIDEO AVEC LES GESTES DES MAINS

Module :

Traitement numérique des images

Réaliser par :

Mariam OMRANE

Dorra CHAARI

Khalil KOLSI

Table des matières :

I. Problème & application.....	2
II. Méthodes et techniques utilisé	2
III. Hypothèses.....	4
IV. Implémentation.....	5
V. Limites	5
VI. Travaux futurs possibles	6

Table des figures :

Figure 1 : Capture d'image et conversion en niveaux de gris	3
Figure 2 : Blurring and Thresholding	4
Figure 3 : Détection de contour.....	4
Figure 4 : Détection des défauts de convexité	5

I. Problème & Application :

Notre projet était motivé par la capacité de notre algorithme à reconnaître certains types de formes de main. En nous basant sur un flux vidéo en direct, notre programme visait à reconnaître les formes des mains en temps réel pour manipuler un lecteur de vidéo. Les formes de main que nous choisissons d'identifier à notre programme sont un poing fermé, un signe de paix (deux doigts), trois doigts, quatre doigts et une main (cinq doigts). Certaines choses que nous voulions également mettre en œuvre que notre code expliquerait était la position de la main. Nous voulions un programme polyvalent capable de détecter les formes des mains, quelle que soit leur position (qu'elles soient latérales ou orientées vers le bas).

II. Méthodes et techniques utilisé :

Pour réaliser la reconnaissance des gestes de main, nous avons analysé les gestes de la main par détection de contour, dessin de coque convexe et recherche des défauts de convexité.

*les étapes suivi :

Étape 1 : Capturer les images et convertir les en niveaux de gris

Nous capturons les images de la main et les convertissons en niveaux de gris.

Question : Pourquoi niveaux de gris ?

Réponse : Nous convertissons une image de RVB en niveaux de gris, puis en binaire, afin de trouver la partie de l'image qui nous intéresse le plus pour le traitement de l'image.

En faisant cela, notre décision devient binaire : « oui, le pixel est d'intérêt » ou « non pixel n'a pas d'intérêt ».



Figure 1 : Capture d'image et conversion en niveaux de gris

Étape 2 : Blurring and Thresholding (Flou et seuillage)

En brouillant, nous créons une transition en douceur d'une couleur à une autre et réduisons le contenu des contours. Nous utilisons le seuillage pour la segmentation d'images, afin de créer des images binaires à partir d'images en niveaux de gris.



Figure 2 : Blurring and Thresholding

Étape 3 : Rechercher des contours

Pour trouver les bouts des doigts. Nous dessinons une coque convexe. La coque convexe est le convexe qui enferme la région de la main.



Figure 3 : Détection du contour

Étape 4 : Trouver les défauts de la coque convexe

Nous trouvons maintenant les points convexes et les points de défaut. Les points convexes sont généralement le bout des doigts. Les points de défaut constituent le point de déviation le plus profond sur le contour. De cette manière, nous pouvons trouver le nombre de doigts étendus, puis nous pouvons effectuer différentes fonctions en fonction du nombre de doigts étendus.



Figure 4 : Détection des défauts de convexité

III. Hypothèses :

Les hypothèses que nous avons formulées pour que notre système se comporte de manière fiable sont les suivantes :

1. Une seule main serait affichée (de sorte que toute la peau sous le poignet ne soit pas visible).
2. L'arrière-plan (les objets qu'il contient) est évidemment distinct de la couleur de la peau en termes de valeurs RVB.
3. Les doigts de la main s'affichent lorsque vous faites des gestes.
4. La main est le plus gros objet du cadre proportionnellement à tout autre objet.

IV. Implémentations :

***Environnement matériel :**

On utilise un ordinateur qui contient une webcam intérieure.

***Langage utilisée :**

Pour développer notre projet, nous avons utilisé python3 comme langage de programmation et nous avons utilisé les bibliothèques suivantes :

- opencv (cv2)
- numpy
- math
- pyautogui (bibliothèque qui contrôle par programme la souris et le clavier)
- time

***La tâche réalisée par chaque geste :**

- Deux doigts levés : mute ou unmute.
- Trois doigts levés : augmenter le volume.
- Quatre doigts levés : diminuer le volume
- Cinq doigts levés : lire ou pause

V. Limites :

Si votre main n'est pas correctement éclairée, tous les doigts ne seront pas capturés. Par conséquent, vous avez besoin d'un éclairage correct et vous devez également vous assurer que votre main est correctement positionnée de sorte qu'elle ne soit pas dans l'ombre.

Comme notre algorithme de reconnaissance des mains est basé sur des défauts de convexité, nous ne sommes pas en mesure de détecter un seul doigt. Lorsqu'il y a un doigt, il n'y a pas d'autres points pour détecter un angle. Ainsi, par défaut, un doigt est reconnu comme un poing (puisqu'il a des angles nuls).

Une autre limite à l'utilisation des défauts de convexité est que, si vous tenez un nombre quelconque de doigts ensemble, ils ne s'enregistreront pas comme des doigts individuels. Pour que notre algorithme fonctionne correctement, les doigts doivent être évasés.

Si un autre objet se trouve dans l'image est plus gros que votre main (c'est-à-dire si vous vous penchez et que votre visage est dans l'image), l'algorithme dessine des contours et des défauts sur celui-ci et tente de trouver le nombre de doigts.

VI. Travaux futurs possibles :

Certaines améliorations qui pourraient être apportées à l'avenir consisteraient à affiner la méthode de détection de la peau afin que divers objets ne s'enregistrent pas. Améliorer cela signifie que notre programme peut être utilisé dans n'importe quel environnement sans souci d'interférences négatives.

Une autre amélioration serait l'ajout de la fonctionnalité de détection d'un seul doigt. Cela serait très utile pour détecter si quelqu'un pointe ou donne éventuellement un coup de pouce à la caméra. En outre, nous pourrions affiner notre méthode de détection d'angle afin qu'elle puisse déterminer les doigts de votre main que vous tenez debout. Donc, si vous teniez votre pouce et votre petit doigt en l'air ("je vous aime" en langage des signes), il serait capable de détecter que ces deux doigts sont différents de ceux qui vous tiennent debout, l'index et le majeur. Avec cet ajout, vous pouvez potentiellement implémenter une variété de signaux de la main différents et le programme les reconnaîtra indépendamment du nombre de doigts levés.