

Rapport d'étude de la librairie HandVu

1. Introduction

Le but de ce travail est de réaliser l'étude de la librairie **HandVu**.

Le présent rapport s'articulera sur les points suivants :

- Présentation et description de la librairie
- Caractéristiques de la librairie
- Installation de la librairie
- Fonctionnement de la librairie
- Conclusion

2. Présentation et description de la librairie

HandVu est une librairie permettant la détection de la main, le tracking et la reconnaissance de postures de cette dernière en temps réel.

Ce programme a été développé par M.Mathias Kölsh à l'université de Santa Barbara

Toutes ces opérations s'effectuent à l'aide d'une caméra connectée à l'ordinateur.

3. Caractéristiques

Les caractéristiques de HandVu sont les suivantes

- Travaille avec différents types de caméra
- Réagit de différentes manières suivant le changement de lumière
- Est sensible au « back ground »
- Délai de détection : 5-150ms per 640x480 frame (à 3GHz)
- Robuste et permet un déploiement sans contrainte de la scène visuelle.
- Open Source (programme écrit en C++)

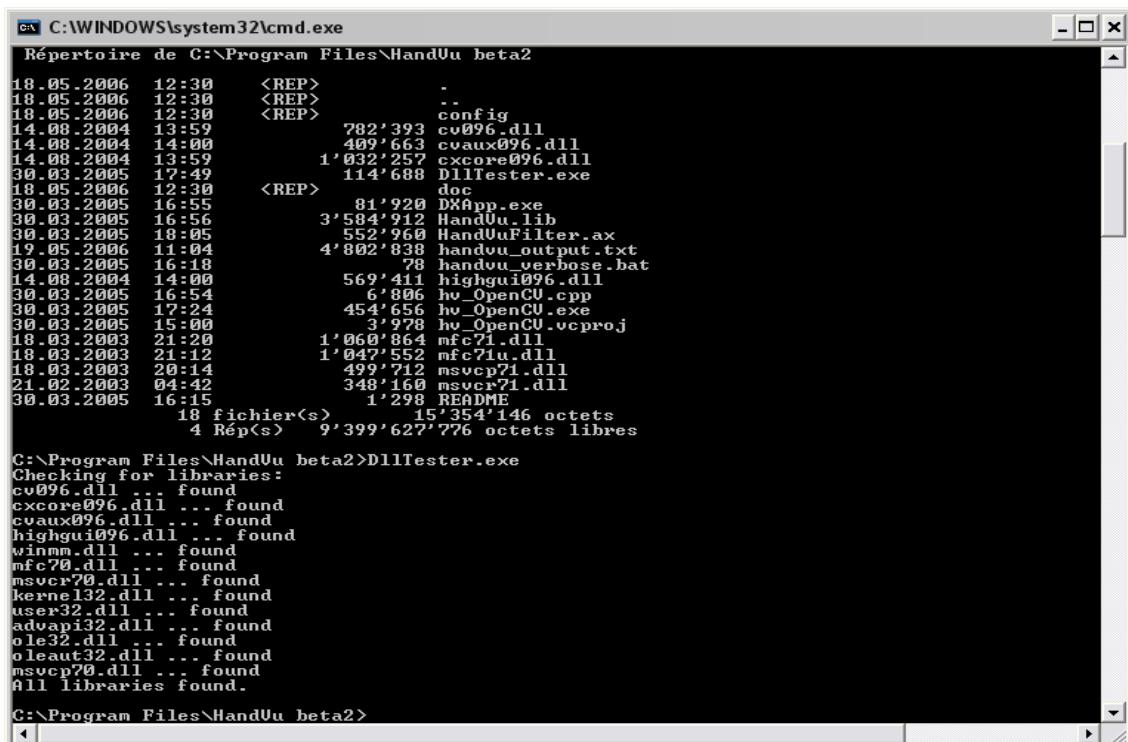
4. Installation

L'installation du produit se fait en exécutant le fichier MSI se trouvant sur le site de l'auteur, nous avons utilisé la version HandVu Beta 2, actuellement le produit en est à la version beta3, il tourne sur les plate-formes windows et Linux

Pour fonctionner, HandVu requiert la librairie OpenCV de Intel

Il est à noter qu'il y a des différences majeures entre les versions 4 et 5 d'OpenCV et que cette dernière version n'est pas compatible avec les versions précédentes. Une fois OpenCV beta4 installé, les problèmes de compatibilité furent résolus.

- télécharger le fichiers HandVuBeta2.msi
- Ce fichier exécute l'installation du produit sur la machine
- Télécharger OpenCV version 4
- Faire un contrôle des librairies en ouvrant un éditeur de ligne de commande et exécuter le fichier DLLTester comme ceci



```
C:\WINDOWS\system32\cmd.exe
Répertoire de C:\Program Files\HandUu beta2
18.05.2006 12:30 <REP>      -
18.05.2006 12:30 <REP>      ..
18.05.2006 12:30 <REP>      config
14.08.2004 13:59          782'393 cv096.dll
14.08.2004 14:00          409'663 cvaux096.dll
14.08.2004 13:59          1'032'257 cxcore096.dll
30.03.2005 17:49          114'688 DLLTester.exe
18.05.2006 12:30 <REP>      doc
30.03.2005 16:55          81'920 DXApp.exe
30.03.2005 16:56          3'584'912 HandUu.lib
30.03.2005 18:05          552'960 HandUuFilter.ax
19.05.2006 11:04          4'802'838 handvu_output.txt
30.03.2005 16:18          78 handvu_verbose.bat
14.08.2004 14:00          569'411 highgui096.dll
30.03.2005 16:54          6'806 hv_OpenCV.cpp
30.03.2005 17:24          454'656 hv_OpenCV.exe
30.03.2005 15:00          3'978 hv_OpenCV.vcproj
18.03.2003 21:20          1'060'864 mfc71.dll
18.03.2003 21:12          1'047'552 mfc71u.dll
18.03.2003 20:14          499'712 msucr71.dll
21.02.2003 04:42          348'160 msucr71.dll
30.03.2005 16:15          1'298 README
          18 fichier(s)          15'354'146 octets
          4 Rép(s)          9'399'627'776 octets libres

C:\Program Files\HandUu beta2>DLLTester.exe
Checking for libraries:
cv096.dll ... found
cxcore096.dll ... found
cvaux096.dll ... found
highgui096.dll ... found
winmm.dll ... found
mfc70.dll ... found
msucr70.dll ... found
kernel32.dll ... found
user32.dll ... found
advapi32.dll ... found
ole32.dll ... found
oleaut32.dll ... found
msucp70.dll ... found
All libraries found.

C:\Program Files\HandUu beta2>
```

- Il se peut que certaines bibliothèques ne soient pas trouvées.

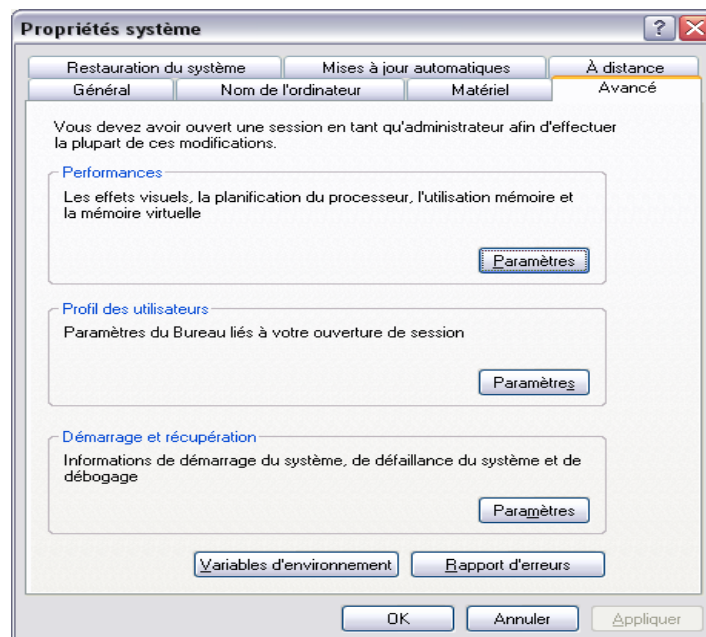
Dans ce cas, il faudrait aller les chercher dans OpenCV et les copier dans le répertoire d'installation de HandVu

Il se peut que certains problèmes surviennent après ces étapes, en effet, en lançant HandVu, ce dernier nous informait que la bibliothèque HandVuFilter.ax n'était pas trouvée.

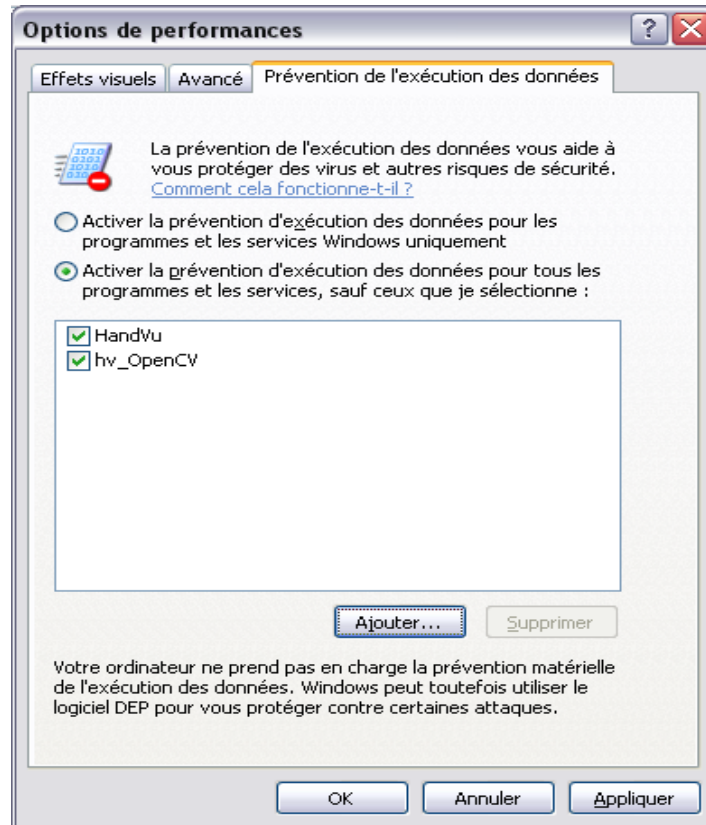
Après réflexion et diverses recherches sur Internet, nous avons pu découvrir que cette erreur était liée au buffer overflow.

En effet, certains processeurs de types AMD ou Intel de génération récente sont équipés d'une technologie nommée EVP (Enhanced Virus Protection) [AMD EVP] qui propose une aide hardware au système d'exploitation pour éviter les buffer overflows.

Nous avons essayé de corriger cette erreur dans HandVu mais n'avons pas réussi, c'est pourquoi nous l'avons contournée en désactivant l'utilisation d'EVP dans windows comme ceci



Puis nous avons ajouté HandVu et OpenCv comme ceci



Après cette manipulation, nous avons réussi à lancer HandVu

Il se peut que cette dernière étape ne soit pas nécessaire sur certaines machines

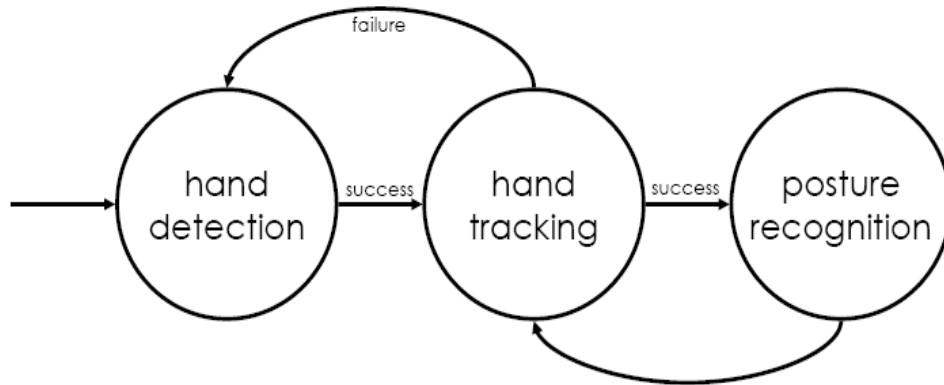
5. Fonctionnement

Camera

HandVu fonctionne avec divers types de caméra ayant une résolution d'au moins 320x240

Pour notre part, nous avons utilisé les caméras **Philips ToUCam PRO II** et **Intel T4000**

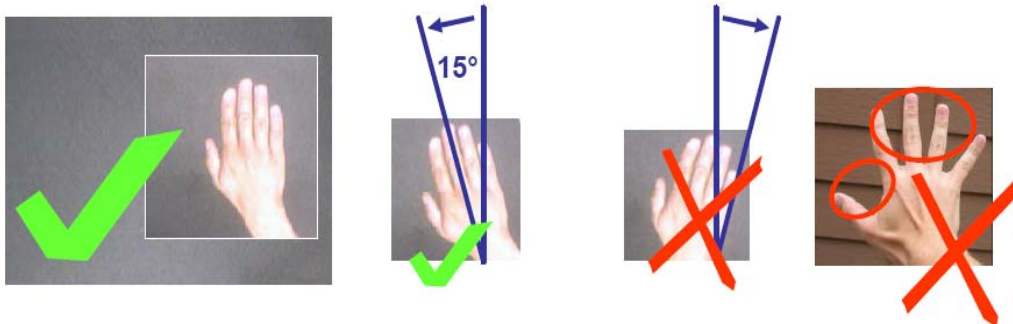
Le schéma ci-dessous illustre le fonctionnement de **HandVu**



Comme on peut le voir sur la figure ci-haut, le fonctionnement de HandVu se fait en trois étapes :détection de la main, tracking et reconnaissance de la posture de cette dernière.

Hand detection

La main est détectée seulement dans les postures standards et il s'avère nécessaire d'éviter des gestes brusques durant ce processus



HandTracking

Une fois la main détectée, l'utilisateur peut la bouger dans n'importe quelle posture.

Pour un tracking de la main satisfaisant, il s'avère nécessaire d'opérer dans un environnement possédant de bonnes caractéristiques lumineuses.

Il faudrait éviter les mouvements trop rapides de la main ou un changement brusque des postures de la main

Posture recognition

HandVu est capable de reconnaître 6 postures différentes de la main.

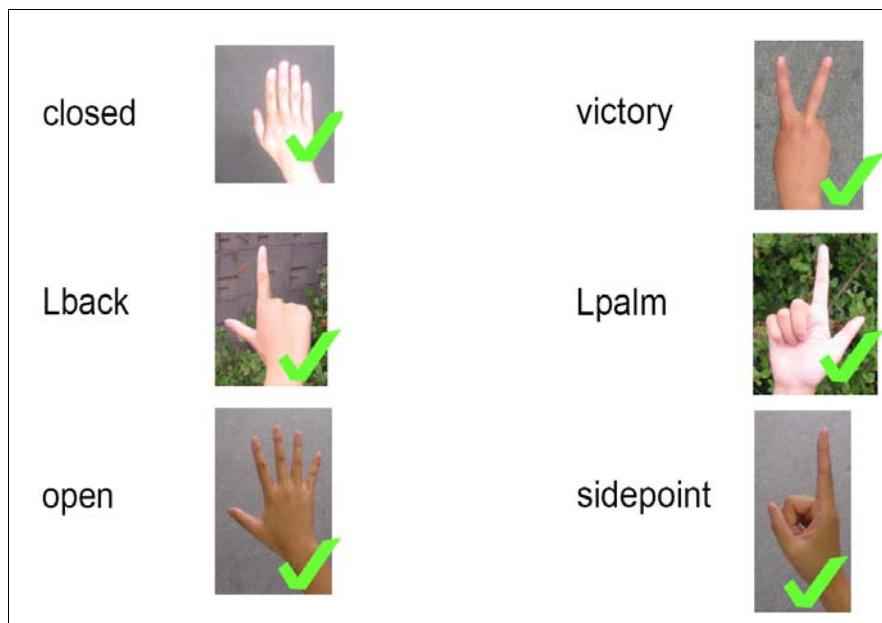
Ces 6 postures de reconnaissance peuvent être effectuées à n'importe quel moment pendant le tracking de la main et elles seront reconnues.

Il est à noter que toutes les postures doivent se faire dans un plan parallèle au plan de l'image vers le haut et avec un sens de rotation de pas plus de 15 degrés dans le sens inverse des aiguilles d'une montre.

Les différentes postures sont les suivantes :

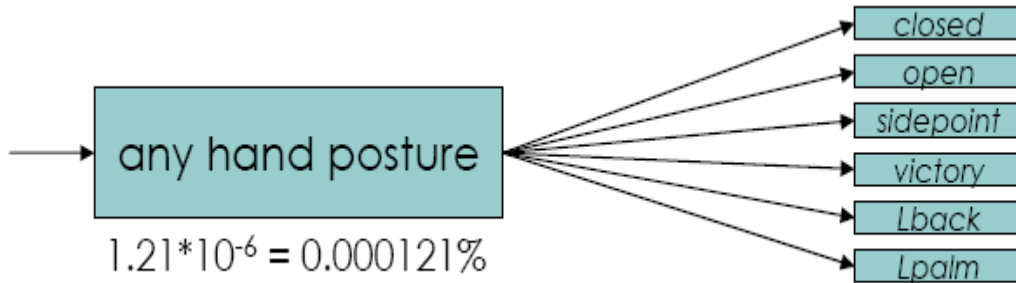
- Closed
- Victory
- Lpalm
- Lback
- Open
- Sidepoint

La figure suivante illustre les 6 postures

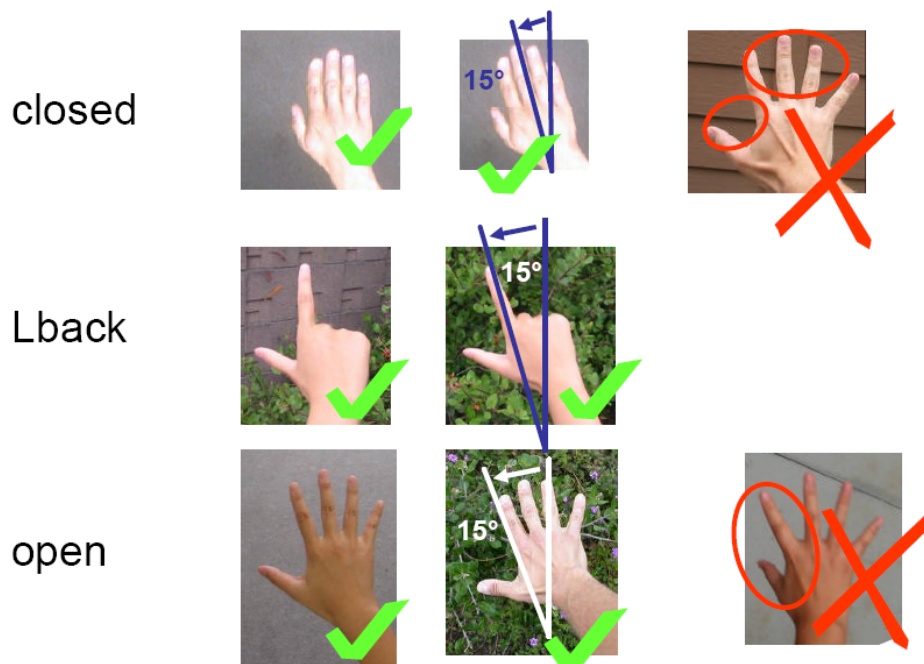


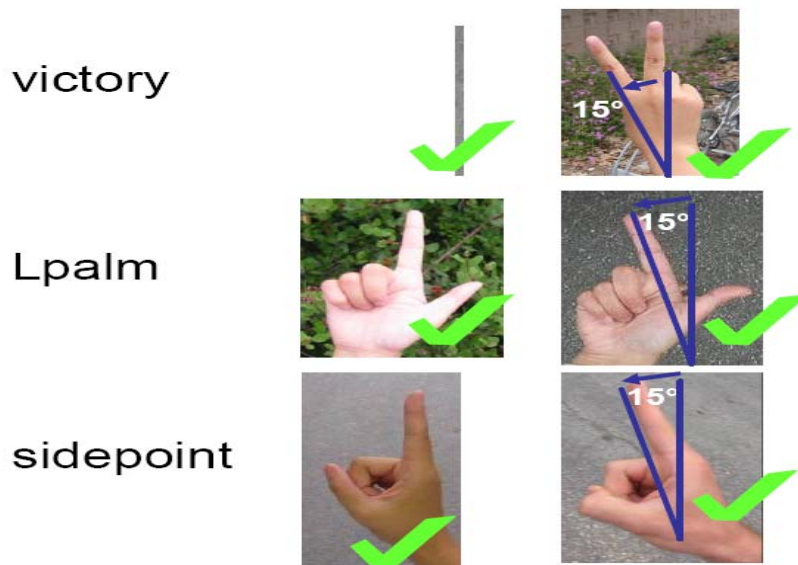
Nous avons pu remarquer que seulement deux postures sont vraiment faciles à reproduire (closed et Lpalm) et que pour les quatre autres, il faut passablement d'entraînement pour pouvoir déclencher des événements.

La reconnaissance des différentes postures de la main suit la hierarchie suivante :



La figure ci-après détaille la reconnaissance des différentes postures





6. Performances

Durant la phase de prise en main, nous avons été surpris de l'inconsistance des performances de HandVu. Parfois, le logiciel répondait de manière très précise, les postures étaient faciles à faire et rapidement reconnues. Il suffisait cependant de quelques changements mineurs au niveau de l'environnement lumineux ou de la disposition de la caméra pour obtenir des résultats désastreux.

7. Conclusion

Malgré les grosses difficultés que nous avons rencontrées durant l'installation et la phase de prise en main d'HandVu, nous avons remarqué que cette librairie est riche et peut être utilisée comme déclencheur d'événement avec d'autres applications écrites en langages C ou Java par exemple utiles dans différents domaines, on peut très bien imaginer une application qui permet de manipuler un jeu par l'intermédiaire des différentes postures de la main reconnues par HandVu.

8. Webographie et Bibliographie

- <http://www.movesinstitute.org>
- HandVu Postures Mathias Kolsh, University of Santa Barbara
- Computer Vision for interaction Mathias Kolsh , University of Santa Barbara september 2004
- Vidéo de présentation du projet Chirmouse (projet de semestre d'hiver 2005-2006)