



ÉCOLE POLYTECHNIQUE DE L'UNIVERSITÉ DE TOURS

64, Avenue Jean Portalis

37200 TOURS, FRANCE

Tél. (33)2-47-36-14-14

Fax (33)2-47-36-14-22

www.polytech.univ-tours.fr

Parcours des écoles d'Ingénieurs Polytech

Rapport de projet S4

Application mobile pour la détection des émotions

Auteur(s)

Alexandre Turpin

[\[alexandre.turpin@etu.univ-tours.fr\]](mailto:alexandre.turpin@etu.univ-tours.fr)

Pierre Savary

[\[pierre.savary@etu.univ-tours.fr\]](mailto:pierre.savary@etu.univ-tours.fr)

Encadrant(s)

Mohamed Slimane

[\[mohamed.slimane@univ-tours.fr\]](mailto:mohamed.slimane@univ-tours.fr)

Donatello Conte

[\[donatello.conte@univ-tours.fr\]](mailto:donatello.conte@univ-tours.fr)

Polytech Tours

Département DI

Table des matières

Introduction	1
1 Le stress	2
1.1 Qu'est ce que le stress ?	2
2 L'état de l'art	3
2.1 Première étude : Stress and anxiety detection using facial cues from videos [1] .	3
2.1.1 Provoquer un état de stress	3
2.1.2 Capturer les facteurs de stress	3
2.1.3 Résultat de l'expérience	4
2.2 Deuxième étude : Stress Detection using Wearable Physiological Sensors [2] . . .	5
2.2.1 Provoquer un état de stress	5
2.2.2 Capturer les facteurs de stress	6
2.2.3 Résultat de l'expérience	6
2.3 Troisième étude : Détection multimodale du stress pour la conception de logiciels de remédiation [3]	7
2.3.1 Provoquer un état de stress	7
2.3.2 Capturer les facteurs de stress	7
2.3.3 Résultat de l'expérience	9
2.4 Résumer de l'état de l'art	9
3 Programme de détection du stress	10
3.1 OpenCv	10
3.1.1 Haarcascade	11
3.2 Le programme de détection	12
3.2.1 Le classifieur du programme	16
4 Application mobile	17
5 Les limites et continuités du projet	18
5.1 Les limites	18
5.2 Les continuités	18
Conclusion	19
Annexes	20
A Liens utiles	21
B Fiche de suivi de projet PeiP	22

Table des figures

1.1	Image de stress	2
2.1	Détection de l'ouverture et du clignement des yeux avec AAM	4
2.2	Trier Social Stress Test	5
2.3	bracelet BioNomadix	6
2.4	Points fournis par FaceTracker	7
2.5	Quantité de mouvement par rapport à une silhouette	8
3.1	Utilisation de Haarcascade	11
3.2	Points obtenue avec Dlib	12
3.3	Les "imports" de notre programme	12
3.4	Transformation des points en coordonnées	13
3.5	Fonctions dlib	13
3.6	Affichage des points sur le visage	14
3.7	Tracé des distances sur le visage	14
3.8	Exemple d'analyse d'ouverture des yeux	15
3.9	Exemple d'utilisation de SVM	16

Introduction

Aujourd'hui, l'informatique nous permet de faire des choses impensables il y peu. Notamment dans le domaine de l'estimation, en effet on peut estimer les émotions d'une personne. Durant ce projet nous nous sommes focalisé sur le stress. L'objectif de celui-ci était de réaliser une application mobile pour l'estimation de l'état de stress d'une personne à partir d'une vidéo. Pour ce faire nous avons du comprendre comment fonctionne le stress, qu'est-ce qu'y le déclenche et comment le repérer. Ensuite, on a réaliser un programme permettant de récupérer ces facteurs de stress grâce à une vidéo. Enfin, nous devons exploiter nos résultats pour en déduire l'état de la personne, soit si elle est stressée ou non.

Chapitre 1

Le stress

1.1 Qu'est ce que le stress ?

Dans un premier temps nous devons connaître exactement ce qu'est le stress. Cependant il ne faut pas confondre stress et anxiété. En effet, ces deux termes sont liés mais n'ont pas la même signification. Le stress est un phénomène physiologique et psychologique qui est provoqué par une pression ou une agression de notre environnement. Ces pressions sont appelées "stresseurs", ils peuvent être le travail, un conflit, un bruit ... L'anxiété renvoie quant à elle à la tendance de tout être humain à anticiper et à grossir les difficultés, parfois à s'en créer ou à en imaginer, alors qu'il n'y en a pas, ou qu'elles sont mineures.

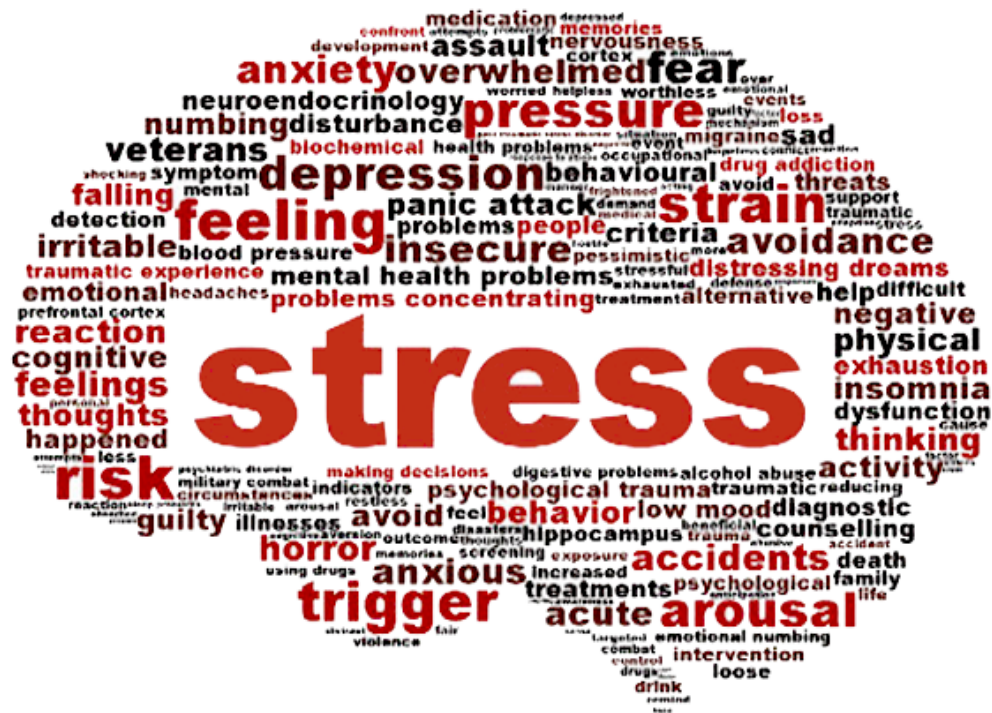


FIGURE 1.1 – Image de stress

Nous devons donc à partir d'une vidéo pouvoir repérer le stress. Pour cela il existe des facteurs de stress permettant de jauger l'état de stress d'une personne. Pour connaître ces facteurs de stress nous avons fait un état de l'art sur les expériences déjà effectué.

Chapitre 2

L'état de l'art

2.1 Première étude : Stress and anxiety detection using facial cues from videos [1]

2.1.1 Provoquer un état de stress

Dans cette expérience, les sujets ont été 23 adultes dont 16 hommes et 7 femmes âgés de 30 à 55 ans. Pour évaluer le stress de ces personnes, ils leur ont fait passer un test divisé en quatre parties.

- Tout d'abord, le sujet doit rester neutre pendant une minute devant la caméra pour servir de référence au reste du stress. Puis il effectue une description de lui-même dans une langue étrangère pendant une minute et enfin il lit un texte de 85 mots émotionnellement neutre pendant 30 secondes.
- Ensuite, le sujet reste neutre pendant encore 30 secondes et doit ensuite se rappeler pendant 30 secondes des événements anxieux puis 30 autres secondes d'événements stressants.
- Suite à cela, le sujet doit regarder des images (15 secondes par image) neutres et stressantes pendant 2 minutes, puis le sujet doit réaliser une tâche mentale, pendant 2 minutes, de lire un mot indiquant une couleur avec le mot écrit dans une autre couleur (exemple le mot rouge écrit en bleu).
- Enfin, le sujet doit rester neutre pendant 30 secondes et doit ensuite regarder 3 vidéos de 2 minutes dont une relaxante, une d'un film d'aventure et une stressante.

Cette expérience est filmée en vidéo de 14 minutes et 30 secondes.

2.1.2 Capturer les facteurs de stress

Durant cette expérience, les chercheurs ont décidé de déterminer le stress avec les facteurs de stress des lèvres, des yeux, des mouvements de la tête et de la fréquence cardiaque. Pour cela ils ont utilisé le système Active Appearance Models(AAM) pour détecter les yeux et extraire les données utiles à la détection de stress. Ici, l'utilisation de l'AAM permet de placer six repères au niveau des yeux et dix points au niveau de la bouche. Puis la largeur des yeux est utilisée pour déterminer si la personne est stressée en comparant celle obtenue lors des différentes étapes de l'expérience par rapport à celle obtenue lorsque le sujet restait neutre. La largeur sert aussi à détecter le clignement des yeux lorsque celle-ci diminue brusquement (les valeurs ont été vérifiées et filtrées pour éliminer les pointes artificielles créées lorsque le système perdait le repérage des yeux). Les différents résultats de l'expérience sont ensuite traités par Repeated measures ANOVA.

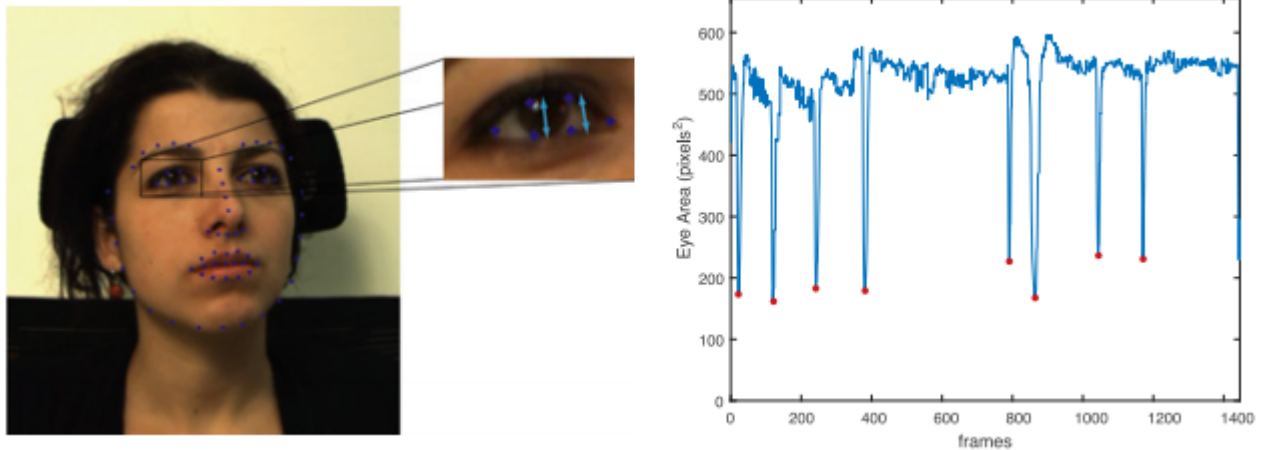


FIGURE 2.1 – Détection de l'ouverture et du clignement des yeux avec AAM

De plus, l'Active Appearance Models est aussi utilisé pour déterminer les mouvements de la tête en déterminant 5 repères sur le nez, puis toutes les 100 images, le programme calcule la distance euclidienne entre les points de l'image et les points de l'image de référence. L'expérience a montré que la vitesse et les mouvements de la tête augmentent lors des phases stressantes. Ensuite, ils utilisent le ROI (region-of-interest) au niveau de la bouche pour calculer la variation des intervalles de temps entre les extrémités de la bouche (VTI) qui est utilisé pour estimer la périodicité des mouvements en se basant sur l'observation que les mouvements rythmiques produiraient des variations proches de zéro. Une VTI plus élevée implique une diminution de la rythmicité des mouvements des lèvres associée à des niveaux accrus de stress et de l'anxiété (cette fonctionnalité n'a pas été utilisée dans les étapes 1 et 3 où les mouvements de la bouche associés avec le fait de parler fort pouvaient entraîner une confusion avec l'activité liée à l'émotion). Enfin, ils utilisent aussi la vidéo pour calculer la fréquence cardiaque basée sur la photopléthysmographie en utilisant le ROI. La photopléthysmographie étant le fait d'envoyer de la lumière par une LED infrarouge puis la lumière émise par notre LED est plus ou moins réfléchi en fonction de la circulation du sang.

2.1.3 Résultat de l'expérience

Une fois toutes les données collectées, ils les ont passées dans un classifieur via l'algorithme "SVM". L'algorithme va ensuite sortir l'état de la personne étant soit neutre, relaxé ou stressé.

2.2 Deuxième étude : Stress Detection using Wearable Physiological Sensors [2]

2.2.1 Provoquer un état de stress

Pour provoquer un état de stress, ils utilisent la méthode « Treves Stress Test » ou « Trier Social Stress Test » (TSST). Le TSST consiste à donner une tâche neutre, suivi d'une prise de parole en publique, puis d'une tâche cognitive et enfin d'une tâche neutre. Les tâches neutres sont des tâches qui ne stress pas la personne comme par exemple répondre à des questions du quotidien comme « Qu'est-ce que tu penses du temps d'aujourd'hui ? » ou « Tu es venu en voiture ? ». La prise de parole en publique de 5 minutes est une simulation d'entretien d'embauche. Après cela vient la tâche cognitive, on demande à la personne de compter à l'envers de 13 en 13 en partant de 1022 (soit $1022 - 1019 - 1016 - 1013 - 1010 - \dots$).



FIGURE 2.2 – Trier Social Stress Test

Les sujets arrivent et reçoivent un briefing. Après le briefing, les sujets doivent remplir une feuille pour donner leur état de stress avant l'expérience. Puis on leur met le bracelet avec les électrodes. Ensuite vient des questions neutres qui permettent de déterminer une base, l'état neutre du sujet. On demande ensuite au sujet de s'asseoir près d'un bureau et de préparer une présentation pour un entretien d'embauche pendant 3 minutes. Une fois les 3 minutes écoulées, il prend sa feuille et fait sa présentation devant un groupe de personne. Pendant 5 minutes on encourage le sujet à parler, à leur première pause on lui demande de continuer. A la pause suivante on lui pose des questions typiques d'un entretien d'embauche comme : « Quels sont vos qualités et vos défauts ? », « Où vous voyez-vous dans 5 ans ? » et d'autres. On explique ensuite la tâche cognitive au sujet et il commence à 1022. Quand le sujet dit un mauvais nombre il doit recommencer à 1022. A la fin de la tâche cognitive, on repose des questions neutres pour lui laisser un petit temps pour se relaxer. Pour finir on redemande au sujet d'évaluer son état de stress après l'expérience.

2.2.2 Capturer les facteurs de stress

Dans cette expérience, ils capturent les données grâce au bracelet BioNomadix module de Biopac.



FIGURE 2.3 – bracelet BioNomadix

Ce bracelet possède aussi deux électrodes sur deux doigts qui mesure les signaux de l'« electro-dermal activity (EDA) » et des capteurs BVP (Blood Volume Pulse) . L'EDA permet de mesurer la conductance de la peau qui est une indication de la transpiration de la personne. Le BVP est obtenu en envoyant un signal lumineux sur la peau et en mesurant le taux d'absorption de la lumière par la peau. La quantité de lumière qui est retournée est proportionnel au volume de sang dans la peau. Ensuite ils ont transféré les données sur le logiciel « AcqKnowledge software » qui va comparé les données. Ils calculent la variabilité de fréquence cardiaque en calculant la variation entre les battements du cœur mesurés et ceux « normaux » d'une base de données.

2.2.3 Résultat de l'expérience

Le TSST a été effectué sur 5 hommes et femmes âgées de 18 à 39 ans. On compare donc leur EDA et BVP durant l'expérience par rapport à la référence que l'on a pris au début de l'expérience. Ce qui nous donne un score entre -1/1, plus on se rapproche de -1 plus la personne est stressé et plus on se rapproche de 1 moins la personne est stressé.

2.3 Troisième étude : Détection multimodale du stress pour la conception de logiciels de remédiation [3]

2.3.1 Provoquer un état de stress

Dans cette expérience on ne sait pas comment le stress est provoqué chez les personnes mais elle a été faite sur différentes catégories de personnes composé d'hommes et de femmes :

- Deux expériences sur des sociophobes et témoins d'un âge moyen de 35 ans.
- Quatre expérience sur des étudiants d'un âge moyen de 26 ans
- Deux expériences sur des chercheurs de pays différents.

Pendant ces expériences, ils récupèrent plusieurs indices de stress.

2.3.2 Capturer les facteurs de stress

Tout d'abord, ils enregistrent les sujets grâce à un micro-cravate. Ils analysent l'audio par fenêtre glissante. Chaque fenêtre comporte N secondes d'enregistrement, et les fenêtres se chevauchent de façon à ce que N/2 secondes soient incluses dans deux fenêtres consécutives. Ensuite, pour calculer le stress ils utilisent les 384 descripteurs audio du set emoIS09 d'OpenSMILE.

Au niveau du visage, les expressions faciales sont contrôlées par les zones des yeux et de la bouche. Ils s'intéressent donc particulièrement aux froncements et haussement de sourcils, et aux sourires et moues. Les indices décrivant la position et le mouvement de ces parties du visage sont largement employé pour la reconnaissance d'expressions faciales, par annotation manuelle, par modèles d'apparence (Active Appearance Model, "AAM") ou encore par motif binaire local (Local Binary Patterns, "LBP"). Pour détecter le visage, ils utilisent le logiciel FaceTracker. Celui-ci permet de détecter les sourcils, les yeux, le nez, la lèvre et le menton grâce à 66 points.

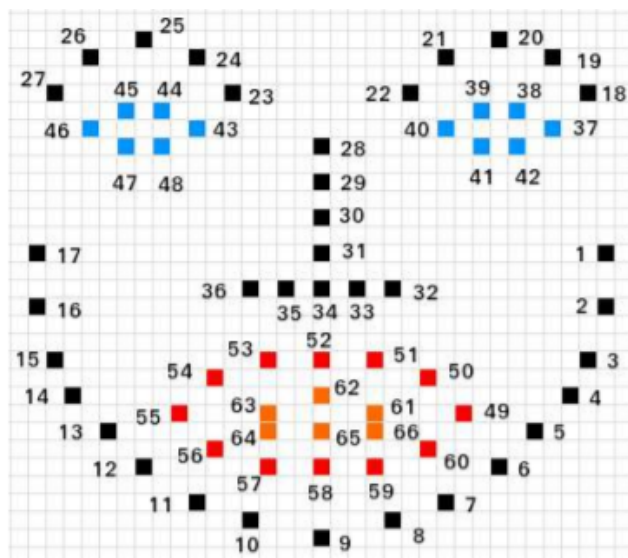


FIGURE 2.4 – Points fournis par FaceTracker

A partir des points extraits à l'aide de FaceTracker, nous calculons pour chaque image les distances suivantes : arche des sourcils, écart entre les sourcils, écart entre les sourcils et les yeux, ouverture de la bouche, rapport entre l'épaisseur de la lèvre supérieure et l'épaisseur de la lèvre inférieure. Les trois premières mesures quantifient les mouvements des sourcils, les deux dernières ceux de la bouche. Ensuite, ils calculent 7 descripteurs quantifiant les expressions faciales :

- Arche sourcils
- Ecart sourcils
- Ecart sourcils – yeux
- Ouverture bouche
- Rapport lèvres
- Largeur et hauteur boîte englobante
- Abscisse et ordonnée boîte englobante

La boîte englobante sert à normaliser la taille du visage de la personne. Enfin, pour vérifier que le visage de la personne est de face, ils utilisent le modèle HaarCascade d'OpenCV. Si la personne n'est pas de face, ils considèrent l'image comme non utilisable. Ensuite, ils vont capturer la posture. Afin de qualifier les mouvements des sujets, utilisent l'indice de contraction (CI) et la quantité de mouvement (QM). L'indice de contraction décrit dans quelle proportion le sujet observé est ramassé sur lui-même, en particulier à quel point ses bras sont proches de son corps. Il correspond au rapport entre l'aire de la silhouette du sujet et l'aire de la « boîte englobante » de cette silhouette (le plus petit rectangle contenant l'ensemble des pixels de la silhouette). Pour une image donnée i , l'indice de contraction est : $CI(i) = A(bi) - A(si)$ où $A(bi)$ est l'aire de la boîte englobante de la silhouette dans l'image i , et $A(si)$ est l'aire de la silhouette dans l'image i . Cet indice décrit la posture du sujet à un instant donné. Ils calculent la dérivée des CIs pour obtenir sa variation. Puis ils calculent 7 descripteurs statistiques sur les M valeurs du CI^* et de sa dérivée : moyenne, déviation standard, minimum, maximum, étendue, asymétrie, et aplatissement. La QM décrit le déplacement d'un corps dans l'espace sur une période de temps donnée ; il est calculé sur une suite d'images successives, en notant pour chaque image les pixels occupés par la silhouette pour construire la « silhouette du mouvement », une image représentant la forme et la position du corps sur la succession d'images considérée. Il s'agit d'une variante des images mouvement-énergie (Motion-Energy Images) qui exclut la silhouette de la dernière image pour ne retenir que la silhouette du mouvement. La QM correspond au rapport de l'aire de la silhouette de « mouvement » sur l'aire de la silhouette « mouvement et corps ». Plus ce rapport est élevé, plus le mouvement aura été important pendant la série d'images considérée.



FIGURE 2.5 – Quantité de mouvement par rapport à une silhouette

2.3.3 Résultat de l'expérience

Au final, les différentes expériences ont permis d'évaluer le stress avec différentes échelles, une échelle à 10 niveaux, une à 3 niveaux(stress faible, moyen et fort) et une en binaire(stress ou non). L'évaluation du stress était calculée par des experts et auto-évalué par le sujets, grâce à cela ils ont pu déterminé ce qui est évident que les échelles avec 10 et 3 niveaux étaient moins efficaces et que le sujet et les experts étaient le plus souvent d'accord pour l'échelle en binaire.

L'algorithme SVM a été utilisé pour déterminer le stress par auto-apprentissage à partir de la base de donnée obtenu au cours des expériences en séparant les différents facteurs de stress.

2.4 Résumer de l'état de l'art

Cet état de l'art nous a permis de renseigner sur les différents facteurs de stress existant et sur les méthodes qui permettent de les récupérer. Grâce à ça, nous allons pouvoir choisir ce qu'on allait utiliser pour notre projet.

Facteur de stress	Description
Les yeux	Au niveaux des yeux on peut relever deux informations. Dans un premier temps, plus l'ouverture des yeux est grande plus une personnes peut être stressé. On peut calculer cela directement sur une vidéo. Ensuite, plus une personne cligne des yeux, plus elle est stressée. On pourra donc calculer la fréquence de clignement par minute pour évaluer le stress.
La tête	Au niveau de la tête plus elle bouge plus la personne est stressé pour calculé cela on fait des mesures avec la position du nez sur le visage.
La peau	On peut calculer le stress grâce à la température de la peau et à la conductivité de la peau mais cela nécessite un équipement extérieur (par exemple le bracelet :Bio-Nomadix model BN-PPGED)
La voix	Le stress va modifier la voix d'une personne et son débit de parole. On peut donc capturer la voie de la personne avec un micro pour savoir si une personne est stressé.
Autres	Il existe d'autre moyen pour évaluer le stress, comme avec la fréquence cardiaque en effet plus celle-ci est élevé plus on est stressé. De plus l'augmentation du taux de cortisol dans le sang ou la salive, du rythme respiratoire permettent aussi de savoir si une personne est stressée.

Pour notre projet nous avons choisi de détecter le stress qu'avec une caméra car nous n'avons pas les dispositifs nécessaires pour capturer les facteurs de stress de la peau. Nous avons donc pris comme facteurs de stress :

- Ouverture des yeux
- Fréquence de clignement des yeux
- Mouvement de la tête

Chapitre 3

Programme de détection du stress

Pour notre programme nous allons utiliser utiliser Python.



Pourquoi Python ?

Python va nous être utile car il est multiplateforme et est très pratique dans l'étude de données.

3.1 OpenCv

Pour faire du traitement d'image nous allons utiliser la bibliothèque OpenCv. En effet celle-ci est spécialisée dans le traitement d'image et de vidéo, elle permet notamment la lecture, écriture, l'affichage d'une vidéo et la détection de visages.



3.1.1 Haarcascade

Dans un premier temps, il nous fallait détecter le visage d'une personne dans une vidéo. Pour cela on utilise le classifieur Haarcascade d'OpenCv. Celui-ci nous a permis de récupérer les coordonnées du visage de la personne et de ses yeux.

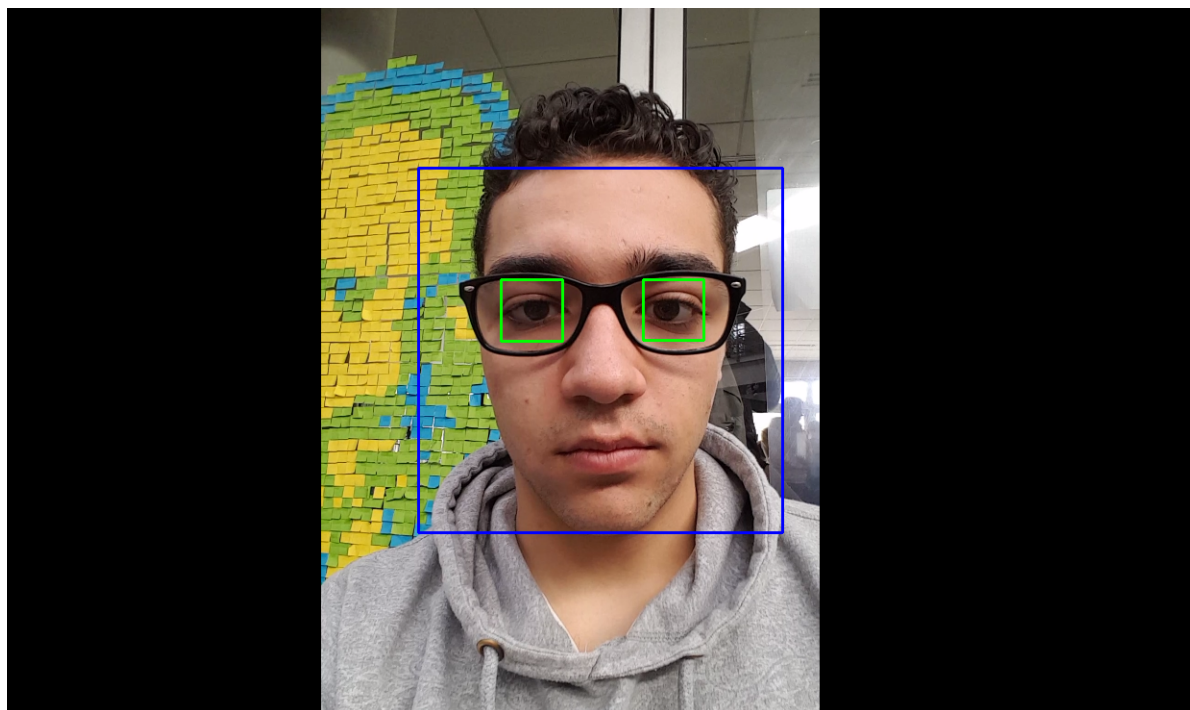


FIGURE 3.1 – Utilisation de Haarcascade

Cependant, nous avons réalisé qu'avec Haarcascade nous sommes limités car nous ne pouvons pas extraire les données que nous souhaitons vu que nous avons que la position des yeux sur le visage. C'est à dire l'ouverture des yeux, la fréquence de clignements des yeux et le mouvement de la tête. Nous avons donc continué nos recherches pour au final utiliser le package Dlib.

3.2 Le programme de détection

On utilise Dlib car il nous permet de tracker le visage d'une personne avec des points.

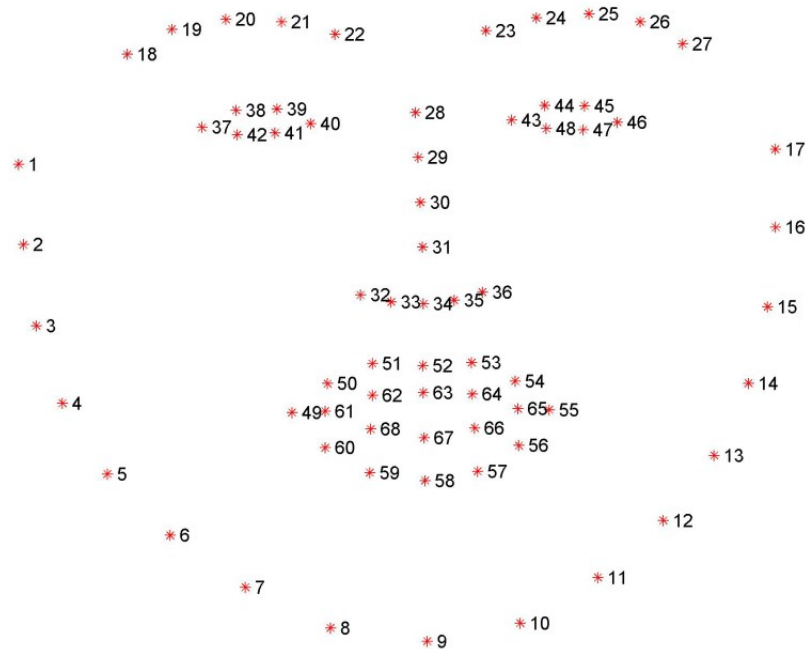


FIGURE 3.2 – Points obtenue avec Dlib

Voici les parties importantes du programme :

```
3 import cv2
4 import numpy as np
5 import dlib
6 import matplotlib.pyplot as plt
7 from matplotlib import style
8 style.use("ggplot")
9 from sklearn import svm
```

FIGURE 3.3 – Les "imports" de notre programme

Tout d'abord on importe les packages que l'on va avoir besoin. Ici, "cv2" correspond à OpenCv, "numpy" nous sert à manipuler des matrices et des tableaux, "dlib" va nous servir à mettre les points sur la vidéo, "matplotlib" permet de réaliser des graphiques et "sklearn" correspond à scikit-learn qui va nous permettre de réaliser un classifieur.

```

12 def land2coords(landmarks, dtype="int"):
13
14     coords = np.zeros((68, 2), dtype=dtype)
15
16     for i in range(0, 68):
17         coords[i] = (landmarks.part(i).x, landmarks.part(i).y)
18
19     return coords

```

FIGURE 3.4 – Transformation des points en coordonnées

Cette fonction transforme les points obtenus grâce à Dlib en coordonnées pour pouvoir les utiliser plus tard. Pour ce faire on crée une matrice de 2 colonnes et 68 lignes qu'on remplit avec les coordonnées des points.

```

40 face_detector = dlib.get_frontal_face_detector()
41 landmark_predictor = dlib.shape_predictor('shape_predictor_68_face_landmarks.dat')

```

FIGURE 3.5 – Fonctions dlib

On appelle face-detector la fonction renvoyant les coordonnées du visage et landmark-predictor la fonction renvoyant les coordonnées des points du visage.

```

43 vid = cv2.VideoCapture(0)

```

Cette ligne permet de lire la vidéo. Dans les parenthèses on indique le chemin vers une vidéo ou on utilise le "0" pour utiliser la caméra principal de l'ordinateur.


```

66     frame_gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
67
68     face_boundaries = face_detector(frame_gray,0)
69
70     for (enum,face) in enumerate(face_boundaries):
71         x = face.left()
72         y = face.top()
73         w = face.right() - x
74         h = face.bottom() - y
75
76         cv2.rectangle(frame, (x,y), (x+w, y+h), (120,160,230),2)
77
78         landmarks = landmark_predictor(frame_gray, face)
79         landmarks = land2coords(landmarks)
80
81         for (a,b) in landmarks:
82             cv2.circle(frame, (a, b), 2, (255, 0, 0), -1)

```

FIGURE 3.6 – Affichage des points sur le visage

Pour reconnaître le visage avec Dlib, l'image doit être en noir et blanc donc on la transforme avec OpenCv (`cv2.cvtColor`). Une fois les coordonnées de la tête récupérées on les stocke dans des variables `x,y,w,h` et on affiche un rectangle à ces coordonnées entourant le visage. Ce rectangle est notre ROI (Region of interest) à partir de laquelle on recherche les points du visage à l'intérieur. Puis on associe la position des points à une matrice de coordonnées. Enfin, on affiche les points avec des petits cercles.

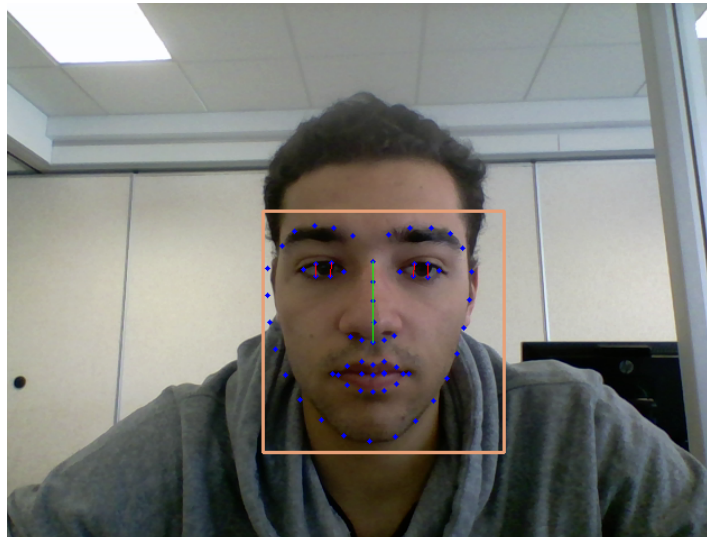


FIGURE 3.7 – Tracé des distances sur le visage

Ensuite, on va calculer la distance entre les deux paupières, c'est à dire entre les points 38-42, 39-41, 44-48, 45-47. Ces distances permettent de calculer l'ouverture des yeux et la fréquence de clignement des yeux. On vient ensuite calculer le déplacement du nez entre deux images en fonction des coordonnées des points 28 et 34.

Pour effectuer nos calculs il faut qu'on récupère le durée de la vidéo. Ce que l'on fait en divisant le nombre d'image de la vidéo par le nombre d'images par seconde. Ensuite, on calcul le nombre de clignement. On prend dans un premier temps le milieu de l'ouverture des yeux soit la plus grande valeur moins la plus petite sur l'ensemble de la vidéo. On considère que chaque valeur au dessus de ce milieu ne peut pas correspondre à un clignement des yeux. Ensuite pour affiner le résultat on calcul la moyenne de l'ouverture des yeux sur 5 images que l'on compare au milieu (max - min) sur 20 images.

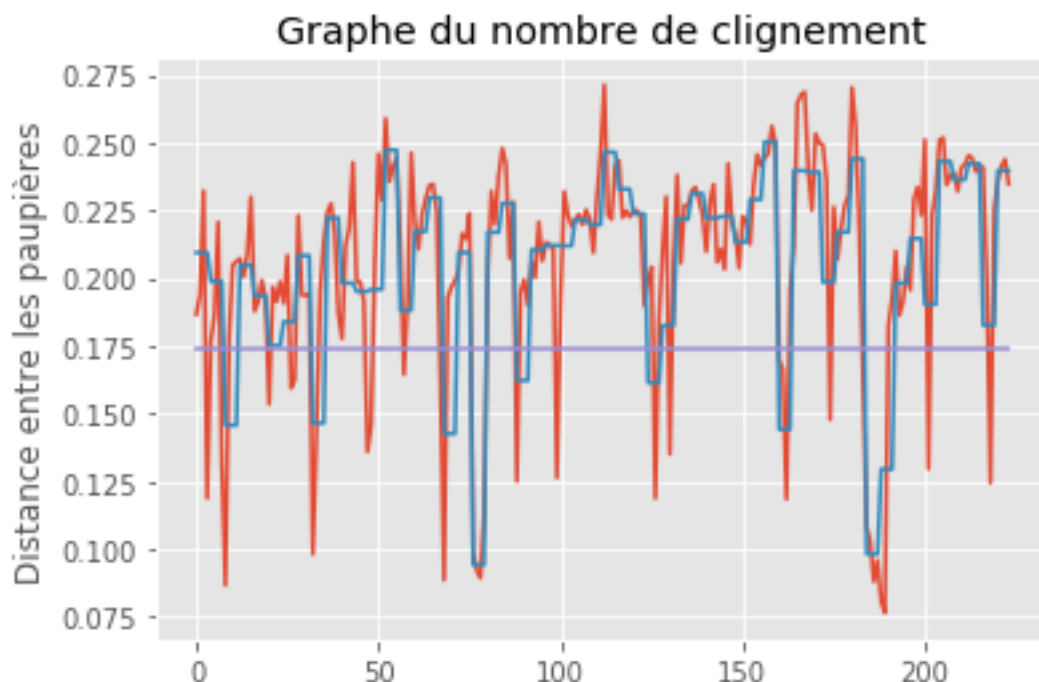


FIGURE 3.8 – Exemple d'analyse d'ouverture des yeux

On obtient donc le nombre de clignement dans la vidéo, ce qui nous permet de déterminer la fréquence de clignement. On calcul la moyenne de l'ouverture des yeux et du mouvement de la tête sur la vidéo.

3.2.1 Le classifier du programme

Un classifier consiste à attribuer une classe à chaque objet en se basant sur des données obtenues au préalable. On utilise l'algorithme SVM (Support Vector Machines) de scikit-learn.

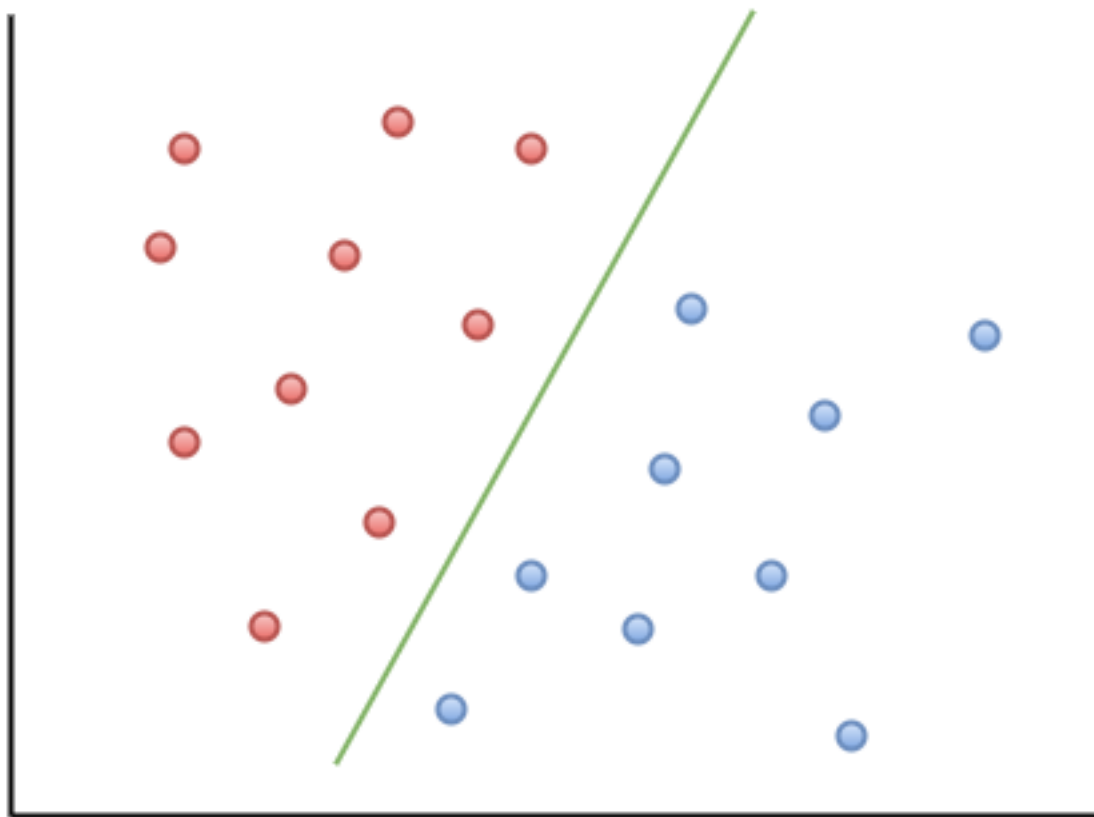


FIGURE 3.9 – Exemple d'utilisation de SVM

On définit la base de données du classifier comme étant les résultats obtenues de six vidéos, 3 non-stressées, 3 stressées. Le programme doit alors prédire dans quel ensemble se situe les résultats obtenues en analysant une nouvelle vidéo.

Chapitre 4

Application mobile



Nous avons essayé de créer une application utilisant le programme sur une vidéo capturée sur le téléphone. Pour cela nous avons utilisé le logiciel Android studio pour créer une application android.

Dans un premier temps il fallait demander les autorisations du téléphone pour utiliser la caméra.

```
<manifest ... >
    <uses-feature android:name="android.hardware.camera"
                android:required="true" />
    ...
</manifest>
```

Nous avons aussi réussi à programmer un bouton pour effectuer une action. Cependant nous n'avons par réussi à enregistrer une vidéo depuis l'application.

Chapitre 5

Les limites et continuités du projet

5.1 Les limites

Le projet est néanmoins limité. En effet nous avons rencontré des limites :

- On ne pouvait pas récupérer certains facteurs de stress comme ceux recueillis par le bracelet (fréquence cardiaque, débit sanguin, conductivité et température de la peau). Ainsi que le rythme respiratoire et le taux de cortisol pour des raisons pratiques.
- Il est dur de mesurer le stress chez une personne à cause de l'« effet Blouse Blanche », c'est le fait d'être stressé car quelqu'un vous observe et mesure votre stress.
- Il est difficile de différencier certains comportements comme le fait d'être timide avec le stress ou encore la peur et l'anxiété (appréhension du danger alors qu'il n'y en a pas). Les humains sont tous différents, donc les paramètres peuvent grandement varier selon les individus.
- Le programme est lui aussi limité avec la détection d'un clignement qui peut en compter plusieurs au lieu d'un. Mais aussi que l'ouverture des yeux et du mouvement de la tête ne sont pas des facteurs de stress pertinents.

5.2 Les continuités

Avec plus de temps et de moyen, nous aurions pu améliorer la détection de stress :

- Nous aurions pu ajouter d'autres paramètres au programme basés sur les autres facteurs de stress que nous n'avons pas utilisés qui auraient permis d'affiner le programme.
- Avec plus de temps nous aurions pu améliorer la base de données du classifieur et nous aurions pu finir l'application Android.
- Pour utiliser le programme sur l'application nous aurions pu passer par internet en demandant l'autorisation du manifeste.

Conclusion

Avec ce projet nous avons étudié les émotions plus particulièrement le stress et nous avons essayé de le traduire grâce à l'informatique. Cependant nous nous sommes rendus compte que même si c'est possible cela reste limité car nous avons tous une manière différente d'exprimer le stress. Nous avons aussi pu découvrir le domaine de la reconnaissance faciale que nous pensions inaccessible à notre niveau. Nous avons aussi amélioré notre niveau en programmation python et notre travail d'équipe.

Bibliographie

- [1] "Stress and anxiety detection using facial cues from videos" par G. Giannakakis, M. Pedaditis, D. Manousos, E. Kazantzaki, F. Chiarugi, P.G. Simos, K. Marias, M. Tsiknakis. (2016)
- [2] "Stress Detection using Wearable Physiological Sensors" par Virginia Sandulescu, Sally Andrews, David Ellis, Nicola Bellotto, Oscar Martinez Mozos. (2015) <http://eprints.lancs.ac.uk/76167/1/proof.pdf>
- [3] "Détection multimodale du stress pour la conception de logiciels de remédiation" par Mariette Soury (2014) <https://tel.archives-ouvertes.fr/tel-01082865/document>

Annexe A

Liens utiles

Voici une petite liste d'url intéressantes au sujet de ce projet :

- www.polytech.univ-tours.fr
- <https://www.python.org/>
- <https://opencv.org/>
- <http://scikit-learn.org/stable/>
- <http://dlib.net/>
- <https://developer.android.com/studio/index.html>

Annexe B

Fiche de suivi de projet PeiP

Séance n° 1 du 19/01/2018	<p>Lecture de l'article "Biomedical Signal Processing and Control" :</p> <ul style="list-style-type: none">— l'étude se base sur la détection de 3 états différents (neutre, relaxé ou stressé) à travers différents facteurs de stress.— l'expérience se base sur l'analyse de l'activité de la bouche, des yeux, des mouvements de la tête et de la fréquence cardiaque.— l'article explique en introduction la notion de stress et d'anxiété sur laquelle ils se basent au cours de l'expérience. En effet le stress et l'anxiété sont en soit la même chose mais l'anxiété est un stress que l'on éprouve avant un événement alors que le stress est ce que l'on ressent pendant l'événement .Il explique aussi les différents effets du stress que l'on peut remarquer.— Ensuite l'article présente comment ils avaient mené l'expérience et les différentes phases pour relever des données chez différents sujets ainsi que la manière de calculer le stress d'un sujet à travers toutes les données(différents algorithmes).— Enfin, l'article présente les différents résultats qu'ils ont obtenu. <p>Résumé de l'entretien du 19/01/2018 : Le travail à effectuer est la lecture d'autres articles sur la détection des émotions et de chercher une définition du stress. A travers ces articles, il faudra faire un état de l'art pour trouver les différents moyens d'évaluer le stress chez une personne. Ensuite nous choisirons un de ces principes pour pouvoir l'appliquer dans une application mobile codée en python grâce à OpenCV.</p>
------------------------------	---

<p>Séance n° 2 du 26/01/2018</p>	<p>Nous avons continué la lecture d'article sur le stress et la détection de celui-ci.</p> <p>Premier article : http://www.psychologies.com/Moi/Problemes-psy/Anxiete-Phobies/Reponses-d-expert/Quelle-difference-y-a-t-il-entre-stress-et-anxiete.</p> <p>Définition du stress : "La réaction se passant dans notre corps et notre esprit lorsque nous sommes soumis une pression ou une agression de notre environnement (ce que l'on appelle des "stresseurs" : travail, conflit, bruit...)". Définition de l'anxiété : "L'anxiété renvoie quant à elle à la tendance de tout être humain à anticiper et à grossir les difficultés, parfois à s'en créer ou à en imaginer, alors qu'il n'y en a pas, ou qu'elles sont mineures."</p> <p>Deuxième article : http://eprints.lancs.ac.uk/76167/1/proof.pdf</p> <p>Le but des recherches de l'article est de créer des nouveaux équipements pour détecter plus facilement l'état mental et physique chez une personne au quotidien. En particulier, la dépression et l'anxiété. Les différents facteurs de stress sont la fréquence cardiaque, le débit sanguin, la température de la peau, la dilatation des pupilles, la conductivité de la peau. Pour le détecter, ils utilisent une machine qui utilise le "deep learning" pour détecter le stress avec des bracelets. Ceux-ci sont équipés de capteurs physiologique permettant de récupérer des données sur la conductance de la peau pour savoir si une personne transpire, ainsi que le débit sanguin. Ces données permettent de classer la personne dans l'une des deux catégories : "stressé" ou "non stressé". Avec toutes ces données on calcul avec un algorithme un score compris entre -1 et 1 où -1 correspond à une personne "non stressé" et 1 à une personne "stressé". Ce calcul s'effectue en temps réel toutes les 0.1 sec. Les paramètres de l'expérience consistaient à stresser les personnes en les faisant parler en publique et résoudre des tâches cognitives.</p> <p>Troisième article : https://tel.archives-ouvertes.fr/tel-01082865/document</p> <p>Définition du stress : «Le stress est une réaction physiologique à une situation perçue comme menaçante pour l'individu stressé. C'est la réponse « Fight or Flight », une réaction du système nerveux orthosympathique à une situation compromettant la survie de l'individu afin de préparer le corps pour une réponse offensive ou défensive pour échapper au danger.»</p> <p>Deux états peuvent déclencher la réaction physiologique de stress : la peur, et l'anxiété. Les meilleurs méthodes pour évaluer le stress sont les variations des rythmes cardiaque et respiratoire, la conductance de la peau, et le taux de cortisol présent dans le sang ou la salive. Cependant il est dur de mesurer le stress chez une personne à cause de l'« effet Blouse Blanche », c'est le fait de stresser car quelqu'un vous observe et mesure votre stress.</p> <p>Enfin nous avons fait quelques recherches sur Open CV.</p>
--------------------------------------	--

Séance n° 3 du 02/02/2018	<p>Aujourd'hui nous avons continué de lire des articles sur le stress et nous avons fait un résumé sur les facteurs de stress pouvant être utilisés pour évaluer l'état de stress d'une personne :</p> <ul style="list-style-type: none"> — Les yeux : Au niveau des yeux on peut relever deux informations. Dans un premier temps, plus l'ouverture des yeux est grande plus une personne peut être stressée. On peut calculer cela directement sur une vidéo. Ensuite, plus une personne cligne des yeux, plus elle est stressée. On pourra donc calculer la fréquence de clignement par minute pour évaluer le stress. — La tête : Au niveau de la tête plus elle bouge plus la personne est stressée pour calculer cela on fait des mesures avec la position du nez sur le visage. — La peau : On peut calculer le stress grâce à la température de la peau et à la conductivité de la peau mais cela nécessite un équipement extérieur (par exemple le bracelet :BioNomadix model BN-PPGED) — La voix : Le stress va modifier la voix d'une personne et son débit de parole. On peut donc capturer la voix de la personne avec un micro pour savoir si une personne est stressée. — Autres : Il existe d'autres moyens pour évaluer le stress, comme avec la fréquence cardiaque en effet plus celle-ci est élevée plus on est stressé. De plus l'augmentation du taux de cortisol dans le sang ou la salive, du rythme respiratoire permet aussi de savoir si une personne est stressée. <p>Enfin nous avons installé OpenCv et commencé à chercher comment capturer le visage d'une personne sur une vidéo.</p> <p>Questions : Nous aimerions faire un point sur le choix des facteurs pour déterminer le stress et comment calculer le degré de stress à partir de ces différents facteurs.</p>
Séance n° 4 du 09/02/2018	<p>Nous avons rédigé l'état de l'art pour vous le présenter mardi 13/02 à 17h, puis nous avons continué nos recherches sur OpenCv pour détecter les yeux et le visage d'une personne à partir d'une vidéo.</p>
Séance n° 5 du 16/02/2018	<p>Aujourd'hui nous avons repris les différentes méthodes pour détecter le stress et nous avons décrit leurs principes, c'est à dire comment se déroule l'expérience. Qu'utilise-t-on pour faire l'expérience, dans quelles conditions sont mises les personnes passant l'expérience. On a aussi regardé s'ils utilisaient une base de données déjà existante pour comparer leurs résultats et s'ils donnaient en sortie un pourcentage de stress ou des catégories de stress (exemple : "stressé" , "non-stressé").</p>
Séance n° 6 du 23/02/2018	<p>Aujourd'hui nous avons continué à chercher les différentes méthodes pour détecter le stress et leurs principes. Je vous joins ce que l'on a fait avec ce mail.</p>

Séance n° 7 du 09/03/2018	Aujourd'hui, nous avons commencé à étudier la programmation de différents logiciels permettant la reconnaissance faciale sur OpenCv (exemple : Active Appearance model : AAM). Puis nous avons regardé comment appliquer cela sur mobile.
Séance n° 8 du 16/02/2018	Aujourd'hui, nous avons réussi à traquer les visages sur une vidéo en utilisant Haar cascade d'OpenCV. Cela nous donne donc un rectangle sur le visage de la personne et sur les yeux de la personne. Cependant, nous n'arrivons pas à traquer le visage avec des points, nous essayons avec le package "dlib". Ensuite au niveau de l'application Android, on a commencé à regarder comment créer une application avec Android Studio et on a le code pour capturer une vidéo. Questions : Nous aimerions avoir un rendez-vous avec vous pour nous aider avec le code pour traquer le visage avec des points. Nous sommes disponible le mardi 20 et mercredi 21 matin ou après 16H.
Séance n° 9 du 19/02/2018	Aujourd'hui nous avons réussi à traquer le visage avec des points. Nous avons calculé l'ouverture des yeux et nous l'avons affiché dans un tableau. De plus nous avons continué à regarder la programmation sur android studio pour enregistrer une vidéo.
Séance n° 10 du 23/03/2018	Aujourd'hui, nous avons fini le de coder le programme qui nous donnent nos variables. Nous avons ensuite cherché des vidéos de personnes stressées et non stressées. Puis nous avons commencé à étudier le fonctionnement de scikit-learn pour le classifier.
Séance n° 11 du 30/03/2018	Aujourd'hui, nous avons fait des vidéos de personnes non stressé il nous reste à trouver un moyen de stressé des personnes pour les filmer et utiliser les vidéos pour faire le classifier.
Séance n° 12 du 06/04/2018	Aujourd'hui, nous avons fini le programme. Nous avons dans un premier temps régler le calcul de clignement des yeux car on avait un plus de clignement que ce que l'on devait avoir. Puis avons utilisé 3 vidéos de personnes stressées et 3 vidéos de personnes non stressées pour faire le classifier avec scikit-learn. Il nous reste plus qu'à refaire quelque test avec d'autres vidéos.

Application mobile pour la détection des émotions

Rapport de projet S4

Résumé : Ce projet consistait à créer une application mobile permettant d'estimer si une personne était stressée ou non. Pour ce faire on a utilisé différents facteurs de stress comme le clignement des yeux, l'ouverture des yeux et la quantité de mouvement de la tête. Pour récupérer ses facteurs à partir d'une vidéo nous avons utilisé les packages OpenCv et Dlib de python pour reconnaître le visage de la personne et l'entourer de points. Ensuite, grâce à scikit-learn on a fait un classifieur qui va permettre de dire si la personne est stressée en fonction des données trouvées.

Mots clé : stress, facteurs de stress, python, OpenCv, reconnaissance faciale, Dlib, scikit-learn, classifieur

Abstract : This project consisted in creating a mobile application allowing to consider if a person was stressed or not. To do this, we used different stress factors as the blinking rate, eye aperture and head movement. To get back those factors from a video we used the package OpenCv and Dlib of python to recognize the face of the person and surround it with points. Then, thanks to scikit-learn we are going to make a classifier which is going to tell us if people is put under stress according to the data found.

Keywords : stress, stress factor, python, OpenCv, facial recognition, Dlib, scikit-learn, classifier

Auteur(s)

Alexandre Turpin

[alexandre.turpin@etu.univ-tours.fr]

Pierre Savary

[pierre.savary@etu.univ-tours.fr]

Encadrant(s)

Mohamed Slimane

[mohamed.slimane@univ-tours.fr]

Donatello Conte

[donatello.conte@univ-tours.fr]

Tours

Département DI

Ce document a été formaté selon le format EPUProjetPeiP.cls (N. Monmarché)