

Université de Fribourg – Département d'Informatique

Interfaces Multimodales

*Emotional Photo Shooting*

Projet réalisé par : Said Mechkour, Raphael Boesch et Razvan Oprea

## Table des matières

1. Introduction.....	3
2. Bibliothèques utilisées.....	3
2.1. OpenCV.....	3
2.2. Sphinx.....	3
3. Présentation du système.....	4
3.1. Algorithme utilisé.....	5
3.2. Fonctionnement du système.....	5
4. Conclusions.....	6

## 1. Introduction

Le projet « Emotional Photo Shooting » a été réalisé dans le cadre du cours d'approfondissement *Interfaces Multimodales*. Le but du cours est de donner une connaissance théorique et pratique sur les méthodes de conception, réalisation et d'évaluation des interfaces multimodales.

L'idée de départ du projet a été de réaliser un système d'interaction multimodale qui permet de prendre une photo dès qu'il détecte une émotion. Les modes d'entrée naturels, la voix et l'expression faciale seront captés à l'aide d'une camera respectivement d'un micro. Le système fournira en sortie une photo. Suite aux consultations avec l'enseignant Denis Lalanne on a ajouté une autre fonctionnalité à notre système. Celle-ci permet de diriger un avatar à l'aide des gestes de l'utilisateur. Plus précisément quand l'utilisateur fait un certain geste celui-ci est répété par l'avatar. En ce moment les gestes reconnues par l'avatar sont: sourire, tête en haut, tête en bas, tourner la tête à gauche, tourner la tête à droite, ouvrir la bouche et rester en position libre.

## 2. Bibliothèques utilisées

### 2.1. OpenCV

Pour la détection faciale et émotionnelle on a utilisé [OpenCV](#) qui est une bibliothèque open source et multiplateforme, développée par Intel. La bibliothèque est divisée en quatre grandes parties qui contiennent plus que 500 algorithmes écrits en C/C++ permettant le traitement d'images en temps réel. Le grand avantage d'OpenCV est donné par sa vitesse de calcul et le fait que les implémentations tiennent compte des résultats de recherche les plus récents.

### 2.2. Sphinx

Pour la partie d'interaction vocale on a choisi de travailler avec [Sphinx-4](#) qui est un logiciel de reconnaissance vocale open source implémenté en Java. L'idée de départ était de commander le système par les commandes vocales suivantes :

- *Start* – débiter la détection de l'émotion
- *Accept* – sauvegarder la photo
- *Repeat* – prendre une nouvelle photo sans sauvegarde de la photo actuelle

Après l'installation de Sphinx et les premiers tests on s'est rendu compte que le taux de reconnaissance des ordres était très faible et que le risque de confusion concernant une commande est élevé. Une autre observation est celle de la nécessité d'un microphone de haute qualité.

On a observé que le premier ordre donné est dans la plus part des cas reconnu et que le problème intervient après. Pour contourner ce problème on aurait pu utiliser un bouton qui bloque l'interaction avec Sphinx après le premier ordre. Plus précisément quand on veut donner un ordre au système on va appuyer le bouton. De cette façon on sera toujours dans la situation d'un premier ordre donné qui a des fortes chances d'être reconnu. Pour ces raisons on a décidé de ne plus utiliser dans notre système la reconnaissance vocale et à la place des ordres vocaux on a opté pour l'utilisation des boutons (le clavier de l'ordinateur dans notre cas).

## 1. 3. Présentation du système

### 3.1. Algorithme utilisé

Pour notre implémentation on a utilisé l'algorithme de Lucas-Kanade qui est un des algorithmes de tracking le plus populaires.

L'idée de base de l'algorithme Lucas-Kanade repose sur trois hypothèses :

1. Luminosité constante – l'intensité lumineuse se conserve entre deux images successives.
2. Persistance temporelle ou « petits mouvements » - le mouvement d'un cadre à l'autre doit être « petit »
3. Cohérence spatiale – les voisins d'un point doivent rester les mêmes.

Un des problèmes de l'algorithme est du à l'utilisation des fenêtres de petite taille. Si les mouvements sont grands on risque de déplacer les points à l'extérieur de la fenêtre locale et d'ici l'impossibilité pour l'algorithme de les retrouver.

L'utilisation des fenêtres de grande taille n'est pas une solution pour résoudre le problème soulevé par le paragraphe précédent. Ainsi l'utilisation des fenêtres larges est contraire à une des trois hypothèses de l'algorithme plus précisément la « Coherence spatiale ».

La solution est donnée par l'approche hiérarchique ou pyramidale (figure 1). Cette méthode respecte toutes les hypothèses de l'algorithme de Lucas-Kanade. La première fenêtre située dans la partie supérieure de la pyramide est d'une taille inférieure aux suivantes et mais fournira moins de détails que les suivantes. En effet plus on va vers la base de la pyramide plus le niveau de détail fournit augmente.

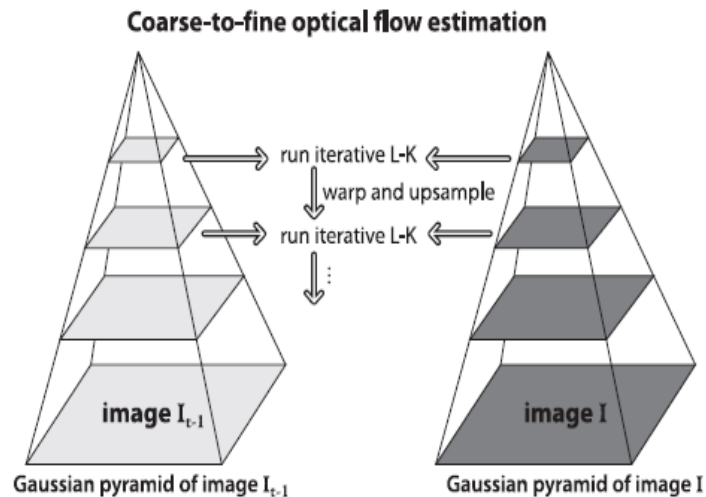
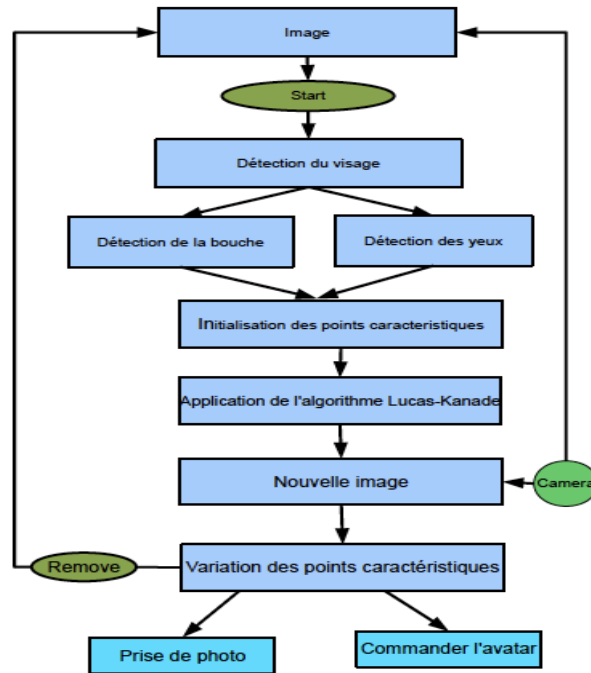


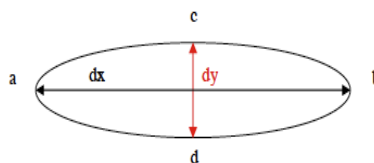
Figure 1

### 3.2. Fonctionnement du système

La figure suivante présente l'enchaînement des phases nécessaires pour arriver à obtenir une photo ou commander l'avatar.



Une fois le programme exécuté des images seront capturées et affichées dans une fenêtre sans qu'aucun traitement soit appliqué. A ce stade le programme est en attente, en effet il doit être « informé » que l'utilisateur est prêt (il ne doit pas bouger et son visage est encadré par le cercle). Pour transmettre au système qu'il est prêt l'utilisateur doit presser la touche « p » du clavier. Une fois pressée cette touche, la détection du visage ainsi que de la bouche et des yeux commence. Cette action permet d'extraire les points qui seront utilisés par l'algorithme de Lucas-Kanade. Le processus de traitement ne fait que la suivie des points d'une image à une autre. La détection des émotions se réalise en se basant sur les variations entre ces points. Par exemple afin de détecter un sourire on utilise le modèle suivant:



On peut observer que par rapport à la position de la bouche avant le sourire, la variation horizontale  $dx$  augmente significativement alors que la variation verticale  $dy$  diminue.

Une fois une émotion détectée le système va sauvegarder une photo. Comme on a déjà mentionné le système permet aussi de commander un avatar. En effet les gestes captés par le système seront répétés par l'avatar.

## 4. Conclusions

La réalisation de ce projet nous a permis de mieux comprendre des notions qui ont été traitées dans la partie théorique du cours et aussi de se familiariser avec la conception des systèmes d'interactions multimodales et aussi avec les difficultés qui peuvent être rencontrées.