

Université ABDELMALEK ESSAADI
école National des Sciences appliquée de Tanger

département de mathématique et informatique

Master Cyber Sécurité et cybercriminalité (MCSC)



Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Color Image Sequences

Najah Issam

Abdelaziz laayachi

année universitaire : 2019/2021

M. Ahmed Idrissi

Introduction

Reconnaissance des signes de la langue par le mouvement ou la posture de la main est un domaine actif dans la reconnaissance des gestes recherche pour l'interaction Homme-Machine (HCI). UNE le geste est un modèle spatio-temporel qui peut être statique ou dynamique ou les deux comme dans la Reconnaissance des Signes de la langue (SLR). Au cours des dernières années, plusieurs méthodes ont été suggérées pour la reconnaissance de la langue des signes de la main mouvement, mais ceux-ci diffèrent les uns des autres dans leur des modèles. Certains de ces modèles sont Analyse syntaxiques , Réseau de neurones, Modèles de Markov cachés etc.

HMM sont largement utilisés dans la lecture de manuscrite, la reconnaissance du parole, marquage d'une partie du discours et machine Traduction. Un autre avantage de l'utilisation discrète HMM est que les distributions de sortie sont automatiquement appris par le processus de formation. Vassilia et. Al.ont développé un système qui pourrait reconnaître à la fois les Phrases isolé et continu en langue des signes grecque (GSL) où le vecteur d'orientation est extrait des images et utilisé en phrases en entrée de HMM. Ho-Sub et. Al.a introduit une méthode de reconnaissance des gestes de la main qui utilise des caractéristiques combinées de localisation, angle et vitesse pour déterminer le vecteur discret qui est utilisé comme entrée pour HMM. Cette méthode se répète les alphabets A à Z, les numéros 0 à 9, six commandes de modification et six éléments de dessin. Nianjun et. Al. a proposé une méthode pour reconnaître les 26 lettres de A à Z en utilisant un HMM différent topologies avec différents états. Mais, ces méthodes exécuter hors ligne sur un fond non complexe. Nguyen et. Al. a introduit un système de reconnaissance des geste de la main pour reconnaître le geste en temps réel dans des environnements sans contrainte et le système était testé à un vocabulaire de 36 gestes y compris l' American Sign Language (ASL) alphabet et chiffres. Nobuhiko et. Al.introduit une méthode pour obtenir des fonctionnalités à partir de séquence d'image où une personne effectue les signes du langage japonais(JSL) dans un contexte complexe et pour reconnaître le mot JSL. Les deux précédents s'exécutent en temps réel sur un contexte complexe, mais ils étudient la la posture de la main, pas la trajectoire de mouvement du la main comme en elle est notre système.

Nous développons un système de reconnaissance des alphabets (A - Z) et les nombres (0 - 9) en temps réel à partir des séquences des images en couleur par la trajectoire de mouvement d'une seule main en utilisant HMM. Notre système dépend des principales étapes suivants ; utilisant GMM pour la détection de la couleur de la peau, le l'orientation entre deux points consécutifs est extrait comme fonctionnalité de base, l'algorithme BW pour le training et des algorithme avancé pour tester en conjonction avec Modèle LRB. De plus, chaque alphabet et chaque nombre est basé sur 30 vidéos (20 pour la training et 10 pour le testing) où les images d'entrée sont capturées par un Caméra stéréo Bumblebee avec une distance focale de 6 mm pendant environ 2 à 5 secondes à 15 images par seconde avec Résolution d'image 240×320 pixels sur chaque image. Le taux de reconnaissance obtenus lors de la training et de testing des gestes sont respectivement de 99,16% et 94,72%.

Chapitre 1

SYSTÈME DE RECONNAISSANCE DES GESTES

Notre système est conçu pour reconnaître les alphabets et nombres en temps réel à partir d'une séquence d'images stéréo en couleur par la trajectoire de mouvement d'une seule main en utilisant HMM. Pour l'initialisation automatique, la couleur et les informations 3D sont utilisées sur la base des points 3D afin de surmonter les difficultés de régions qui se chevauchent. En particulier, le système de reconnaissance des gestes se compose de trois étapes principales (Fig.1 et Fig.10) :

- * Segmentation et prétraitement automatiques ; la main est segmentée, localisée et génère sa trajectoire de mouvement (trajectoire gestuelle) en utilisant GMM pour la détection de la couleur de la peau.
- * Extraction de caractéristiques ; déterminer le vecteur discret qui est utilisé comme entrée pour HMM par l'orientation quantification.
- * Classification ; la trajectoire du mouvement de la main est reconnue en utilisant un vecteur discret, modèle LRB et algorithme avant de HMM.

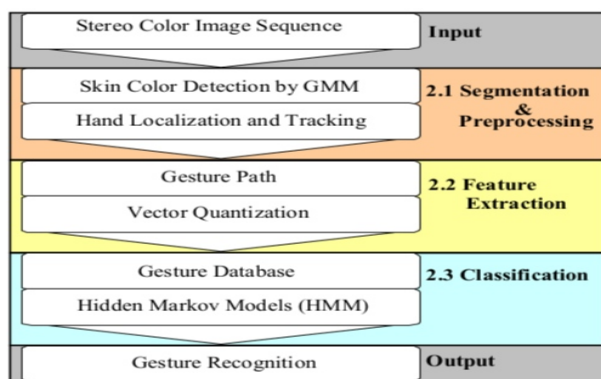


FIGURE 1.1 – Système de reconnaissance de gestes utilisant HMM.

Le geste graphique de la main se compose de 26 alphabets caractères de A à Z et 10 chiffres arabes de 0 à 9 où les formes gestuelles sont représentées sur la Fig.2.



FIGURE 1.2 – Formes de geste alphabets et chiffres qui sont utilisés dans notre système à partir de graphiques à la main mouvement.

1.1 Segmentation et prétraitement

La segmentation et le prétraitement automatiques sont un étape importante de notre système où le la segmentation de la main s'effectue en 3D et les informations sur les couleurs. Pour l'élimination des restes erreurs, les opérations morphologiques sont utilisées comme prétraitement. Cette étape comprend deux étapes ; dans le première étape, la couleur de la peau est détectée en utilisant le modèle gaussien de mélange (GMM) sur l'espace colorimétrique $YCbCr$. Dans la deuxième étape, la main est localisée et suivie par en utilisant une analyse de blob pour la région de la main.

1.1.1 Détection de la couleur de la peau via un GMM

L'espace colorimétrique $YCbCr$ est utilisé dans notre système où Y canal représente la luminosité et les canaux (Cb, Cr) se référer à la chrominance [Ask04a, Noo06a]. Nous ignorons Canal Y afin de réduire l'effet de la luminosité variation et ensuite utiliser uniquement les canaux de chrominance qui représentent pleinement la couleur. Dans un plan de chrominance, la couleur de la peau humaine se petite zone (Fig. 3 (a)), donc chaque pixel est classé comme peau ou non peau en utilisant le modèle gaussien. Le La technique GMM commence par la modélisation de la peau et non skin en utilisant une base de données de skin et non skin pixels respectivement. Une grande base de données de pixel de peau est utilisé pour former le modèle gaussien où la moyenne matrice de vecteur et de covariance de la base de données caractériser le modèle. Dans notre système, nous collectons des images contenant des pixels de la peau humaine (Fig. 4) et également des images pour des pixels sans peau (Fig. 5).

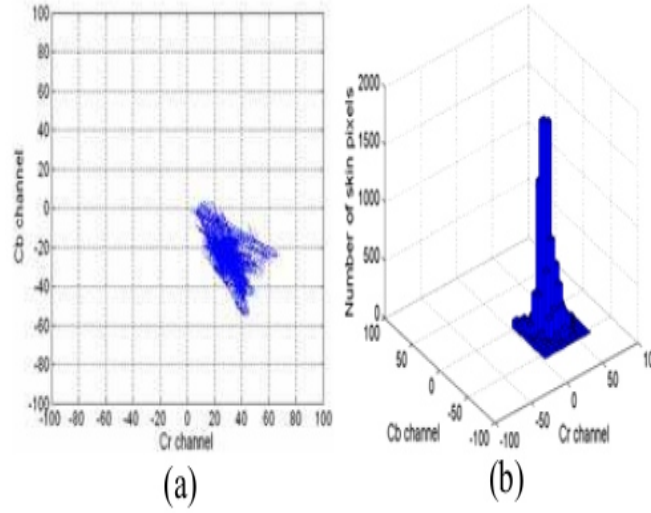


FIGURE 1.3 – The distribution and histogram of the skin training data (a) Distribution values of skin pixels for training data (b) Histogram for skin training data over (C_b, C_r) channels.

Une variante de l'algorithme de clustering k-means pour les clusters gaussiens effectue la formation modèle pour déterminer la configuration initiale.



FIGURE 1.4 – Database of skin pixel where these cropped images were collected from the World Wide Web for different races.

Supposons que $x=[C_b, C_r]^T$ représente la chrominance vecteur d'un pixel d'entrée. La probabilité de pixel de peau sur le vecteur x pour le modèle de mélange est linéaire combinaison de ses probabilités et est calculé comme suit :



FIGURE 1.5 – Database of non skin pixel where these cropped images were collected from the World Wide Web for different background.

$$p(x|skin) = \sum_{i=1}^K p(x|i) \cdot p(i) \quad (1)$$

où K est le nombre de composants gaussiens, est estimé par un algorithme constructif utilise les critères de maximisation de la fonction de vraisemblance. $p(i)$ est le poids du mélange et $p(x|i)$ est le modèle de densité gaussien pour la i ème composante.

$$p(x|i) = \frac{e^{-1/2(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}}{(2\pi)^{f/2} \sqrt{|\Sigma_i|}}$$

où μ_i et Σ_i représentent le vecteur moyen et le matrice de covariance de i ème composante respectivement et f est la dimension de l'espace d'entité, $x \in \mathbb{R}^f$.

Après que K est décidé, qui prend la valeur 4 dans notre expérience, l'algorithme de maximisation des attentes (EM) est utilisé pour estimer la probabilité maximale de paramètres (moyenne, covariance et poids du mélange) qui données d'entraînement. L'algorithme EM commence par l'initiale valeurs des paramètres où l'algorithme k-means est utilisé pour déterminer la configuration initiale comme dans le tableau 1.

$$\sum^K p(i) = 1 ; \quad 0 \leq p(i) \leq 1$$

Cluster	Mean μ	Covariance Σ	Weight
1	[119.5 ; 144.1]	$\begin{pmatrix} 35.81 & -13.55 \\ -13.55 & 14.88 \end{pmatrix}$	0.2422
2	[110.3 ; 153.2]	$\begin{pmatrix} 13.34 & 2.12 \\ 2.12 & 5.73 \end{pmatrix}$	0.2612
3	[98.6 ; 165.9]	$\begin{pmatrix} 46.09 & -21.65 \\ -21.65 & 46.82 \end{pmatrix}$	0.1668
4	[103.1 ; 157.3]	$\begin{pmatrix} 16.83 & -1.26 \\ -1.26 & 16.94 \end{pmatrix}$	0.3298

FIGURE 1.6 – Table 1. Gaussian Mixture Model for skin color.

De plus, les paramètres initiaux sont maintenant anciens paramètres, puis l'algorithme EM estime la nouvelle valeurs des paramètres. Dans la prochaine itération, les nouveau paramètres deviennent vieux et ce processus est répété jusqu'à ce que la convergence soit atteinte (c-à-d. le changement de log- vraisemblance entre deux itérations est inférieur à un seuil). L'algorithme EM est fait par le les deux étapes suivantes :

1-Étape d'attente

$$p^{old}(i | x_n) = \frac{p^{old}(x_n | i) \cdot p^{old}(i)}{\sum_{j=1}^K p^{old}(x_n | j) \cdot p^{old}(j)}$$

2-Étape de maximisation

où N est le nombre de points de données x_n . Pour le probabilité $p(x | non-skin)$, les pixels de couleur non skin sont modélisé comme un gaussien unimodèle afin de réduire la complexité de calcul de la probabilité du calcul de skin (tableau 2).

Pour la segmentation de la peau des mains et du visage en stéréo séquences d'images, un algorithme est utilisé, qui calcule les informations de profondeur en plus les informations des couleurs du peau. Les informations détaillées peut être recueilli par mesure stéréo passive basée sur la corrélation croisée et les données d'étalonnage connues des caméras. Plusieurs clusters sont composés des points 3D résultants. l'algorithme du clustering peut être considéré comme une sorte de région croître en 3D en utilisant deux critères ; couleur de peau et la distance euclidienne. De plus, cette méthode est plus robuste à l'éclairage désavantageux et occlusion partielle qui se produit en environnement de temps réel (par exemple, en cas de reconnaissance de geste). Pour plus de détails, le lecteur peut se référer à [Nie07b]. Par les informations détaillées fournies par le système de configuration de la caméra (Fig. 6 (c)), le problème de chevauchement entre

$$\mu_i^{new} = \frac{\sum_{n=1}^N p^{old}(i | x_n) x_n}{\sum_{n=1}^N p^{old}(i | x_n)} \quad (5)$$

$$\Sigma_i^{new} = \frac{\sum_{n=1}^N p^{old}(i | x_n) (x_n - \mu_i^{new})(x_n - \mu_i^{new})^T}{\sum_{n=1}^N p^{old}(i | x_n)} \quad (6)$$

$$p^{new}(i) = \frac{1}{N} \sum_{n=1}^N p^{old}(i | x_n) \quad (7)$$

Mean μ	Covariance Σ
$[58.65; 48.24]$	$\begin{pmatrix} 13.22 & -8.37 \\ -8.37 & 19.01 \end{pmatrix}$

FIGURE 1.7 – Table 2. Unimodel Gaussian for non skin color.

les mains et le visage est résolu depuis les régions de la main sont plus proches de la caméra que du région du visage. Pour supprimer les valeurs aberrantes (bruit, composants parasites) de l'image du probabilité de la peau, nous utilisons l'opération morphologique (médiane filtre, érosion et dilatation) car il y a de petites régions proches de la peau mais n'appartenant pas à la peau humaine. De plus, les pixels des trous sont rempli sur le bord extérieur d'une image qui n'est pas connecté à l'arrière-plan. Ainsi, la couleur de la peau est détecté (mains et visage). La figure 6 (a) montre le premier cadre de la séquence d'images.

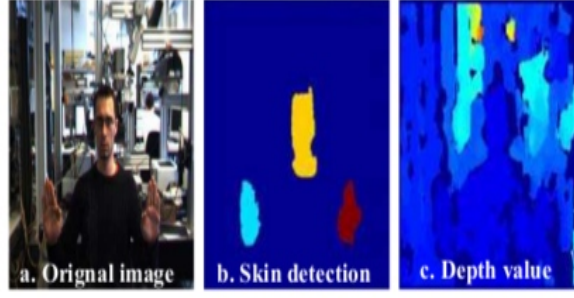


FIGURE 1.8 – Skin segmentation. (a) First frame of image sequence (b) Labeled skin color detection after using morphological operation (c) Depth information of the original image from a Bumblebee stereo camera.

1.1.2 Détection et suivi des mains

Une fois l'image de la peau étiquetée déterminée (Fig.6(b)), la localisation des deux mains est trouvée par sélection des deux petites zones (Fig. 7 (a)) où le visage représente la grande zone et le plus éloigné de la caméra. De plus, nous utilisons une analyse de blob pour déterminer la zone limite, le cadre de délimitation et le point centroïde de chaque région de la main. Notre attention se concentre sur le mouvement d'une seule main pour détecter la trajectoire graphique de la main pour un alphabet spécifique ou nombre. Par conséquent, nous sélectionnons une zone de recherche dans le cadre suivant (Fig. 7 (b)) autour du cadre de délimitation qui est déterminé à partir de la dernière image afin de suivre la main et réduire la zone d'intérêt (AOI). S'il y a plusieurs régions cutanées extraites dans un zone de recherche de la main, une grande région est sélectionnée cette région représente tout au plus un coup de main.

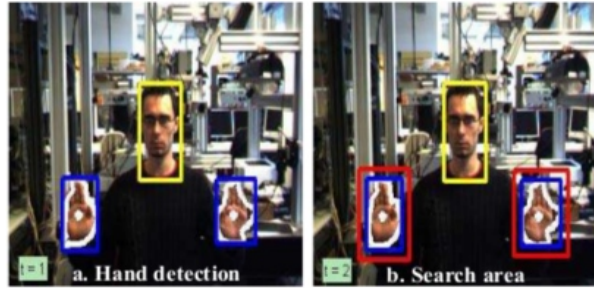


FIGURE 1.9 – Hand localization and search area (a) Hand localization with a boundary area, bounding box and centroid point (b) Search area around the hands in the next frame.

Ainsi, la nouvelle boîte englobante est calculée et le point centroïde est déterminé. Par itération de ce processus, la trajectoire de mouvement de la main soi-disant le chemin du geste est généré à partir de la connexion du centroïde points (Fig.8).

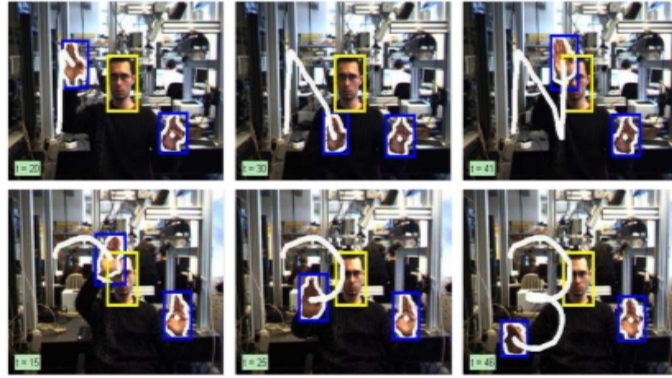


FIGURE 1.10 – Gesture path for alphabet N and number 3.

2.2- Extraction de caractéristiques.

- La sélection de bonnes fonctionnalités pour reconnaître le chemin des gestes de la main joue un rôle important dans les performances du système. Il existe trois caractéristiques de base comme: l'emplacement, l'orientation et la vitesse.
+ les recherches précédentes [Has01a, Via04b] ont montré que la caractéristique d'orientation est la meilleure en termes de résultats.

mais nous remettrons à cette caractéristique principale de notre système.

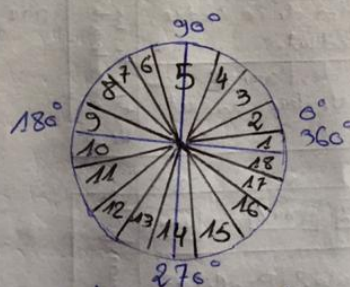
+ une trajectoire gestuelle est un modèle spatiotemporel qui se compose de points centralisés (x_{hand}, y_{hand}) . ainsi l'orientation est déterminée entre deux points consécutifs de la trajectoire du geste de la main.

$$\theta_t = \arctan \left[\frac{y_{t+1} - y_t}{x_{t+1} - x_t} \right] ; t = 1, 2, \dots, T-1$$

$T \rightarrow$ représente la longueur du ~~mon~~ chemin du geste

- l'orientation θ_t est divisée par 20° afin de quantifier la valeur de 1 à 18

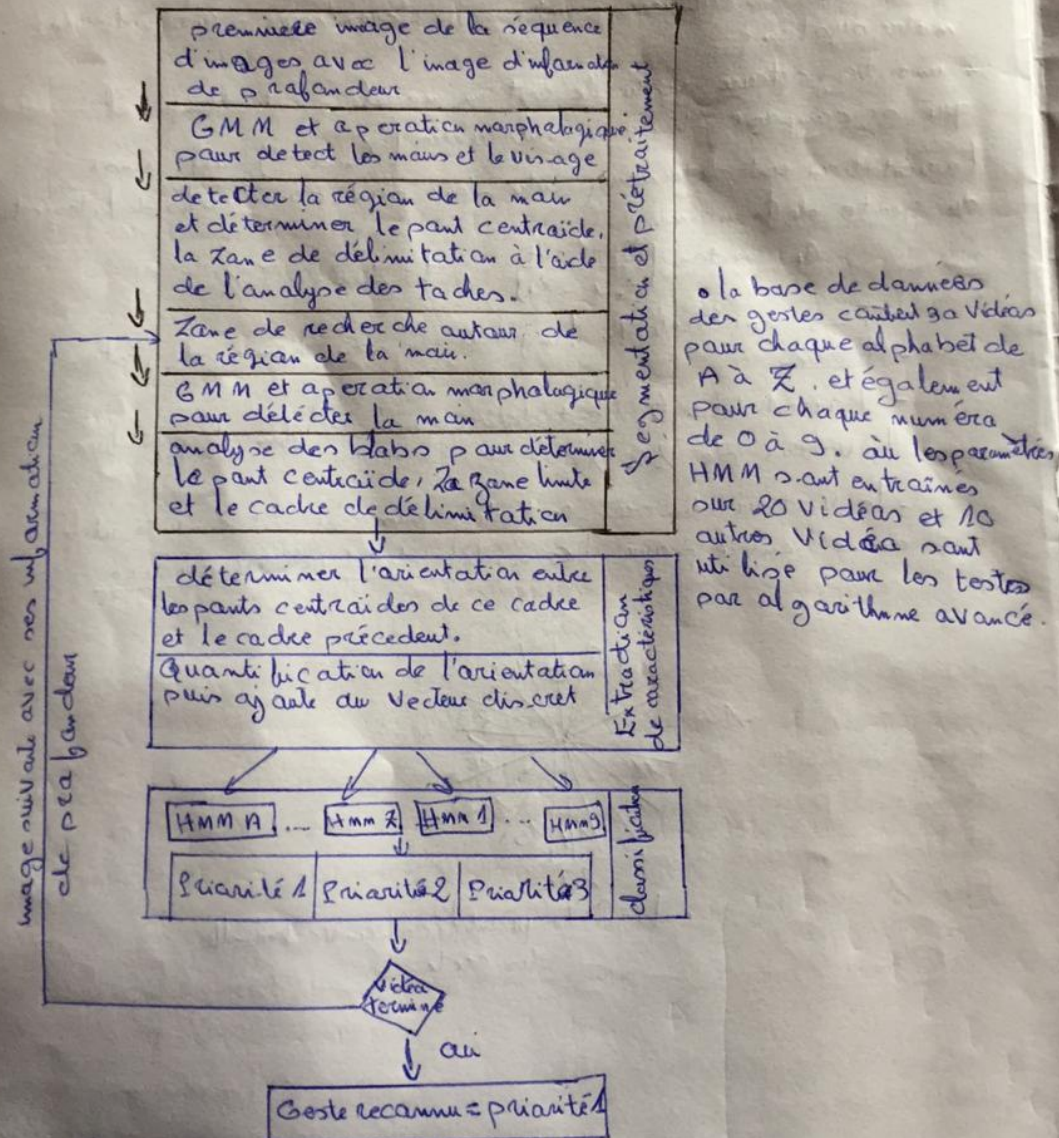
- le vecteur discret est déterminé et est ensuite utilisé comme entrée pour le modèle de Markov caché



palage de quantification vectorielle discrète.

2.3. classification

- dans cette étape, on a utilisé l'algorithme de Baum-welch (BW) pour faire un train complet pour les paramètres initialisés de HMM par un vecteur discret.
- la trajectoire gestuelle du mouvement de la main utilise un modèle à bandes gauche-droite avec 9 états, un vecteur discret en conjonction avec l'algorithme Forward et la construction d'une base de données de gestes.

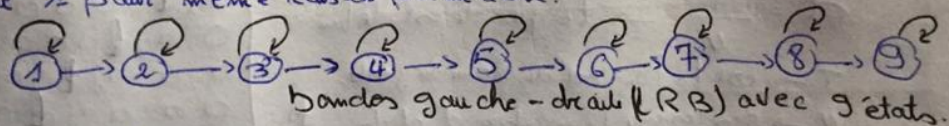


Système en temps réel pour la reconnaissance des gestes alphabétiques et numériques

2.3.1. Modèle de Markov Caché.

(3u)

- un HMM modélise un automate à états cachés dans lequel chaque état a une certaine probabilité de transition vers chacun des autres états. chaque transition engendre une observation. l'observation suit une loi de probabilité associée à l'état courant.
→ les observations peuvent être discrètes, ou elles peuvent être continues.
- le modèle de Markov est un triple $h = (A, B, \pi)$
 - $A \rightarrow$ la matrice de probabilité de transition de chaque état vers chaque état.
 - $B \rightarrow$ l'ensemble des fonctions de probabilités des observations associées à chaque état.
 - $\pi \rightarrow$ est le vecteur des probabilités d'émission initiales
- + l'ensemble des états $S = \{s_1, s_2, \dots, s_N\}$ (où N le nombre d'états).
- + une probabilité initiale pour chaque état $\pi_i, i = 1, 2, \dots, N$ telle que $\pi_i = P(s_i)$ à l'étape initiale.
- + une matrice de transition N par N $A = \{a_{ij}\}$ où a_{ij} est la probabilité d'une transition de l'état s_i à $s_j, 1 \leq i, j \leq N$ et la somme des entrées de chaque ligne de la matrice A doit être 1 car il s'agit de la somme des probabilités d'effectuer une transition d'un état donné à chacun des autres états.
- + l'ensemble d'émission possible (une observation) $O = \{o_1, o_2, \dots, o_T\}$ où T est la longueur du trajet gestuel.
- + l'ensemble des symboles discrets $V = \{v_1, v_2, \dots, v_M\}$ où M représente le nombre des symboles discrets.
- + une matrice d'observation N par M $B = \{b_{im}\}$ où b_{im} donne la probabilité d'émettre le symbole v_m de l'état s_i et la somme des entrées dans chaque ligne de la matrice B doit être 1 pour même raison perméable.



l'évaluation, le décodage et la formation sont les principaux problèmes de HMM et ils peuvent résoudre en utilisant respectivement l'algorithme Forward-Backward, l'algorithme Viterbi, l'algorithme Baum-welch.

En outre HMM a trois topologies :

- entièrement connecté (modèle ergodique) où n'importe quel état peut être atteint à partir de n'importe quel autre état.
- Modèle gauche-droite de sorte que chaque état peut revenir à lui-même ou aux états suivants.
- bandes gauche-droite (LRB) qui permet également à chaque état de revenir à lui-même ou à l'état suivant uniquement.

2.3.2. initialisation des paramètres HMM

+ pourquoi nous utilisons le modèle à bandes gauche-droite avec 9 états ?

le modèle LRB est restreint et simple pour les données d'apprentissage qui servent en mesure de faire correspondre les données au modèle.

Étant donné que le modèle entièrement connecté a de nombreuses transitions plutôt que le modèle LRB, ses données de structure peuvent facilement perdre. cela s'applique également au modèle gauche-droite, de plus des recherches antérieures ont montré que l'utilisation de 9 états pour le modèle LRB est la meilleure en termes de résultats.

Nous avons donc utilisé le modèle LRB avec 9 états dans notre système.

la sélection de bons paramètres initiaux pour HMM permet d'obtenir de meilleurs résultats de reconnaissance. le vecteur initial π prend la valeur suivante :

$$\pi = \{1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0\}^T$$

c'est parce que nous utilisons 9 états et afin nous assurer que cela commence à partir du premier état.

la matrice A dépend de la durée d des états pour chaque nombre et est déterminée par :

$$d = \frac{T}{N} \rightarrow \text{to que de est défini}$$

$T \rightarrow$ est la longueur du chemin du geste.

$N \rightarrow$ représente le nombre d'états.

$$a_{ij} = 1 - \frac{1}{d}$$

$$b = 1 - a_{ij}$$

$$A = \begin{pmatrix} a_{11}d & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{22}d & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{33}d & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44}d & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{55}d & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{66}d & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{77}d & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{88}d & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{99}d \end{pmatrix}$$

La matrice B est le paramètre important de notre système et est déterminé par l'équation $B = \{b_{im}\}$; $b_{in} = \frac{1}{\text{nombre d'états}}$
 Les états HMM étant discrets, tous les éléments de la matrice B peuvent être initialisés avec la même valeur pour tous les états différents.

2.3.3 - Algorithme Baum - Welch et Forward.

- l'algorithme Baum - Welch joue un rôle très important dans notre système car il est utilisé pour faire un train complet pour les paramètres HMM initialisés.

cet algorithme estime la nouvelle matrice A , la nouvelle matrice B et le vecteur π dont les entrées sont un vecteur discret et les paramètres initialisés.

→ étant donné que notre système est formé sur 20 vidéos pour chaque alphabet et numera, la valeur de la matrice A et de la matrice B pour eux est moyenne.

→ Selon l'algorithme Forward les 20 autres vidéos pour chaque alphabet et numera sont testées car cet algorithme est construit sur un vecteur discret, une matrice A , B et un vecteur π comme entrées pour lui.

→ l'algorithme discret est utilisé pour résoudre le problème d'évaluation en calculant le $P(O|h)$ qui donne la probabilité de l'observation.

définissons les valeurs (alpha) comme suit :

$$\alpha_t(i) = P(O_1 = o_1, \dots, O_t = o_t, S_t = i | h)$$

calcul de la valeur directe

$$\begin{cases} \text{- étape ① : } \alpha_1(i) = \pi(i) \cdot b_{im} \\ \text{- étape ② : } \alpha_{t+1}(j) = \sum_{i=1}^N \alpha_t(i) \cdot a_{ij} \cdot b_{j+1} \end{cases}$$

Chapitre 2

RÉSULTATS EXPÉRIMENTAUX

Notre système a proposé de bons résultats pour reconnaître les alphabets et les nombres en temps réel à partir de séquences d'images en couleur par la trajectoire de mouvement d'une seule main en utilisant HMM. Dans nos résultats expérimentaux, chaque alphabet de A à Z et chaque nombre de 0 à 9 était basé sur 30 vidéos dont 20 pour la formation et 10 pour les tests. En d'autres termes, notre base de données contient 720 vidéos pour les gestes d'entraînement et 360 vidéos pour tester les gestes. Le système a été implémenté en langage Matlab et les images d'entrée ont été capturées par un système de caméra stéréo Bumblebee qui a une distance focale de 6 mm pendant environ 2 à 5 secondes à 15 images par seconde avec une résolution d'image de 240×320 pixels sur chaque image.



la figure exprime ,Sortie système pour l'alphabet R, où à $t = 28$ la priorité élevée est l'alphabet F, à $t = 45$ la priorité élevée est l'alphabet P et à $t = 70$ le résultat est R.

La figure montre la sortie de notre système pour l'alphabet R. Les critères suivants ont évalué notre résultat comme suit :

Les données de test sont considérées comme, $\tau=10$,pour chaque alphabet ou chaque numéro où ces données de test incluent un geste valide v et un geste non valide tel \bar{v} que :

$$\tau = v_j + \bar{v}_j ; \quad j = 1, 2, \dots, 36$$

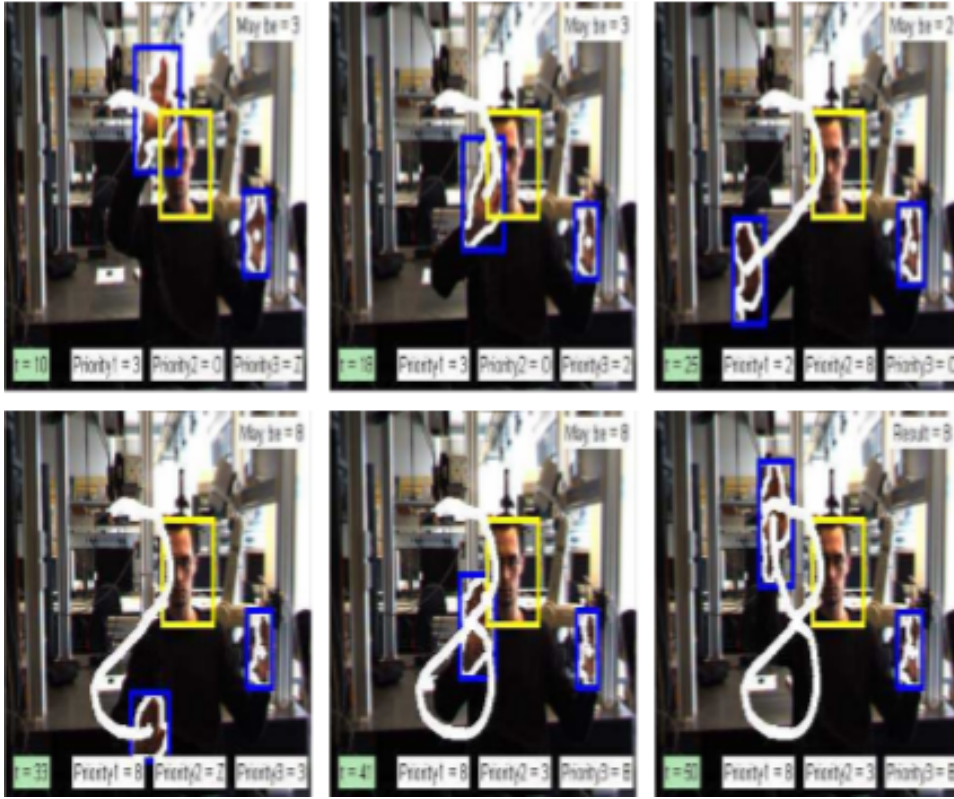
où j représente l'index des alphabets de A à Z et des nombres de 0 à 9. Le pourcentage valide

pour chaque alphabet et chaque numéro est calculé par l'équation la première ci-dessous et le pourcentage total pour toutes les données de test est déterminé par la deuxième équation ci-dessous.

$$\eta_j = \frac{v_j}{\tau} \cdot 100$$

$$\mathfrak{R} = \frac{1}{36} \sum_{j=1}^{36} \eta_j$$

où η est le résultat de chaque alphabet ou d'un nombre et \mathfrak{R} représente la valeur de toutes les données de test. De même, les données d'entraînement ont été calculées avec 20 vidéos pour chaque alphabet et numéro. Pour chaque vidéo graphique à la main, la priorité la plus élevée a été calculée par algorithme direct pour reconnaître l'alphabet ou le nombre dans notre système réel, image par image ci-dessous.



la figure ,Sortie système pour le numéro 8, où à $t = 18$ la priorité élevée est le numéro 3, à $t = 25$ la priorité élevée est le numéro 2 et à $t = 50$ le résultat est 8. La reconnaissance a été obtenue sur la formation et les gestes de test avec 99,16% et 94,72 % respectivement où le rendement des données de formation est supérieur à celui des données de test dans notre système.

CONCLUSION

Cet article propose un système pour reconnaître les alphabets (A - Z) et les nombres (0 - 9) à partir de séquences d'images en couleur par la trajectoire de mouvement d'une seule main à l'aide de HMM qui convient à une application en temps réel. Le système comprend trois étapes principales. La première étape est la segmentation et le prétraitement automatiques, où la main est localisée et suivie pour générer sa trajectoire gestuelle en utilisant GMM pour la détection de la couleur de la peau. Dans la deuxième étape dite d'extraction de caractéristiques, le vecteur discret est obtenu par quantification de l'orientation où ce vecteur est utilisé comme entrée pour HMM. La dernière étape est la classification qui peut reconnaître le geste graphique de la main en utilisant le modèle LRB, l'algorithme BW et l'algorithme Forward. Notre base de données contient 720 vidéos pour la formation et 360 vidéos pour les tests. Les résultats montrent que ; un taux de reconnaissance moyen est de 94,72% et 99,16% pour les vidéos de test et de formation respectivement.