UNIVERSITÉ CAEN NORMANDIE

LEGRAIN MIKAEL [21706525]
PETIT TITOUAN [21706474]

MASTER 2 INFORMATIQUE

Analyse de la façon de taper au clavier par le son

Table des matières

Projet Annuel *Christophe Rosenberger & Emmanuel Giguet*

Réalisé durant l'année de M2 Informatique 2021/2022

Introduction

Étant donné les délais serrés et la réévaluation des objectifs à la baisse, nous avons réalisé un changement assez radical de direction pour assurer un minimum de positivité dans les résultats de notre travail. Au lieu de passer par l'étape d'identification individuelle des touches frappées au clavier pour par la suite identifier la dynamique de frappe, nous avons décidé de réaliser une analyse acoustique de l'enregistrement audio et d'essayer de reconnaître un individu via les caractéristiques du son qu'il produit en écrivant au clavier.

Les différents types de claviers

Même si cela peut sembler évident, il est bon de rappeler tous les aspects que nous aborderons lors de ce rapport. Lorsque l'on utilise le clavier d'un ordinateur, le son produit par la frappe de chaque touche est différent. Nous pouvons prendre l'exemple des touches "espace" ou "entrée" qui sont plus grosses que les autres mais la différence est en réalité bien plus fine. Chaque touche individuelle produit un son qui lui est propre, même si la subtilité est difficilement appréciable par notre audition. Même si le sujet peut assez simplement s'étendre aux claviers tactiles, nous nous concentrerons ici aux claviers physiques.

Un point important à définir est que tous les claviers n'émettent pas le même son. Même si globalement, deux claviers identiques sont relativement proches, une multitude de technologies de claviers existe. Nous pouvons définir trois familles principales :

- Les claviers **membranes** sont les claviers "classiques". Ce sont les claviers les plus répandus et ceux auxquels nous sommes habitués. Ils sont très populaires car très simples dans leur conception ce qui les rend abordables. Leur nom provient d'une membrane en caoutchouc qui vient s'appuyer sur un circuit imprimé. Lorsqu'une touche est appuyée, cette partie élastique se compresse et permet de réaliser un contact électrique, indiquant la pression. C'est cette spécificité qui leur apporte l'un de leurs plus gros avantages mais également leur inconvénient. Ils sont silencieux donc idéaux dans des bureaux partagés par plusieurs personnes mais cette simple membrane en caoutchouc les rend parfois peu agréables à utiliser.

- Les claviers **mécaniques** sont eux bien plus complexes dans leur conception mais permettent une précision et une rapidité largement supérieurs aux claviers à membrane. Leur conception se base sur un "switch" présent sous chaque touche qui réalise un mouvement d'activation bien plus ample car contenu dans un mécanisme semblable à un interrupteur, ce qui résulte en une frappe bien plus tactile, agréable et bien souvent, bruyante. Ces derniers ne sont pas nouveaux, au contraire, mais leur popularité explose ces dernières années. Cette technologie offre une frappe largement plus agréable, réactive mais également identifiable par le son d'activation des "switch". Cette technologie reste moins populaire car la conception plus complexe les rend largement plus solides mais coûteux.

- Les claviers **portables** peuvent être de l'une ou l'autre des familles précédentes, voire des deux. En effet, ce sont des claviers extrêmement populaires car ils sont fournis avec les ordinateurs portables. Leur contraintes liées à la finesse nécessaire pour tenir à l'intérieur des machines ont mené à de nouvelles approches hybrides des différentes technologies dont nous venons de parler. Dans l'ensemble, nous pouvons noter que ces claviers sont très souvent plus silencieux que les autres étant donné leur épaisseur.

Pour en revenir au sujet, nous pouvons alors expliquer que nous mènerons nos expérimentations sur ces trois technologies afin de pouvoir comparer les profils. Les claviers mécaniques, très bruyants, sont alors plus adéquats à notre sujet mais nous verrons si la précision peut être suffisante avec des claviers plus silencieux.

1 La biométrie acoustique

1.1 Identification de la personne via la manière dont elle frappe au clavier

La biométrie représente la mesure biologique ou les caractéristiques physiques qui peuvent être utilisées pour identifier les individus. Il existe des centaines de formes de technologies biométriques, comme la reconnaissance faciale, l'identification des empreintes digitales, ou encore l'identification des empreintes rétiniennes. On distingue deux catégories de technologies biométriques : les mesures physiologiques et les mesures comportementales. Dans notre cas, nous allons nous intéresser à cette dernière catégorie, et plus précisément à la dynamique de frappe au clavier.

La dynamique de frappe au clavier permet de vérifier l'identité d'un individu en analysant le rythme auquel il tape, en analysant le pression exercée sur les touches ou la vitesse de frappe. Chaque individu possède une méthode unique pour taper au clavier. Nous analyserons cette donnée biométrique en utilisant le son.

1.2 Création d'une base de données biométrique via un enrôlement de volontaires

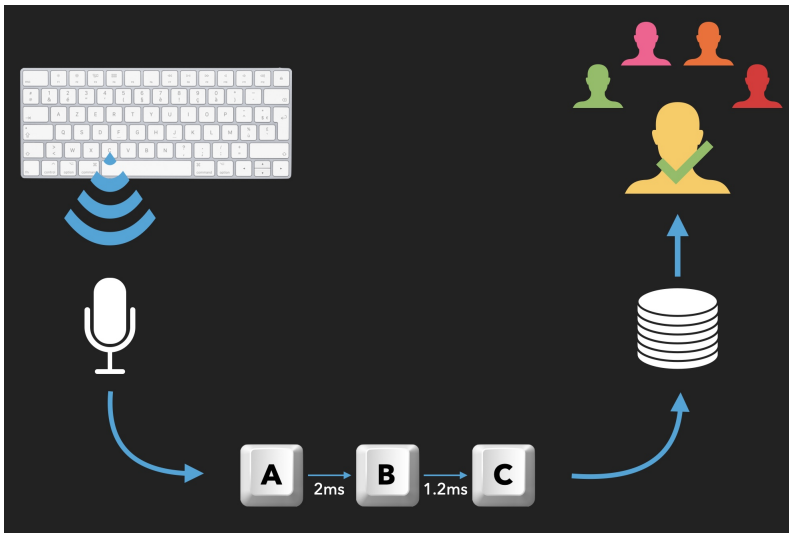
Afin de faire des analyses et une étude sur ce sujet, nous avons besoin de données biométriques afin de les comparer. Pour cela, nous allons devoir recueillir ces données grâce à un enrôlement de personnes volontaires. Le processus d'enrôlement se déroule en deux parties.

Dans un premier temps, le volontaire répond à un questionnaire (5.2) contenant des questions avec des réponses neutres, toujours en nombre pair pour avoir des réponses non neutres et dégager une tendance. Si le cas contraire n'est pas spécifié, chaque question à choix multiple ne doit contenir qu'une seule réponse.

Dans un seconds temps, nous passons à la phase d'enregistrement. Nous recueillons donc l'enregistrement du son produit lorsque le volontaire tape au clavier. Les conditions dépendent des attentes et données cherchant à être extraites. Dans tous les cas, des mesures redondantes sont une bonne pratique pour pouvoir revenir sur l'enrôlement à posteriori si les résultats présentent des anomalies. Pour ce projet, nous pensions qu'un enregistrement vidéo serait intéressant.

1.3 Reconnaître un individu dans la base

Afin de reconnaître un individu qui est enrôlé dans notre base de données, l'idée est de suivre le chemin suivant :



Grâce à ce schéma, nous sommes en capacité d'associer un individu à un sujet présent dans notre base de données.

1.4 Créer un protocole d'authentification

Maintenant que l'on sait comment identifier une personne et une fois la base de données biométrique remplie, nous avons vu précédemment comment reconnaître un individu dans cette base. Il y a deux approches possibles pour l'exploitation de ces données.

La première serait de faire un outil biométrique permettant de reconnaître un individu en utilisant non pas une connaissance mais un trait physique. La particularité de ce trait physique est qu'il est très difficilement reconnaissable par l'être humain tandis que c'est une formalité pour un ordinateur.

Pour la seconde approche, nous avons pensé à créer un protocole d'authentification. Imaginons une entreprise contenant 20 individus et une base de données contenant les enregistrements de la dynamique de frappe de ces derniers. Afin que chacun puisse s'identifier à son poste de travail, l'utilisateur tape une phrase, et le système associera l'utilisateur à son poste de travail.

2 Reconnaissance de la frappe via le son

Afin de commencer avec le plus de cartes en main, nous avons réalisé un état de l'art en début de projet. Ce dernier avait comme but de nous documenter sur les différentes stratégies qui existent aujourd'hui pour reconnaître un individu, uniquement via le son produit lorsqu'il frappe au clavier. Différentes études et projets (voir références) nous ont permis d'envisager plusieurs approches. Dans tous les cas, l'objectif était de comprendre ce qui est écrit avant d'analyser la manière dont cela a été écrit.

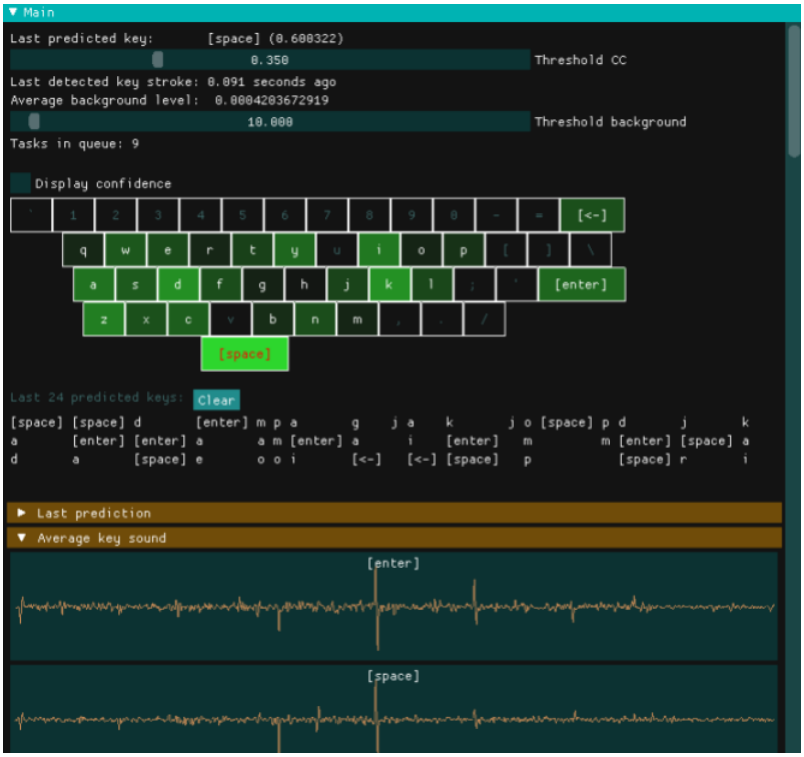
2.1 Trianguler le son via plusieurs micros

Une étude ([1]) a montré qu'il était possible de trianguler la position de la source du son émis par le clavier, soit la touche enfoncée, en utilisant différents microphones placés aux alentours de celui-ci. Nous pouvons alors imaginer des applications d'espionnage, de surveillance ou d'écoute clandestine liée à cette idée mais la complexité de son déploiement, le manque de temps que nous avions et le peu d'informations techniques pour aboutir à de tels résultats nous ont alors mené vers la deuxième approche.

2.2 Créer un modèle d'apprentissage automatique (keytap)

Celle-ci n'utilise qu'un seul microphone et écoute simplement le son depuis un seul endroit. Toute identification réside alors dans l'analyse. En isolant le son de chaque touche, il est alors possible de réaliser un modèle d'apprentissage automatique et ainsi, d'identifier chaque son via de "l'intelligence artificielle". Cette deuxième idée est bien plus documentée et surtout, implémentée dans différents projets [9].

Keytap2 est dérivé du projet original Keytap qui est un projet que nous avons trouvé sur github. Ce dernier se base sur les caractéristiques linguistiques de l'anglais et permet de reconnaître des phrases écrites avec un assez bon degré de précision mais n'est pas adapté pour nous. En effet, l'idée est de comprendre avec précision les touches pressées mais un mot de passe, par exemple, ne rentre pas dans ce cadre alors que c'est l'une des possibilités d'élargissement de nos objectifs. De plus, il est nécessaire de fournir une phrase relativement longue pour que la détection soit correcte. Là encore, notre idée est de faire frapper l'utilisateur une phrase de passe relativement courte dans le cadre de l'authentification.



Nous avons alors décidé de nous concentrer sur le projet Keytap qui semblait prometteur car précis et simple d'utilisation. Ce dernier isole le son produit par chaque touche du clavier, réalise un modèle d'apprentissage et reconnaît individuellement chaque touche pressée.

Nous avons alors réalisé de nombreux essais avec Keytap. Nous avons enregistré individuellement chaque touche du clavier un nombre de fois différent. D'abord 5, puis 20 et nous avons alors essayé de monter à 100 enregistrements de chaque touche, soit 100*26+3*100 (pour les touches espace, entrée et retour arrière). Nous avons également essayé différentes configurations d'enregistrement. Nous avons changé de système d'exploitation (MacOS 12.1, Windows 10, Windows 11 et Ubuntu 21), le micro (intégré à l'ordinateur, externe voir un microphone studio) et enfin le clavier (mécanique, membrane ainsi qu'intégré à différents ordinateurs portables). Malgré toutes ces configurations, les résultats de Keytap n'ont pas été au niveau espéré. Dans le meilleur des cas, nous avons réussi à faire comprendre le programme de chaque frappe réalisée mais la déduction faite était dans les alentours des 30% de précision. Même si les touches espace et entrée étaient relativement isolées de par leur sonorité, le reste des touches était très souvent mélangé. Les résultats de tous nos tests nous ont menés à la conclusion que Keytap n'était pas adapté pour notre projet. Le problème reste non identifié, nous avons maximisé les conditions de tests pour isoler chaque variable pour savoir d'où pouvait venir l'erreur, sans succès.

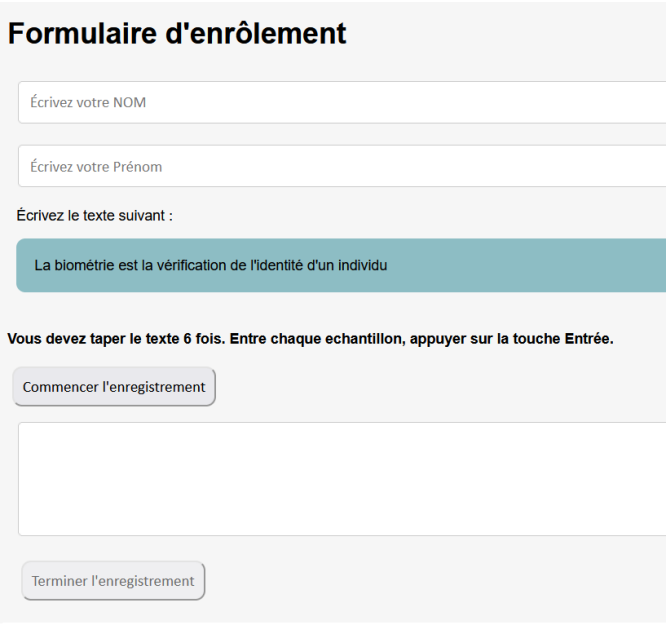
Malheureusement, nous avons choisi d'abandonner le projet Keytap sur lequel nous avons consacré beaucoup de temps. Nous misions beaucoup sur cet outil ce qui a fait que le temps restant après ce changement de direction a vu nos objectifs grandement revus à la baisse. Nous avons tout de même pu nous inspirer de l'approche faite par Keytap pour nous intéresser à une nouvelle approche plus archaïque utilisant la librairie d'étude acoustique python pyAudioAnalysis.

3 Changement de direction

Etant donné les délais serrés et la réévaluation des objectifs à la baisse, nous avons réalisé un changement assez radical de direction pour assurer un minimum de positivité dans les résultats de notre travail. Au lieu de passer par l'étape d'identification individuelle des touches frappées au clavier pour par la suite identifier la dynamique de frappe, nous avons décidé de réaliser une analyse acoustique de l'enregistrement audio et d'essayer de reconnaître un individu via les caractéristiques du son qu'il produit en écrivant au clavier.
En utilisant la librairie pyAudioAnalysis [7] et en s'inspirant des exemples basiques d'implémentation présents sur un article d'introduction au sujet [1]. La stratégie proposée vise à catégoriser des enregistrements musicaux en 2 genres musicaux. En reprenant l'approche et les grandes lignes de cette idée, nous avons un nouvel objectif réalisable et potentiellement intéressant.

3.1 Enregistrement du son produit lors de la frappe

Avec cette nouvelle approche, il nous fallait uniquement l'enregistrement du son produit lorsqu'un texte est écrit mais nous n'avons pas spécialement d'intérêt à garder la trace de chacune des touches qui ont été appuyées lors de l'enregistrement contrairement à l'approche précédente. En effet, nous analysons ici les caractéristiques acoustiques propres à chacun lorsque plusieurs personnes enregistrent la même phrase plusieurs fois mais les fautes de frappe ou petites variations du texte écrit impactent peu.



Afin de simplifier cette capture, nous avons mis au point un outil web, utilisable sur le navigateur Mozilla Firefox, afin d'enrôler chaque individu. L'interface se résume simplement à l'acquisition du nom et du prénom de l'individu, et de l'enregistrement d'une phrase tapée au clavier : "La biométrie est la vérification de l'identité d'un individu". Nous avons choisi cette phrase car celle-ci propose des combinaisons de lettres variées, elle est relativement longue pour notre utilisation mais devait être assez courte car nous demandons au sujet d'enregistrer 6 fois cette phrase.

D'un point de vue technique, une fois que l'individu a inscrit son nom et son prénom, celui-ci doit cliquer sur le bouton " Commencer l'enregistrement ", ce qui déclenche un premier signal sonore. Ensuite, entre chaque phrase tapée, l'utilisateur doit appuyer sur la touche "Entrée" afin de déclencher un nouveau signal sonore. Une fois l'enregistrement terminé, un dernier signal sonore est émis, cela signe la fin de l'enrôlement de l'individu. De notre côté, nous récupérons le fichier *NOM_PRENOM_record.ogg*. Une fois tous nos sujets enregistrés, il nous faut donc ouvrir les fichiers avec le logiciel Audacity, puis découper chaque fragment d'enregistrement correspondant à la phrase écrite. Nous obtenons donc six enregistrements par personne, ce qui nous permet de calculer des paramètres pour chaque enregistrement.

3.2 Calcul des paramètres propres aux enregistrements

Une fois chaque enregistrement audio effectué, nous pouvons nous intéresser à l'analyse. Grâce à l'exemple de prise en main de pyAudioAnalysis cité plus tôt, nous avons pu suivre un chemin similaire et reprendre les grandes idées du code proposé. L'analyse acoustique proposée par cette librairie consiste à diviser l'audio en images. Ces images sont des segments qui sont paramétrables. Chaque segment est encore une fois divisé en plus petites parties sur lesquelles les mesures vont être réalisées. La division en plusieurs échantillons se nomme le windowing ou framing. Ces différentes fenêtres se nomment les short-term windows. Elles font dans notre cas 50ms mais cette durée peut varier selon l'utilisation attendue. La longueur des segments est d'une seconde. Là encore, ces valeurs peuvent être modifiées selon l'usage mais ce compromis est relativement adapté à notre cas. En effet, il y a environ 20 fenêtres par seconde. Si on considère une moyenne de vitesse de frappe de 40 à 70 mots par minute et des mots allant de 5 à 10 lettres en moyenne, cela donne un nombre de lettres par seconde allant de 3 à 11. En choisissant 20 échantillons par seconde, nous avons alors un découpage qui a de grandes chances d'isoler plus ou moins correctement chaque frappe individuelle.

Chaque segment subit alors une analyse permettant d'identifier de nombreux paramètres audio. Ces derniers vont de l'entropie à la vélocité en passant par le rythme. Le programme calcule alors des moyennes sur chacun des fragments et permet de mesurer un vecteur final de 138 paramètres audio propres à chaque enregistrement. Ces empreintes uniques peuvent alors être comparées les unes aux autres et permettent de constituer un modèle si l'on regroupe ensemble différentes analyses de différents enregistrements.

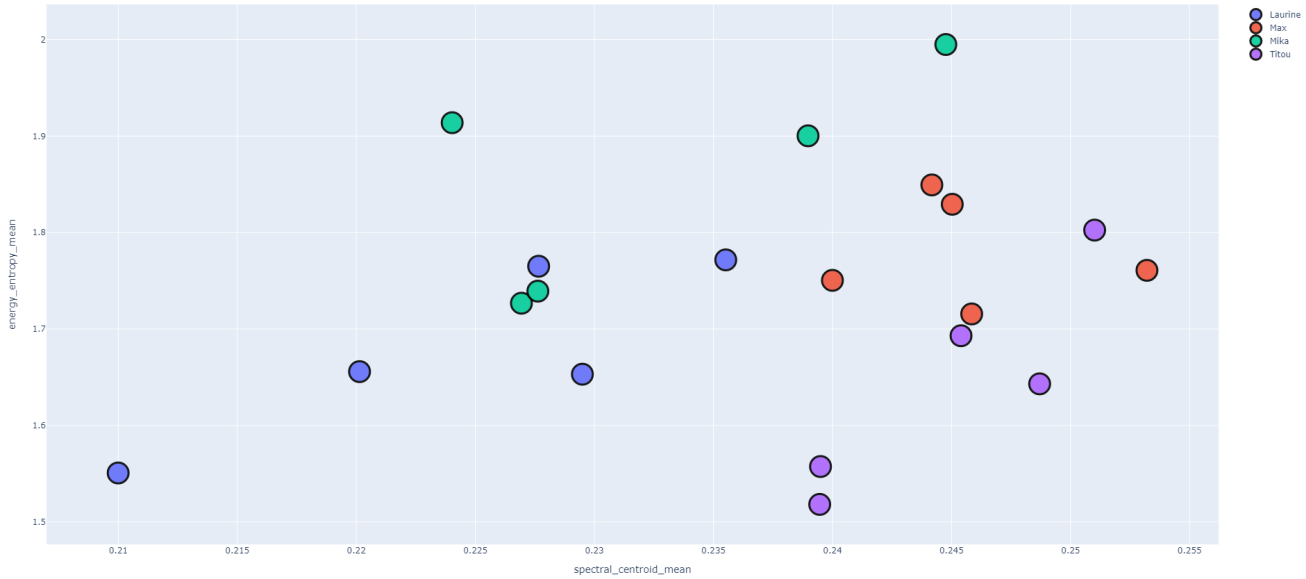


FIGURE 1 – Nuage de points représentants nos classes

3.3 Création des modèles SVM-RBF

PyAudioAnalysis permet par défaut de réaliser des modèles de classification et de régression. Il est possible d'utiliser les KNN, des forêts d'arbres décisionnels mais ce qui nous intéresse majoritairement est la possibilité des machines à vecteurs de support (SVM) et particulièrement des SVM RBF. En reprenant l'exemple de la classification par genre musical, nous avons très simplement pu remplacer le métal et le classique par deux individus. De cette manière, les caractéristiques propres aux individus A et B sont calculées et nous obtenons un modèle SVM-RBF entre A et B. L'idée est de savoir via ce modèle si un enregistrement audio "c" appartient à A ou B. La manière dont ce calcul est réalisé correspond à une distance ou un vecteur entre A et c et B et c. Nous pouvons alors considérer le résultat obtenu comme un score d'assurance < 1.

Nous avons alors organisé les enregistrements d'apprentissage de tous les volontaires enrôlés de notre base d'individus à identifier dans un dossier portant leur nom. Le nom de ce dossier correspond au nom du modèle qui en découle. La limite de l'utilisation de SVM-RBF est qu'il n'est possible de comparer que deux modèles à la fois, du moins par défaut. En effet, plusieurs stratégies existent pour générer des modèles de classifications de plus que 2 classes mais nous n'avons pas eu le temps de s'y pencher.

Nous avons alors fait au plus simple, à savoir faire des couples de 2 individus de notre base afin de générer un modèle par binôme. Ainsi, dans le cas où A, B et C sont enrôlés dans notre base, nous réalisons les modèles entre A et B, A et C ainsi que B et C.

Dans notre cas, tous les enregistrements de chaque dossier sont analysés et un modèle est alors généré pour chaque couple formable. Cette approche est simple dans son fonctionnement mais demande un temps de pré-calcul relativement élevé. Cela veut dire que nous réalisons (2 parmi n) modèles distincts où n est le nombre d'individus de notre base. Nous obtenons alors 10 modèles distincts pour 5 individus, 15 pour 6 ou bien 45 pour 10 personnes. Cette approche permet d'obtenir des résultats intéressants mais demande un temps non-négligeable de pré-calcul. Heureusement, une fois les modèles générés, il n'y a plus besoin d'y retoucher et leur exploitation est très peu coûteuse en temps.

3.4 Stratégie d'identification des modèles

Après avoir expliqué notre génération des modèles, nous allons voir comment nous identifions la correspondance d'un fichier à une des classes issues de nos modèles. L'approche initiale permet de savoir dans le cas où un modèle A-B existe si c correspond plutôt à la classe A ou B, nous avons ici simplement multiplié le nombre de couples. Il suffit alors de calculer le score d'un fichier d entre A et B, A et C puis entre B et C et de regarder le résultat moyen de ces tests. La classe ayant le score le plus élevé sera alors celle qui sera la plus proche de notre fichier d. Nous pouvons de plus noter que de manière assez logique, la somme de ces scores est de $n/2$ où n est le nombre de classes.

Ainsi, nous pouvons pour un fichier donné, calculer son score par rapport à tous les modèles.
Dans notre approche, nous avons demandé aux volontaires d'enregistrer 6 fois la même phrase plus une à deux supplémentaires sur lesquelles nous reviendrons plus tard. En réalité, les 5 premiers enregistrements permettent de constituer la classe correspondant à chaque individu. Nous obtenons, pour chaque personne, une matrice correspondant aux statistiques liés aux paramètres propres à chaque classe. Le sixième enregistrement nous permet de prendre un audio exclu de cette classe, donc des modèles et ainsi de voir si notre approche permet de le rattacher à la personne correspondante.

4 Des résultats prometteurs

Nous pouvons à présent rentrer dans le vif du sujet. Après avoir défini la manière dont nous réalisons l'identification, intéressons-nous aux résultats.

Même si l'approche a finalement grandement dérivé de l'approche initiale, nous pensons être tout de même parvenus à des résultats intéressants. Nous pouvons par avance exprimer une légère prudence sur l'exactitude des chiffres que nous allons exprimer car ils sont relatifs à nos conditions d'enregistrement de d'expérimentation et notre panel d'individu n'était pas suffisant pour tirer de réelles conclusions sur l'efficacité.
En ayant 4 personnes dans la base, nous pouvons alors estimer que notre programme permet d'identifier chaque individu sans erreur avec un score d'assurance moyen de 0.85 avec des variations allant de 0.78 à 0.89. Cela signifie que le 6ème enregistrement correspond systématiquement à la classe à laquelle il est rattaché. Cela semble relativement logique mais il est bon de détailler chacune des parties de nos expérimentations. Ce score veut donc dire qu'il est possible d'identifier les caractéristiques propres à chaque enregistrement et qu'il est également simple de définir qui en est à l'origine.

La réserve que l'on peut à présent avoir sur ces résultats est qu'ils proviennent d'une simple classification d'un son parmi différents sons pratiquement identiques étant donné que le contenu écrit était le même. Nous pouvons penser que la longueur du son, donc la vitesse de frappe ou encore la force avec laquelle chaque personne appuie sur les touches du clavier peuvent être à l'origine de cette identification. De simples variations d'un même enregistrement suffisent à obtenir de tels résultats.

Nous avons alors, pour clarifier si oui ou non un individu est identifiable avec cette stratégie, utiliser des enregistrements supplémentaires réalisés lors de l'enrôlement qui étaient un texte différent de l'original. Les personnes ont ainsi écrit "Ceci est une autre phrase". Le seul but de ce deuxième texte et qu'il varie du premier sur la longueur, le lexique et les enchaînements de lettre utilisés. En réalisant à présent la même identification que précédemment, nous pouvons estimer que la personne semble bel et bien être à l'origine des variations propres à chaque enregistrement. Si nous reprenons les 4 individus précédents, seul un enregistrement n'est pas correctement attribué sur plus d'une quinzaine d'enregistrements comprenant du texte varié. Les scores d'assurance sont par contre bien plus bas que précédemment même si la précision semble au rendez-vous. Nous obtenons ici, pour le texte libre, des scores d'assurance dans les alentours de 0.65.

Nous pouvons en déduire que les résultats semblent à priori prometteurs mais nous ne pouvons en tirer de conclusion définitive. Il faudrait pour confirmer notre théorie réaliser plus de tests. Nous pouvons imaginer un enregistrement de plus d'individus, de leur demander d'autres exemples de textes libres à identifier mais également de faire varier différents facteurs de nos tests afin d'isoler chaque variable pour confirmer que celle qui est la plus importante est la personne derrière le clavier.

5 Mise au point des différents protocoles

La fin de ce rapport relate de l'application de notre stratégie de deux manières différentes. En effet, même si nous utilisons les mêmes données, l'exploitation varie pour revenir sur nos objectifs initiaux.

5.1 Identification de l'individu présent dans la base

La première approche est tournée vers l'identification d'un individu dans la base. Pour ce faire, nous prenons simplement un fichier à identifier, calculons les paramètres pour construire la matrice qui lui est propre avant de le confronter à tous les modèles de notre base. L'idée est assez simple, chaque fichier est ajouté dans un dictionnaire et chaque comparaison avec nos modèles donnera un score à deux individus de notre base, soit les deux à la base du modèle comparé. Cette comparaison est faite entre les fichiers à identifier et tous les individus. Nous calculons une fois toutes les comparaisons faites la moyenne des scores obtenus et nous en déduisons que la personne identifiée comme correspondant au fichier à identifier est donc celle qui a la moyenne la plus élevée.

5.2 Authentifier un individu via le son qu'il produit lors de la frappe au clavier

La deuxième approche est relativement proche mais n'exploite pas l'identité des modèles pour la comparaison. L'objectif de cette partie est l'authentification d'un individu afin de vérifier s'il est bien la personne qu'il prétend être. L'utilisation d'un tel mécanisme pourrait être envisagée pour autoriser l'accès d'un système à une personne non pas en lui demandant un secret comme la majorité des système d'authentification actuels à mots de passe mais pas non plus avec une méthode biométrique forte, mais en utilisant une biométrie comportementale.
Même si notre implémentation d'un tel mécanisme n'est pas complète et utilisable étant donné le manque de temps de la fin de ce projet, nous avons tout de même pu expérimenter sur la possibilité d'utiliser notre approche pour réaliser ce genre d'application. Notre programme demande alors à l'individu cherchant à s'authentifier son nom, qui correspond ici à un authentifiant et lui demande, normalement, de taper une phrase au clavier pour enregistrer le son qu'il produit en le faisant. Dans notre cas, l'implémentation de cette étape n'a pas pu être réalisée non plus, nous cherchons alors un enregistrement correspondant dans les fichiers à identifier. La comparaison est par la suite faite entre la personne cherchant à s'identifier et tous les individus ayant un modèle commun avec ce dernier. Le dictionnaire est alors différent de l'approche précédente car seule la personne à identifier aura plusieurs scores et verra un calcul de moyenne réalisé. Les autres individus de la base n'auront qu'une seule comparaison avec la personne et un seul score suffit à savoir si elle correspond ou non.
Si le programme identifie correctement la personne derrière l'enregistrement comme étant la même que celle qui cherche à s'authentifier, alors l'accès est accordé.

Conclusion

Pour conclure, nous pouvons dire que malgré le manque de temps et le changement de direction en fin de projet, nous sommes parvenus à réaliser une bonne partie de nos objectifs initiaux.
En effet, même si la décomposition de l'approche visant à utiliser une stratégie d'analyse de la dynamique de frappe extraite via le son a été modifiée en une analyse du son lui-même, nous sommes parvenus à extraire des paramètres propres à chaque individu et nous avons réussi à réaliser à la fois une identification d'individus enrôlés dans une base et une esquisse d'authentification.

Même si nous ne sommes pas en mesure de présenter des protocoles déployables en l'état, nous avons déjà réalisé une bonne partie des expérimentations, de la recherche et documentation qui permet de mener à de meilleurs résultats.

Les résultats que nous avons sont très prometteurs et s'inscrivent dans une démarche de recherche qui n'en est qu'au début. Nous pensons que notre travail peut être réutilisé et amélioré pour parvenir à l'élaboration d'un outil d'authentification et de reconnaissance efficace et globalement utilisable dans de nombreuses situations.

Références

[1] Jia-Xuan Bai, Bin Liun, Luchuan Song "I Know Your Keyboard Input: A Robust Keystroke Eavesdropper Based-on Acoustic Signals" *MM '21 : Proceedings of the 29th ACM International Conference on Multimedia*,October 2021.

[2] Joseph Roth, Xiaoming Liu, Arun Ross, *Biometric Authentication via Keystroke Sound*,Department of Computer Science and Engineering , Michigan State University, East Lansing MI 48824

[3] S Syed Idrus, Estelle Cherrier, Christophe Rosenberger, *Fusion et biométrie douce pour la dynamique de frappe au clavier*,GREYC - Groupe de Recherche en Informatique, Image et Instrumentation de Caen,2016

[4] Pin Shen Teh , Andrew Beng Jin Teoh , and Shigang Yue1, *A Survey of Keystroke Dynamics Biometrics*, 2013

[5] Applied machine learning, *User Verification based on Keystroke Dynamics: Python code*, 2017

Références GitHub :

[6] Theodoros Giannakopoulos, *Intro to Audio Analysis: Recognizing Sounds Using Machine Learning*, September 12th 2020

[7] Theodoros Giannakopoulos, *pyAudioAnalysis*, 2015

[8] Goncalopp, *keystroke dynamics*, 29 January 2016

[9] Georgi Gerganov, Ggicci, Nicholas Wheeler, Nico Sonack, *Keytap*, December 2021

[10] Georgi Gerganov, Ggicci, Nicholas Wheeler, Nico Sonack, *Keytap2 - acoustic keyboard eavesdropping based on language n-gram frequencies*, December 2020

Annexes

Résultats du terminal pour le script biometry.py.
Les fichiers **NOM6.wav** sont ici les 6èmes enregistrements réalisés lors de l'enrôlement. Le texte écrit par chaque personne est donc le même que celui constituant le modèle. Les fichiers **NOM2_x.wav** ont eux aussi été enregistrés au même moment mais le texte écrit était différent. Il était libre mais généralement bien plus court.

```
Max2_2.wav
Laurine : 0.324
Max : 0.615
Mika : 0.57
Titou : 0.49
Max2_2.wav correspond to Max with an assurance of 0.615

Max6.wav
Laurine : 0.258
Max : 0.892
Mika : 0.454
Titou : 0.395
Max6.wav correspond to Max with an assurance of 0.892

Mika2_2.wav
Laurine : 0.361
Max : 0.434
Mika : 0.65
Titou : 0.554
Mika2_2.wav correspond to Mika with an assurance of 0.65

Mika6.wav
Laurine : 0.337
Max : 0.345
Mika : 0.886
Titou : 0.432
Mika6.wav correspond to Mika with an assurance of 0.886

Titou2_2.wav
Laurine : 0.364
Max : 0.408
Mika : 0.599
Titou : 0.629
Titou2_2.wav correspond to Titou with an assurance of 0.629

Titou2_3.wav
Laurine : 0.361
Max : 0.378
Mika : 0.545
Titou : 0.716
Titou2_3.wav correspond to Titou with an assurance of 0.716

Titou6.wav
Laurine : 0.49
Max : 0.249
Mika : 0.442
Titou : 0.819
Titou6.wav correspond to Titou with an assurance of 0.819
```

Questionnaire de pré-enrollement :

Nom	
Prénom	
Âge	
Fréquence d'utilisation d'un clavier	☐ Tous les jours ☐ 5 à 6 jours par semaine ☐ 3 à 4 jours par semaine ☐ 1 à 2 jours par semaine ☐ Une fois par semaine ☐ Quelques fois par mois maximum ☐ Ne souhaite pas répondre
Temps d'utilisation d'un clavier par jour	☐ Plusieurs heures ☐ À peine quelques heures ☐ Une heure ☐ Quelques minutes ☐ Ne souhaite pas répondre

Les habitudes d'utilisation d'un clavier (utilisation d'un ordinateur)
Plusieurs cases peuvent être cochées
	☐ Travail
	  ☐ Bureautique
	  ☐ Programmation/Développement
	  ☐ Milieu Artistique
	☐ Écriture
	  ☐ Livres
	  ☐ Bande dessinées
	☐ Internet
	☐ Divertissement (spécifier si envie)
	☐ Jeux Vidéo
	☐ Mails
	  ☐ Persos
	  ☐ Professionnels
	☐