



# RAPPORT de stage

Master « Sciences et Numérique pour la Santé »

Délivré par l'Université de Montpellier

Préparé au sein de l'INRIA CAMIN

Lieu : **Bâtiment 5 du campus St-Priest, Montpellier**

Spécialité : **Ingénierie des Dispositifs pour la Santé**

Présenté par **BLANQUINQUE JÉRÉMY**

## TITRE DES PROJETS

- **Limiteur de volume pour casque audio**
- **Dispositifs d'aide aux personnes malvoyantes**

Durée du projet : 01/02 au 31/07

Encadrants de projet :

- AZEVEDO COSTE Christine
- PISSARD-GIBOLLET Roger
- SIJOBERT Benoit

# Table des matières

<b>I.</b>	<b>Remerciements</b>	3
<b>II.</b>	<b>Introduction</b>	4
	1. Contexte du stage	4
	2. Objectifs et cibles du stage	4
<b>III.</b>	<b>Projet limiteur de volume</b>	5
	1. Objectifs et cibles du projet	5
	2. Contexte physique de l'étude	6
	3. État de l'art des solutions existantes	8
	4. Simulations et montage	11
	5. Réalisation de la solution retenue	17
<b>IV.</b>	<b>Projet ReadForMe</b>	22
	1. Description du système et de ses faiblesses	22
	2. État de l'art	24
	3. Librairie développée et Framework de tests	25
	4. Résultats	30
<b>V.</b>	<b>Conclusion</b>	33
<b>VI.</b>	<b>Bibliographie</b>	34
<b>VII.</b>	<b>Annexe</b>	35

## I. Remerciements

Je tiens à remercier toutes les personnes qui ont contribué de près ou de loin à ce stage et qui m'ont aidé lors de la rédaction de ce rapport. Ainsi, j'adresse tout d'abord mes remerciements à Azevedo Coste Christine et Pissard-Gibollet Roger de m'avoir fait confiance tout au long de ce stage et pour leur encadrement, puis des stagiaires/doctorants/chercheurs de l'équipe Inria CAMIN de Montpellier et de l'Inria de Grenoble pour m'avoir soutenu et apporté de l'aide quand j'en avais besoin. Particulièrement, j'aimerais remercier William Lucie et Schlosser-Perrin Felix, deux doctorants de l'équipe CAMIN, de m'avoir accueillie chaleureusement dans leur bureau de travail, ce qui m'a permis de me sentir à l'aise très rapidement. Évidemment merci à Sijobert Benoît et au Humanlab Saint-Pierre de m'avoir fait confiance ainsi de m'avoir proposé et permis de travailler sur ces projets lors de ce stage. J'aimerais aussi personnellement remercier la famille de l'adolescent atteint d'autisme qui fait tout pour que leur fils puisse être heureux en lui permettant de faire ce que j'aime beaucoup aussi : écouter de la musique. Pour moi, ça aura été un honneur et un plaisir si j'ai pu l'aider d'une quelconque façon.

Tout ce stage et plus globalement toutes ces années d'études ne seraient rien sans le soutien inconditionnel de mes différents professeurs ainsi que de mon entourage, mes amis, mon colocataire, ma famille, mes parents, mes frères. Même s'ils n'ont pas toujours compris ce que j'étudiais et dans quelle voie je pouvais m'orienter, ils m'ont toujours soutenu et poussé en avant pour avant tout faire ce que j'aime. J'ai toujours pu senti leur présence, même dans des moments difficiles et de doutes, mais si je n'ai pas lâché, c'est grâce à eux, alors merci d'être là pour moi.

## **II. Introduction**

### **1. Contexte du stage**

L'Inria est un institut de recherche en informatique et mathématiques. L'équipe Inria CAMIN [1] mène une action exploratoire, nommée Humanlab inria (HLI) [2], qui a pour ambition de créer un modèle de collaboration entre l'INRIA et les Humanlabs. Les Humanlabs [3] sont des espaces collaboratifs de fabrication numérique ou de réparation d'objets, ouverts à des personnes présentant un handicap pour leur permettre de s'approprier la technologie pour leur usage propre. En effet, les Humanlabs présentent alors des projets concrets en open-source que tout amateur peut réaliser et reproduire. Malgré cette prétendue simplicité, il est parfois utile voire nécessaire d'utiliser des technologies complexes, rendues elles aussi open-source avec le temps. Par exemple dans des dispositifs pour compenser des déficits sensoriels comme la vue ou l'ouïe, des algorithmes et des dispositifs de traitement de signaux sonores ou visuels sont très utiles. Mon travail a donc consisté à utiliser ces derniers dans deux cas très concrets, afin d'identifier et de mettre en œuvre les technologies adéquates pour répondre aux besoins.

Ce stage s'inscrit dans cette collaboration entre l'action exploratoire et le Humanlab Saint-Pierre situé à Palavas [4]. Il fait suite à un mois de projet que j'ai effectué lors du premier semestre du Master 2 [5]. Ce rapport comprendra donc des éléments de ce projet basé sur un limiteur de volume, appelé par la suite projet ChutChut. Par conséquent la première partie du rapport portera sur la réalisation concrète du dispositif élaboré lors du projet préliminaire et la seconde partie portera sur la découverte du projet ReadForMe, mis en lumière lors d'un Fabrikarium [6] dont je dois améliorer le logiciel de traitement visuel.

Pour le projet ChutChut dont j'avais fait l'état de l'art lors de mon projet précédent, j'ai implémenté et testé une solution analogique qui répondait aux besoins de robustesse et de simplicité. Pour le projet ReadForMe, après l'étude du dispositif, j'ai amélioré son fonctionnement et sa robustesse en le rendant moins sensible à l'éclairage et l'orientation de la feuille qui doit être lue. Pour cela, j'ai effectué un état de l'art des pré-traitements visuels dans la numérisation de textes et implémenté une solution finale pour le système.

### **2. Objectifs et cibles du stage**

Les objectifs du stage sont donc multiples. Le premier sera de réaliser entièrement un dispositif électronique de limitation de volume. En effet, lors du projet précédent, j'avais étudié et simulé plusieurs schémas électroniques suite à un état de l'art sur les technologies existantes. Le principal objectif de cette première partie de stage sera de prototyper ces montages sur plaquettes expérimentales afin de tester avec de réels composants si ces solutions sont envisageables, de

choisir laquelle répond à nos critères pour ensuite la compacter sous forme de carte de circuit imprimée PCB. La dernière étape sera de réaliser le boîtier en imprimerie 3D pour en faire un dispositif fonctionnel et robuste, simple d'utilisation et incassable.

Une fois cela fait, le deuxième objectif du stage est de, comme précisé plus haut, d'améliorer le traitement visuel du projet ReadForMe qui est actuellement en version V3, avec une V1 faite lors du Fabikarium 2020 puis une V2 lors de l'événement « Focus Vision ». Il s'agit d'un dispositif permettant, à l'aide d'un Raspberry et d'une caméra, de prendre en photo des documents écrits et de les retranscrire dans un premier temps en texte brut et dans un second en audio écoutable par l'utilisateur. Ce dispositif est ainsi conçu pour les personnes malvoyantes, leur permettant de lire des documents, carte de restaurant etc... Il utilise deux technologies, la première étant « l'océrisation » permettant de retranscrire du texte à partir d'une image et la deuxième est la synthèse vocale, ce qui permet de traduire du texte en audio via des haut-parleurs.

### **III. Projet limiteur de volume**

Ce projet est destiné à un adolescent atteint d'autisme. En effet, ce dernier ne se rend pas toujours compte du volume d'écoute de sa musique, d'où le besoin de limiter électroniquement le volume. L'adolescent a pour habitude d'écouter de la musique grâce à un casque audio filaire branché directement à une chaîne Hi-Fi ou un ordinateur. De plus, l'adolescent étant susceptible de crise de colère, il faut que le dispositif soit robuste mécaniquement.

Outre le fait de l'autisme de l'adolescent, écouter de la musique à fort volume est dangereux pour la perception auditive de tous. En effet, d'après un article des laboratoires Unisson [7], écouter de la musique trop fort est dangereux pour l'oreille interne de l'humain, surtout sur de longue durée. De graves symptômes et séquelles peuvent alors apparaître, comme des fatigues auditives (bourdonnements et acouphènes), de l'hyperacousie (le son devient gênant voire douloureux) et même une surdité irréversible. Les normes de santé définissent alors le seuil à 85dB maximum pour une écoute inférieure à 8 heures, en sachant que les téléphones modernes peuvent naturellement proposer des écoutes à plus de 100dB en affichant simplement un message dissuasif. Comme on peut difficilement mesurer le décibel sonore une fois sorti du casque, notre dispositif devra donc à minima réduire de 20 décibels électroniques le signal de sortie.

#### **1. Objectifs et cibles du projet**

Il existe une multitude de casques pour enfants avec ce qu'on appelle un « limiteur de volume », souvent basé à 85dB SPL (Sound Pressure Level, correspondant à la pression sonore maximale). Cependant, il est rare de trouver sur le marché des casques grand public avec un limiteur de volume

intégré aussi important, ils sont souvent aux alentours des 95-100 décibels, ce qui reste bien trop élevé pour le cadre de notre étude.

L'objectif du projet global est donc d'établir un dispositif électronique externe pour moduler directement le signal reçu par le casque à la sortie de l'ordinateur et/ou de la chaîne Hi-Fi. Lors de mon projet précédent, j'ai : fait un état de l'art des solutions grand public et professionnelles contenant des limiteurs de volumes, puis proposer des solutions plus ou moins génériques dans le domaine du Do It Yourself (DIY) et si possible vérifier l'adaptabilité des solutions pour par exemple une connexion filaire ou Bluetooth du dispositif audio. Pour cela, j'ai utilisé un logiciel de simulation électronique nommé LTSpice pour expérimenter et simuler le dispositif. Lors du stage, je prototyperai sur breadboard les différents schémas pour choisir le schéma le plus adapté. Et finalement, pour la partie prototypage et développement, nous utiliserons le logiciel KiCad permettant la transition simulation vers PCB puis permettant le routage des pistes électroniques.

Les contraintes seront alors, de par l'analyse de l'état de l'art, d'expérimenter et de prototyper un dispositif permettant de limiter le volume d'écoute de l'utilisateur à un seuil de 85 décibels maximum. Le dispositif doit être dans une gamme de prix raisonnable et pour rester dans l'optique et la démarche des HumanLabs, le dispositif doit être reproductible, simple d'utilisation et de conception (démarche du Do It Yourself).

## **2. Contexte physique de l'étude**

Cette partie figure déjà dans mon rapport originel, mais j'ai préféré la laisser pour garder une certaine cohérence avec la suite du projet, cependant certaines choses ont changé depuis. Que ce soit en électronique ou en musique, la mesure du son se fait grâce à l'unité du décibel. Il s'agit en effet d'une échelle logarithmique représentant le niveau d'intensité acoustique défini par la formule suivante :  $IdB = 10 \cdot \text{Log}(I/10^{-12})$ . Où  $I$  est l'intensité d'un son perçu sur une surface en Watts/mètre<sup>2</sup> et  $\text{Log}$  le logarithme décimal en base dix.

### **L'encodage audio, la transmission et la stéréo**

Lors de la transmission filaire de la musique vers notre dispositif, il sera important de savoir si elle est analogique ou numérique et si le traitement qu'on souhaite lui appliquer se fera dans le domaine analogique ou numérique pour une transmission filaire ou non filaire. De nos jours, la majorité des communications se font grâce à une connexion électrique coaxiale, plus communément appelée prise Jack. Cette dernière permet une communication analogique de notre signal, transportant à la fois la piste gauche et la piste droite de notre musique (figure 1). À cela peut s'ajouter une quatrième piste pour le transport de la piste issue d'un micro, à noter que l'ordre des pistes peuvent changer selon certains standards.

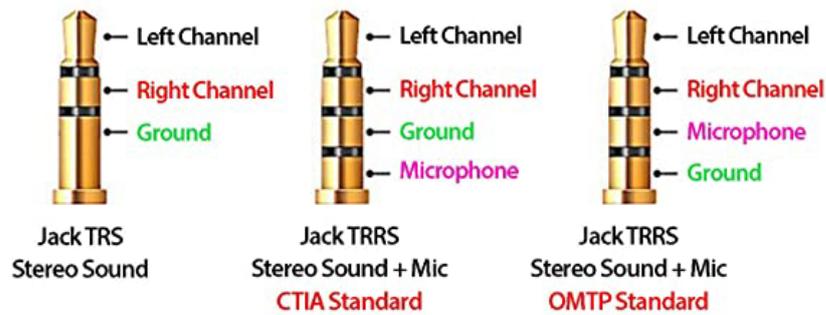


Figure 1 : Connectique prise Jack

Il existe aussi d'autres types de transmission filaire comme la transmission numérique via le protocole USB mais aussi la transmission par fibre optique permettant un débit bien plus élevé. Cependant nous utiliserons dans ce projet une communication filaire standard avec prise Jack.

Une grande majorité des dispositifs audio (casques, enceintes) est désormais disponible dans le commerce possédant une connexion et/ou une transmission sans fil. En effet, les technologies sans fil (Bluetooth, Wi-Fi) sont très utiles, moins encombrantes et moins fragiles que les fils (surtout pour la fibre optique). Ainsi, dans le cadre d'une future étude, nous pourrions nous pencher sur un dispositif pouvant à la fois fonctionner sans fil pour les casques utilisant cette technologie.

### Fonctionnement des compresseurs

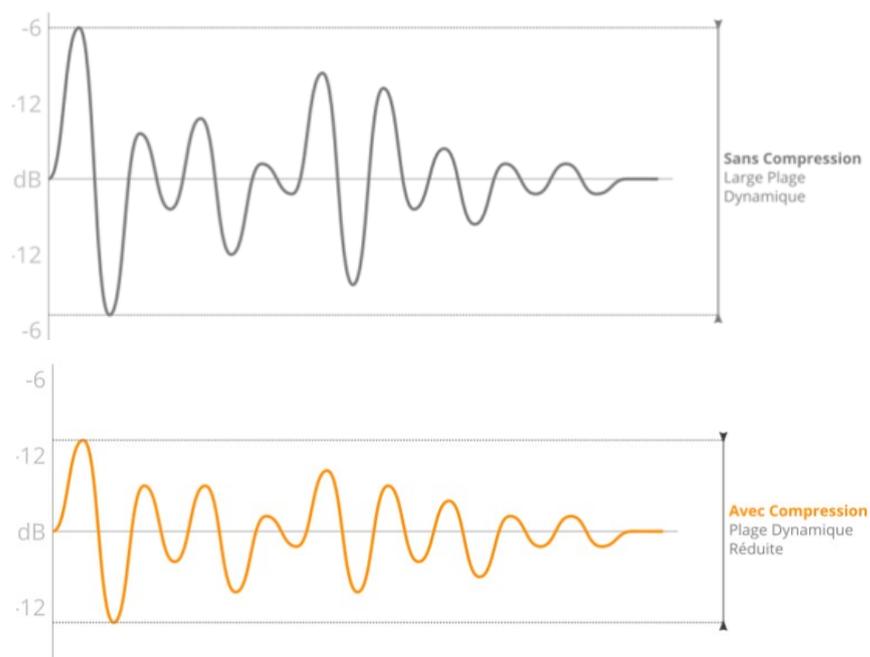


Figure 2 : Représentation graphique d'un compresseur

En musique et plus particulièrement en mixage, les professionnels utilisent déjà des compresseurs et des limiteurs [8] pour égaliser les différentes pistes des instruments et des voix pour obtenir un rendu agréable à l'oreille, c'est ce qu'on appelle le Mastering (figure 2). Ces compresseurs et limiteurs peuvent être soit simplement numériques dans des logiciels de productions de musique (FL Studio par exemple) ou alors pour du mixage analogique qui sont souvent inclus dans les amplificateurs ou préamplificateurs. Dans les deux cas, le fonctionnement de ces compresseurs et limiteurs sont essentiels pour notre projet pour ainsi non pas couper la musique à partir d'un certain seuil mais pour simplement la compresser et ainsi réduire le volume général une fois sorti des haut-parleurs du casque.

### 3. État de l'art des solutions existantes

Pour répondre aux objectifs de notre étude, il nous faut un dispositif qui permette de limiter le volume sonore. Il peut ainsi s'agir du casque audio en lui-même, ou alors d'un dispositif externe qu'on viendra brancher entre la sortie audio et le casque. Une autre piste serait une solution logicielle à notre problème, mais cela fonctionnera uniquement avec un lecteur média d'un ordinateur ou d'un téléphone.

#### Les casques limiteurs de volume



Figure 3 : EasySMX Casque audio enfant avec limiteur de volume

Comme mentionné dans la partie « objectifs du projet », il existe une multitude de casques pour enfants avec un limiteur de volume intégré au seuil de 85dB recommandés par les normes de santé, comme celui représenté à la figure 3. Ils sont très peu coûteux (entre 15€ et 30€) mais ne conviennent évidemment plus à un adolescent de par sa taille, son design, son ergonomie, sa qualité sonore...

Il existe alors des casques pour adultes avec des limiteurs de volume. Pour exemple, celui de la figure 4 possède un limiteur de volume au seuil de 96dB, ce qui est donc beaucoup trop élevé pour nos objectifs du projet. De plus, le prix est quant à lui bien plus élevé : il dépasse souvent les 100€. Un autre point négatif est qu'il s'agit de casques professionnels donc les connexions ne sont souvent pas les mêmes, il faut donc des adaptateurs en plus. Enfin, un casque audio sera difficilement reproductible dans une démarche de DIY.



Figure 4 : Beyerdynamic DT-770 Pro avec limiteur de volume

### Les dispositifs limiteurs de volume

Une des premières solutions potentiellement envisageables est une solution logicielle (figure 5) téléchargeable sur téléphone, gratuite et simple d'utilisation. En effet cette application permet de plusieurs façons moduler les volumes sortant du téléphone, comme par exemple en choisissant les sorties audio (casque pour la musique, haut-parleur pour les appels) et permet aussi de modifier le volume de chaque application. La dernière fonctionnalité utile est qu'elle permet d'imposer une limite aux volumes de sortie. Le problème étant que notre utilisateur a pour habitude d'écouter de la musique via une chaine Hi-Fi, cette solution pourra quand même lui servir dans lors de l'utilisation d'un téléphone portable.



Figure 5 : Application limiteur de volume

Dans certains domaines où l'utilisation de casque audio est fréquente comme l'audiovisuel, le journalisme, les centres d'appels etc., la majorité des employés sont incités à utiliser ce genre de dispositif pour protéger les utilisateurs de casques des niveaux excessifs. Ils sont alimentés par le signal audio lui-même, ils sont simples d'utilisation et de branchement, comme par exemple le modèle Canford à la figure 6. Cependant ce dispositif présente plusieurs problèmes liés à notre étude : tout d'abord tous les casques ne sont pas éligibles pour la bonne réduction du bruit, de plus il faut que chaque casque soit testé individuellement pour calibrer la limitation sonore, ce qui augmente considérablement le prix d'installation. Enfin, la faisabilité et la reproductibilité de l'installation et de la conception pour notre projet sera compliquée.



*Figure 6 : Canford – Limiteurs pour casques audio*

Une solution plus universelle et modulable possible est celle de la figure 7. Il s'agit du même fonctionnement que le dispositif électronique précédent sauf que celui-ci est compatible avec tout type de casque et il n'y a pas besoin de calibration. Cependant le prix est bien plus élevé (200€), mais il s'agit de la meilleure solution pour l'instant trouvée. Comme il s'agit d'une « boîte noire », il ne pourra pas répondre à nos contraintes de reproductibilité mais on peut désormais orienter notre étude dans l'objectif de reproduire ce dispositif à des frais réduits, avec des composants électroniques facilement obtenables.



*Figure 7 : LIM100 Limiteur de niveau pour casque*

On va donc désormais se focaliser sur une transmission filaire analogique (prise Jack) de la musique et d'un traitement entièrement analogique de notre signal, le but final étant de prototyper notre dispositif sous la forme d'un boîtier analogique. Ainsi, on peut commencer à explorer les solutions existantes les plus simples pour répondre à nos contraintes et objectifs.

La solution la plus simple de compréhension et de réalisation est un pont diviseur de tension entre le signal d'entrée et de sortie. Il est nécessaire cependant de connaître l'impédance du casque mais rajouter une résistance (en réalité deux pour la piste gauche et droite) entre la sortie de la chaîne Hi-Fi et le casque permettra simplement et analogiquement de réduire la valeur du signal. Cependant il s'agira d'une valeur constante de gain mais cela pourra être pallié avec des résistances variables.

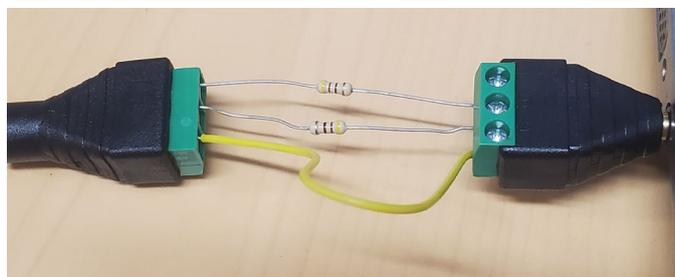
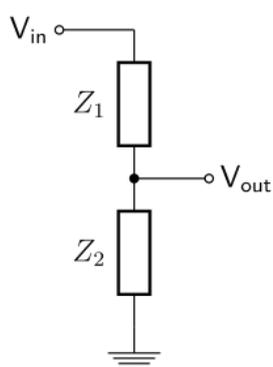


Figure 8 : Schéma et montage du pont diviseur de tension

Ainsi, sur la figure 8, la résistance  $Z_2$  représente l'impédance du casque et la résistance  $Z_1$  celle qu'on rajoute dans notre dispositif. Par exemple, si l'impédance du casque est de 16 Ohms et celle de notre résistance de 20 Ohms, on aura ainsi un signal de sortie de  $\frac{16}{16+20} = 44\%$  du signal de départ.

Les problèmes d'un dispositif de ce genre sont qu'il dépend de l'impédance du casque de l'utilisateur, qu'il agit uniquement sur l'amplitude du signal et non sur les décibels ressentis (on pourrait toujours être au-dessus des 85 décibels sans s'en rendre compte) et aussi on aurait une grosse perte d'énergie due à l'effet Joule, mais le dispositif serait très simple en termes de faisabilité et de reproductibilité. On va donc continuer dans cette direction de dispositif électronique simple en essayant d'adapter notre dispositif à nos propres contraintes et objectifs.

Pour conclure cet état de l'art, nous avons pu constater qu'il existait déjà des casques limiteurs de volume et des dispositifs limiteurs de volume. Cependant, les casques sur le marché ne sont pas adaptés à notre adolescent pour leurs dimensions, prix et domaine d'utilisation. On a alors pu constater que les dispositifs existants sont intéressants dans leurs conceptions et leurs fonctions, ils sont de bonnes pistes pour créer notre propre dispositif de boîtier analogique.

#### 4. Simulations et résultats

Dans cette partie, nous allons utiliser le logiciel LTSpice qui est un logiciel informatique de simulation de circuit électronique analogique, gratuit et open source. Il nous permettra alors de concevoir et théoriser notre dispositif sous forme de schéma électrique et ainsi de le tester. Il permet aussi d'importer et d'exporter directement des fichiers .wav, donc des fichiers audios de musique.

L'exportation nous permettra de détecter la diminution ainsi que les déformations du signal à l'oreille.

Pour commencer, nous allons utiliser la base de l'amplification en électronique : les ALI (Amplificateur linéaire intégré). Il s'agit d'un composant électronique simple permettant de réaliser des opérations mathématiques (addition, soustraction...) mais aussi des fonctions mathématiques récurrentes en traitement du signal (filtrage, gain...).

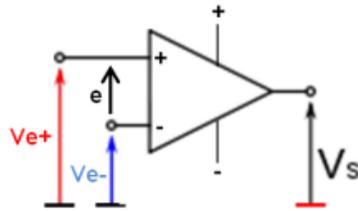
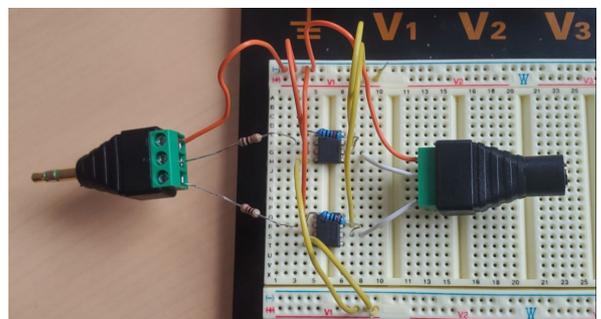
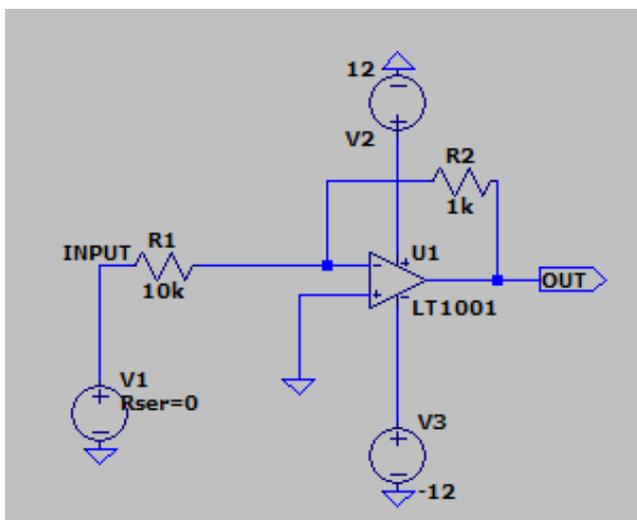


Figure 9 : Schéma électrique d'un ALI

Il est composé de deux entrées, une non inverseuse (+) et une inverseuse (-), et d'une sortie. Il possède aussi deux bornes d'alimentation. La fonction de base d'un ALI est d'amplifier la différence de potentiel « e » entre les tensions d'entrées, on a donc  $e = V_{e+} - V_{e-}$  et  $V_s = A.e$ , avec A la valeur d'un gain dépendant du reste du circuit.

Le premier montage est un amplificateur inverseur. Celui-ci permet d'obtenir un gain inférieur à 1 donc une atténuation.



$$\frac{V_e}{R_1} = -\frac{V_s}{R_2}$$

$$\Leftrightarrow \frac{V_s}{V_e} = -\frac{R_2}{R_1}$$

$$\Rightarrow V_s = -\left(\frac{R_2}{R_1}\right)V_e$$

Figure 10 : Schéma électrique, montage et fonction de transfert d'un amplificateur inverseur

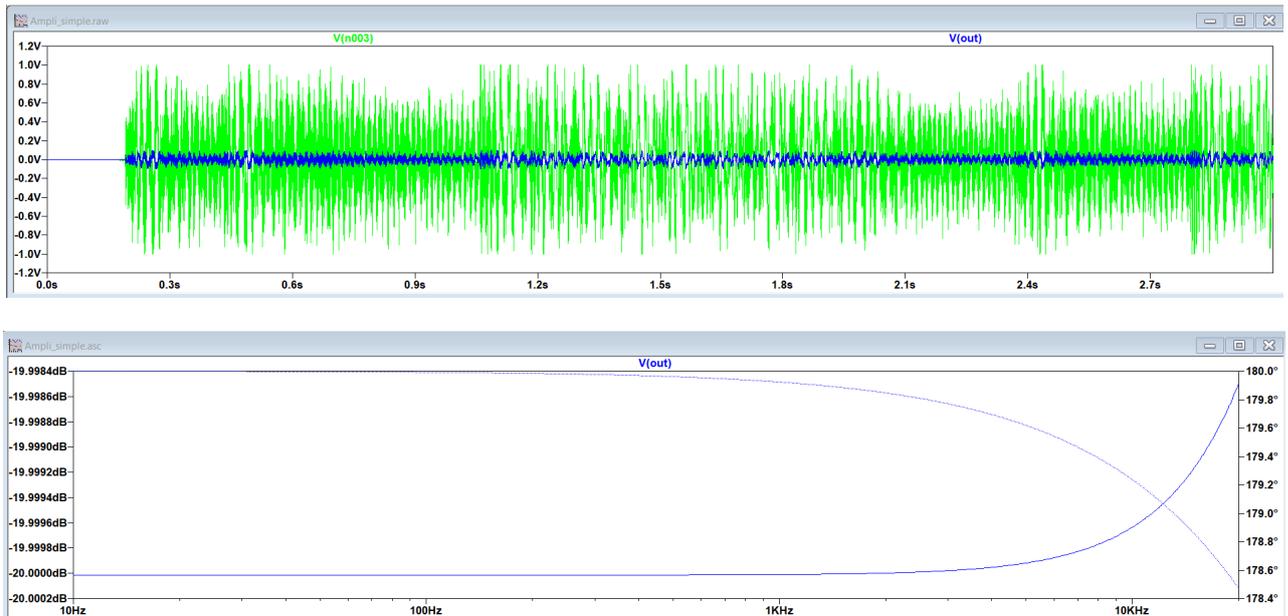


Figure 11 : Schéma de l'amplificateur inverseur et la réponse temporelle (bleue) du signal d'entrée (vert) ainsi que sa réponse fréquentielle

Ainsi, sur l'exemple du dessus, nous avons une atténuation de 0.1 soit 90% du signal d'entrée. Il s'agit donc en réalité dans le domaine fréquentiel d'une diminution de 20 décibels. Le signal est inversé mais cela ne semble pas impacter la qualité audio une fois écoutée. Comme pour le montage précédent, il serait intéressant d'ajouter des résistances variables pour ainsi contrôler le gain. Un des problèmes majeurs est que comme le montage utilise des composants actifs, ils doivent donc être alimentés par une source extérieure, mais on parlera de cette contrainte plus tard.

Le deuxième montage est un compresseur semi-actif utilisant des transistors. Il s'agit d'un montage assez connu dans le milieu des amateurs de musique analogique, notamment utilisé pour le mastering d'instruments de musique comme une guitare ou une basse directement branchée à l'ampli.

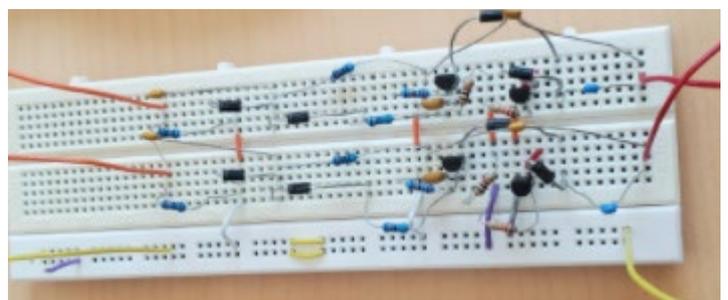
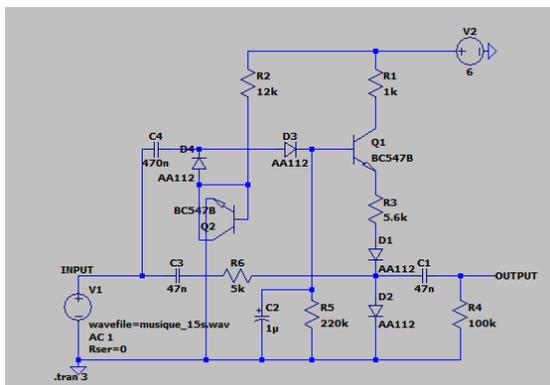


Figure 12 : Schéma et montage du compresseur semi-passif

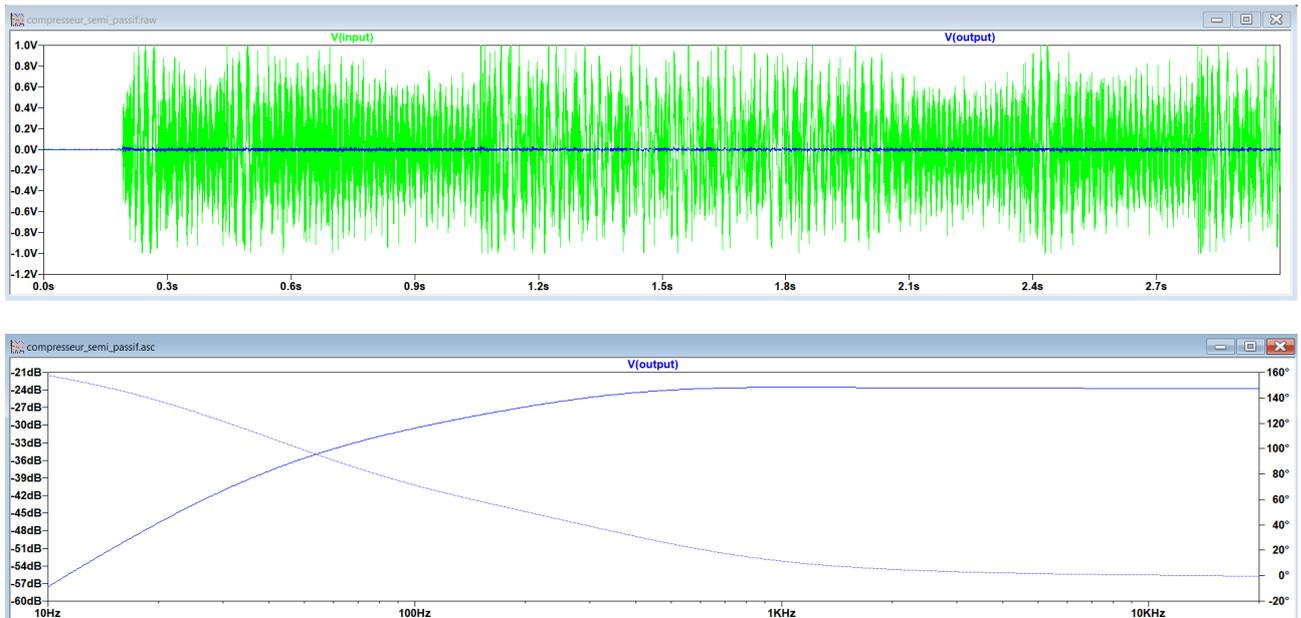


Figure 13 : Réponse temporelle et fréquentielle du compresseur semi-passif

Premièrement, le gain de ce montage est intrinsèquement lié aux modèles des quatre diodes présente dans le montage. En effet, selon leur tension de seuil, elles peuvent plus ou moins atténuer le signal. Sur la figure 13 ci-dessus, j'ai sélectionné les diodes avec de faibles tensions de seuil ce qui créer donc une forte atténuation. Cependant comme on peut l'observer cette atténuation n'est pas linéaire, les basses fréquences sont plus atténuées que les hautes. Cela peut sembler anodin mais on peut facilement entendre la différence, cela crée une sorte de distorsion de la musique et les aigus deviennent plus forts que les graves. De plus, ce montage aussi nécessite une alimentation externe, mais une seule de 6 volts.

Le dernier montage que j'ai simulé et monté est un montage d'amplificateur complètement passif à base d'un pont de diodes et d'une résistance de retour permettant d'ajuster le gain de ce dernier.

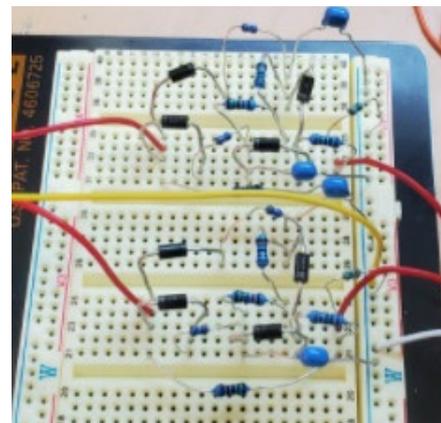
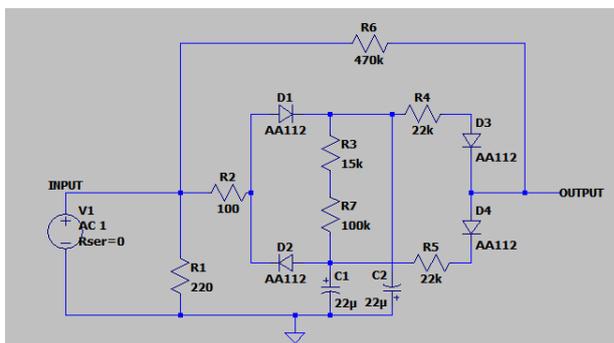


Figure 14 : Schéma et montage du compresseur passif

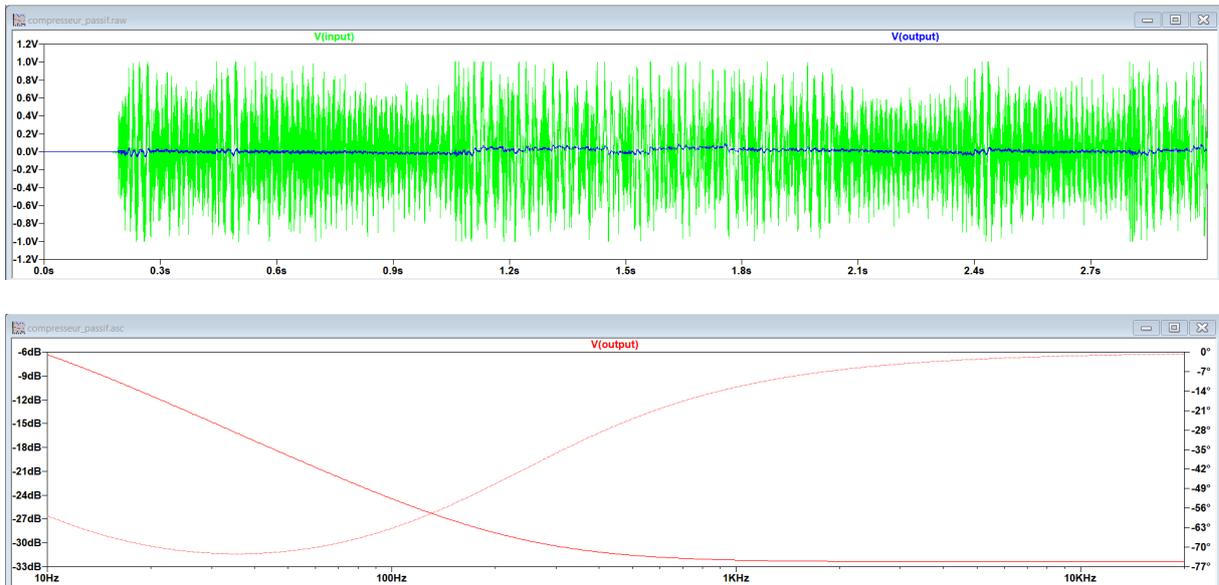


Figure 15 : Réponse temporelle et fréquentielle du compresseur passif

On peut tout d'abord constater l'utilisation d'un pont de diodes permettant d'écarter le signal dans des enveloppes, puis les atténuer avec les résistances R4 et R5. C'est avec la résistance de retour R6 qu'on va jouer sur l'atténuation du signal, on a donc décidé de le remplacer par un potentiomètre linéaire entre 1 et 10k Ohms. Ainsi, ce montage permet une atténuation du signal d'entrée sans réelle distorsion notable pour l'oreille humaine, le gain se stabilise autour des 100Hz à 30dB donc ce qu'il y a avant n'est déjà peu audible dans les musiques. Il n'a aussi pas besoin d'alimentation externe comme les précédents montages ce qui le rend bien plus pratique à mettre en place.

Un dernier point annexe montrant que le logiciel LTSpice n'est pas la meilleure solution pour élaborer des dispositifs liés à la musique et au son est son traitement de la stéréo. En effet, j'ai effectué une batterie de tests quant à l'appréhension de la stéréo par le logiciel. J'ai alors créé une piste audio avec deux canaux différents, montré sur la figure 16.

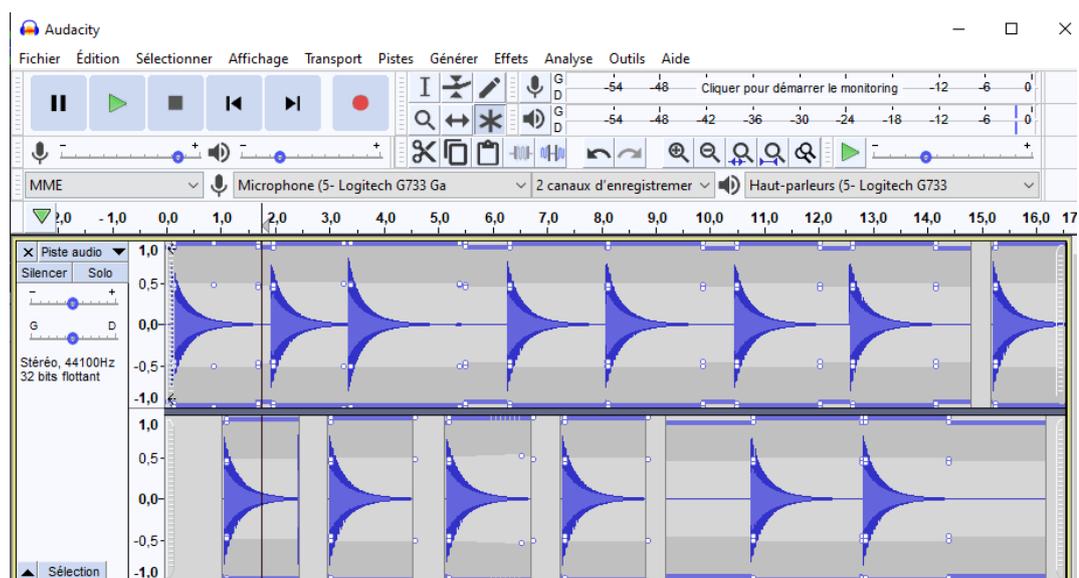


Figure 16 : Piste audio comportant deux canaux différents

Puis lors de l'importation du fichier audio, du passage dans un montage quelconque et de l'exportation, on obtient la figure 17 en sortie : On remarque alors que le logiciel ne travaille que sur le canal 1 de la piste et que lors de l'exportation il supprime le canal 2. Sachant que dans la majorité des encodages des musiques modernes il y a toujours dorénavant de la stéréo donc 2 pistes, cela pose problème pour notre étude. Une des solutions serait alors de couper notre piste en amont, la simuler pour chaque canal puis reformer la piste.

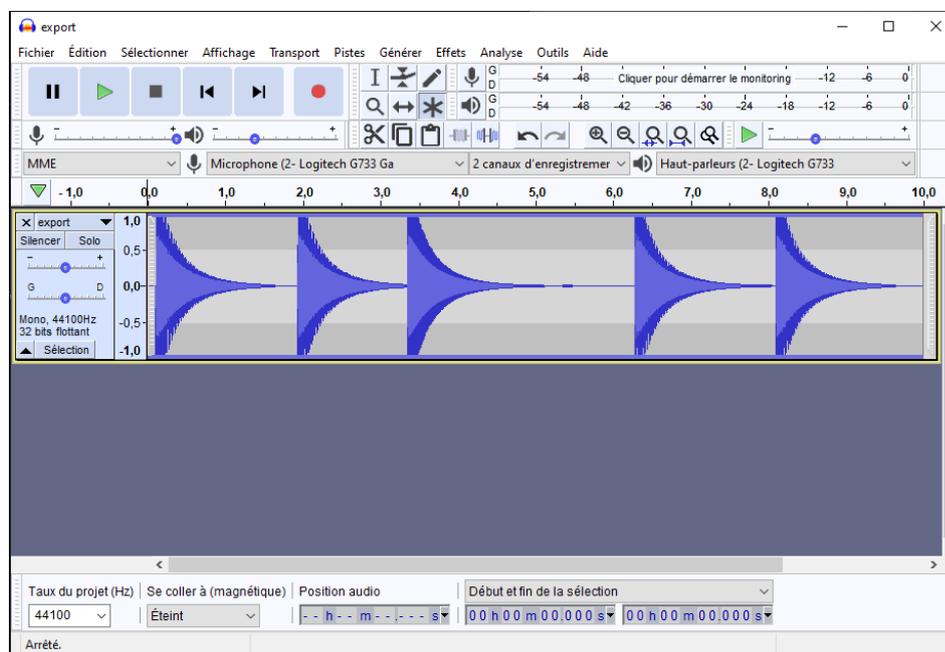


Figure 17 : Exportation d'un fichier audio par LTSpice

Ce dernier point soulève ainsi un problème de taille lors de la réalisation de notre dispositif : comment gérer électroniquement la stéréo ? Pour un montage comme un pont diviseur de tension, il sera simple de traiter les canaux indépendamment et donc d'avoir 2 fois le montage, mais pour des montages plus complexes, donc plus de composants, cela pourra engendrer des problèmes comme la taille du dispositif. Il faudra ainsi tester si le traitement pourra se faire l'un après l'autre ou pas, si un décalage / déphasage apparaît etc...

Après réflexion, nous avons décidé de simplement doubler les montages pour la piste gauche et la piste droite. Cette solution semble être la plus efficace et la plus simple à mettre en œuvre pour des montages à composants simples comme les nôtres.

La solution retenue est donc le quatrième montage présenté précédemment. Il semble convenir à toutes nos contraintes : il est plutôt simple, ne demande pas des composants complexes ou chers, il atténue bien de plus de 20dB (variable avec un potentiomètre en fonction des impédances des casques), il est entièrement passif donc ne nécessite pas d'alimentation externe et ne distord pas le signal de sortie. On va donc continuer avec ce modèle et la prochaine étape est de le retranscrire sur une plaque PCB, dans le but de réduire sa taille et ainsi pouvoir le faire rentrer dans un boîtier transportable.

## 5. Réalisation de la solution retenue

La première étape pour réaliser le montage retenu est de le retranscrire sur le logiciel gratuit et open-source KiCad. Comme on peut voir sur la figure 18, j'ai alors dédoublé le montage en ajoutant les deux prises Jack femelle ainsi que les 2 potentiomètres. Entre-temps, j'ai aussi pris l'initiative d'utiliser des diodes Schottky, très connues dans ce milieu et parfait pour des montages de ce genre.

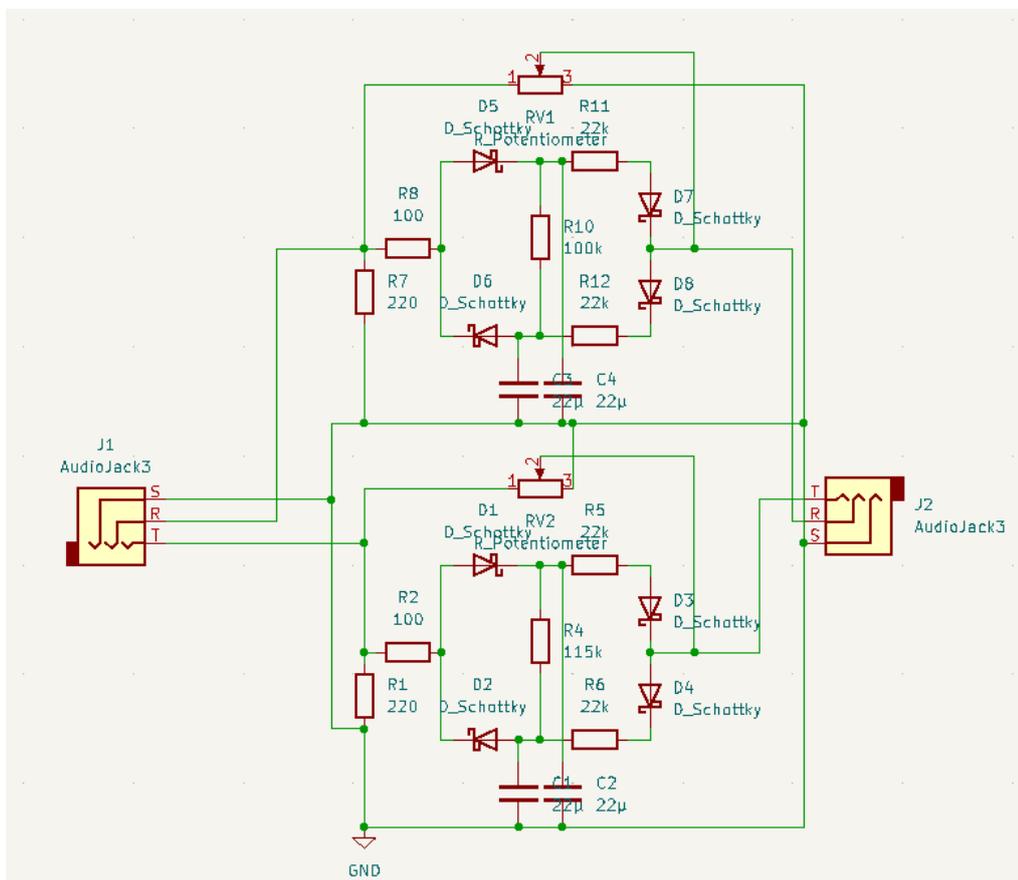


Figure 18 : Schéma électrique sur KiCad du compresseur passif

Ensuite, l'étape suivante est de choisir pour chaque composant son « empreinte » (footprint en anglais) qui est la représentation du composant une fois sur le PCB, indiquant où on doit le placer. Il faut donc faire attention à la taille et à la forme de nos composants de départ, où sont les broches à souder pour les Jack/potentiomètres etc.. Ce qui donne la figure 19 une fois l'exportation schéma/PCB faite. On peut donc reconnaître les Jacks à 3 broches, les diodes, les potentiomètres en haut etc..

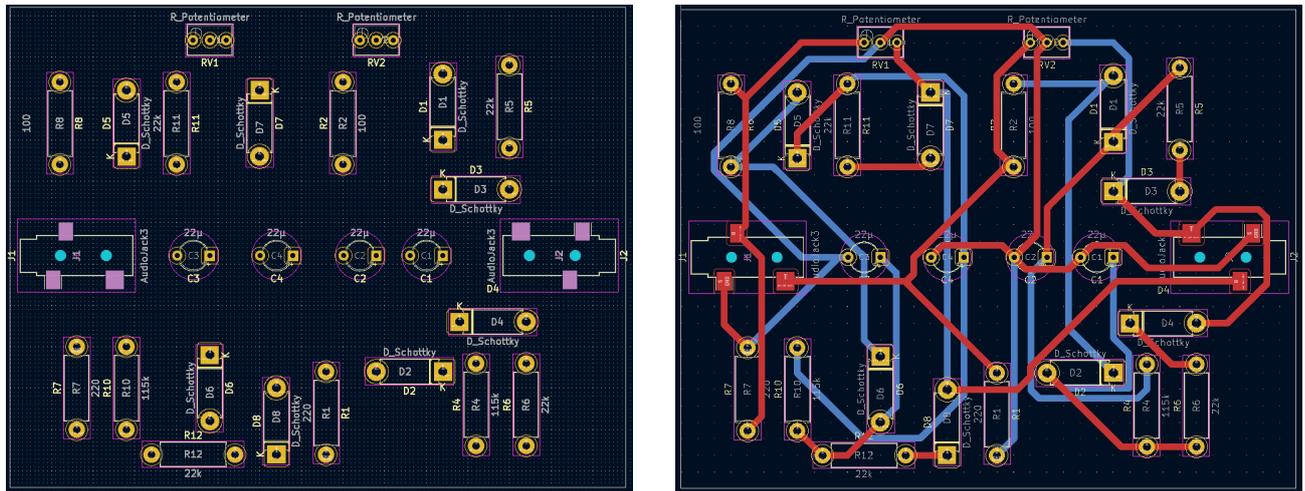


Figure 19 : Schéma électrique version PCB avec et sans les pistes en cuivre

La tâche suivante est de relier chaque broche des composants aux autres composants, comme on l'a indiqué lorsqu'on a construit le schéma électronique. On a choisi de faire un PCB sur les deux faces sans multicouches car le schéma est simple, nous avons donc deux pistes (bleu et rouge sur la figure 19) pour relier tous les composants entre eux.

Pour gagner en place et en facilité, une autre solution est de faire un plan de masse (en bleu sur la figure 20) sur une des faces : toutes les masses des composants y sont directement reliées, on a donc plus qu'une piste à gérer. On a aussi ajouté quatre trous pour le passage de vis, visible sur la figure 20, permettant de fermer le boîtier.

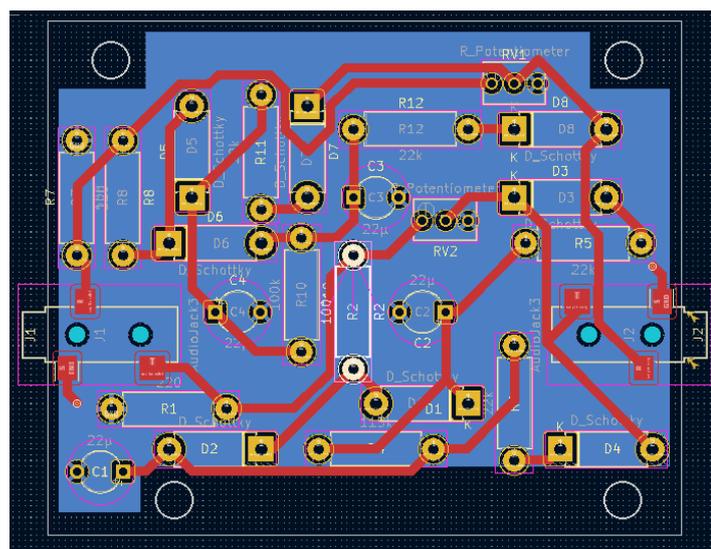


Figure 20 : Schéma électrique version PCB avec plan de masse et trous

On peut finalement avoir une vue 3D grâce à KiCad pour nous donner une idée des dimensions de notre carte avec et sans composants (figure 21).

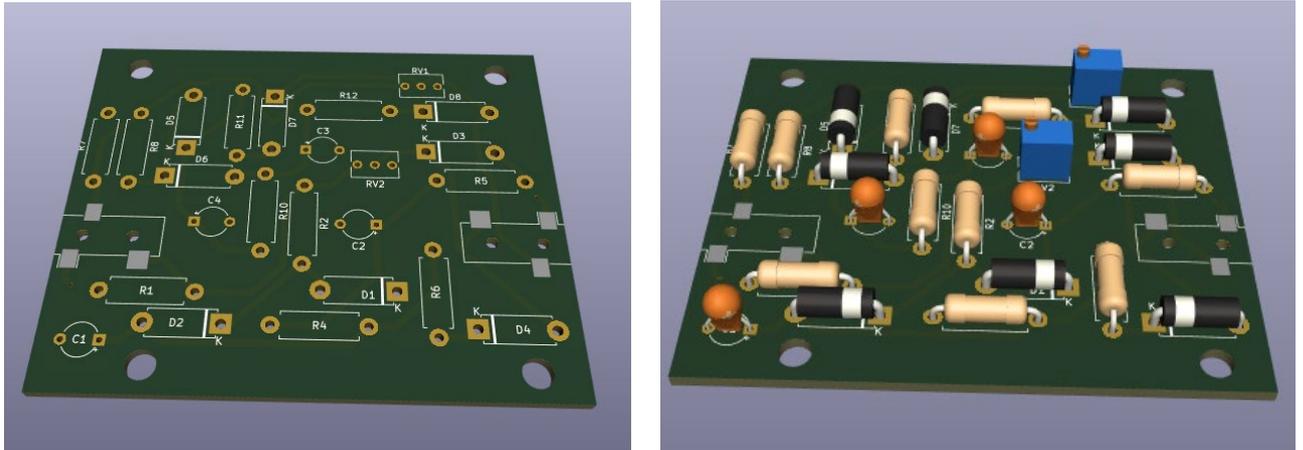


Figure 21 : Représentation 3D de notre carte, sans et avec composants

Suite à cela, nous avons passé commande sur le site pcbway [9] pour imprimer ces fameuses cartes électroniques, et nous avons reçu un paquet de 5 cartes comme celles montrées sur la figure 22.

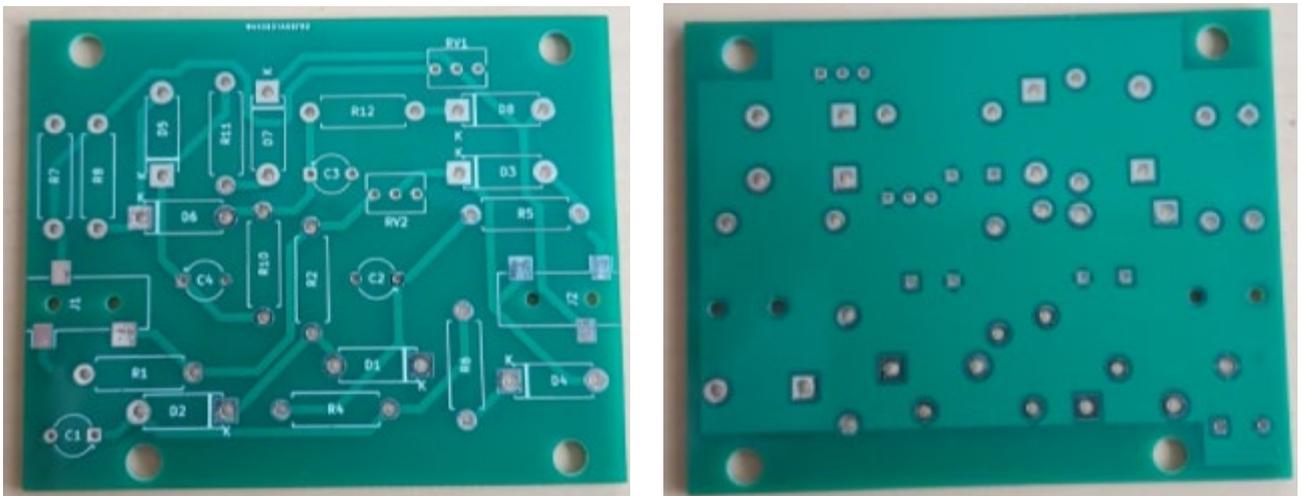


Figure 22 : La carte électronique une fois imprimée, face recto et verso

Et c'est lors du Fokus Magic Control où j'ai pu notamment rencontrer quelques personnes de l'Inria de Grenoble dont Pissard-Gibollet Roger qui m'aide depuis le début de mon stage, que j'ai pu récupérer les cartes et que j'ai pu y souder les composants de mes plaquettes de tests. On retrouve donc sur la figure 23 les différents composants assemblés et soudés par mes soins.

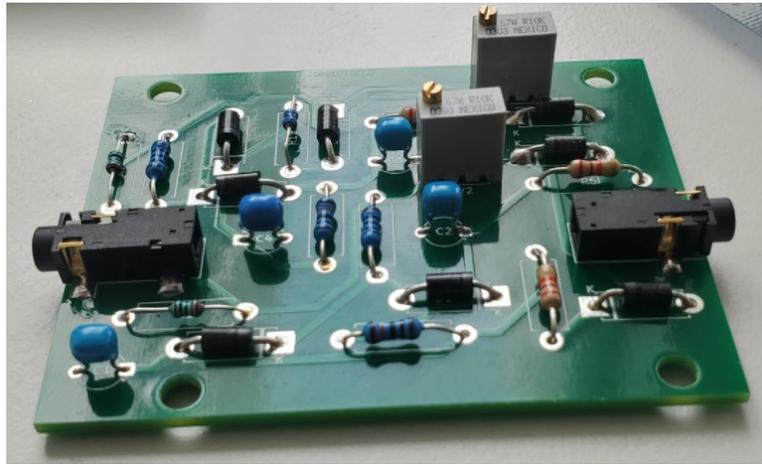


Figure 23 : La carte électronique une fois les composants soudés dessus

Ensuite, pour le boîtier, nous avons reçu une pré-version imprimée par Sijobert Benoît du HumanLab de Palavas, contenant deux embouts pour bloquer le câble pour éviter de débrancher les prises Jack (figure 24 en noir), malheureusement il n'était pas à la bonne hauteur donc nous en avons imprimé un autre durant le Fokus Magic Control (figure 24 en rose).



Figure 24 : Les différents prototypes de boîtiers imprimés

On a retiré les extrémités pour faciliter les phases de tests, mais ce boîtier devra être réimprimé par la suite pour qu'il soit en matériaux inoffensifs car l'adolescent a tendance à mâchouiller les câbles donc potentiellement aussi le boîtier.

J'ai alors effectué quelques tests notamment vis-à-vis de la distorsion mais aussi en fonction des valeurs maximales et minimales des potentiomètres. J'y ai donc injecté un bruit blanc d'amplitude

1 sur la figure 25 et j'ai mesuré la sortie en la branchant sur l'entrée micro de mon ordinateur. On obtient donc la figure 26 qui sont les valeurs minimales et maximales obtenable.

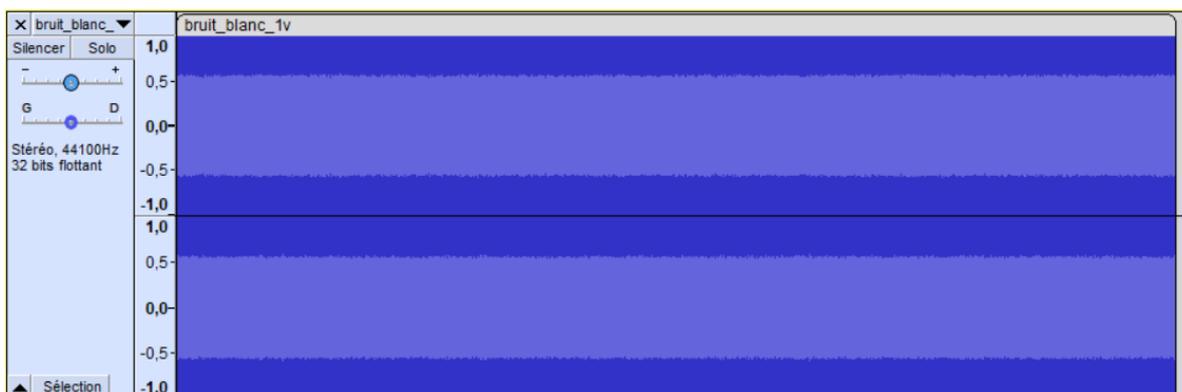


Figure 25 : Bruit blanc d'amplitude 1

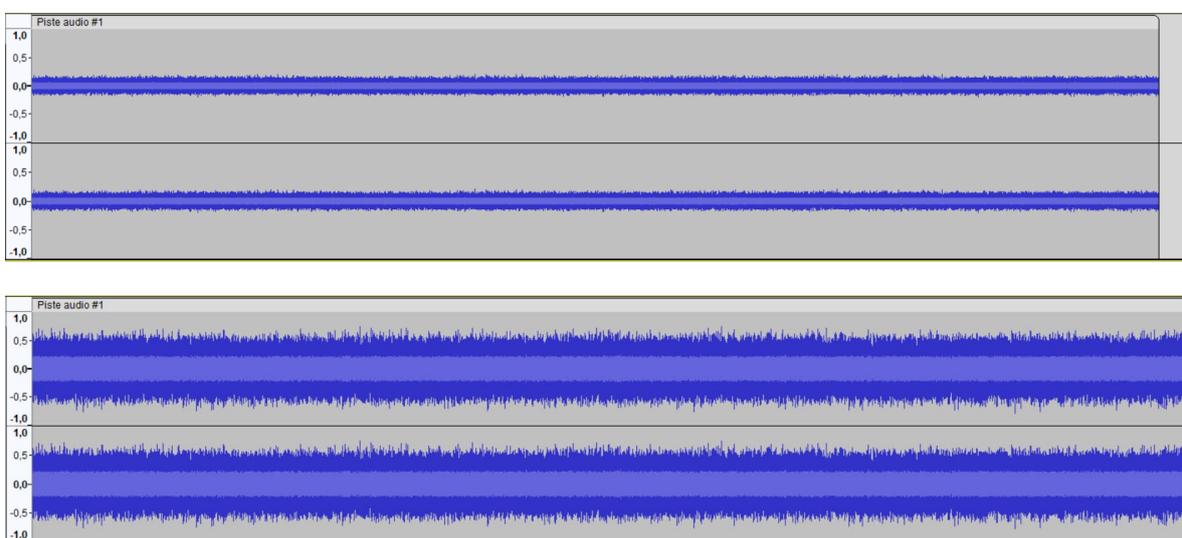


Figure 26 : Réponse temporelle du bruit blanc injecté dans le dispositif

Il est par conséquent possible d'avoir un minimum d'environ 0,15 du signal d'entrée et un maximum de 0,5 du signal d'entrée, sans distorsion notable.

Cela met fin à ce projet. Malheureusement je ne pense pas avoir le temps de tester le dispositif avec les porteurs du projet, mais je prendrai soin à envoyer les cartes déjà imprimées et soudées au HumanLab de Palavas pour qu'ils puissent tester et me faire un retour sur le dispositif, ce qui marche, ce qu'il faut modifier etc... Pour ma part, les futurs points d'amélioration seraient de remplacer les deux potentiomètres par un seul à deux entrées et sorties ce qui serait plus simple à configurer. Un autre point serait de potentiellement réduire encore plus la taille de la carte pour pouvoir imaginer une autre façon de contraindre les câbles (en les enroulant autour ou en collant les 2 Jack par exemple) mais aussi de pouvoir implémenter le dispositif directement dans le casque,

pouvant ainsi le rendre utilisable sur des casques sans fil. La dernière chose est, comme expliquer plus tôt, de refaire les boîtiers dans un matériau non nocif, de même pour les vis et les boulons.

Une dernière parenthèse que j'aimerais faire est que le traitement du signal en analogique dans notre cas n'est peut-être pas la seule solution possible. En effet, une autre approche de cette problématique serait d'utiliser un traitement numérique du son qui aurait été beaucoup plus modulable mais plus complexe à mettre en place. Cependant, il reste le problème de la transmission qui est forcément analogique et filaire avec une prise Jack, donc pourquoi pas se pencher sur l'utilisation de convertisseur analogique numérique pour de futures expérimentations, ou d'autres méthodes pour des casques sans fil, en Bluetooth ou non.

## IV. Projet ReadForMe

### 1. Description du système et ses faiblesses

Ce projet a pour but premier d'aider des personnes mal-voyantes en élaborant une 'machine à lire'. Il existe des solutions commerciales mais très chères à l'achat, de l'ordre de plusieurs milliers d'euros. Ainsi, le développement matériel et logiciel est réalisé en Open-Source pour que d'autres personnes puissent fabriquer et adapter ces dispositifs à leurs besoins, dans une optique de DiY (Do it Yourself). Le but du stage vis-à-vis de ce projet est d'intégrer des algorithmes pour améliorer les résultats déjà obtenus lors de la version précédente, que ce soit au niveau des prises de vue ou du traitement logiciel.

Le dispositif est composé d'une Pi Camera, d'un Raspberry 4 et d'une structure permettant d'orienter la caméra, avec quelques boutons de contrôles, qu'on peut voir sur la figure 27. La caméra viendra prendre en photo le texte que l'utilisateur souhaite lire, puis le Raspberry traitera l'image pour en faire ressortir uniquement le texte (océrisation) et pourra ensuite le transcrire en audio (synthèse vocale) grâce à un haut-parleur. Tout au long du projet, j'utiliserai des sets d'images déjà prises avec la caméra pour pouvoir effectuer des tests en série sur un grand nombre de photo.

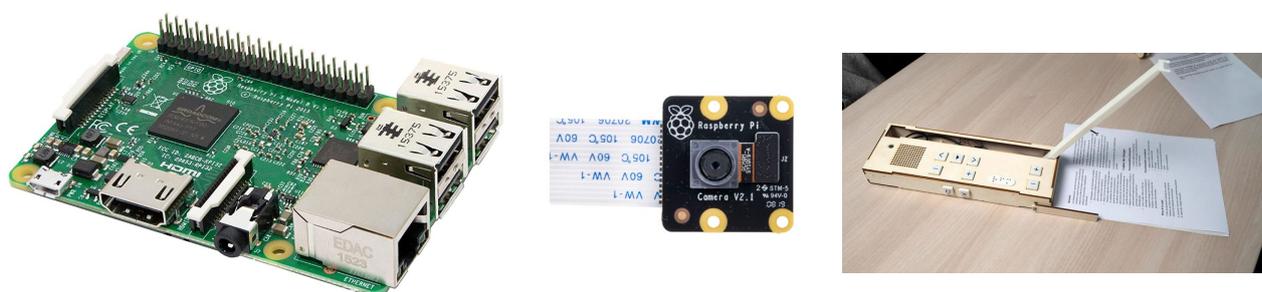


Figure 27 : Matériel à disposition : Raspberry 4, PiCamera v2.1 et le dispositif accueillant le tout

Je vais donc me concentrer sur l'ajout d'un bloc pré-traitement dans la chaîne de traitement, comme présenté sur la figure 28 en rouge car les résultats de ces 3 versions ne sont pas convenable, j'en ai donc déduis qu'il faudra améliorer la qualité des images d'entrées pour faciliter la conversion en texte qui est faite avec la librairie open-source PyTesseract [10] qui est simple à utiliser et plutôt performant, et qui est compatible avec OpenCV [11], la librairie que je vais utiliser pour effectuer le pré-traitement.

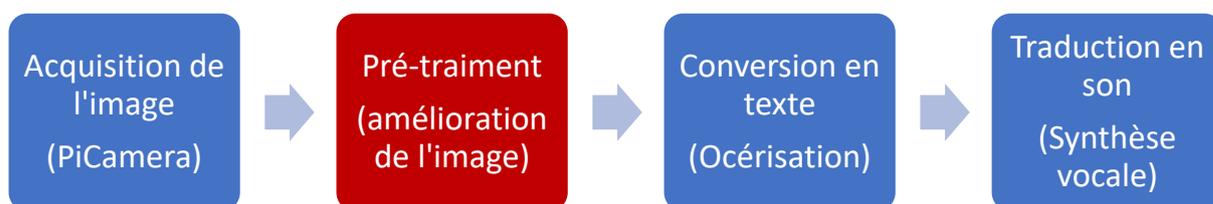
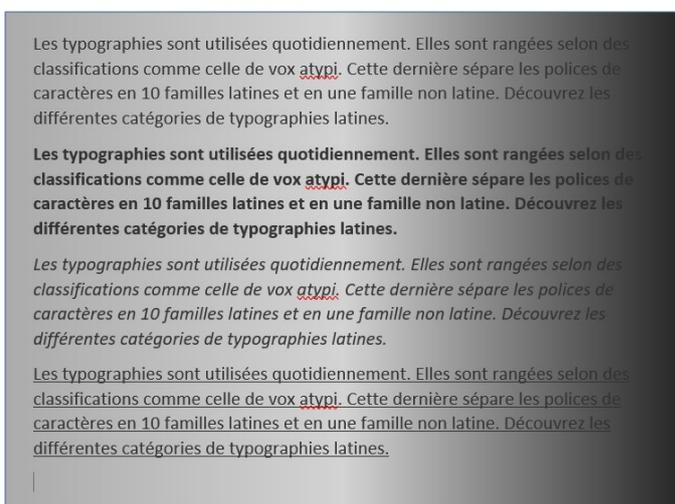


Figure 28 : Schéma-bloc des différentes étapes du dispositif

La première étape a été de déterminer les paramètres qui influaient sur les résultats de Tesseract, en reprenant les images prises lors des précédentes versions du projet et en créant de nouvelles, toutes différentes avec des défauts facilement identifiables pouvant nuire au résultat de l'océrisation. Donc tout d'abord j'ai testé avec différentes polices d'écritures, de taille, de rotation, des textes en gras/italiques/soulignés, différents niveaux de gris, tous les accents, etc... Vous pouvez trouver des exemples d'images que j'ai utilisés sur la figure 29.



Attirez l'attention du lecteur avec une citation du document ou utilisez cet espace pour mettre en valeur un point clé. Pour placer cette zone de texte n'importe où sur la page, faites-la simplement glisser.

Figure 29 : Exemples d'images créées par moi-même pour tester la robustesse du logiciel

Après ces tests, j'ai pu en déduire que les paramètres qui gênent Tesseract dans l'océration sont : les écritures manuscrites mais cela est plutôt rare dans notre cas, les photos pouvant être interprétées comme du texte, les lettrines, mais surtout les textes inclinés et les textes avec des différences de contrastes. Mon travail sera donc de déterminer et tourner automatiquement les textes mais aussi d'effacer tout contraste de lumière sur les images.

## 2. Etat de l'art

Il existe déjà sur le marché une multitude de dispositifs et de solution performante dans ce genre, comme des applications sur smartphone, notamment celle de Microsoft nommée Seeing AI (figure 30). Cependant, cela ne correspond pas au cahier des charges car un téléphone est souvent difficile à utiliser pour une personne mal-voyante, et leur usage n'est pas forcément adapté dans le cadre d'école ou de centre d'accueil. De plus, ces services ont souvent besoin d'une connexion internet continu ce qui peut déranger certaines personnes.



*Figure 30 : Logo du logiciel Seeing AI de Microsoft*

Ensuite, il existe des dispositifs électroniques disponible à la vente comme le Téléagrandisseur parlant Vocatex+ ou alors des dispositifs portables plus pratique comme le Stylo Orcam Read Smart, présentés à la figure 31. Tous deux sont des produits de la marque Mieux-Voir, Le Téléagrandisseur est vendu au prix de 4550€ et le stylo au prix de 2110€. Comme on peut le constater, leur prix est souvent très élevé, d'où l'envie d'en faire une solution DIY open-source pouvant être reproduit par toute personne dans le besoin et n'ayant pas forcément les moyens.



Figure 31 : Téléagrandisseur parlant Vocatex+ et le Stylo Orcam Read Smart

### 3. Framework de test et librairie développés

Tout d'abord, le framework de test est composé, comme dit précédemment, d'images déjà prises par une PiCamera, d'images de document PDF comme des articles de journaux ou scientifique et des images que j'ai moi-même créés contenant des anomalies (orientation, contraste, police...), ainsi que leur retranscription en fichier texte. Ainsi pour juger et quantifier les résultats, nous utiliserons et calculerons la distance de Levenshtein [12] entre les fichiers textes préalablement écrits et les fichiers textes produits par Tesseract. Nous utilisons la distance de Levenshtein car celle-ci est une distance au sens mathématique du terme : elle donne une mesure de la différence entre deux chaînes de caractères. Il s'agit ainsi du coût minimal pour transformer la première chaîne en la deuxième en effectuant les seules opérations élémentaires comme la substitution, l'insertion ou la suppression. Si ces deux chaînes sont identiques, la distance sera de 0 alors que si des caractères sont différents les uns des autres, cette distance va croître en fonction du nombre d'erreurs de Tesseract.

Pour le premier paramètre causant des erreurs, c'est-à-dire les images tournées d'un certain angle, Tesseract possède déjà une fonction nommée « image\_to\_osd » permettant de ressortir plusieurs informations de l'image : la langue détectée, la police, le nombre de pages, mais surtout la valeur en degrés de la page. J'ai donc développé une fonction permettant de prendre cette valeur en degrés et de tourner la page pour qu'elle soit bien droite, car Tesseract utilise les lignes pour détecter où sont les caractères et les mots, notamment pour faire de la lecture en colonne (article de journaux par exemple). Ainsi la première étape de traitement sera de tourner toutes les pages pour que le texte soit parfaitement horizontal lors de l'océrisation, peu importe l'angle pris avec la caméra, comme montré à la figure 32.

Attirez l'attention du lecteur avec une citation du document ou utilisez cet espace pour mettre en valeur un point clé. Pour placer cette zone de texte n'importe où sur la page, faites-la simplement glisser.



Attirez l'attention du lecteur avec une citation du document ou utilisez cet espace pour mettre en valeur un point clé. Pour placer cette zone de texte n'importe où sur la page, faites-la simplement glisser.

Figure 32 : Processus de rotation du pré-traitement

La deuxième étape est quant à elle bien plus compliquée : enlever les ombres et les contrastes d'une image. En effet, ce paramètre semble essentiel pour Tesseract car si l'éclairage n'est pas bon, l'océrisation contiendra beaucoup de fautes. Il faut donc utiliser des outils de traitement d'images pour isoler le texte du fond, c'est ce qu'on appelle la segmentation. Il existe alors bon nombre de méthodes avec un ou plusieurs paramètres modifiables, et j'en ai testé plusieurs pour déterminer lequel sera le plus apte à améliorer notre pré-traitement.

La première méthode que j'ai pu tester est une simple dérive de fond. Elle consiste simplement à isoler le fond d'une image grâce à une fermeture morphologique et d'un élément structurant en forme de cercle plus grand qu'un caractère. Une fois l'élément structurant passé sur toute l'image, nous obtenons simplement le fond de l'image, visible sur la figure 33. Puis une fois ce fond isolé, nous pouvons soustraire l'image de base au fond (Top Hat) pour obtenir seulement les caractères sur fond noir puis seuiller et inverser pour n'obtenir que le texte sur fond blanc, sans contraste dû à la caméra. Cette méthode fonctionne bien pour des objets, mais pour des caractères, nous pouvons remarquer sur la figure 34 qu'on perd en précision : certains caractères ne sont plus entiers, des « o » et des « l » sont coupés etc... ce qui pose énormément de problèmes une fois l'océrisation faite, ce qui n'améliore absolument pas le résultat. L'objectif est donc de trouver une méthode de segmentation performante qui n'altère pas la qualité et la netteté de l'image.

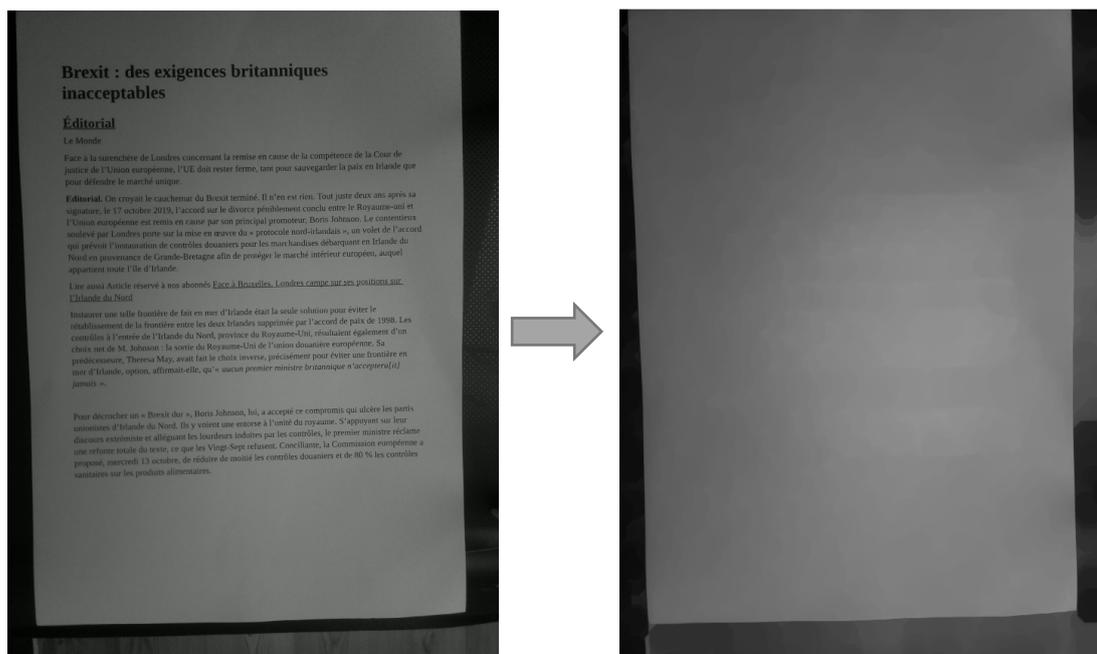


Figure 33 : Image de référence et son fond isolé grâce à une fermeture

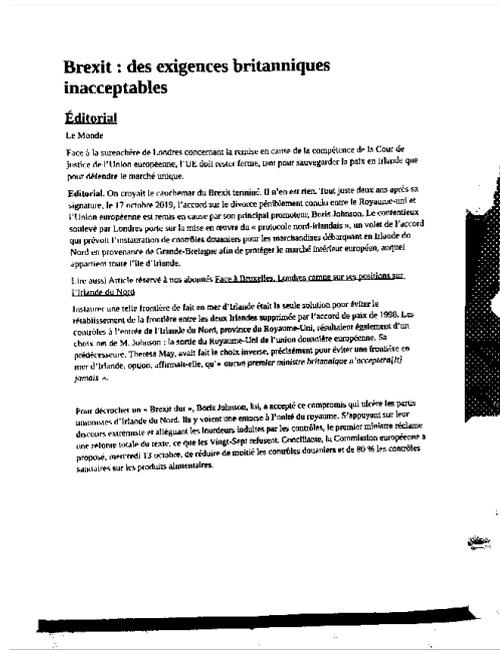
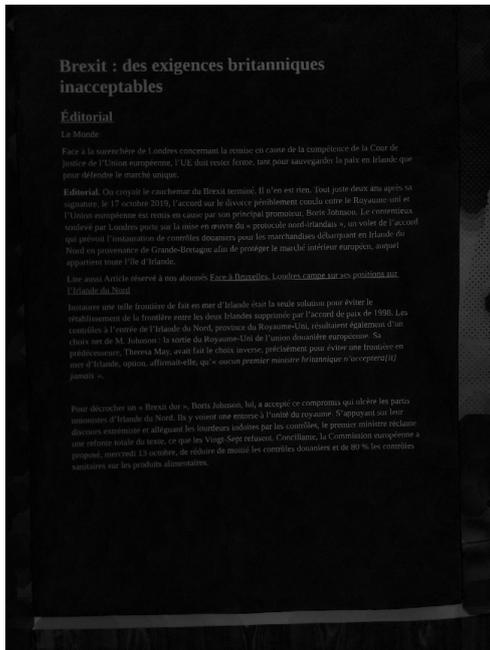


Figure 34 : Image issue de la dérive de fond et la même image une fois seuillée

Ainsi, la deuxième méthode que j'ai testée, tout aussi connue, est la méthode des K-means. Il s'agit d'un algorithme non supervisé de clustering, c'est-à-dire d'une méthode d'apprentissage qui va essayer de trouver des patrons communs dans l'image, notamment en regroupant les choses qui se ressemblent (pour nous, les caractères) dans K groupes. Dans notre cas, on aura 2 classes de pixels à classifier : ceux appartenant au fond et ceux appartenant aux caractères.

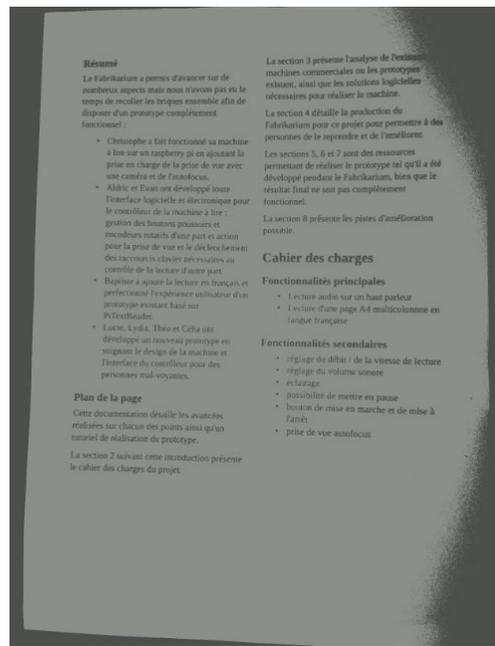
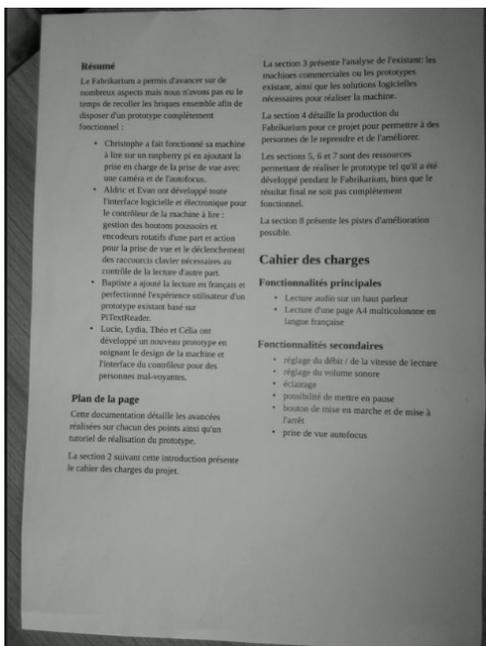


Figure 35 : Image de référence et sa segmentation grâce à la méthode des K-means

Ainsi, sur la figure 35, nous pouvons déjà remarquer qu'on perd beaucoup moins en précision. En effet, les caractères restent quasiment intacts et discernables. Cependant, comme nous pouvons le constater dans les coins de l'image, le contraste de l'image peut lui aussi rentrer dans les classes s'il devient trop sombre. Ainsi les ombres dans les coins supérieurs et inférieurs droit sont associés à la classe du texte, obstruant ainsi quelques mots. Cependant, la méthode des K-means possède plusieurs paramètres à modifier qui pourraient potentiellement améliorer le résultat, mais j'ai préféré chercher d'autres méthodes spécifiques à la segmentation de texte. En plus de cette méthode, j'en ai testé d'autres que je ne vais pas trop détailler. Parmi elles se trouvent des algorithmes segmentant en fonction de la colorimétrie de l'image (que ce soit dans le domaine RGB mais aussi HSV, L\*a\*b...) ou alors des contours en utilisant le gradient. Malheureusement ces méthodes donnaient aussi des résultats plutôt médiocres pour notre cas de figure.

La méthode que j'ai retenue est celle du seuillage adaptatif [13]. Tout d'abord, le seuillage est le simple fait de parcourir l'image et pour chaque pixel d'une couleur en dessous ou au-dessus d'un certain seuil (il s'agit du plus souvent de la moyenne du niveau de gris de l'image), on lui attribue la couleur blanc ou noir. Cependant le seuil est arbitraire, et celui-ci peut donc varier selon les images, comme montré à la figure 36, et est appliqué sur toute la page. Très vite le fond peut devenir du texte et vice-versa.

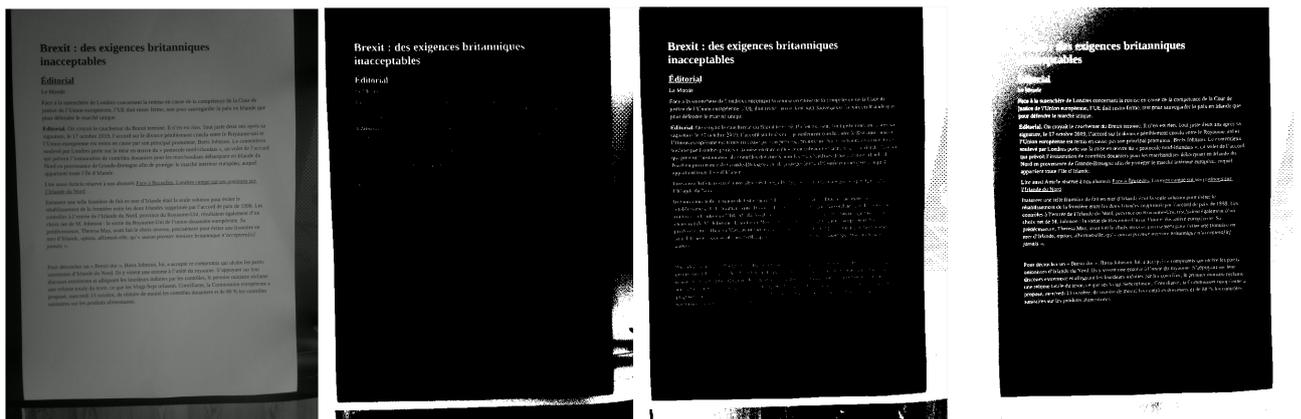


Figure 36 : Image de référence et son seuillage global aux seuils 25, 50 et 75

Ainsi, à l'instar du seuillage classique, le seuillage adaptatif ne possède pas de seuil fixe :

*Seuillage global :*

$$si \text{ le pixel } F(x,y) > S \text{ alors } F(x,y) = 1 \text{ sinon } 0$$

*Seuillage Adaptatif :*

$$si \text{ le pixel } F(x,y) > S(x,y) \text{ alors } F(x,y) = 1 \text{ sinon } 0$$

On peut cependant lui donner une valeur en entrée pour pouvoir modifier la précision des seuils. Mais chaque coordonnée de pixel  $F(x,y)$  possède un niveau de seuil  $S(x,y)$  calculé pour chaque pixel. Pour cela, l'image est tout d'abord découpée en bloc  $N \times M$  ( $16 \times 16$  dans notre cas), puis on calcule la moyenne dans chaque bloc (intégrale de la distribution en niveau de gris des pixels) pour obtenir un seuil local. Puis on calcule le seuil final de chaque pixel avec une moyenne pondérée des seuils locaux des blocs voisins en fonction de leurs distances, représenté par la figure 37.

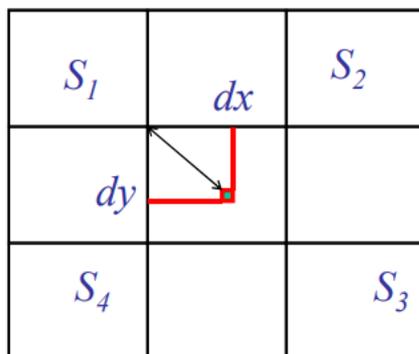


Figure 37 : Représentation des blocs et des distances pris en compte dans le calcul

On obtient donc la formule suivante pour calculer le seuil d'une zone, avec  $dx$  et  $dy$  les distances du pixel de la région et  $S_1, S_2, S_3$  et  $S_4$  les seuils locaux des voisins :

$$S(x, y) = dx \cdot dy \cdot S_1 + dx \cdot (1 - dy) \cdot S_2 + (1 - dx) \cdot dy \cdot S_3 + (1 - dx)(1 - dy) \cdot S_4$$

Cette méthode, bien que très gourmande en puissance de calcul dû au calcul des intégrales, est beaucoup plus efficace que les précédentes. Il s'agit par exemple de la seule méthode pouvant traiter correctement l'image de la figure 38.

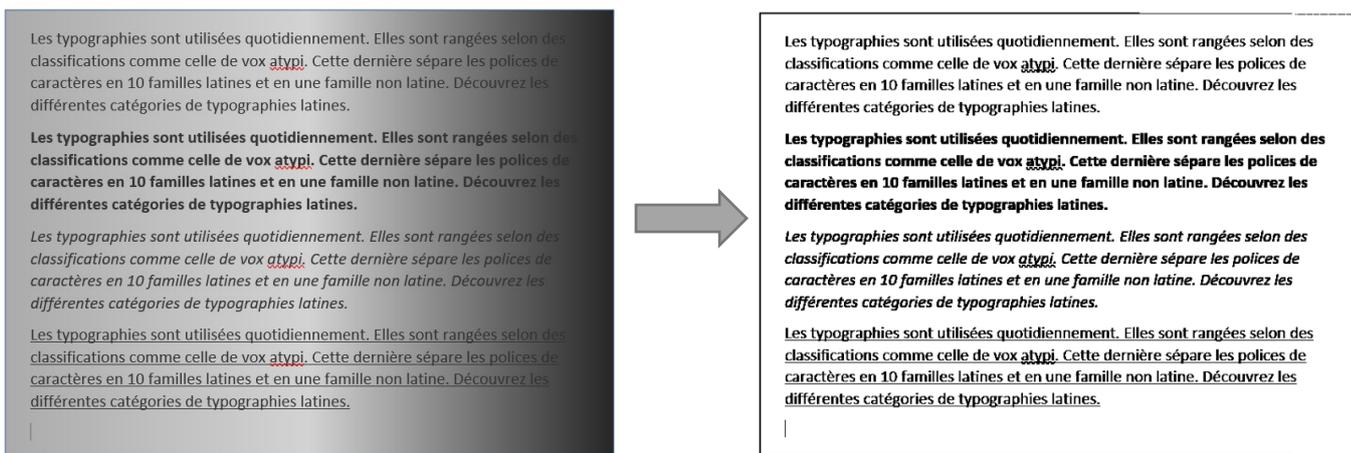


Figure 38 : Image de référence et son seuillage adaptatif

Sur une image prise par la PiCamera, nous obtenons la figure 39. On peut observer que contrairement à la dérive de fond, les caractères n'ont pas perdu en netteté ou en précision. Une fois l'océrisation faite, nous obtenons une distance de Levenshtein de 25, c'est-à-dire un pourcentage de réussite de 99,1%. Les dernières fautes qu'il reste sont majoritairement dû aux accents « é », « à », « ù » etc... qui sont difficilement reconnaissables par le logiciel, ainsi que par des sauts de lignes non voulu.

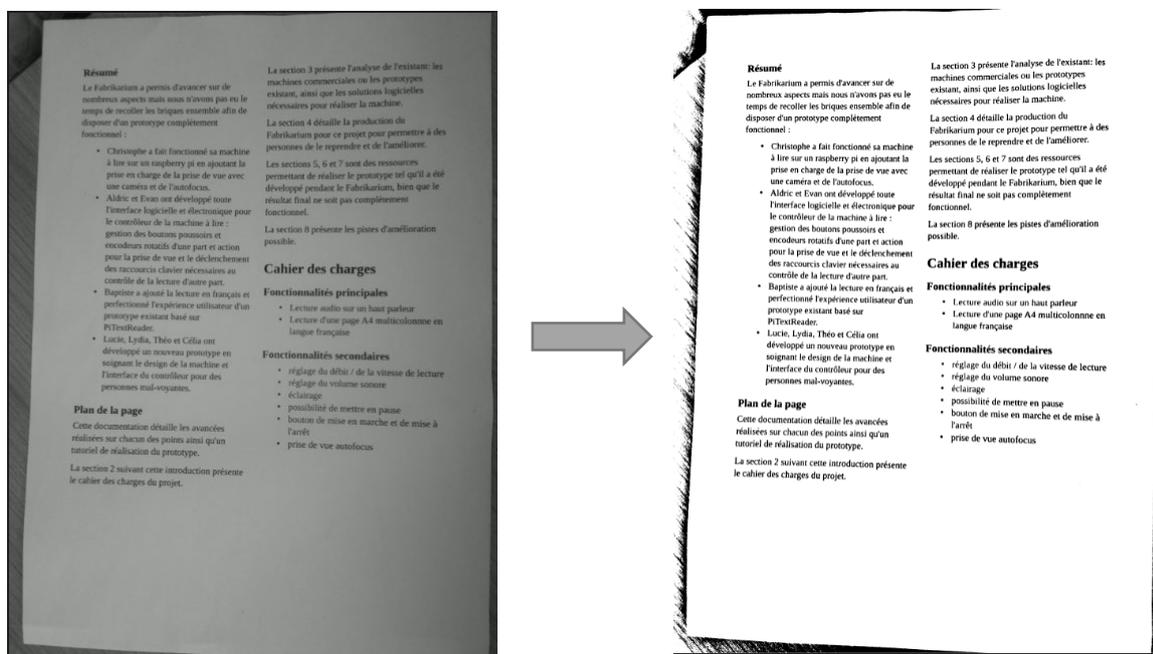


Figure 39 : Autre image de référence et son seuillage global avec un pré-seuil à 20

#### 4. Résultats

Cependant, le résultat de 99,1% est le meilleur que j'ai pu obtenir. Même si globalement le traitement améliore grandement les résultats, certains paramètres perturbent encore l'océrisation. Par exemple, les images dans les textes (comme dans les articles de journaux) sont très difficilement prises en charge. De plus, même si le pré-traitement est censé enlever tout contraste sur l'image, si le contraste de base est très élevé entre certaines zones, les résultats ne seront pas forcément meilleurs car la définition des caractères pourra être affectée. Un autre point négatif est que, comme dit précédemment, le temps de calcul est beaucoup plus long. Le temps classique d'une océrisation d'une image complète de résolution 3280x2464 sur Raspberry 4 est d'environ 30 secondes. Avec le seuillage adaptatif, nous atteignons les 60 secondes, soit une minute avant d'avoir un résultat.

Cependant nous pouvons tout d'abord encore améliorer les résultats. Comme dis précédemment, nous pouvons introduire un paramètre, appelé pré-seuil, pouvant forcer le seuillage à être plus doux

ou plus dur. Ainsi pour chaque image, il y a une valeur de ce paramètre pour laquelle la distance est minimum. Par exemple sur la figure 40, avec le pré-seuil en abscisse et la distance en ordonnée, nous pouvons voir que la distance est minimum pour plusieurs valeurs, dont 20 sur une des images à ma disposition.

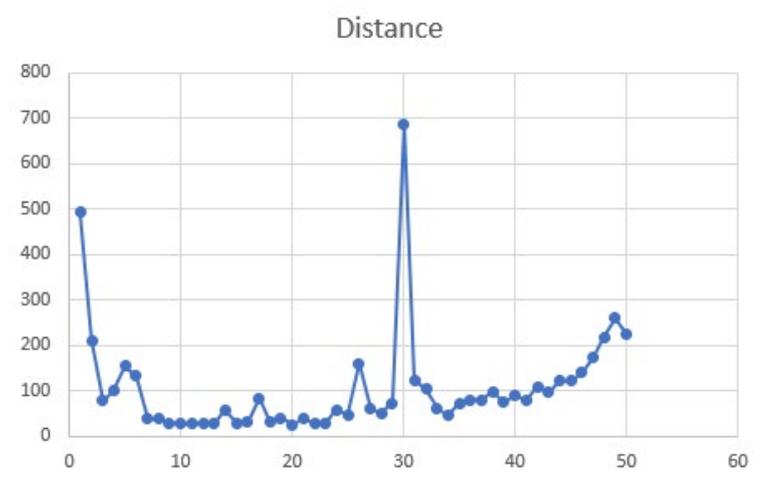


Figure 40 : Schéma électrique d'un ALI

J'ai donc par la suite utilisé un pré-seuil de 20 sur toutes les images de ma base de données, et en moyenne nous obtenons une amélioration de la distance de Levenshtein de 4,85%. Cependant, ce chiffre est une moyenne, pour certaines images nous avons même l'inverse de l'effet escompté : le pré-traitement augmente la distance de Levenshtein.

Ce dernier point permet de mettre en lumière le principal problème de ce projet, qui est traité, par ailleurs, par un bénévole de MyHumanKit : la prise de vue est tout aussi importante que le pré-traitement. Ainsi, on peut grandement améliorer les résultats en jouant sur l'optique et la vision. Pour cela, on doit ajouter de l'éclairage pour éviter tout contraste, mais aussi des pointeurs laser pour être sûr que l'utilisateur met la feuille droite. Enfin, un dispositif permettant de rendre parallèle le plan de la caméra et le plan de la feuille pour éviter toute profondeur de champ, mais ce dispositif a déjà été réalisé lors des versions précédentes.

Une autre amélioration possible, d'un point de vue temporel cette fois, serait de réduire la résolution de l'image. Cependant, j'ai pu tester quelque référence de résolution présente dans le monde de l'audiovisuel (voir annexe) et effectivement on réduit drastiquement le temps de calcul mais la précision d'océrisation aussi. Une alternative serait d'utiliser des résolutions non conventionnelles mais tout en gardant le ratio intact, ainsi on peut trouver un juste milieu entre le temps de traitement et la qualité de l'océrisation et, intrinsèquement, la compréhension audio du texte.

## V. Conclusion

Pour conclure ce stage, j'ai dans un premier temps pu découvrir le monde de l'électronique bien plus poussé que pendant mes cours, avec des contraintes bien spécifiques et surtout très différentes de celles que j'ai pu avoir par le passé. Tous les objectifs du premier projet ChutChut ont été remplis, et les connaissances que m'a apportées ce projet lors de la conception de l'état de l'art, de la découverte des phénomènes physiques, des simulations et de la conception de réelle carte électronique vont m'apporter beaucoup dans mes projets futurs. Je sais aussi désormais que le domaine du son et de la musique est bien plus vaste et complexe qu'on ne le pense, ainsi que le nombre de technologies, de techniques, de critères, de normes etc... utilisées regroupent beaucoup de domaines scientifiques différents. De plus, les méthodes de conception des HumanLabs (telles que le DIY) sont à la fois très intéressantes mais rajoutent un certain challenge lors de la résolution des problèmes rencontrés.

Je suis très heureux d'avoir pu finir ce projet de bout en bout, même si je n'aurais peut-être pas la chance de le faire valider par les porteurs du projet, notamment par l'adolescent atteint d'autisme. Cependant, j'ai pu apprendre beaucoup de choses sur la modélisation, la schématisation et la conception électronique et qu'elle n'est pas toujours simple et compatible entre différents logiciels, ce qui rend certaines simulations compliquées à mettre en œuvre.

Ensuite, j'ai aussi été très heureux de travailler sur le projet ReadForMe. Premièrement car il se rapproche plus de ma formation d'ingénieur spécialisé en traitement d'images et donc de mon parcours professionnel, mais aussi car le porteur du projet, malvoyant, est très impliqué dans ce projet. Il a d'ailleurs participé lors de la réalisation des 3 versions précédentes, alors j'espère que ma nouvelle version pourra l'aider.

J'ai ainsi pu appliquer mes connaissances en traitement d'images, mais j'ai vite été ramené à la réalité et à la difficulté que cela implique. Tout d'abord, il existe une infinité de méthodes et d'algorithmes pour une infinité de cas, avec un ou plusieurs paramètres à optimiser. Ensuite, même si une méthode marche bien sûr une image, cela ne veut pas dire qu'elle marchera pour une autre. Dans mon cas, certaines avaient une amélioration de 15% alors que d'autres une diminution de 3%.

D'un point de vue de la santé, j'ai aussi pu découvrir qu'il existe très peu de dispositifs ou de moyen disponible pour des personnes atteintes de handicap comme l'autisme ou la malvoyance, afin de simplement faire ce qu'on fait tous plus ou moins : écouter de la musique et lire. Le marché des dispositifs médicaux pratiques et portables lié aux maladies est quasiment inexistant, ou alors à des prix exorbitants, alors que des milliers voir des millions de familles en auraient bien besoin.

## VI. Bibliographie

[1] Site internet de l'équipe Inria CAMIN :

<https://team.inria.fr/camin/>

[2] Site internet de la collaboration entre l'inria et le humanlab :

<https://project.inria.fr/humanlabinria/>

[3] Site internet de l'Humanlab Saint-Pierre :

<https://www.humanlabsaintpierre.org/>

[4] Page du projet ReadForMe :

[Projets:Read For Me V3 - wikilab \(myhumankit.org\)](Projets:Read For Me V3 - wikilab (myhumankit.org))

[5] Drive du rapport du projet :

<https://drive.google.com/file/d/1E68OI03BDzPkr101nY6WHrUsVLEN8dkg/view?usp=sharing>

[6] Page du Fabrikarium :

<https://wikilab.myhumankit.org/index.php?title=Projets:Read For Me V3>

[7] Site internet des laboratoires Unisson :

<https://www.laboratoires-unisson.com/faq/audition/est-ce-dangereux-ecouter-musique.html>

[8] Article sur les compresseurs et limiteurs :

<https://www.projethomestudio.fr/reglages-compresseur-audio/>

[9] Site internet spécialisé dans l'imprimerie de carte électronique PCB :

<https://www.pcbway.com/>

[10] Site internet de la librairie PyTesseract :

<https://pypi.org/project/pytesseract/>

[11] Site internet de la librairie OpenCV :

<https://opencv.org/>

[12] Cours de l'université de Rouen sur les distances entre mots :

<http://www-igm.univ-mlv.fr/~lecroq/cours/distances.pdf>

[13] Cours de l'INSA Lyon sur le prétraitement des images :

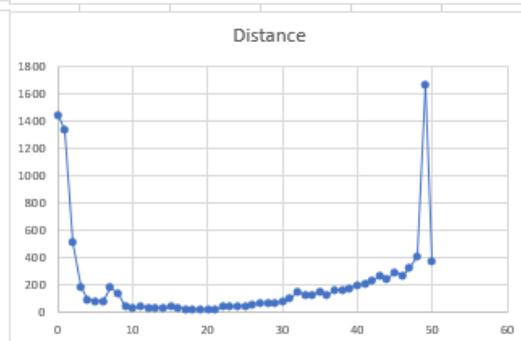
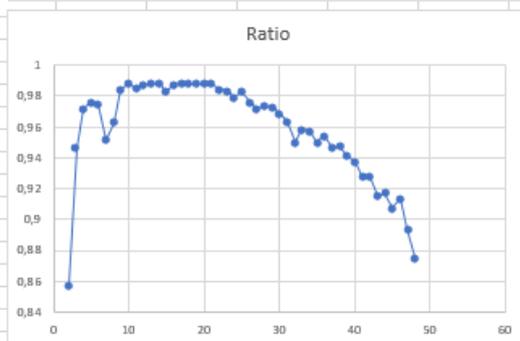
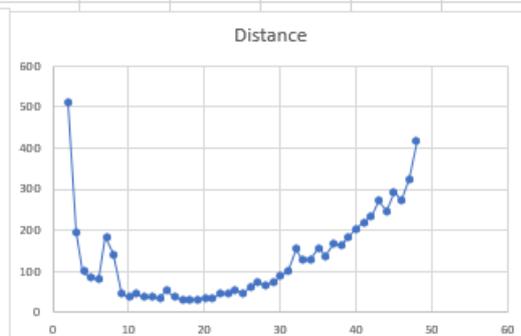
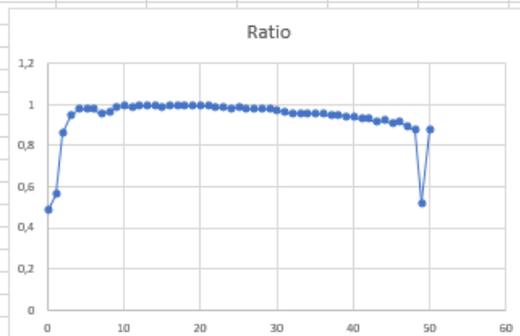
[https://perso.liris.cnrs.fr/~veglin/Master2\\_2011.pdf](https://perso.liris.cnrs.fr/~veglin/Master2_2011.pdf)

## VII. Annexe

### Détails des résultats pour le pré-traitement

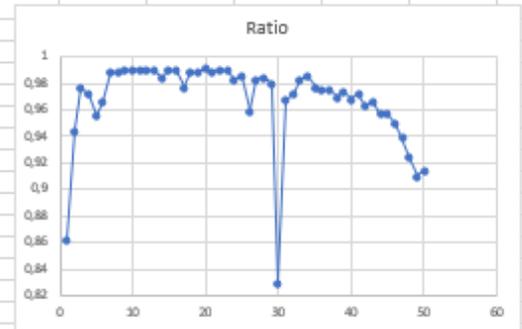
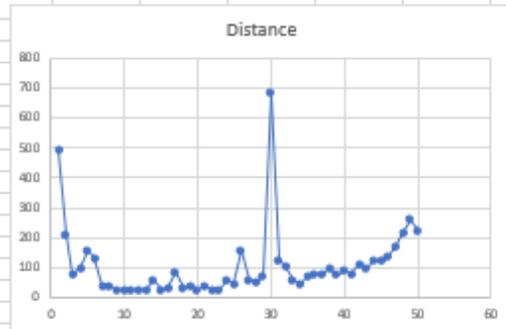
#### Graphiques des distances et des ratios selon différents seuils pour l'image nuit15

Image :	Nuit15			
Seuil	Distance	Ratio	Temps	
0	1448	0,4808	58,9	
1	1334	0,5619	60,9	
2	511	0,8569	63,7	
3	195	0,9465	63,8	
4	99	0,9717	61,2	
5	85	0,9758	61	
6	84	0,9748	60,9	
7	184	0,9511	61,2	
8	142	0,9628	59,5	
9	49	0,9842	59,4	
10	36	0,9877	59,6	
11	45	0,9849		
12	38	0,9872		
13	37	0,9877		
14	35	0,9884		
15	53	0,983		
16	37	0,9874		
17	32	0,9884		
18	32	0,9879		
19	32	0,9879		
20	33	0,9879		
21	33	0,9879		
22	44	0,9843		
23	46	0,9833		
24	56	0,9791		
25	47	0,9827		
26	61	0,9761		
27	72	0,9721		
28	68	0,973		
29	73	0,9727		
30	88	0,9691		
31	101	0,9641		
32	155	0,9501		
33	126	0,9583		
34	129	0,9571		
35	157	0,9503		
36	139	0,9544		
37	167	0,9464		
38	166	0,9481		
39	182	0,941		
40	204	0,9377		
41	221	0,9279		
42	234	0,9283		
43	273	0,915		
44	248	0,9169		
45	293	0,9071		
46	272	0,9134		
47	324	0,8942		
48	418	0,8749		
49	1674	0,5174		
50	387	0,8725		



### Graphiques des distances et des ratios selon différents seuils pour l'image Jour04

Image	Jour04		
Seuil	Distance	Ratio	Temps
0	1405	0,5402	
1	434	0,862	
2	211	0,9445	
3	77	0,9763	
4	100	0,9717	
5	155	0,9557	
6	132	0,9658	
7	38	0,9877	
8	38	0,9877	
9	29	0,9896	
10	26	0,9905	
11	26	0,9908	
12	29	0,9901	
13	27	0,9903	
14	57	0,9835	
15	28	0,9903	
16	30	0,9898	
17	84	0,9777	
18	32	0,9891	
19	39	0,9882	
20	25	0,9915	
21	38	0,9882	
22	29	0,9903	
23	28	0,9903	
24	58	0,9828	
25	47	0,9856	
26	160	0,9596	
27	59	0,9831	
28	50	0,9849	
29	73	0,9796	
30	688	0,8293	
31	123	0,9682	
32	104	0,972	
33	61	0,9821	
34	44	0,9856	
35	73	0,9777	
36	80	0,9754	
37	79	0,9755	
38	37	0,97	
39	74	0,9742	
40	90	0,9686	
41	78	0,9716	
42	109	0,963	
43	95	0,9655	
44	122	0,9568	
45	121	0,957	
46	139	0,9491	
47	175	0,9392	
48	218	0,9246	
49	260	0,9104	
50	224	0,9145	



### Images tests, PDF, photo...

text	Avant		Après			Diff	Diff %	Moy diff	2,52128483
1	175	0,26	158	0,36		0,27777778	27,777778		
2	322	0,83	27	0,98		0,15306122	15,3061224		
3	21	0,97	42	0,96		-0,01041667	-1,04166667		
4	22	0,96	19	0,97		0,01030928	1,03092784		
5	78	0,98	92	0,97		-0,01030928	-1,03092784		
6	24	0,989	22	0,99		0,0010101	0,1010101		
6_2	68	0,975	100	0,968		-0,0072314	-0,7231405		
6_3	37	0,986	142	0,96		-0,02708333	-2,70833333		
7	35	0,986	25	0,991		0,00504541	0,50454087		
7_2	22	0,992	71	0,979		-0,01327886	-1,3278856		
8	220	0,95	187	0,96		0,01041667	1,04166667		
8_2	179	0,96	260	0,93		-0,03225806	-3,22580645		
9	47	0,986	49	0,984		-0,00203252	-0,20325203		
10	7	0,987	8	0,985		-0,00203046	-0,20304569		

### Image prise avec la Picamera

Text	Avant		Après			Diff	Diff %	Moy diff	7,2010606
1	86	0,976	25	0,991		0,01513623	1,5136226		
2	78	0,978	37	0,988		0,01012146	1,01214575		
3	87	0,975	46	0,985		0,01015228	1,01522843		
4	61	0,928	27	0,99		0,06262626	6,26262626		
5	520	0,856	44	0,984		0,1300813	13,0081301		
6	532	0,85	33	0,988		0,13967611	13,9676113		
7	513	0,85	28	0,989		0,14054601	14,0546006		
8	287	0,922	31	0,989		0,0677452	6,77451972		

### Résultats en fonction des résolutions

Résolutions différentes : image=jour04, seuil=20

[3280,2464] (native): 25, 0.9915

60.01s

[1920,1080]: 88, 0.9734

19.5s

[1280,720]: 1721, 0.3519

13.78s

[720,480]: trop faible