



MESURE DE VARIABILITÉ DES ENVIRONNEMENTS SONORE DE JEUX VIDÉO

Thomas DEBEUGNY

Mémoire de Master 2 - Spécialité Son

Directeur de mémoire interne : Laurent MILLOT

Responsable universitaire : Corsin VOGEL

Rapporteur.e.s : Alan BLUM

Novembre 2021

Résumé :

La variabilité sonore est une problématique récente dans l'industrie du jeu vidéo. Les ambitions grandissantes de ce média en terme de réalisme et d'immersion font qu'un soin particulier est apporté à la diversité des décors, des moyens d'interaction, des effets visuels, mais aussi à la variabilité des réponses et des accompagnements sonores. Ce mémoire se propose d'étudier les différents moyens utilisés dans l'industrie pour créer le renouvellement des environnements sonores. L'objectif à terme est de pouvoir mesurer et quantifier leur variabilité, en lien direct avec la sensation de répétition ou de variation à l'écoute de ces ambiances. En utilisant des enregistrements de sons d'ambiances générés directement par les jeux issus de notre corpus, nous étudierons plusieurs techniques de traitement du signal.

Dans un premier lieu, nous utiliserons les indications visuelles des spectrogrammes en lien avec l'écoute des ambiances pour détecter des indices sur la composition des environnements sonores. Ensuite nous utiliserons le procédé d'intercorrélation pour trouver les répétitions d'un même son au sein d'une ambiance. Enfin nous nous intéresserons à l'utilisation du profil matriciel et de la bibliothèque Stumpy pour tester un détecteur automatisé d'information sur les signaux temporels.

Mots clefs : Jeux vidéo, ambiance, immersion, traitement du signal, variabilité.

Abstract

Sound variability is a quite recent problematic in the video game industry. Many developers care about generating diversity to landscapes, interactions, visual effects as well as sounds, due to growing ambitions for realistic and immersive experiences. This study reviews the multiple tools used in the industry to create variability within a soundscape. The aim is to eventually create a tool able to measure this variability in relation with the sensation of diversity throughout the listening of those soundscapes. We will study several signal processing approaches by using records of soundscapes captured within the chosen games.

First we will use spectrograms and their visual indications to understand how those soundscapes are composed, then we will use the intercorrelation technique to find the repetitions of a same sound within a soundscape. Finally, we will examine how to use the matrix profile and the Stumpy library to create an automatic data mining program practicable on our soundscapes.

Key words : Video games, soundscape, immersion, variability signal processing,

Remerciements

Je tiens à remercier chaleureusement Laurent Millot pour son investissement, son écoute et sa disponibilité tout au long de l'écriture de ce mémoire. Ces conseils sur la rédaction et sur l'utilisation du langage Python, ainsi que sa contribution au codage de l'analyse IDS et de l'intercorrélation furent d'une grande aide.

Merci à Corsin Vogel pour ses conseils.

Merci à Coline, Renaud, Céline et Nico pour leur présence.

Merci à ma famille pour leur soutien indéfectible.

Introduction

L'immersion est une composante essentielle du jeu vidéo. L'interactivité qui fait l'unicité de ce média souligne souvent ses capacités immersives sans précédent. Au fil des progrès techniques provoquant son impressionnante évolution, ce média s'est à de nombreuses occasions approché d'une conception plus réaliste de la représentation audiovisuelle. Le jeu vidéo s'est ainsi beaucoup approché de son parent, le cinéma, en s'inspirant de la complexité visuelle et sonore du réel.

Or le jeu vidéo est un média vaste et ces propositions sont nombreuses et diverses. Tandis que certains jeux continuent à proposer des expériences héritées de l'arcade, où concentration, finesse d'exécution et réflexes sont de rigueur, on observe aujourd'hui des jeux qui, au contraire, se développent sur un temps plus long, où l'exigence du jeu se concrétise plutôt par de la prise de décision stratégique et organisationnelle. En outre, on trouve également de nombreux jeux proposant des expériences où l'exploration, la déambulation, voire la flânerie sont possibles, si ce n'est encouragées. Il semble que de nombreux jeux proposent un mélange de toutes ces façons de jouer. Un jeu de rôle occidental typique exigera réflexes et doigté lors de ces phases de combat, stratégie et réflexion pour la création et l'évolution de son personnage et prendra soin d'offrir au joueur de magnifiques décors à parcourir entre deux quêtes.

Le jeu vidéo est donc pluriel mais évolue systématiquement dans une sphère audiovisuelle interactive. Cette même interactivité entraîne de fait une multiplication des possibilités à chaque partie. La tournure que prennent les événements étant dépendante de notre bon vouloir, dans les limites imposées par le jeu, il n'est pas rare de voir des joueurs passant énormément de temps sur un même jeu, que ce soit par souci de perfectionnement, de découverte ou de complétionnisme. Ce fait bien connu de l'industrie entraîne les éditeurs et développeurs à penser leurs jeux de façon à ce qu'ils soient « rejouables », autant dans l'aspect interactif que dans l'aspect audiovisuel.

La multitude des propositions ludiques, l'importance de la rejouabilité et les ambitions réalistes du média ont profondément modifié les caractéristiques audiovisuelles des productions vidéoludiques, notamment la façon dont on imagine l'univers sonore des jeux que l'on crée.

La particularité de la composition d'une bande son de jeu vidéo provient directement des contraintes de ce média. Tandis que la non-linéarité impose de pouvoir entretenir l'espace sonore pour des temps potentiellement infinis, les limites de poids des jeux dû aux supports ou à la volonté des éditeurs imposent souvent une quantité restreinte de matière sonore au regard de cette problématique. De plus, avec l'ambition grandissante des développeurs et le perfectionnement des outils technologiques, les exigences en matière d'immersion sonore des univers vidéoludiques n'ont cessé de s'accroître. L'importance de la variabilité sonore est désormais prit en considération dans de nombreux jeux. Autant concernant la musique, les ambiances ou les effets, on attache de l'importance à créer une multitude d'interventions sonores possibles afin de créer une couverture sonore constamment renouvelée et adaptée à la non-linéarité du jeu. Une façon pour les développeurs d'effectuer un nouveau pas dans la quête de réalisme et de richesse immersive, tout en offrant aux joueurs un contenu renouvelable et donc bien moins lassant pour qui désirerait s'attarder dans ces univers.

Ce mémoire propose de s'intéresser à la variabilité des sons d'ambiance dans les jeux vidéo en monde ouvert. Pour ce faire nous aurons recours à divers outils nous permettant d'analyser la formation des bandes sonores de jeu vidéo. A partir d'un corpus d'environnements sonores issus des jeux sélectionnés, nous comparerons l'efficacité de ces outils d'analyse dans l'objectif de pouvoir mesurer et quantifier la variabilité des environnements sonores au sein d'un jeu vidéo.

Table des matières

Mesure de Variabilité des Environnements Sonore de Jeux Vidéo.....	1
Résumé :.....	3
Abstract.....	4
Remerciements.....	5
Introduction.....	6
Chapitre 1 - Le son dans le jeu vidéo.....	10
A – Une industrie en évolution.....	10
1. L'industrie du jeu vidéo de 1972 à aujourd'hui.....	10
2. Evolution du son dans le jeu vidéo.....	16
B – Problématiques liées au son.....	18
1. La non linéarité.....	18
2. Quelques critères d'appréciation des jeux aujourd'hui.....	21
3. Définition environnements sonores.....	24
C – Recherches préliminaires.....	26
1. Moteurs Audio.....	26
2. Techniques de variabilités sonores.....	26
Chapitre 2 - Mesures Préliminaires.....	30
A – Analyse Spectrogramme.....	30
1. Choix des jeux et ambiances.....	30
2. Écoute et Analyse du spectrogramme.....	37
B – Utilisation de l'intercorrélation entre les ambiances et les sons élémentaires.....	60
1. Analyse IDS.....	60
2. Intercorrélation.....	62
3. Autocorrélation de l'intercorrélation.....	71
4. Perspectives.....	78
Chapitre 3 - La Bibliothèque Stumpy.....	80
A – Introduction.....	80
1. Présentation générale.....	80
2. Le Profil Matriciel.....	81
B – Application au corpus d'ambiances sonores.....	91
1. Détection de similarités et/ou d'anomalie.....	91
2. Segmentation Sémantique.....	100
3. Détection de motif connu.....	104
4. Perspectives.....	107
Conclusion.....	109
Annexes.....	111
Annexe 1 : Enregistrement des ambiances des jeux.....	111
Annexe 2 : Extractions des sons des fichiers des jeux.....	115
Bibliographie.....	116

Chapitre 1 - Le son dans le jeu vidéo

A – Une industrie en évolution

1. L'industrie du jeu vidéo de 1972 à aujourd'hui

L'industrie du jeu vidéo représente aujourd'hui l'un des plus gros marchés du divertissement audiovisuel. En 2019, le chiffre d'affaires réalisé par celle-ci était de 120 milliards de dollars. A titre de comparaison, l'industrie du cinéma a, elle, généré 42,5 milliards de dollars. En France, le chiffre d'affaires du jeu vidéo représente 5,3 milliards d'euros en 2020, en nette croissance par rapport aux années précédentes. [10]

Repoussant sans cesse ses limites grâce aux progrès technologiques, le jeu vidéo est passé au fil des années de gadget numérique à produit culturel important.



Figure 1: Borne d'arcade Street Fighter

Ce médium a su trouver un public de plus en plus varié en proposant diverses façons de consommer ses produits. Depuis sa démocratisation en 1972 avec le fameux *Pong*, les jeux étaient alors principalement consommés sur bornes d'arcade. Jusque dans les années 80, on observait une appréhension plus compétitive dans l'interaction proposée, avec des jeux à confrontation directe (*Pong*, *Street Fighter*...) ou des jeux à score (*Pacman*, *Space Invaders*,...)

C'est avec l'essor des consoles de salon suivant l'arrivée de la Famicom en 1983 au Japon et la NES, son équivalent européen sorti en 1985, que l'entreprise Nintendo dévoile au grand public une nouvelle façon de consommer le jeu vidéo. C'est avec des jeux comme *Super Mario Bros* (1985) ou *The Legend of Zelda* (1986) que beaucoup découvrent dans le jeu vidéo un potentiel narratif plus développé, couplé à une réalisation audiovisuelle efficace et évocatrice servant la narration. On propose aux joueurs un univers nouveau, souvent enchanteur, à explorer.



Figure 2: *The Legend Of Zelda* - Nes - Ecran titre

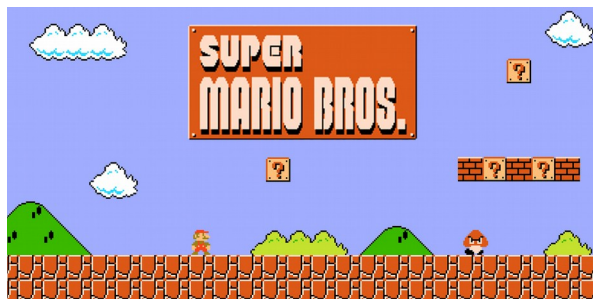


Figure 3: *Super Mario Bros* - Nes - Ecran titre

Les limites techniques présentes à l'époque dans la création d'espaces audiovisuels virtuels n'ont pas empêché les premiers artisans du jeu vidéo de développer avec talent des images esthétiques et des musiques d'une richesse indéniable. Cette créativité, s'imposant face aux contraintes particulières de ce médium, a d'ailleurs inspiré de nombreux artistes, reprenant des codes similaires pour réaliser des œuvres picturales dites en *Pixel Art* (art du pixel), ou de la musique 8-bit aussi appelé *Chiptune* (De *Chip* pour « puce informatique » et *tune* pour « mélodie ») . (figure 4)



Figure 4: ©CamilleUnknown

La perspective d'une immersion dans un univers riche par le biais d'un avatar était née, et les progrès techniques n'ont cessé d'aller dans ce sens depuis. La promesse d'un univers toujours plus grand, aux graphismes plus époustouffants, regorgeant de détails, avec un niveau de richesse proche du réel dans la diversité des paysages explorables ou dans les interactions avec l'environnement, représentait alors un cap à suivre pour certains développeurs, et un rêve pour certains joueurs.



Figure 5: Metal Gear Solid - Playstation

L'avènement progressif de la 3D, jusqu'à sa démocratisation a permis un grand pas en avant dans la promesse d'une immersion toujours plus forte. Notamment avec l'arrivée des consoles de jeux 32 et 64 bits, comme la Playstation avec des jeux comme *Tomb raider* (1996) ou *Metal Gear Solid* (1998) ou la Nintendo 64 avec les fameux *Super Mario 64* (1996) ou *The Legend of Zelda, Ocarina of Time* (1998).

Depuis, l'industrie s'est développée à une vitesse considérable. La demande des joueurs pour des jeux ambitieux dans leurs propositions ludiques et immersives a encouragé les investisseurs. Par conséquent, la plupart des jeux dits AAA (« triple A », équivalent du blockbuster dans le monde du jeu vidéo, regroupant les projets les plus ambitieux financièrement) de ces 10 dernières années sont des jeux 3D à la première ou troisième personne, proposant l'incarnation d'un avatar dans un univers plus ou moins ouvert, avec une exigence de plus en plus forte dans la richesse des environnements. Aujourd'hui, grâce aux progrès techniques, les univers vidéoludiques des plus réalistes aux plus originaux peuvent se doter d'une représentation audiovisuelle d'une richesse remarquable au regard de la relative jeunesse de l'industrie. On peut citer comme exemple parmi les plus grosses productions de ces dernières années *The Legend of Zelda, Breath of the Wild* (2017), *Grand Theft Auto V* (2013), la série *Assassin's Creed* (2007 – 2020), ou *Death Stranding* (2019)

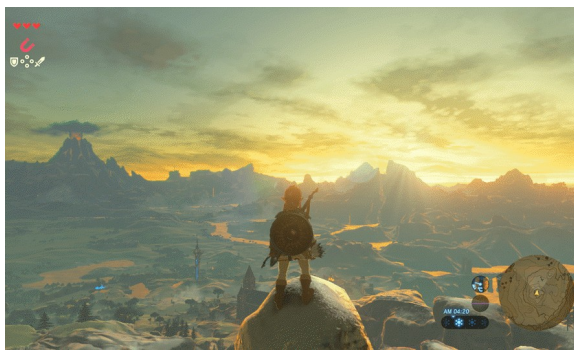


Figure 6: *The Legend Of Zelda: Breath of the Wild* - Switch



Figure 7: *Death Stranding* - PC

2. Evolution du son dans le jeu vidéo

Les capacités techniques d'implémentations sonores dans les jeux vidéo ont, elles aussi, connu une grande évolution depuis leurs débuts. *Pong*, avec ces quelques sons synthétiques accompagnant les rebonds de la balle en témoigne. L'austérité des propositions sonores des premiers jeux peut faire sourire au regard de la qualité des productions actuelles, mais on ne peut que souligner l'incroyable maturation qu'a subi ce médium malgré sa relative jeunesse.

Avant l'avènement du support CD, les sons des jeux étaient bien souvent générés grâce au format MIDI avec une unique piste sonore. Il était en effet impossible d'utiliser des échantillons de sons numérisés directement dans des disquettes ou des cartouches de jeux, notamment à cause de l'espace mémoire de ces supports trop limités. Avant les années 80, les univers sonores des différents jeux étaient alors souvent limités à quelques fonctions essentielles, notamment les feedbacks (indices sonores indiquant au joueur la pertinence ou la réussite de son action), les introductions ou les *game over*. [4]

La Famicom et la NES ont marqué une première révolution en intégrant une puce spécialement dédiée à la synthèse sonore, et en permettant de générer plusieurs pistes sonores simultanément. Les jeux se dotent alors à la fois de bruitages et de musiques, prouesse pour l'époque qui assiste à la naissance de certains des thèmes musicaux les plus emblématiques du médium (Mario, Zelda, Tetris...). Un système similaire est reproduit sur les autres consoles de la génération 8-bit, comme la Gameboy, la Master System ou la Mégadrive. [4]

Cependant, l'un des défauts récurrents des jeux de l'époque, au regard de l'évolution du médium, reste la lassitude ressentie par le joueur subissant la répétition d'un motif musical trop court ou un effet sonore trop répétitif. Bien que certains compositeurs et/ou sound designers aient su relever le défi avec talent, beaucoup demeuraient contraints par l'espace

limité dédié à la programmation sonore dans le poids total du jeu et par les limites imposées par les techniques de synthèse de l'époque. Les différents types de sons générés étaient souvent simplement des formes d'onde basiques (carré, triangle, dents de scie) dont la fréquence pouvait être modifiée, ou des bruits. L'incapacité technique d'avoir une réelle précision dans le modelage des sons imposait souvent aux jeux un accompagnement quasi exclusivement musical, avec quelques effets sonores ponctuels.

La démocratisation du CD avec les consoles 32 et 64 bits a permis d'utiliser les techniques d'échantillonnage et de numérisation du son dans le jeu vidéo. Cela a également permis d'augmenter drastiquement l'espace de stockage disponible. Ainsi, avec l'augmentation de la vitesse de calcul numérique des processeurs, de nombreux studios ambitieux se sont appropriés ces nouvelles technologies pour les mettre au service des jeux et de leurs univers sonores. On assiste à l'utilisation de sons et de musiques préalablement enregistrés, permettant au médium de s'affranchir des sons caractéristiques de ses débuts (les fameux *Blip-blop*) pour proposer une plus grande diversité dans sa dimension audio. On voit naître des jeux qui ne sont plus uniquement accompagnés de musique, mais également de sons d'ambiances et d'effets sonores plus proches d'une conception réaliste des univers dépeints. L'utilisation de voix intelligible fut également grandement facilitée. Toutes ces évolutions semblent marquer le point de départ de l'ambition animant encore aujourd'hui le jeu vidéo à se rapprocher de son parent, le cinéma.

Depuis les années 2000 et jusqu'à aujourd'hui, le développement des moteurs de jeux vidéo a grandement contribué à l'amélioration de la diversité et de la richesse sonore. Grâce aux moteurs comme Wwise ou Fmod, prenant toute la charge de la partie audio des jeux, de la spatialisation à la variabilité, de la synthèse à la gestion de fichiers, du traitement à la hiérarchisation des sons, les sound designer possèdent aujourd'hui des outils puissants pouvant répondre à de nombreuses demandes quel que soit le type de jeux.

B – Problématiques liées au son

1. La non linéarité

Le jeu vidéo comme œuvre audiovisuelle est souvent comparé au cinéma. Son évolution au fil des progrès technologiques comme vu précédemment a été en partie encouragée par une volonté d'arriver à produire un cinéma interactif. Une qualité audiovisuelle s'approchant du réel, mais avec une possibilité d'interagir directement avec l'univers dépeint, et ainsi de plonger plus intensément dans un univers différent. Les similitudes se ressentent aussi dans le rythme de la narration ou les choix ludiques, il est assez clair que le cinéma a une grande influence dans les productions AAA de ces 10 dernières années.

Le lien qui lie ces deux médiums se resserrant d'année en année, on observe également un phénomène inverse. Le cinéma contient dans certaines œuvres des inspirations héritées du jeu vidéo. Outre les univers vidéoludiques originaux adaptés au cinéma (*Final Fantasy*, *Silent Hill*, *Tomb Raider*, *Warcraft* et bien d'autres), certains concepts propres au jeu vidéo servent parfois de base à la création d'un univers cinématographique original. On peut penser à *Edge of Tomorrow* (2014), réalisé par Doug Liman, reprenant le concept de *Die and Retry*, mécanisme de jeu exigeant du joueur la parfaite connaissance du déroulement du jeu et une exécution précise pour terminer un niveau. Dans un autre registre, *Notre Amour Est Assez Puissant*, réalisé par Jonathan Vinel est un court-métrage directement filmé dans un univers de jeu vidéo en trois dimensions.



Figure 8: Extrait de Notre Amour est Assez Puissant - Jonathan Vimel

Néanmoins, l'interactivité qui fait son unicité contient à elle seule de nombreuses problématiques propres à ce médium. Ces mêmes problématiques sont prises en compte lors de la fabrication du jeu, et impactent à différentes échelles chaque aspect du design d'un jeu vidéo.

L'une des conséquences de l'interactivité est la non-linéarité temporelle. Contrairement à un film, les jeux n'ont pas une durée précise. La durée qu'un joueur prendra à terminer un jeu est variable et dépend d'un grand nombre de paramètres. On ne sait que rarement anticiper combien de temps un joueur va rester dans un état donné, combien de temps va-t-il prendre à traverser un lieu, un couloir, un biome, et quels actions va-t-il effectuer durant son passage. Bien que les level-designers (concepteur des niveaux du jeux) aient toujours un certain contrôle dans le cheminement du joueur notamment de par l'architecture des niveaux, l'interface ou d'autres éléments de design, il reste néanmoins une part d'imprévisibilité venant du joueur qui est propre au médium.

En conséquence directe, chaque environnement sonore doit être adaptable en longueur temporelle pour couvrir l'intégralité du temps que le joueur pourra passer dans cet

environnement, si ce n'est un temps théoriquement infini. Prenant conscience de cette problématique, les jeux se sont d'abord contentés de boucler infiniment les pistes sonores pendant une partie, comme Tetris et sa reprise 8-bit de *Casse Noisette* de Tchaïkovski. Mais le jeu vidéo se voulant de plus en plus immersif, il est devenu crucial que l'expérience ne soit pas accompagnée d'une unique boucle sonore figée et facilement perceptible. D'autre part, les interactions effectuées par le joueur au moyen d'une pression sur un bouton se doivent généralement d'être accompagnées ou illustrées d'un effet sonore. Et il est indispensable, dans certaines productions, que ces actions aient une réponse sonore cohérente et non répétitive. En sachant que la répétition d'un même son sans modification peut être facilement identifiable par le joueur (plus ou moins selon le type de son), un jeu décrivant un univers avec une ambition réaliste dans son immersion ne peut se permettre d'offrir toujours le même son en réponse à une action similaire. Le jeu tendra alors à s'approcher du réel, où rares sont les actions dont la réponse sonore est exactement similaire à chaque itération (sauter, marcher, utiliser un outil, etc.).

Il est donc nécessaire pour la cohérence de l'univers que chaque environnement et chaque action ou interaction contiennent une certaine richesse sonore qui les accompagne ou les illustre. Ainsi l'immersion est maintenue et l'aspect réaliste de la proposition vidéoludique reste plausible.

2. Quelques critères d'appréciation des jeux aujourd'hui.

La presse spécialisée en jeu vidéo évalue souvent ces produits non seulement sur leurs qualités ludiques et techniques, mais également sur leur durée de vie et/ou leur rejouabilité. Ces deux termes, bien que l'on préférera l'un ou l'autre selon le type de jeux, traduisent la même idée : la capacité d'un jeu à nous maintenir dans son univers en proposant au joueur plus de contenu à découvrir ou de défis à relever.

Il semble que de nombreux joueurs consomment relativement peu de jeux différents bien que de plus en plus de jeux soient fabriqués et rendus disponibles tous les jours. Cela peut s'expliquer par le prix des jeux. Généralement, un jeu AAA à sa sortie coûte aux alentours de 60 €. D'autre part, sa dimension ludique et son appellation de jeu explique qu'on puisse consacrer plus de temps à un seul de ces produits¹, contrairement aux autres produits culturels que sont les livres ou les films par exemple. Cela peut expliquer la position particulière de la durée de vie et de la rejouabilité comme critère d'appréciation en comparaison à d'autres produits culturels. [14]

Il semble donc établi dans le monde du jeu vidéo que la durée de vie et la rejouabilité sont importantes dans l'appréciation d'un jeu. Autant commercialement dans la publicité² que directement dans la proposition ludique. Les « rogue-lite », ou « rogue-like » par exemple, sont des types de jeux se caractérisant par une rejouabilité énorme grâce à un contenu sans cesse renouvelé par l'utilisation contrôlée d'aléatoire. [16] Les jeux en monde ouvert proposent souvent un univers explorable considérablement vaste, regorgeant de détails, pour une durée de vie estimée pouvant dépasser la centaine d'heures. C'est également le cas pour tous les jeux compétitifs où la rejouabilité est similaire à la pratique régulière d'un sport, où chaque partie est différente au vu des actions et réactions entreprises par les joueurs et leurs adversaires.

Il est néanmoins bon de souligner qu'on observe plus récemment, notamment dans les sphères dites « indépendantes » de l'industrie, un certain attrait pour des expériences vidéoludiques plus courtes, plus condensées, et plus intenses dans leurs discours, dans leur narration ou dans leur proposition ludique. Ces propositions permettent de fournir aux joueurs des expériences marquantes et plus accessibles à des productions moins onéreuses. Ces jeux se consomment alors plus ponctuellement, similairement à un film ou un livre. D'autre

1 «Une dimension nouvelle de l'expérience du jeu s'ouvre : [...] jouer avec un univers qui offre la possibilité extraordinaire d'une répétition à l'infini. « Et si j'avais fait ceci plutôt que cela, ce geste plutôt qu'un autre ? » Si je ne suis pas satisfait, je peux toujours recommencer.» Mathieu TRICLOT, 2017 [14]

2 La publicité télévisée de *The Legend of Zelda : A link to the past* (1991) venait déjà sa durée de vie avec le slogan : « Finirez vous un jour Zelda 3 ? »

part, certains jeux assument la mise en avant de l'aspect narratif au détriment de l'aspect ludique, et donc de la durée de vie apportée par un contenu ludique riche.

Les studios, espérant produire un jeu avec une grande durée de vie ou rejouabilité, prennent généralement soin d'éviter les redondances et les répétitions autant dans les interactions et dans les possibilités ludiques de l'univers que dans les graphismes et l'habillage sonore qui le constituent.

3. Définition environnements sonores

Nous nous proposons dans ce mémoire d'étudier comment les jeux produisent des habillages sonores variables et renouvelables aux divers environnements qui les constituent. Que ce soit dans le but d'améliorer la rejouabilité, par exemple pour les jeux compétitifs, ou d'offrir une plus grande durée de vie et éviter la lassitude des jeux dotés d'un univers explorable riche et vaste.

Les environnements sonores ou ambiances, sont les toiles de fond diégétiques sur lesquelles l'action sonore du jeu prend place. Une ambiance dans un jeu désigne l'habillage sonore d'un lieu donné. Bien que variant selon l'heure, le climat ou le contexte, l'ambiance représente un tissu homogène qui est fondamentalement lié à un lieu, aussi fantasque soit-il. Tout lieu possède une ambiance, et même l'absence de sons d'ambiance constitue en tant que tel un environnement sonore particulier.

Dans le cas particulier d'un jeu, on peut définir l'ambiance sonore comme l'ensemble des sons attachés à un lieu dont le déclenchement n'est pas lié à une quelconque action du joueur autre que sa simple présence dans le lieu. Le vent dans les arbres d'une forêt ainsi que la faune la peuplant, le brouhaha d'une foule sur une place urbaine entourée de bruits de circulations, etc. L'ambiance sonore s'oppose donc aux *feedbacks* et *signes* sonores³.

Bien que la musique évolutive appliquée au jeu vidéo représente un champ d'études très intéressant en lien avec la variabilité sonore, son caractère souvent extradiégétique l'exclut de notre étude. Mais si elle est présente dans la diégèse du jeu, elle peut très bien être constitutive d'une ambiance dans certains lieux particuliers comme une boîte de nuit, un supermarché diffusant de la musique, etc.

3 Les feedbacks et signes sonores sont des effets sonores liés au gameplay du jeu. Les feedbacks indiquent au joueur la réussite ou l'échec d'une action, les signes donnent des indications sur l'état du joueur.

La caractérisation sonore d'un lieu par l'ambiance est essentielle à l'immersion. Elle permet de différencier les environnements explorables en donnant à chacun une identité propre. D'autre part, elle participe à la caractérisation de l'univers. En lui donnant un aspect inquiétant ou chaleureux, animé ou serein, elle participe à la transmission d'une émotion qui va renforcer le propos du jeu. Enfin, elle est essentielle à l'immersion de par sa capacité à englober l'auditeur dans un univers allant au-delà des limites de son écran. [3]

On peut décomposer les ambiances sonores en deux types de sons : les sons continus et les sons isolés. [3]

Les sons continus représentent des éléments constants de l'ambiance, comme un fond d'air d'une pièce en intérieur, un brouhaha d'une foule, un grondement d'une machine ou du vent dans les arbres. Plus longs et plus diffus, ils constituent la base de l'ambiance sonore, assurant sa continuité et sa cohérence.

Les sons isolés représentent quant à eux des éléments ponctuels et précis émergeant des sons continus. Ils habillent et donnent du relief à l'environnement sonore. Cela peut être le cri d'un animal dans un milieu rural, un klaxon dans un milieu urbain, un train qui passe, le tonnerre d'un orage, ou n'importe quel son précis et reconnaissable émergeant d'un tapis sonore plus diffus.

La variabilité des sons d'ambiance est essentielle pour procurer une richesse dans le paysage audiovisuel de l'œuvre, éviter la lassitude et maintenir le joueur immergé dans l'œuvre. Nous allons maintenant étudier les techniques permettant d'augmenter parfois drastiquement le nombre de réponses sonores différentes à un même évènement avec un nombre limité de sons utilisés. Ces techniques compilées dans les moteurs audio permettent de limiter la part du son dans le poids du jeu tout en diversifiant les réponses sonores possibles.

C – Recherches préliminaires

1. Moteurs Audio

Nous nous proposons maintenant d'observer les différents moyens mis en œuvre dans la composition d'un environnement sonore pour que celui-ci puisse correspondre aux critères de richesse sonore exigés par certains jeux. Pour cela, nous observerons dans un premier temps les outils proposés par les moteurs audio.

Bien que certains studios de développement utilisent des moteurs créés en interne par leurs soins, de nombreux jeux bénéficient de l'apport de logiciels extérieurs pour s'occuper de la partie audio d'un jeu. Parmi eux, les plus utilisés semblent être Wwise et Fmod. Ces logiciels ont pour objectifs de restituer ou de créer les éléments sonores d'un jeu, et de composer l'univers sonore au moyen de leurs nombreux outils. Ces fonctionnalités sont multiples, allant de la création de réverbération modelée selon l'espace du jeu, à la spatialisation précise des différentes sources dans l'univers sonore 3D ou encore à la création de musiques ou de séquences sonores évolutives à partir de son importé ou synthétisé. Ces logiciels proposent également de nombreux outils permettant l'augmentation de la variabilité des événements ou des environnements sonores disponibles.

Dans le cadre de ce mémoire, nous nous concentrerons sur Wwise, moteur audio développé par Audiokinetic, entreprise canadienne fondée en 2000. Wwise est utilisé aujourd'hui pour de nombreux types de jeux aux styles, aux univers et aux budgets très variés. Il est resté fréquemment utilisé dans les productions récentes. [2]

2. Techniques de variabilités sonores

Wwise possède différents outils pour offrir une réponse sonore variée à une même action ou une même situation. [2]

L'utilisation de l'aléatoire pour modifier les paramètres de lecture d'un son, par exemple : sa hauteur (pitch), son volume ou la sensibilité d'un effet de filtrage particulier. Wwise permet à l'utilisateur de générer aléatoirement la valeur de nombreux paramètres de lectures de manière contrôlée, en imposant des limites aux valeurs que peut prendre chaque paramètre. Cet outil est très utile notamment pour la répétition de sons ponctuels.

L'aléatoire intervient également dans le choix du son à jouer. Wwise peut utiliser des *Random Containers* (Conteneurs aléatoires) contenant plusieurs sons différents, mais sémantiquement équivalents. Une fois appelé, le conteneur va jouer un de ces sons de manière aléatoire. Cette technique est très utilisée associée à la précédente pour des sons répétitifs et facilement reconnaissables, comme les bruits de pas de l'avatar par exemple. D'autre part, l'outil *Shuffle* présent avec les *Random Containers* permet d'éviter la répétition du même son à l'itération suivante. Ainsi, l'aléatoire reste contrôlé pour éviter que le joueur détecte une répétition.

Ces deux techniques peuvent également être associées au *Sequence Container* (Conteneur de séquence), un objet contenant plusieurs sons qui vont être joués selon une séquence précisée à l'avance. Une playlist (liste de lecture) précise se jouant selon le déroulement du jeu. En associant cette technique avec les deux précédentes, on peut alors diviser chaque son en plusieurs grains et associer ces grains les uns avec les autres aléatoirement pour augmenter drastiquement les résultats possibles à l'appel d'un seul type de son. Cela permet d'avoir une grande richesse sonore associée à un événement en limitant le nombre de sons utilisés.

D'autre part dans le cas des ambiances, ou les différents types de sons ponctuels ne correspondent pas forcément à des événements visuels précis, Wwise permet d'appliquer un délai variable entre la répétition d'un même son.

Enfin, présentons deux derniers outils plus circonstanciels à savoir les *Switch containers* et les *Real Time Parameter Controls (RTPC)*.

Les *Switch containers* permettent de définir quel type de son sera joué selon une condition en jeu : un état, un paramètre ou une position. Prenons l'exemple des bruits de pas. Chaque pas doit avoir une réponse sonore différente selon la surface sur laquelle notre personnage marche ou court. Un pas en forêt sur un tapis de feuilles mortes ne répondra pas similairement à un pas sur une route bétonnée. Le *Switch* permet à partir du paramètre « le joueur marche sur du béton », d'associer les bruits de pas de l'avatar qui y sont associés. Un *Switch* peut être utilisé dans bien d'autres situations, dès qu'un paramètre du jeu peut servir à une réponse sonore particulière. Associé aux autres conteneurs comme le *Random Container*, il permet d'augmenter la précision des réponses sonores du jeu pour une immersion plus forte.

Les RTPC fonctionnent similairement à un Switch Container, en fonction d'un paramètre du jeu. Cependant, ils permettent de faire varier un paramètre quantifiable du son en fonction d'un paramètre quantifiable du jeu, notamment grâce à des représentations graphiques. On observe souvent l'utilisation de RTPC dans les jeux d'action, où à mesure que notre personnage est en danger, les sons environnementaux s'étouffent pour représenter la désorientation du personnage. Dans ce cas précis, le paramètre « santé du personnage » est inversement proportionnel au taux de filtrage des sons environnementaux. Les RTPC sont également utilisés selon la position du joueur pour faire évoluer la spatialisation des sons, ou dès qu'il y a nécessité d'une évolution sonore progressive et adaptable. C'est un outil puissant : à la fois pour l'immersion mais aussi pour transmettre au joueur des informations sur l'état de son avatar ou du monde qui l'entoure.

Malgré nos efforts, il n'a pas été possible d'obtenir des informations auprès des professionnels du milieu du jeu vidéo sur les éléments constitutifs d'une bande son de jeu

vidéo ni de savoir comment les outils présentés sont utilisés voire dans quelles proportions ils sont utilisés. Par conséquent, dans ce mémoire, nous avons essayé de réaliser une analyse détaillée de la composition des environnements sonores d'un jeu. Nous nous sommes concentrés sur l'objectif consistant à déceler l'utilisation de répétitions et de boucles dans la composition des environnements sonores pour plusieurs jeux plus ou moins récents ayant tous des ambitions immersives notables.

L'observation des résultats de ces analyses permettra de déceler quelles techniques sont employées et de quelle manière mais aussi de faire un comparatif rétrospectif de l'évolution de la richesse des environnements sonores au fil des différentes générations de jeux vidéo.

Chapitre 2 - Mesures Préliminaires

A – Analyse Spectrogramme

1. Choix des jeux et ambiances

Les jeux sélectionnés pour réaliser cette étude ont été choisis selon les critères suivants.

Il nous a semblé préférable de se concentrer sur les jeux se déroulant dans un univers en trois dimensions, de par l'importance particulière que prennent les sons d'ambiance dans ceux-ci. En effet, dans ces jeux, on retrouve bien souvent une composition des ambiances adaptées aux décors ainsi qu'une spatialisation fidèle à l'univers 3D dans lequel le joueur évolue. Comme, quel que soit l'univers dépeint, l'immersion dans l'univers fait souvent partie intégrante de l'expérience de jeu, un soin particulier est appliqué à la variabilité des ambiances. Les jeux de type A-RPG (Action Role Playing Games) ou FPS (First Person Shooter), sont *à priori* les plus intéressants à explorer de ce point de vue.

D'autre part, il est nécessaire que la carte du jeu soit relativement ouverte, c'est-à-dire que les différents niveaux et/ou environnements du jeu ne soient pas séparés par une étape hors de la diégèse du jeu, comme un menu ou un temps de chargement. Par ailleurs, nous n'avons pas étudié les jeux où le joueur est contraint dans ses actions par un temps limité ou se conformer à une chorégraphie précise. On s'est intéressé avant tout au renouvellement des ambiances et donc aux capacités qu'ont les jeux à offrir un environnement sonore riche, immersif et adaptable aux actions imprévisibles du joueur. Dans les jeux retenus, le joueur doit donc pouvoir conserver une certaine liberté.

De plus, pour les besoins de cette étude, nous avons sélectionné uniquement des jeux permettant d'isoler les sons environnementaux qui nous intéressent. Nous n'avons pas étudié de jeux comportant de la musique qui ne peut être coupée.

Les ambiances enregistrées et étudiées dans ce mémoire sont disponibles, pour une écoute, afin de permettre au lecteur d'essayer d'identifier par lui même les répétitions des ambiances.

Afin d'établir une analyse rétrospective du niveau de variabilité sonore des ambiances au fil des années, trois jeux de la même série, développés et édités par Bethesda Softworks ont été choisis : *The Elder Scrolls* 3, 4, et 5, nommés respectivement Morrowind, Oblivion et Skyrim. Chacun de ces jeux est un jeu de rôle de fantasy à la première personne avec une proposition ludique similaire, bien que modernisée à chaque nouvel opus : le joueur a la possibilité d'explorer un large espace en trois dimensions sans interruption spatiale ou temporelle. Chaque jeu est le représentant, à son époque, d'un pas en avant dans les capacités d'immersion du joueur dans un univers vaste et riche. Cette série a également beaucoup évolué, autant du point de vue de l'image que du son, en enrichissant ces aspects avec plus de détails et de finesse au fil des opus.

The Elder Scrolls III : Morrowind, troisième jeu de la saga *The Elder Scrolls*, sorti en 2002, se déroule sur une île de 24 km² appelée Vvardenfell, que l'on peut parcourir de long en large sans interruption. Les environnements y sont marécageux et inhospitaliers, parfois menaçants.

Le premier environnement sonore enregistré est un village situé dans un marécage à la faune étrange, le second est une grotte vide avec un feu de camp. (figure 9)



Figure 9: *Morrowind* - Capture d'écran des environnements enregistrés

The Elder Scrolls IV : Oblivion, sorti en 2006, se déroule sur un monde ouvert de 41 km², empli de vallées luxuriantes et de larges forêts. L'espace d'Oblivion est bien plus accueillant que celui de son prédécesseur, ressemblant davantage aux paysages continentaux européens naturels.

Les environnements sonores enregistrés sont un ensemble de collines situées entre un lac et une grande ville et de sombres couloirs d'égouts.(figure 10)



Figure 10: Oblivion - Capture d'écran des environnements enregistrés

Enfin, *The Elder Scrolls V : Skyrim*, dernier opus en date sorti en 2011, prend place dans un univers aux inspirations nordiques, avec de grandes montagnes enneigées et de larges toundras. L'univers explorable, légèrement plus petit que son prédécesseur (37 km²), reste néanmoins très riche s'agissant de la variété des environnements explorables au sein du même espace 3D.

Les ambiances étudiées sont celles d'une grotte sous une montagne enneigée et d'une forêt au pied de cette montagne. (figure 11)



Figure 11: *Skyrim* - Capture d'écran des environnements enregistrés

Afin d' étoffer notre analyse, nous nous sommes aussi intéressés à un jeu plus récent et reconnu pour ses prouesses techniques, *The Witcher 3* (2015), développé par CD Projekt RED.

Ce jeu de rôle et d'action se déroule dans un univers médiéval sombre et dangereux, similaire sur certains points à celui des jeux de la saga *The Elder Scrolls*. Ce jeu dépeint un monde large et riche (136 km²) dans lequel un soin particulier a été apporté aux environnements sonores et à leur implémentation.

Les ambiances choisies sont celles d'un village et d'une clairière, avec évolution de la météo et passage du jour et de la nuit. (figure 9)



Figure 12: *The Witcher 3* - Captures d'écran des environnements enregistrés

Pour observer et analyser les différents effets mis en œuvre au sein du moteur audio sur les sons employés, nous avons réalisé des enregistrements sonores des ambiances afin de pouvoir étudier leur déroulement et leur évolution.

Une version synthétique du protocole d'enregistrement est donnée en annexe afin de faciliter sa réutilisation. Ainsi qu'une explication de la façon dont les sons ont été extraits des fichiers des différent jeux analysés.

2. Écoute et Analyse du spectrogramme

Des premières observations des différents environnements sonores peuvent être faites assez rapidement à l'écoute et grâce au décryptage de leur spectrogramme.

Les spectrogrammes ont été calculés et tracés grâce au logiciel RX7 édité par Izotope, société développant des outils de mixage et de nettoyage audio. RX7 est un outil de nettoyage audio puissant beaucoup utilisé en post production cinéma. Nous nous en sommes servis principalement pour profiter de l'affichage dynamique des spectrogrammes, permettant d'effectuer facilement des zooms et des observations précises concernant la composition spectrale des sons analysés.

Les spectrogrammes considérés correspondent au calcul d'une succession de transformées de Fourier discrètes à court terme, transformées dont on trace à chaque fois le « spectre d'amplitude » (le module). Les transformées de Fourier discrètes ont été calculées pour des fenêtres temporelles de 2048 échantillons.

Afin de limiter la « pollution spectrale » associée à chaque composante fréquentielle potentiellement présente dans le spectre et limiter le masquage des composantes fréquentielles de faible poids par celles de fort poids, nous avons utilisé une fenêtre de pondération de Hann. Nous avons choisi la fenêtre de Hann, comme fenêtre de pondération, car elle réalise un compromis permettant de conserver une certaine finesse dans la localisation fréquentielle (lobe principal pas trop large pour une composante fréquentielle) tout en limitant sensiblement l'amplitude des lobes secondaires. Ce choix classique permet de limiter les erreurs d'interprétation concernant les bosses ou les pics présents dans le spectre d'amplitude et, ainsi, de limiter le risque de considérer un lobe secondaire comme le lobe principal d'une nouvelle composante fréquentielle.

Pour augmenter la capacité de détection des composantes fréquentielles, les spectrogrammes sont tracés avec une échelle logarithmique en décibels pour l'amplitude. Avec la palette des couleurs choisies, les amplitudes de 0 dB sont tracées en orange (blanc pour les versions en niveaux de gris), celles de -120 dB en cyan (noir pour les versions en niveaux de gris).

En abscisse, on trouve le temps indiqué en secondes. En ordonnée, on trouve la fréquence en Hertz avec une échelle non pas linéaire mais logarithmique car potentiellement plus en rapport avec le fonctionnement effectif de la perception auditive humaine.

Tous les spectrogrammes présentés dans la suite ont été tracés sous RX7 en utilisant systématiquement les mêmes réglages. Chacun de ces spectrogrammes comporte deux tracés disposés l'un en dessous de l'autre car les enregistrements d'ambiances sonores correspondent à des signaux stéréophoniques : la voie gauche est tracée au-dessus, la voie droite au-dessous.

Afin de faciliter leur observation, tous les spectrogrammes sont rassemblés au format png, et joints avec ce mémoire.

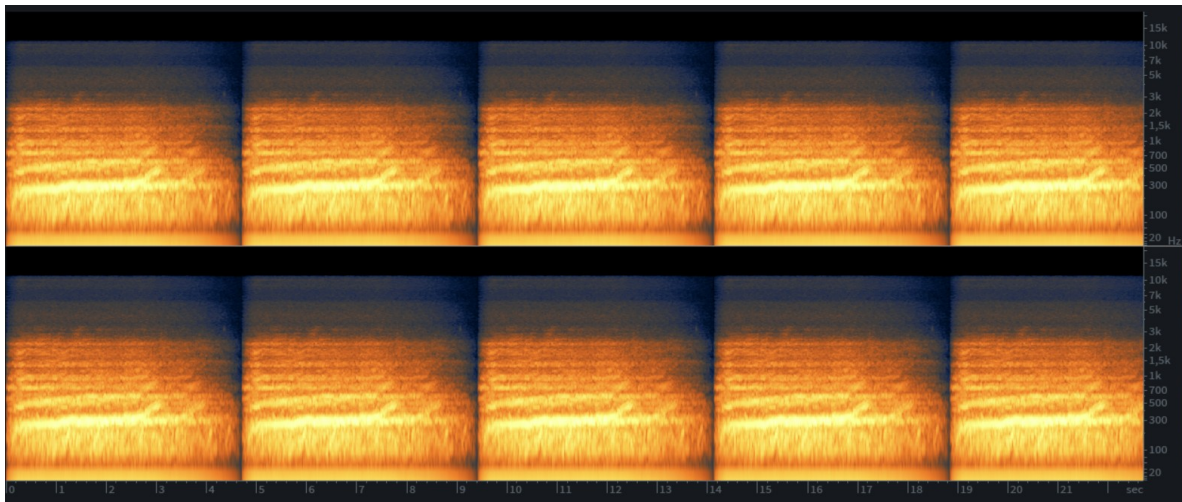


Figure 13: Exemple d’affichage du spectrogramme (durée : 22,8 s)

L’exemple de la figure 13 est représentatif des cas où on peut identifier des répétitions ou des changements rapides dans la composition sonore d’un simple coup d’œil. Mais, comme nous allons le constater, il ne représente qu’un cas caricatural (ou idéal) pour la détection des répétitions ou des changements.

Analyse d'ambiances tirées de Morrowind

La particularité de Morrowind dans les paysages sonores qu'il dépeint, est que chaque son est diffusé depuis sa source dans l'espace 3D, et l'on constate vite une absence totale de sons d'ambiance si l'on s'éloigne des éléments générateurs de sons. Il n'y a donc pas de fond d'air constant lié à l'environnement, seules les sources précises émettent des sons (cours d'eau, feu de camp, faune, etc.) qui s'atténuent jusqu'à disparaître si l'on s'en éloigne. Ces sources étant rarement assez nombreuses pour remplir l'espace, il arrive fréquemment que certains passages soient dénués de tout sons d'environnement. En fait, la musique, assez omniprésente, a pour rôle d'offrir cette toile de fond liant les différents événements sonores, mais il faut souligner que le jeu confronte parfois le joueur à une absence totale de sons d'ambiance diégétiques.

Lors des enregistrements, la détection de l'existence de répétitions était déjà possible à l'écoute. Les analyses des sons effectuées avec le logiciel RX7 permettent de confirmer la présence de répétitions comme nous allons le montrer.

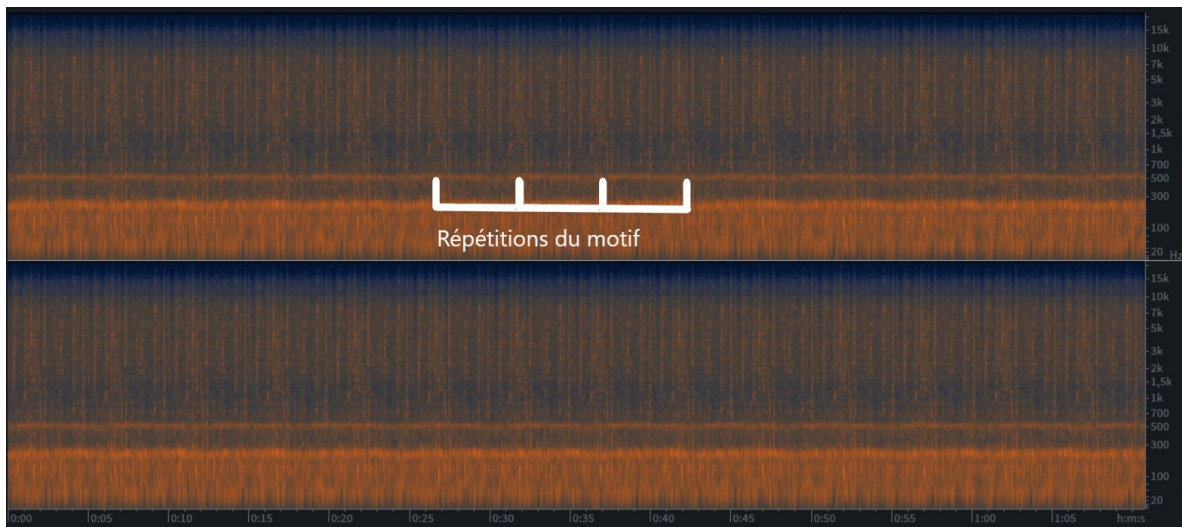


Figure 14: Morrowind - Grotte (durée du spectrogramme : 70 s)

Sons présents : crépitement du feu.

Pour le premier exemple (cf. figure 14), l'ambiance est totalement uniforme, dénuée de sons ponctuels. On entend exclusivement le crépitement du feu de camp.

Une écoute analytique permet de se rendre compte que l'ambiance est en fait constituée de la répétition en boucle d'un unique son durant 5 secondes, ce que confirme l'observation du spectrogramme.

Pour le deuxième exemple (cf. figure 15), l'enregistrement place le joueur entre un cours d'eau et une créature fantastique.

En observant le spectrogramme comme à l'écoute, on constate la présence d'un bouclage assez marqué. Les flèches blanches indiquent les cinq répétitions du cri unique de la créature, séparées à chaque fois d'une période fixe d'environ 25 secondes.

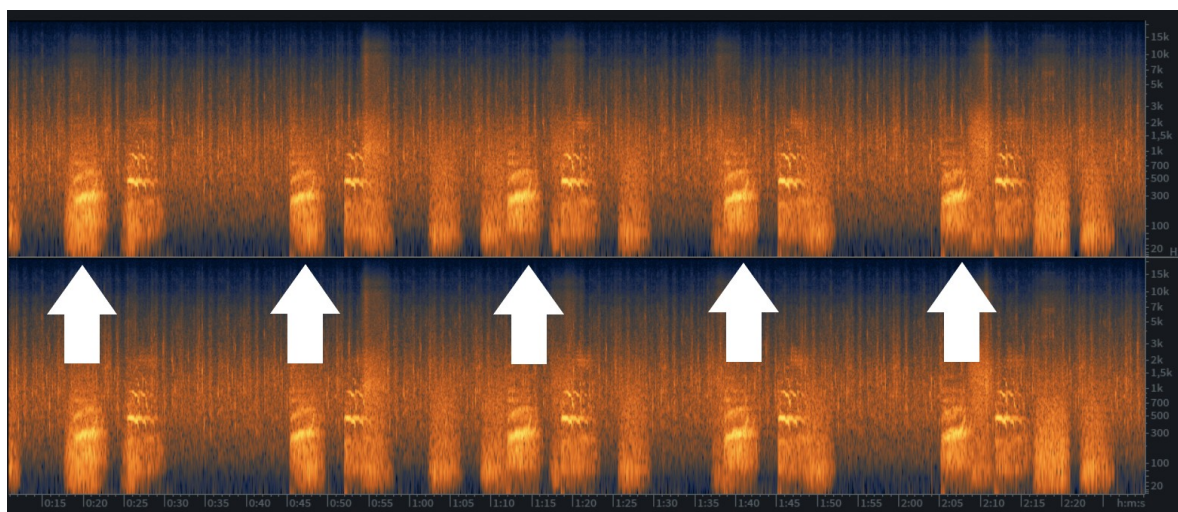


Figure 15: Morrowind – Village (durée du spectrogramme : 2 minutes 30 secondes)

Sons présents :Cris de créature, ressac, clapotement.

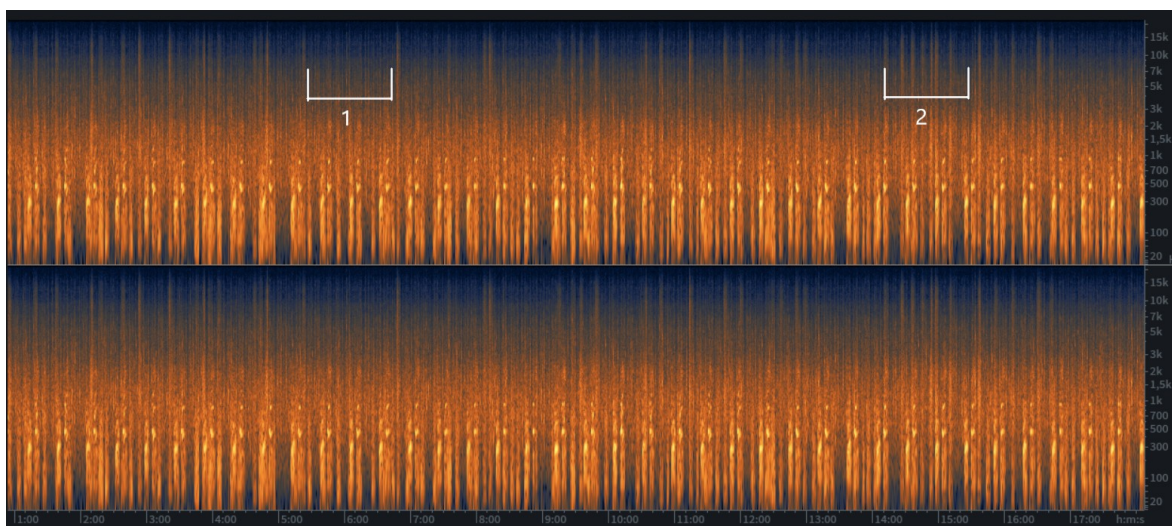


Figure 16: *Morrowind – Village* (durée du spectrogramme : 17 minutes)

Sons présents : cri de créature, ressac, clapotement.

Néanmoins, la présence d'autres sons issus du cours d'eau peut créer des variations dans chaque boucle.

Le cours d'eau est composé par une boucle d'environ 5 secondes représentant un clapotement constant et de divers sons plus ponctuels représentant le ressac. Ces sons, plus ponctuels, sont facilement identifiables sur le spectrogramme car ils occupent les plus hautes fréquences de ce paysage sonore. Une observation globale du spectrogramme dans les hautes fréquences sur une plus grande durée (ici 17 minutes) permet d'observer une apparente absence de bouclage pour ces sons (figure 16). Les zones 1 et 2, respectivement de 5'20" à 6'40" et de 14'00" à 15'20" sont de même durée.

Pourtant, en zoomant (figure 17), on observe de multiples répétitions du même motif dans la zone 2 et une absence totale de ce motif de ressac dans la zone 1. Celles ci sont indiquées par des flèches blanches. On constate donc des différences dans la production des sons d'ambiance qui peuvent être dues soit à un séquençage complexe, soit à l'utilisation de l'aléatoire.

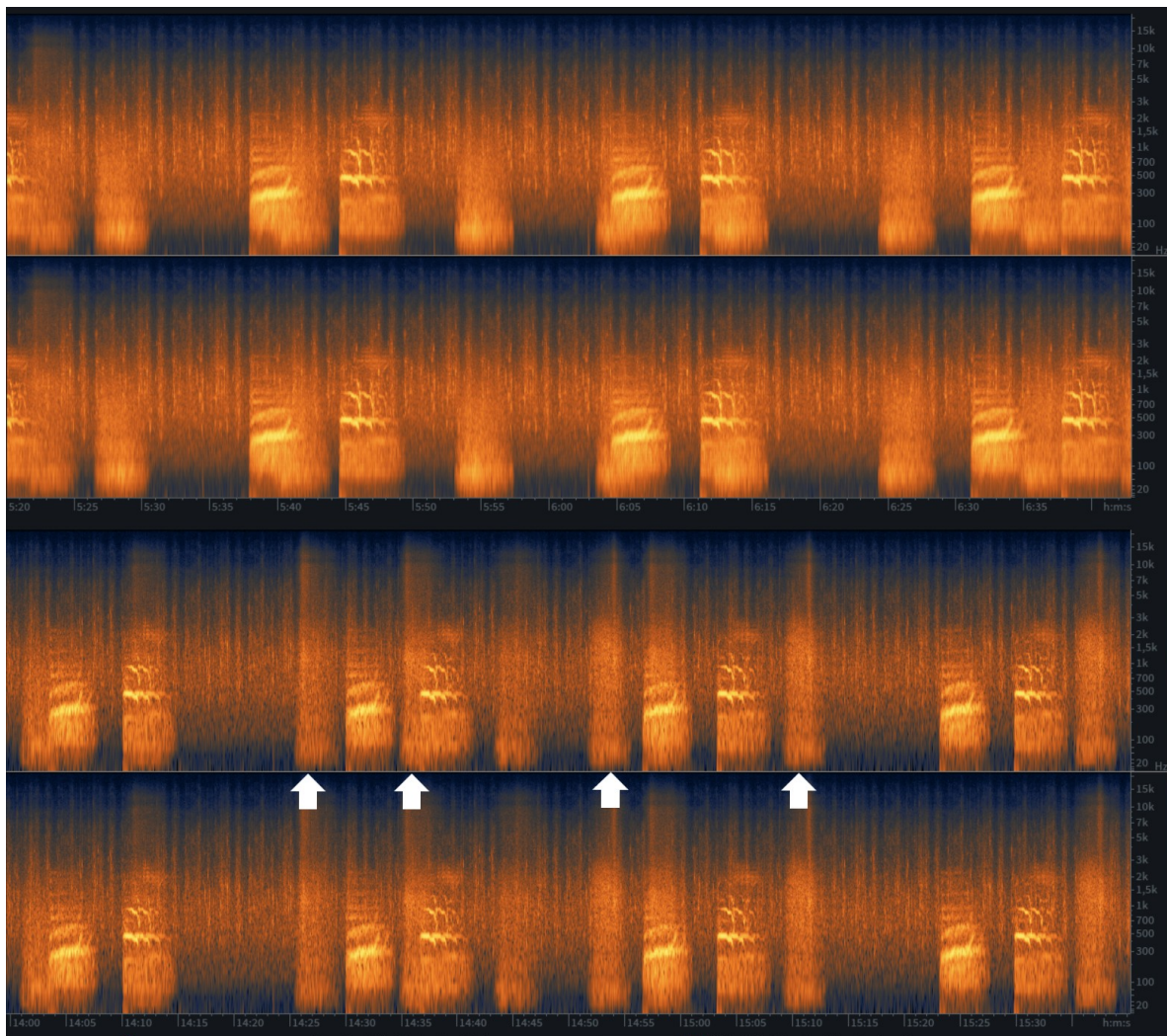


Figure 17: Zoom sur la zone 1 (en haut) et 2 (en bas) du spectrogramme figure 16.

Analyse d'ambiances tirées de Oblivion

Contrairement à son ancêtre, les environnements sonores d'Oblivion sont composés non seulement d'éléments ponctuels provenant de l'espace 3D mais également de sons diffus liants, faisant office de tapis sonore diégétique pour chacun de ces environnements. Néanmoins, en l'absence d'éléments émetteurs de sons proche du joueur, certaines boucles se font ressentir assez vite.

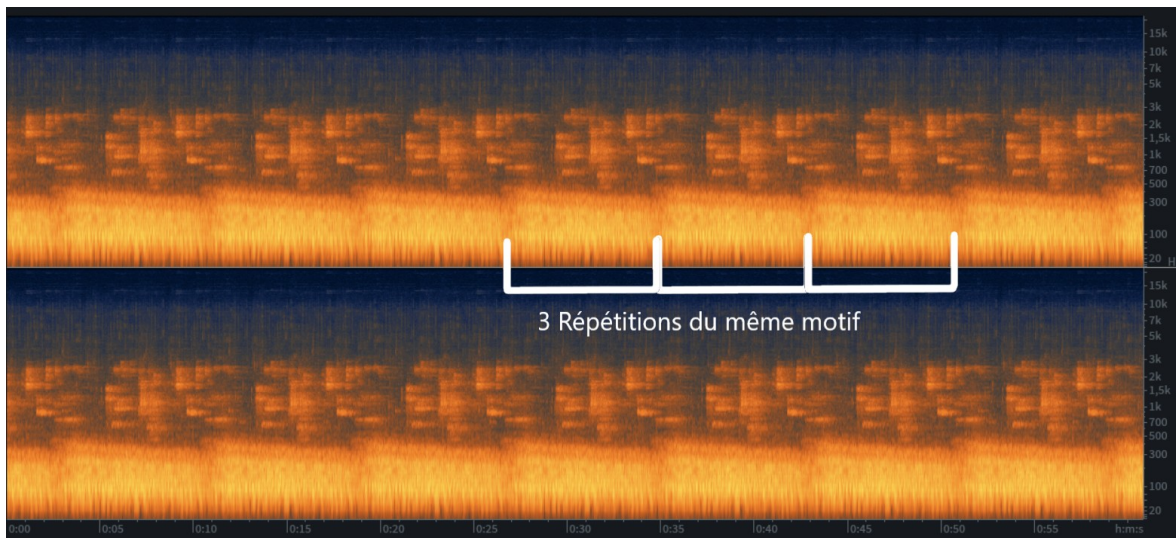


Figure 18: Oblivion – Egouts (durée : 1 minute)

Sons présents : fond d'air, goutte d'eau.

On remarque assez aisément dans la figure 18 le bouclage d'un seul son. L'observation du spectrogramme révèle 7 répétitions pour une durée de 1 minute. La boucle dure environ 8 secondes et reste assez facilement identifiable à l'oreille. Là encore, on peut l'expliquer par la place importante qu'occupe la musique dans l'univers sonore du jeu.

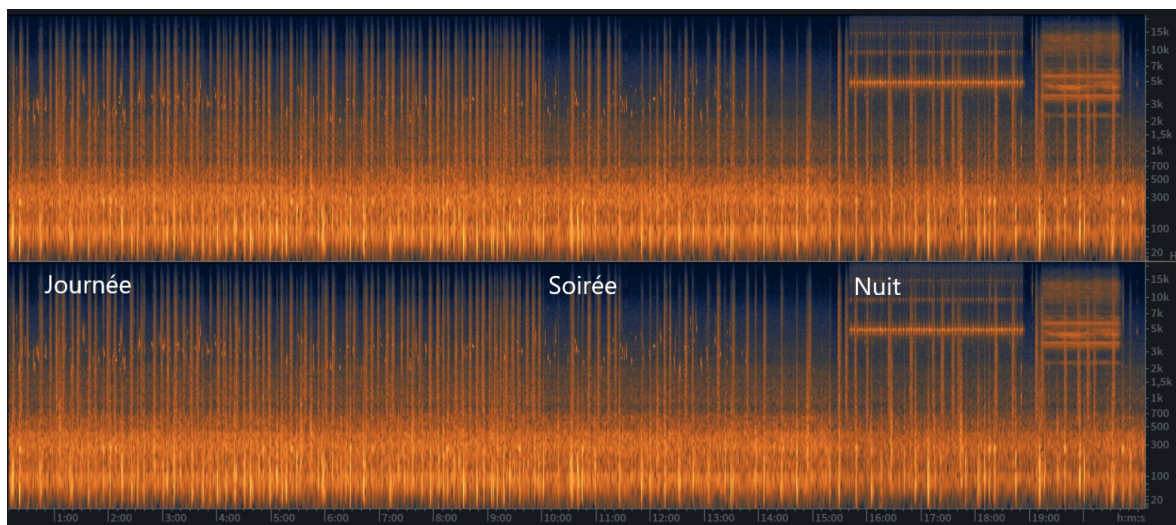


Figure 19: Oblivion – Colline (durée : 15 minutes)

Sons présents : vent, ressac, faune.

Le spectrogramme (figure 19) illustre l'évolution de la composition des ambiances lors d'un passage jour-nuit dans l'univers du jeu. L'ambiance est composée de coups de vent, de ressacs, ainsi que divers bruits d'insectes, d'oiseaux ou d'autres éléments de la faune, indétectable à l'œil nu dans l'univers du jeu.

Plusieurs phases sont distinguables grâce au tracé du spectrogramme. On a d'abord en après-midi, une forte concentration d'oiseaux diurnes en tout genre. La faune diminue en intensité au moment où la soirée commence, à environ 10 minutes. L'agitation du vent devient plus parcellaire. On distingue ensuite plusieurs phases où la faune nocturne prend une place centrale dans le paysage (15'40" et 19'12"). Ces phases se succèdent les unes aux autres en fondus, en gardant le vent et les ressacs.

Ces différentes phases apportent du relief et participent à la diversité des environnements sonores quand bien même on observe uniquement des sons assez répétitifs. Il faut souligner que certains sons propres à la faune nocturne, comme les chants de criquets,

sont, par nature, des sons étonnamment répétitifs et réguliers. Ces sons, en plus d'être très évocateurs, ne nuisent donc pas à l'immersion malgré leur répétition.

L'introduction de plusieurs phases selon le déroulement de la journée ainsi que l'utilisation de la faune permettent à Oblivion d'offrir des extérieurs assez convaincants malgré la faible longueur des sons et leur répétition régulière.

Analyse d'ambiances tirées de Skyrim

Le plus moderne des Elder Scrolls se distingue de ses prédécesseurs grâce à son aspect audio. Bien que la musique garde une place prépondérante notamment dans les scènes d'action, elle reste relativement en retrait à de nombreux moments, plus diffuse et moins mélodique, voire absente. A l'écoute, les environnements sont globalement plus riches, bien que certaines répétitions demeurent détectables.

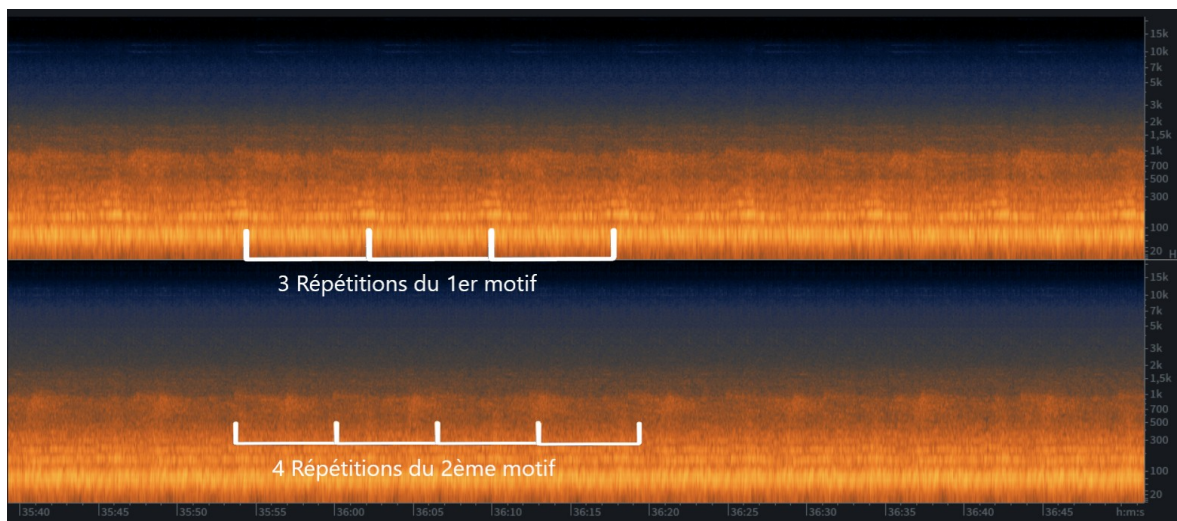


Figure 20: Skyrim – Grotte (durée : 1 minute 12 secondes)

Sons présents : fond d'air.

La figure 20 est intéressante car elle permet d'illustrer l'intérêt de l'observation du spectrogramme par rapport à la seule écoute.

Une écoute analytique permet de distinguer au moins deux patterns qui se répètent, mais ces deux sons étant particulièrement diffus et spectralement proches, il reste assez difficile de les distinguer. Par contre, sur le spectrogramme, les deux motifs sont identifiables : le 1^{er} plus grave dure environ 8 secondes, et le second, identifiable par les pics réguliers vers 1 kHz, dure 6 secondes.

La durée relativement courte de ces sons trahit leurs répétitions. Néanmoins, le fait que ces deux motifs aient chacun une durée différente apporte une sensation de renouvellement immédiat et durable à l'ambiance. En effet, la superposition de deux boucles de durée différente entraîne la répétition différée de chacune de ces boucles. Couplé à la nature diffuse de ces sons, la répétition devient bien moins détectable que lorsque l'on n'a qu'une seule boucle comme pour les jeux Oblivion ou Morrowind, le bouclage de chaque son étant chaque fois masqué par l'autre.

On remarque également (cf. figure 21) la disparité des sons ponctuels qui composent l'ambiance. Il s'agit de sons de courant d'air s'insinuant dans la grotte (repérés par les flèches blanches), parfois couplés à des sons de crépitements des parois rocheuses.

L'écoute analytique de ces événements permet de constater l'utilisation d'un nombre important de sons différents. Quelques répétitions sont identifiables, mais il semble qu'un même événement sonore ponctuel puisse être illustré par un nombre important de sons différents. D'autre part, le temps de réitération entre deux sons ponctuels similaires ou sémantiquement similaires dans cette ambiance peut varier d'une dizaine de secondes à plus d'une minute. Ce caractère aléatoire a pour effet de rendre imprévisible l'apparition de certains sons et ainsi atténuer la sensation de répétition que l'on peut ressentir lors de l'écoute d'un son bouclé.

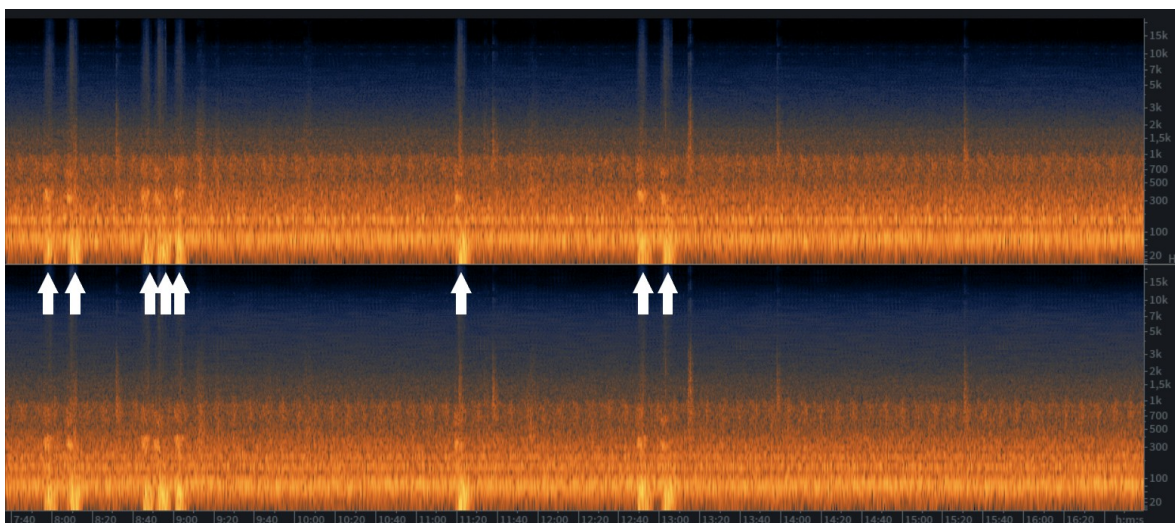


Figure 21: Skyrim – Grotte (durée : 9 minute)

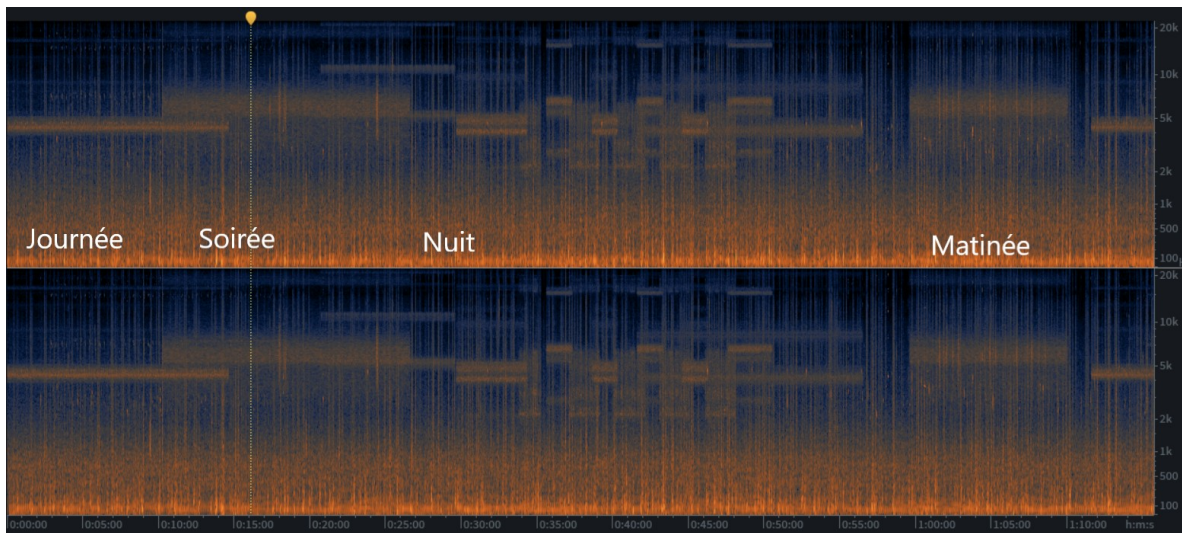


Figure 22: *Skyrim – Forêt* (durée : 1 heure 15 minutes)

Sons présents : fond d'air, vent, faune.

Pour les extérieurs de Skyrim (cf. figure 22) on observe également un soin particulier apporté aux différentes phases de la journée. Le passage jour/nuit et les différentes étapes intermédiaires semblent plus détaillées encore que pour Oblivion.

On remarque similairement à Oblivion un fond d'air constant, principalement composé de souffle et de vent dans les arbres, recouvert de différentes phases où la faune sauvage s'installe dans le paysage sonore selon l'heure donnée dans l'univers du jeu.

On dénombre par ailleurs un nombre plus important de phases différentes qui se succèdent, voire qui se superposent. Une ambiance similaire est utilisée pour les phases d'aubes et de crépuscules, et on assiste durant la nuit à une alternance de plusieurs sons issus de la faune nocturne.

La figure 23 illustre les 32 minutes de la nuit de Skyrim. On y reconnaît les différentes phases de la nuit où les différents insectes nocturnes échangent leurs voix dans le paysage sonore. On peut distinguer plusieurs alternances sur plusieurs bandes spectrales. Ces alternances ne semblent pas suivre un pattern précis, ce qui semble confirmer l'amélioration de la variabilité des ambiances sonores de la saga The Elder Scrolls.

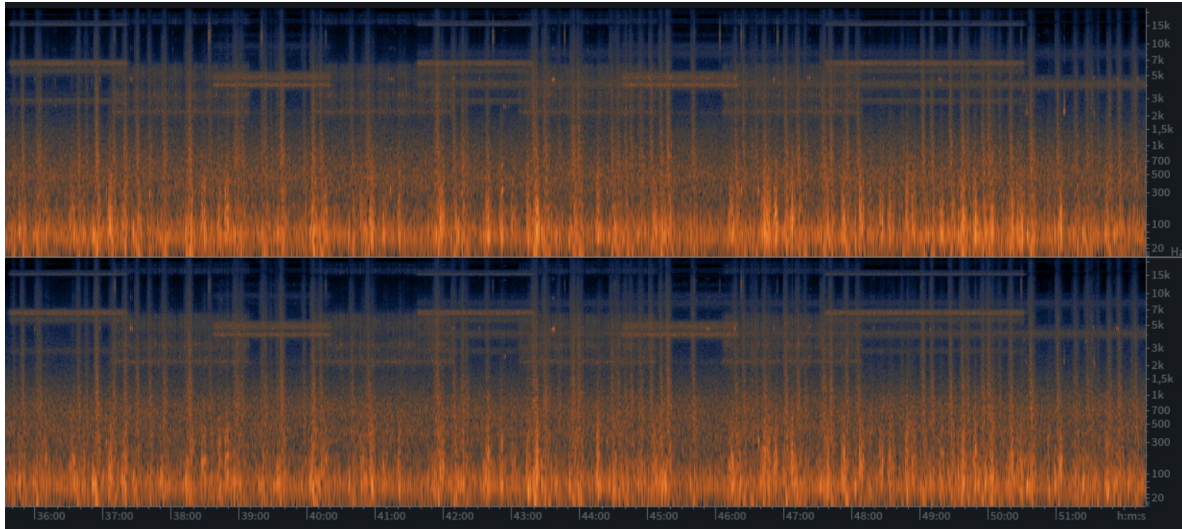


Figure 23: Skyrim - Alternance des différentes ambiances nocturnes (durée : 32 minutes)

Sons présents : fond d'air, vent, faune.

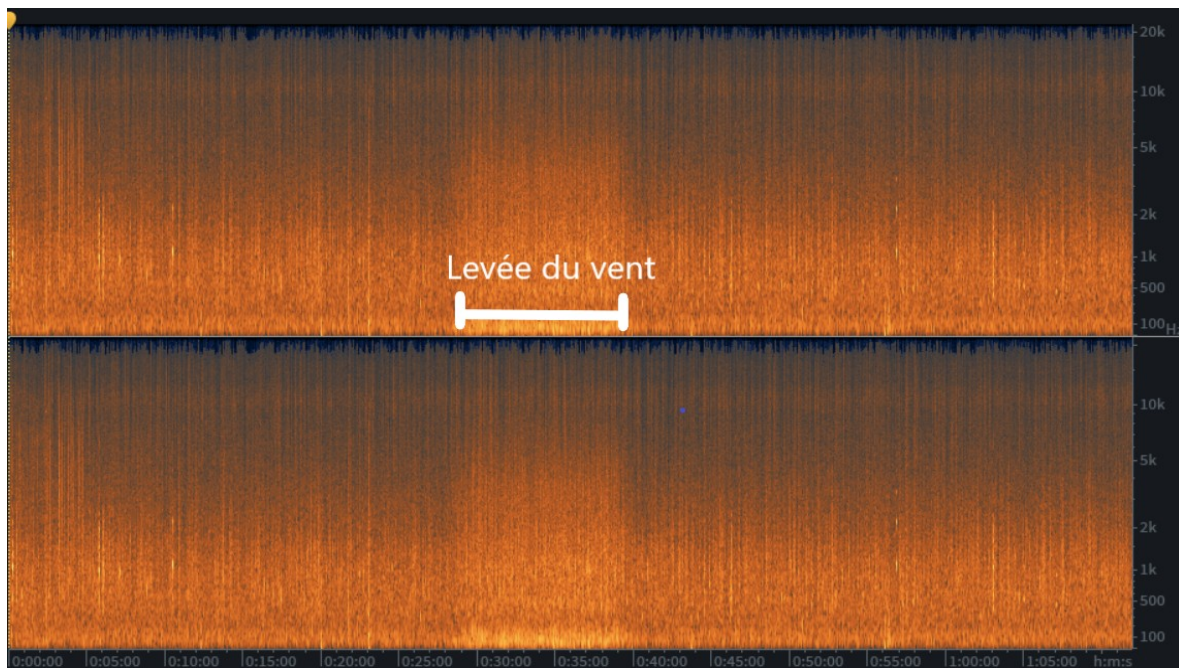


Figure 24: *The Witcher 3 – Village.*

Analyse d'ambiances tirées de *The Witcher 3*

The Witcher 3 est un jeu qui a marqué son époque notamment pour le soin apporté aux environnements et à la construction du monde.

L'enregistrement et l'analyse des spectrogrammes de ces environnements l'ont confirmé puisqu'il est très difficile de repérer un quelconque bouclage ou une répétition à l'oreille comme on peut le constater grâce aux figures 24 et 25.

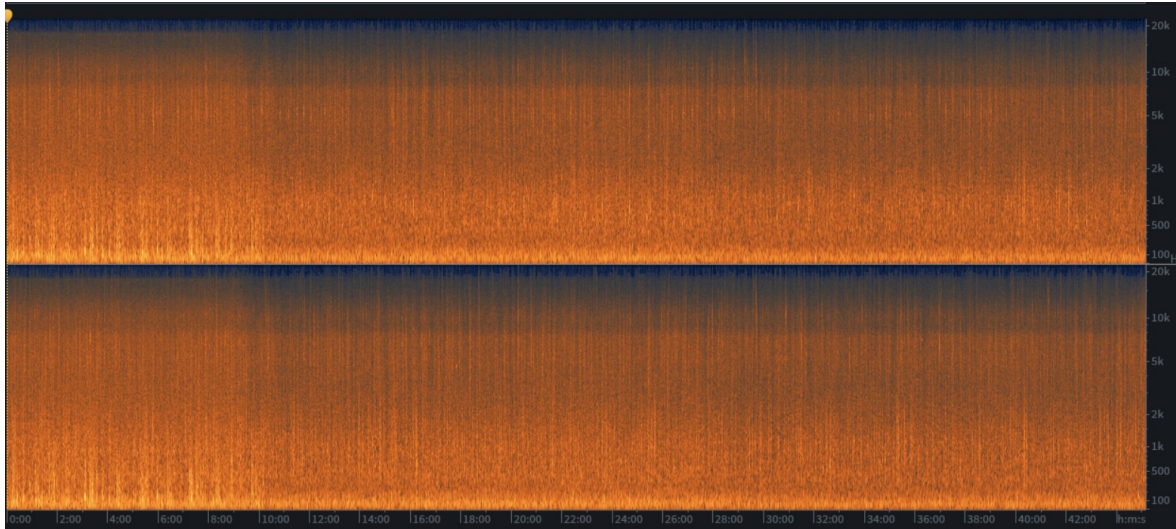


Figure 25: *The Witcher 3 - Forêt*

Les spectrogrammes mettent en valeur les différents changements de phase correspondant aux variations météorologiques (figure 24). Les ambiances nocturnes et diurnes semblent beaucoup moins variables que dans Skyrim ou Oblivion.

On peut déplorer ce manque de variations au sein des ambiances, mais The Witcher 3 se débrouille bien mieux pour masquer le bouclage des sons et l'artificialité des environnements sonores de son univers. Cela peut s'expliquer par l'utilisation de sons beaucoup plus diffus et larges spectralement qui semblent rendre le bouclage moins détectable.

Il reste néanmoins quelques sons ponctuels associés à l'univers du jeu dont on peut identifier les rares répétitions : aboiement de chien, ouverture et fermeture d'une porte.

La limite de l'écoute et de l'observation des spectrogrammes est facilement atteinte avec les jeux plus récents. Les ambiances complexes et chargées spectralement comme le

sont celles de The Witcher 3 sont impossibles à analyser en profondeur avec le spectrogramme pour seul outil tant le nombre de sons joués simultanément peut être important.

Pour aller plus loin dans la recherche d'un moyen de quantifier la variabilité d'une ambiance, nous avons donc eu recours à d'autres techniques de traitement du signal. Dans la suite, nous étudions ce que permet l'utilisation de l'intercorrélation entre les sons de base et les ambiances s'agissant de la détection fine des répétitions d'un son élémentaire dans une même ambiance.

B – Utilisation de l'intercorrélation entre les ambiances et les sons élémentaires

1. Analyse IDS

Dans la suite de ce mémoire, nous présentons le recours à d'autres techniques de traitement du signal : l'intercorrélation et l'autocorrélation. L'objectif visé consiste à essayer de développer un outil capable de détecter avec précision les itérations d'un son élémentaire (ou son de base) présent dans une ambiance.

Nous avons d'abord supposé que la détection serait plus facile en utilisant les techniques de corrélation non pas sur la totalité du spectre audio mais en les appliquant à la zone fréquentielle où le son élémentaire, que l'on cherche à détecter, présente la plus grande énergie. Nous avons donc, dans un premier temps, étudié si le recours à un découpage fréquentiel du son et des ambiances permet une détection facilitée.

L'analyse par Intégration de densité spectrale (dite analyse IDS) proposée par Laurent Millot, [8] [11] permet, lors de son application à un son, de décomposer un signal en plusieurs signaux correspondant à des sous-bandes fréquentielles complémentaires. Nous avons utilisé cette analyse afin de déterminer et se concentrer, pour les détections de répétitions, sur une seule sous-bande fréquentielle : la sous-bande fréquentielle où le son de base présente la plus grande énergie.

Le découpage fréquentiel que nous avons utilisé s'effectue avec 10 sous-bandes larges d'environ une octave, pour des sons échantillonnés à 44.1 kHz. Afin de s'approcher d'une écoute naturelle et dissociative de chaque sous-bande fréquentielle, nous avons choisi un découpage similaire à celui utilisé par Pascal Spitz pour les sons musicaux⁴, mais adapté à l'écoute des ambiances : de 0 à 60 Hz, de 60 à 120 Hz, de 120 à 250 Hz, de 250 à 500 Hz,

⁴ 30Hz – 60Hz, 60Hz – 120 Hz, 120Hz – 250 Hz, 250 Hz – 500 Hz, 500Hz – 1000 Hz, 1000 Hz – 2000 Hz, 2000 Hz – 4000 Hz, 4000 Hz – 8000 Hz et 8000 Hz – 16 000Hz.

de 500 à 1 kHz, de 1 à 2 kHz, de 2 à 4 kHz, de 4 à 8 kHz, de 8 à 16 kHz et de 16 à 22,05 kHz.

La sous-bande fréquentielle contenant les basses fréquences a été étendue jusqu'à 0 Hz et une nouvelle bande pour l'extrême aigu a été ajoutée afin de recouvrir l'ensemble du signal utile échantillonné à 44,1 kHz.

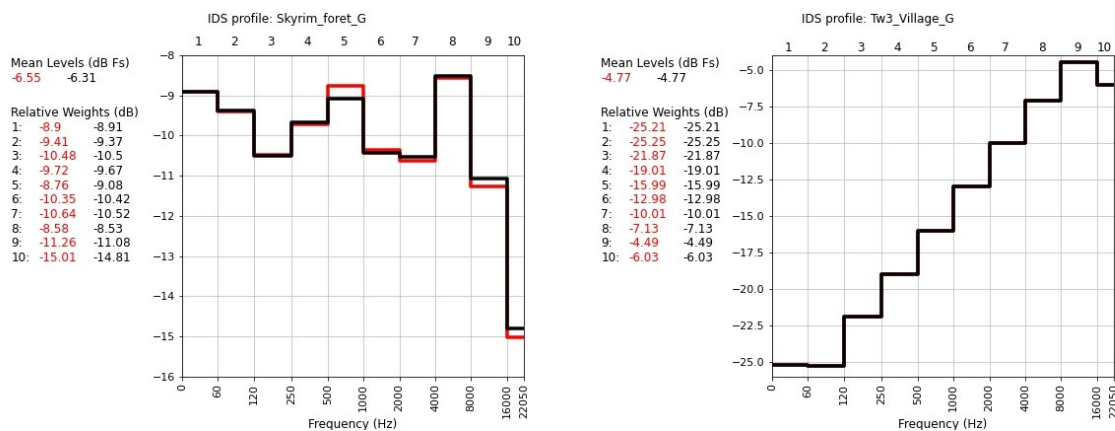


Figure 26: exemples de profils IDS d'ambiance de Skyrim et The Witcher 3.

Le programme fourni par Laurent Millot permet de tracer le profil IDS des sons désirés (cf. les deux exemples proposés en figure 26).

Chaque profil IDS indique à gauche du tracé le niveau moyen en dB Fs pour chacune des voies (la voie gauche en rouge, la droite en noir), ainsi que la valeur du poids relatif de chaque sous-bande en dB relatifs. Les valeurs sont calculées à partir des cumuls des énergies (carrés des valeurs des échantillons) pour chacun des signaux de sous-bande et du signal complet, pour chacune des deux voies. Les poids relatifs correspondent à la valeur en dB du rapport entre l'énergie cumulée dans la sous-bande et l'énergie de la voie du signal complet, pour une voie donnée. Les niveaux moyens correspondent aux énergies cumulées pour les signaux complets associés à chacune des deux voies et exprimés en dB FS.

Les signaux de sous-bande sont calculés par filtrage du signal complet par des filtres « katana » qui réalisent un passage de 0 dB à 80 dB en 5 Hz dans chacune-des bandes de transition entre deux sous- bandes.

L'avantage de l'analyse IDS réside dans la possibilité d'écouter et de manipuler les signaux en sous-bande indépendamment les uns des autres, contrairement à l'analyse par spectrogramme. L'analyse IDS permet également une resynthèse du signal original en effectuant une simple sommation (remixage) de chacune des sous-bandes à leur niveau initial. On peut aussi réaliser une resynthèse partielle en ne sélectionnant qu'une partie des sous-bandes fréquentielles ou modifier le poids des sous-bandes prises en compte pour la resynthèse afin d'augmenter ou de diminuer le poids d'une sous-bande donnée.

Dans la suite de ce mémoire, nous allons utiliser les sons élémentaires situés dans les fichiers des jeux étudiés pour les comparer aux ambiances enregistrées directement pendant une phase de jeu afin de détecter leurs répétitions éventuelles. Ainsi, lors des analyses d'ambiances contenant un nombre important de sons pouvant être joués simultanément, l'analyse IDS sera systématiquement utilisée pour étudier uniquement la bande de fréquences où le son élémentaire est le plus susceptible d'être observé. Nous supposons en effet que le travail de détection par intercorrélation entre l'ambiance et le signal de base recherché sera plus efficace.

2. Intercorrélation

L'intercorrélation est un outil permettant la comparaison entre deux signaux afin d'en identifier les ressemblances. Dans notre cas, en l'appliquant à une ambiance et un son élémentaire contenu dans cette ambiance, le signal d'intercorrélation résultant doit nous permettre d'identifier les occurrences de ce son élémentaire dans cette ambiance.

$$T_{s_1, s_2}(\tau) = \int s_1(t) s_2(t - \tau) dt$$

La fonction d'intercorrélation ci-dessus, dans sa version valide pour les signaux analogiques, permet de comparer l'aire d'un signal de référence (ici s_1) à l'aire d'un autre signal décalé de τ (ici s_2). τ étant une valeur de décalage temporelle. De ce fait, en faisant varier la valeur de τ , on effectue un balayage du signal s_2 sur le signal s_1 en observant comment évolue le produit de la superposition de leurs aires respectives.

Si le signal s_2 est présent dans le signal s_1 , le produit des aires calculé par la fonction d'intercorrélation est maximal lorsque le signal s_2 et une de ses occurrences dans le signal s_1 se superposent. La fonction d'intercorrélation va donc présenter un maximum local en chaque valeur de τ pour laquelle $s_1(t)$ et $s_2(t-\tau)$ sont ressemblants.

Ici, comme nous travaillons avec des signaux numériques, nous utiliserons la version numérique de l'intercorrélation définie par :

$$T_{s_1, s_2}(k) = \sum_{n=0}^{N-1} s_1[n] s_2[n-k]$$

avec k , la valeur de décalage du signal s_2 par rapport au signal s_1 , exprimée en échantillons, et N la longueur du signal s_1 également exprimée en échantillons.

Sur la figure 27, on a tracé deux exemples de signaux permettant d'illustrer ce principe : deux signaux correspondant à des portes rectangulaires unitaires comportant 100 échantillons à chaque fois. Pour le signal s_1 , la porte rectangulaire est centrée sur l'instant $t_1=500$ s, pour le signal s_2 , la porte rectangulaire est centrée sur l'instant $t_2=200$ s.

En figure 28, on a tracé le signal d'intercorrélation entre s_1 et s_2 qui présente un pic à l'instant $t=300$ s. Cela signifie qu'il faut retarder le signal s_2 de 300 s pour synchroniser les deux signaux et obtenir une valeur maximale de l'intercorrélation.

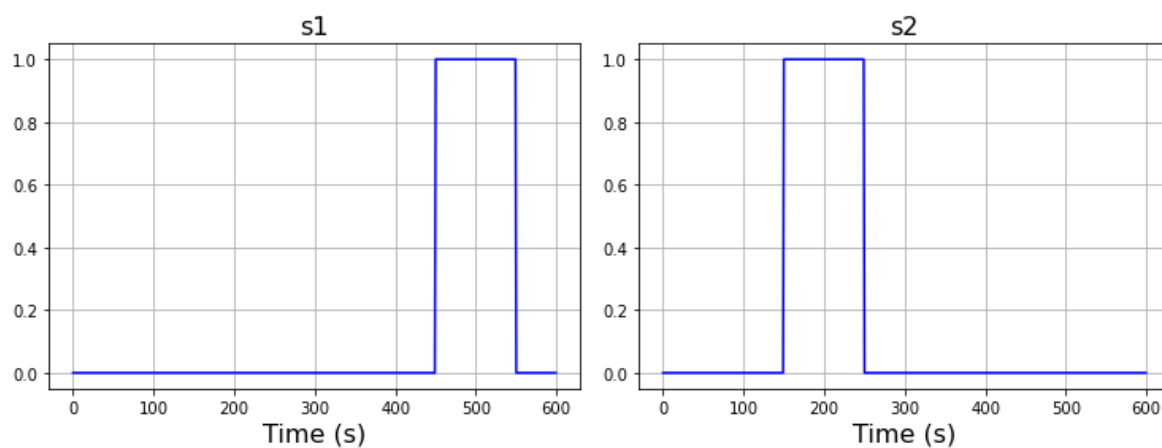


Figure 27: Deux signaux s_1 et s_2 comportant une similarité

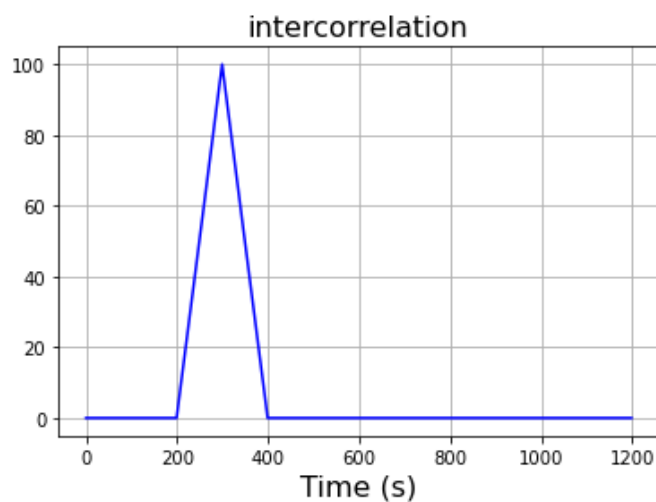


Figure 28: Signal d'intercorrélation entre s_1 et s_2

Ainsi, nous avons supposé que la fonction d'intercorrélation peut nous permettre de comparer les ambiances enregistrées dans les jeux avec les sons élémentaires issus de ces jeux afin de détecter la présence de ces sons élémentaires dans les ambiances, ainsi que la façon dont ceux-ci sont répétés.

En pratique, l'efficacité de cette technique demeure assez aléatoire compte tenu des ambiances spectralement complexes qui constitue notre corpus, comme nous allons le montrer.

Vous trouverez en annexe le programme Python utilisé pour calculer et afficher l'intercorrélation.

La figure 29 illustre ce constat pour le son d'ambiance du village issu de Morrowind et la détection des répétitions du cri de la créature dans l'ambiance.

On peut effectivement apercevoir des pics réguliers susceptibles de trahir la présence du cri dans l'ambiance de village (les flèches rouges pointent certains pics, les segments montrent certaines périodes) mais on constate que les maxima locaux sont bien moins évidents à détecter que dans l'exemple caricatural précédent.

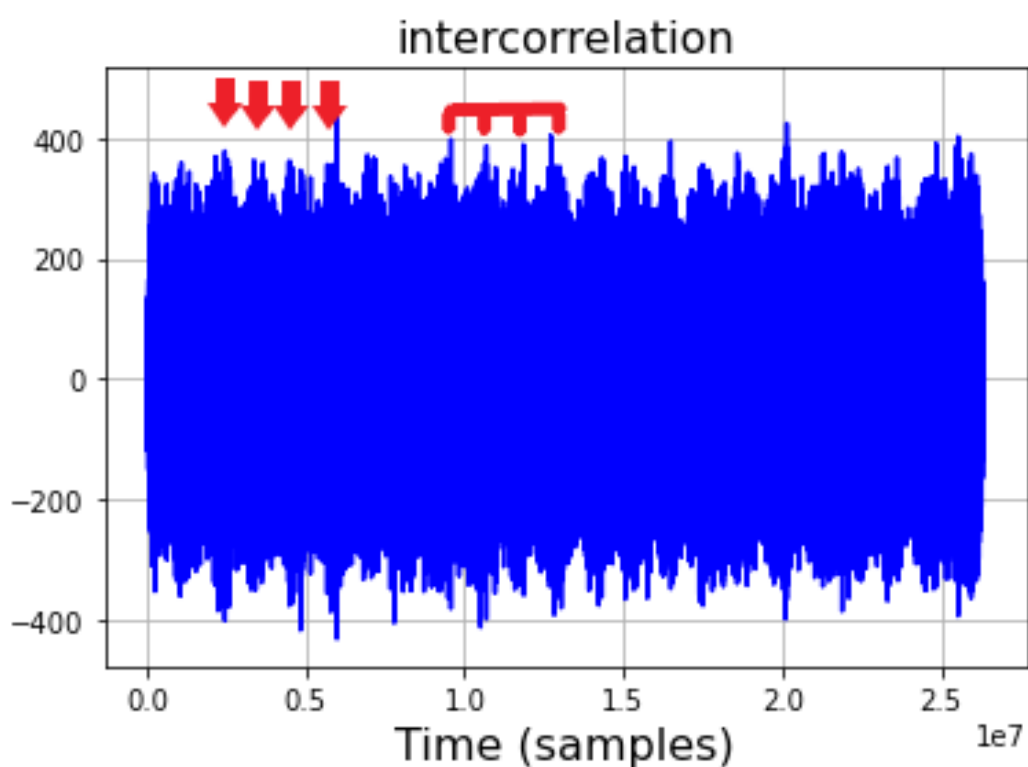


Figure 29: Intercorrélation de l'ambiance village de Morrowind avec le son élémentaire régulier du cri des créatures

La figure 30 correspond au tracé de l'intercorrélation pour le cas de l'ambiance associée aux égouts d'Oblivion dans laquelle un son court est répété à plusieurs reprises. Alors que les répétitions s'entendent de manière évidente à l'écoute, elles demeurent indétectables en observant le tracé d'intercorrélation.

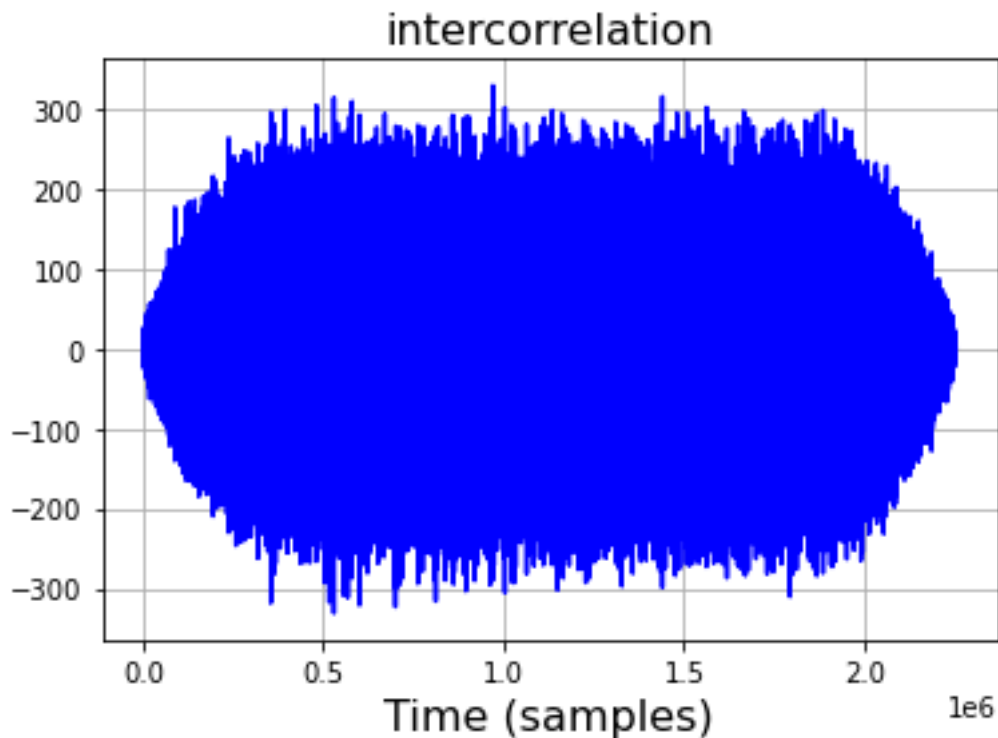


Figure 30: Intercorrélation de l'ambiance égout d'Oblivion avec le fond d'air bouclé qui la constitue.

Ce constat de difficulté de la détection de présence et des répétitions d'un son de base dans beaucoup d'ambiances, nous a amené à envisager l'introduction d'un traitement du signal d'intercorrélation afin d'éliminer le « bruit » et d'essayer de mettre en exergue de manière indéniable les présences ou répétitions des sons de base. Dans la suite, nous étudions ce que donne l'application de l'autocorrélation au signal d'intercorrélation.

3. Autocorrélation de l'intercorrélation

On appelle autocorrélation le calcul de l'intercorrélation d'un signal avec lui-même. Ici, comme nous appliquons l'autocorrélation au signal d'intercorrélation entre l'ambiance et le son élémentaire, nous calculons donc l'autocorrélation de l'intercorrélation définie par :

$$A_{T_{s_1, s_2}}(k) = \sum_{n=0}^{N-1} T_{s_1, s_2}[n] T_{s_1, s_2}[n-k]$$

avec N la longueur du signal d'intercorrélation T en échantillons.

Nous avons supposé que cette astuce devait nous permettre de repérer les itérations d'un même son dans une ambiance malgré le « bruit » correspondant à la contribution à l'intercorrélation des autres sons élémentaires constituant l'ambiance sonore. En effet, l'intercorrélation d'un son d'ambiance avec un son élémentaire apparaissant dans cette ambiance va se traduire par l'apparition de pics à chaque fois que ce son apparaîtra dans l'ambiance. L'autocorrélation permettra, par le glissement du signal d'intercorrélation sur lui-même avec un décalage de k échantillons, de repérer les itérations des pics du signal d'intercorrélation et de les conserver par comparaison aux autres pics.

Sur l'exemple caricatural ci-après (figure 31), on considère un signal test comportant uniquement quelques impulsions numériques de Dirac placés aléatoirement (tracé en haut à gauche), et, on calcule l'intercorrélation de ce signal avec en guise de son élémentaire une simple impulsion de Dirac. Le tracé de l'intercorrélation entre cette ambiance et ce son de base, tous deux caricaturaux, figure en haut à droite tandis que le tracé en bas au centre correspond à l'autocorrélation de l'intercorrélation.

L'analyse du signal d'autocorrélation montre que celui-ci comporte un pic central en 0 et des pics placés de manière symétrique par rapport à ce pic central.

Le pic central correspond au décalage nul pour lequel on observe le maximum de ressemblance puisque le signal est superposé avec lui-même sans décalage. Il n'y a donc pas lieu de le prendre en compte pour l'analyse de la présence du son de base.

Les pics de part et d'autre du pic central trahissent, par contre, la présence de pics dans le signal d'intercorrélation. Par définition, l'autocorrélation est une fonction paire, ce qui explique la présence des pics symétriques par rapport au pic central situé à l'instant nul. Il n'est donc pas nécessaire d'étudier ce qui se passe pour les temps négatifs et il suffit d'étudier la présence des pics pour les instants positifs pour détecter les occurrences du signal de base dans le signal d'ambiance.

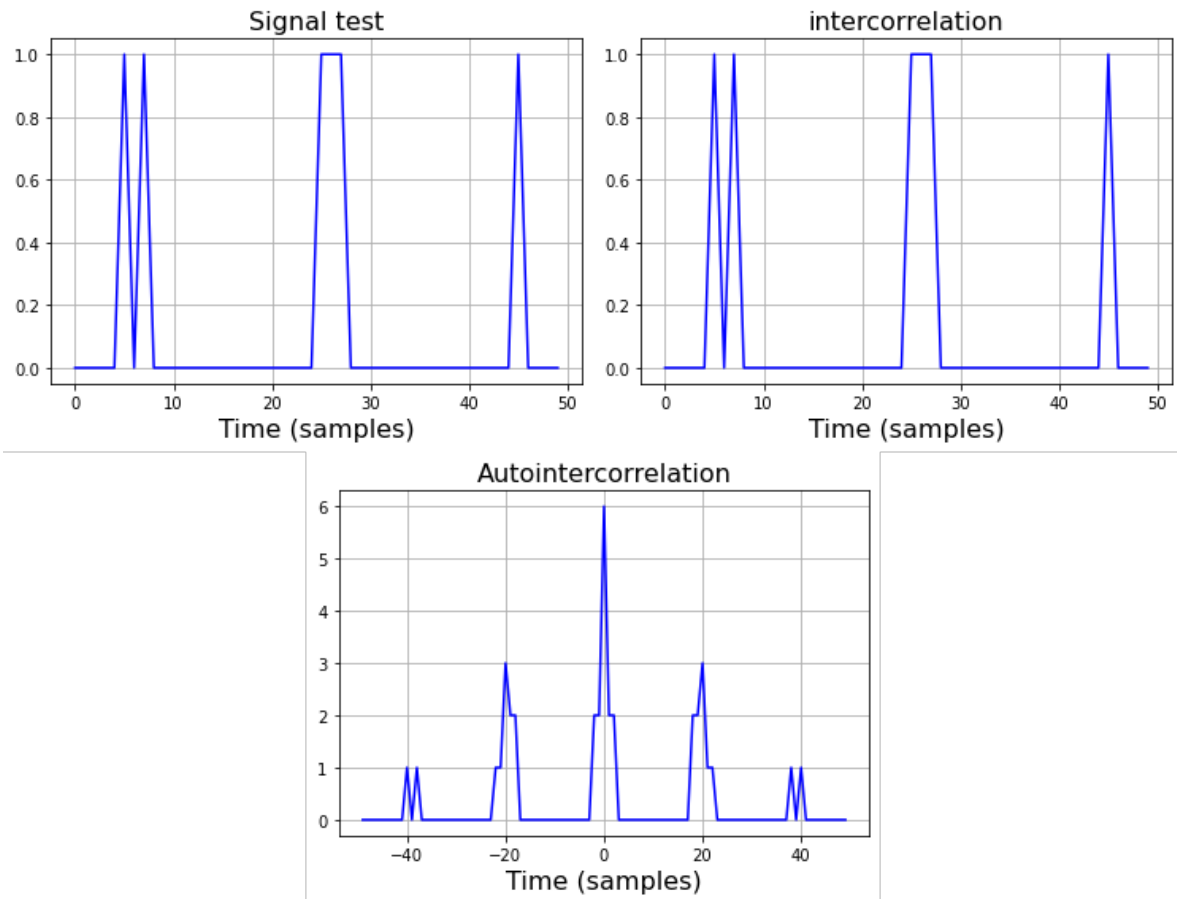


Figure 31: Exemple d'autocorrélation de l'intercorrélaiton avec un signal caricatural

Vous trouverez en annexe le programme Python utilisé pour calculer et afficher l'autocorrélation de l'intercorrélaiton (baptisé « autointercorrélaiton »).

Les pics du signal de base se trouve aux échantillons 5, 7, 25, 26, 27 et 45. Observons maintenant la partie positive du signal « d'autointercorrélaiton » figure 32. Les pics y sont situés à $k = 0, 1, 2, 18, 19, 20, 21, 22, 38$ et 40 . Avec la représentation choisie, les plateaux horizontaux indiquent la succession de plusieurs pics.

Si on ne tient pas compte du pic en $k=0$, le pic de niveau le plus important se trouve en $k = 20$. Il correspond à la superposition des contributions des trois pics du signal de base séparés deux à deux par un décalage de 20 échantillons, à savoir les pics situées en $k=5$, $k=25$ et $k=45$. Cela explique la hauteur de ce pic à $A_T(20) = 3$. Chacune des abscisses des

autres pics du signal « d'autointercorrélation » peut aussi être interprétée comme la détection d'un décalage entre deux des pics du signal test. Le pic trouvé en $k = 38$ représente l'écart entre les pics présents en $k = 7$ et $k = 45$ dans le signal test. Le pic en $k = 2$ représente l'écart entre les pics en $k = 5$ et $k = 7$ ainsi que celui entre les pics en $k = 25$ et $k = 27$, etc.

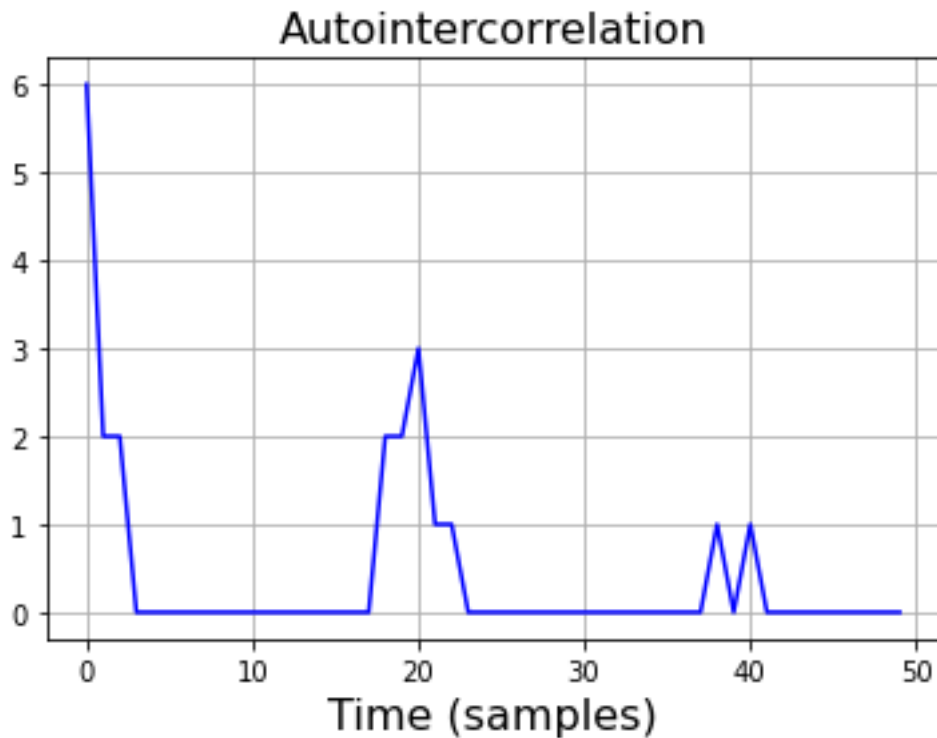


Figure 32: Partie positive du signal "d'autointercorrélation"

La figure 33 illustre le tracé de l'autocorrélation de l'intercorrélation pour l'ambiance des égouts d'Oblivion. Alors que l'on était incapable de détecter la présence de sons élémentaires dans le tracé du signal d'intercorrélation (cf. figure 30), cette présence est révélée par l'autocorrélation du signal d'intercorrélation. Les pics révélant la présence du son élémentaire, précédemment noyés dans le « bruit », se détachent clairement dans le signal d'« autointercorrélation ».

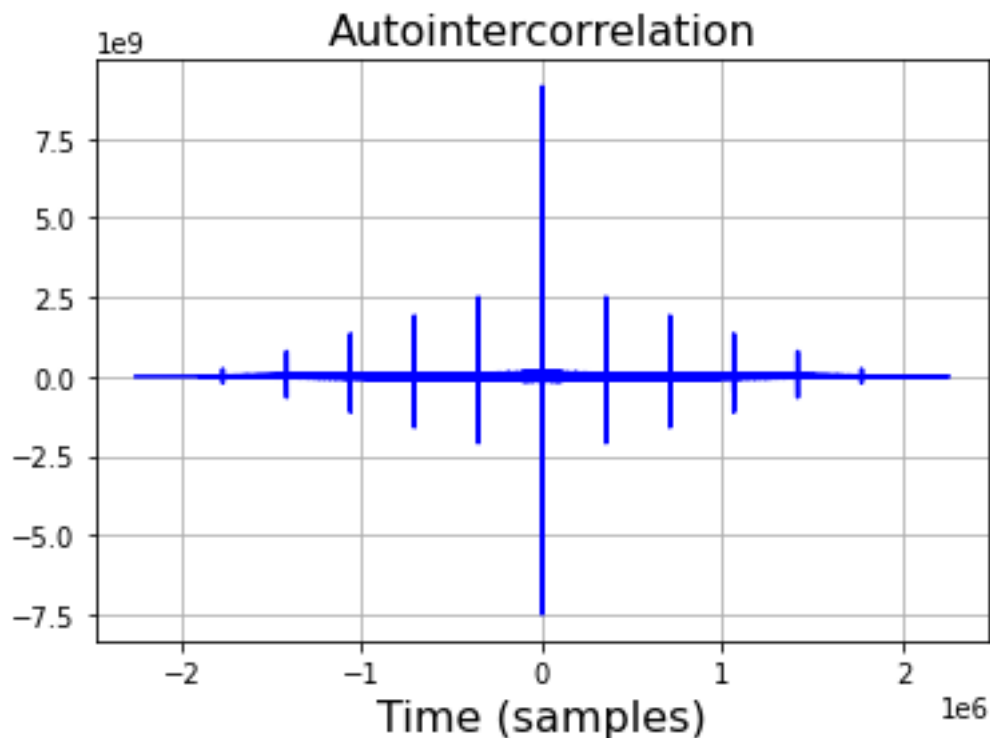


Figure 33: Autocorrélation du signal d'intercorrélation de l'ambiance égouts d'Oblivion

Le tracé de l'« autointercorrélation » permet de déduire l'existence de plusieurs pics similaires donc la présence du son élémentaire testé dans le son d'ambiance. En effet, la présence de pics symétriques de part et d'autre du centre du signal d'« autointercorrélation » indique la présence du son élémentaire dans l'ambiance tandis que la présence d'un unique pic à l'instant zéro indique l'absence du son élémentaire.

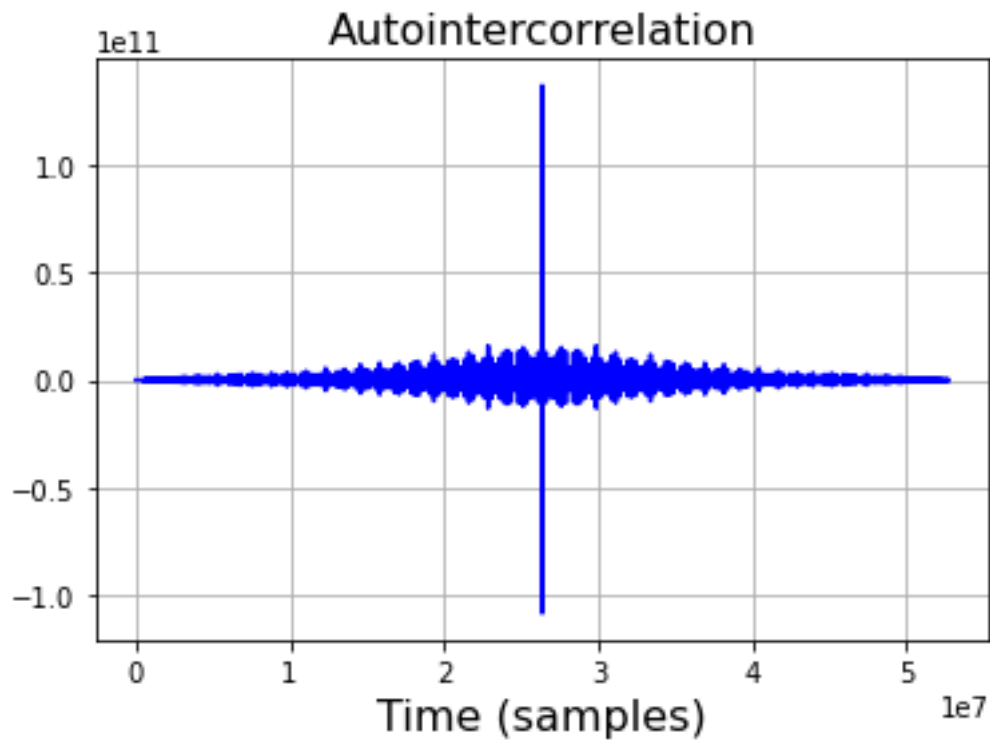


Figure 34: Autocorrélation du signal d'intercorrélation de l'ambiance village de Morrowind

L'étude de l'« autocorrélation » calculée pour le couple ambiance village de Morrowind et cri de créature (cf. figure 34), trahit l'apparition régulière du cri de créature dans l'ambiance puisque l'on observe quelques pics symétriques dans l'« autocorrélation ».

L'utilisation de l'intercorrélation et de l'autocorrélation du signal d'intercorrélation (ou « autocorrélation ») permet de retrouver efficacement les multiples occurrences d'un son élémentaire court dans un son d'ambiance plus long. Encore faut-il qu'il y ait de multiples répétitions d'un son élémentaire identifiable dans l'ambiance en question.

Le tracé de l'« autointercorrélation » de l'ambiance village de The Witcher 3 (figure 35) nous montre une absence de pics autres que le pic central, soit une absence de répétition du son élémentaire choisi, à savoir le son du vent.

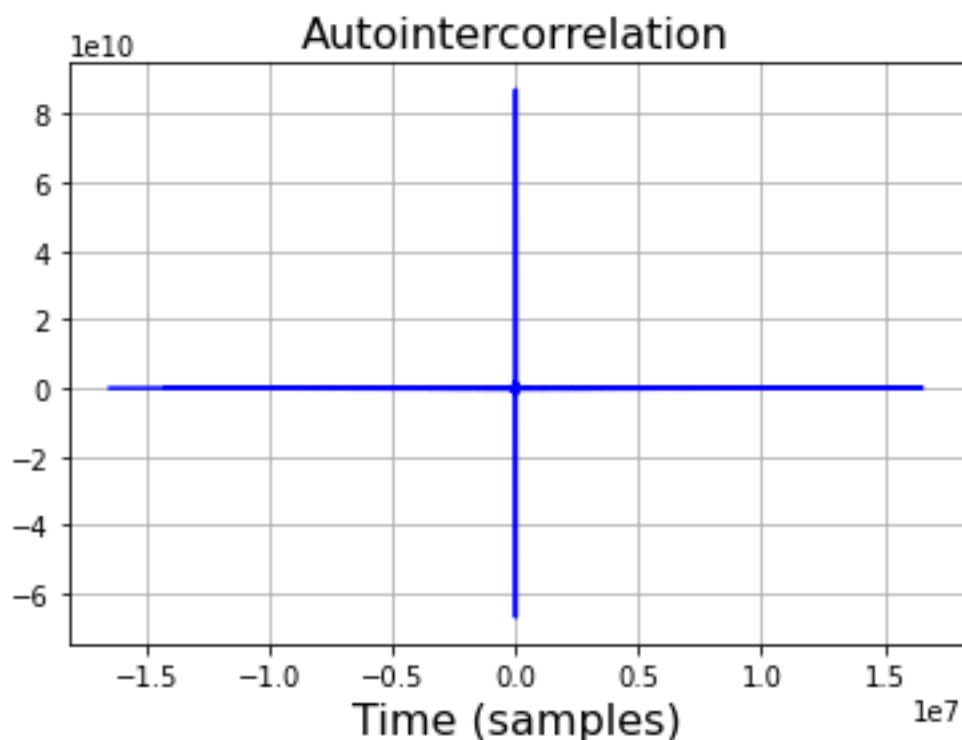


Figure 35: Autocorrélation du signal d'intercorrélation de l'ambiance Village de The Witcher 3

Cette absence de résultat dans le tracé nous montre la robustesse de The Witcher 3 dans la variabilité de ces ambiances. Le nombre important de sons sémantiquement similaires présents dans les données du jeu est révélateur de la façon dont sont composées les ambiances via le moteur audio : il semblerait qu'un même son ne soit jamais utilisé plusieurs fois de suite. The Witcher 3 utilisant Wwise, il semble probable que le choix des sons diffusés dans l'ambiance soit le résultat d'un contrôle aléatoire réglé de manière à éviter ces répétitions. Cela expliquerait l'absence de détection de répétition par le calcul de l'« autointercorrélation ».

4. Perspectives.

L'autocorrélation de l'intercorrélation nous permet de confirmer la présence de sons élémentaires dans une ambiance. L'absence de résultat constitue une preuve intéressante de la robustesse de la variabilité sonore d'une ambiance, au moins pour le contenu sonore sémantiquement équivalent au son élémentaire analysé.

Néanmoins, l'automatisation éventuelle de la recherche et du diagnostic de présence des pics symétriques dans le signal d'« autointercorrélation » pour l'ensemble des sons présents dans les fichiers du jeu et pour de longues ambiances enregistrées risque d'être très gourmande en temps de calcul. En effet, l'étape de filtrage des ambiances par les filtres en sous-bandes fréquentiels nécessite beaucoup de temps de calcul et beaucoup d'espace de stockage si on doit calculer et conserver les signaux de sous-bandes pour toutes les sous-bandes. Et, même si une version portée dans un langage rapide (en langage Fortran compilé par exemple) de l'analyse IDS était disponible, la question de l'espace de stockage des signaux de sous-bandes fréquentielles resterait posée.

Pour réduire le temps de calcul et l'espace de stockage, il faudrait absolument procéder à une étude préliminaire du corpus des sons de base, par analyse IDS de ces sons courts, afin de déterminer la (ou les?) sous-bande(s) à considérer pour essayer d'optimiser la détection de présence et des répétitions d'un son de base. Il ne serait alors nécessaire que de conserver la (ou les) sous-bande(s) intéressante(s) pour chaque signal de base et de ne calculer les signaux de sous-bande de l'ambiance que pour les sous-bandes nécessaires.

Il faudrait aussi étudier comment automatiser l'analyse du signal d'« autointercorrélation » afin que l'on obtienne directement les informations suivantes :

- détection ou non de la présence du signal (travail à partir de la présence de pics autres que le pic central) ;
- positions des pics symétriques pour les temps positifs ;

- détection des pics correspondant aux écarts entre deux répétitions successives en éliminant les pics associés à des répétitions séparées par au moins une autre répétition du son élémentaire. Ceci afin de permettre une détection précise des positions successives de chaque itérations.

Dans la suite, puisque nous sommes dans une approche préliminaire de recherche d'outil, nous nous sommes intéressés à un autre outil d'analyse temporelle qui est annoncé comme de nature à permettre la détection de multiples patterns répétés au sein d'un signal, le profil matriciel, ainsi qu'à une bibliothèque Python implémentant cet outil, la bibliothèque Stumpy.

Il convient de préciser que, si nous avons trouvé cet autre outil d'analyse pendant la phase préalable de recherche bibliographique et pas pendant la seconde moitié du travail de mémoire, nous aurions probablement quand même étudié l'efficacité des outils exposés jusqu'à présent car ils sont classiquement utilisés et assez faciles à mettre en œuvre. D'ailleurs, cette méthode du profil matriciel n'a été proposée qu'assez récemment (2016) et elle n'est probablement pas encore très connue et largement utilisée pour le moment, contrairement aux analyses reposant sur l'intercorrélation.

Chapitre 3 - La Bibliothèque Stumpy

A – Introduction

1. Présentation générale

Stumpy est une bibliothèque open source utilisant un outil appelé le profil matriciel (Matrix Profil) permettant de nombreuses analyses de séries temporelles.

Le profil matriciel est un ensemble de données vectorielles permettant l'analyse d'un signal temporel. Ces données sont la base permettant de « faire parler » les signaux temporels de plusieurs manières. Grâce au travail de Chin-Chia Michael Yeh, Yan Zhu, Abdullah Mueen et Eamonn Keogh, [17] les outils d'analyse des signaux temporels issus du calcul de ces données permettent, entre autres, la segmentation sémantique ou la reconnaissance de patterns, de similarités et d'anomalies au sein d'un signal temporel.

Stumpy calcule le profil matriciel pour des signaux temporels grâce à une méthode, appelée STOMP, proposée par les chercheurs de the University of California, Riverside et de the University of New Mexico. [18]

Pour comprendre ce à quoi correspond le profil matriciel d'un signal temporel, il faut d'abord s'intéresser à la notion de distance euclidienne entre des portions d'une série ou séquence temporelle (on parlera aussi de signal temporel).

2. Le Profil Matriciel

La sous-partie suivante s'inspire des tutoriels présents sur les sites de documentation de Stumpy. [15]

Prenons comme exemple la série (signal) aléatoire qui suit (cf. figure 36), longue de 13 échantillons, composée de la suite d'échantillons suivante :

$s = [0, 4, 3, 6, 2, 10, 12, 5, 4, 4, 3, 8, 2]$.

Bien que n'importe quel signal temporel discret à une dimension puisse être utilisé, nous supposons que ce signal correspond à la numérisation d'un signal sonore, pour simplifier le propos en se projetant directement dans le cas des signaux étudiés dans ce mémoire.

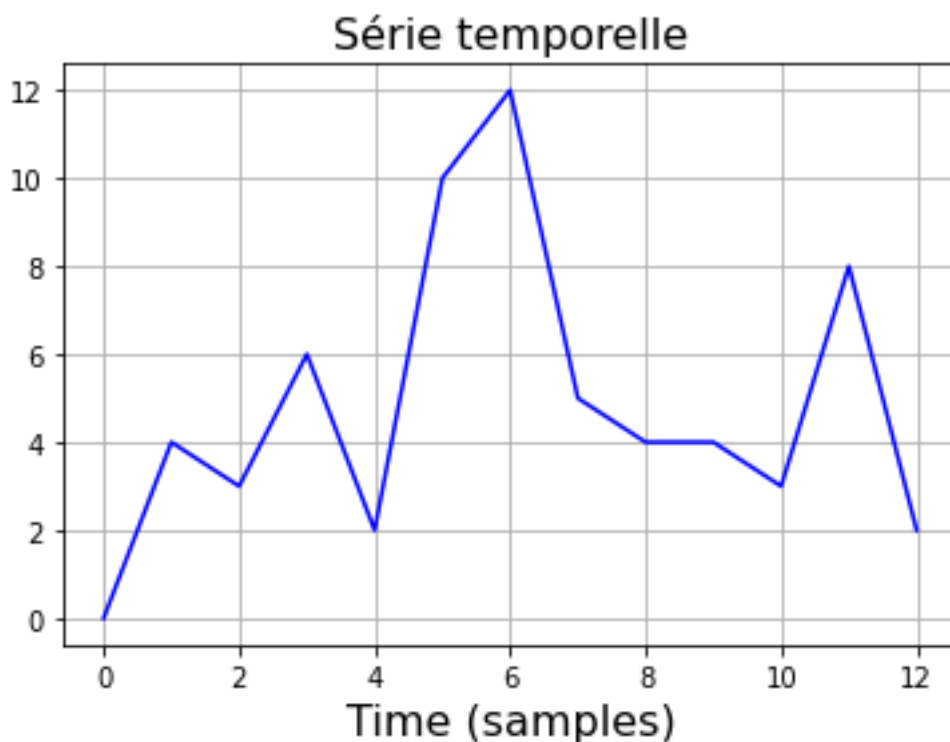


Figure 36: Exemple de série temporelle

L'objectif consiste à comparer les différentes parties de ce signal pour, à terme, être en mesure de détecter des informations utiles lors de l'analyse des ambiances de notre corpus. Pour ce faire, nous commençons par découper cette séquence en de multiples sous-séquences de taille m prédéfinie. Ici, nous avons choisi une taille de 5 échantillons ($m = 4$), ce qui, dans ce cas, nous donne un total de 10 sous-séquences différentes : $s_1 = [0, 4, 3, 6]$; $s_2 = [4, 3, 6, 2]$: ... ; $s_9 = [4, 4, 3, 8]$ et $s_{10} = [4, 3, 8, 2]$.

Dans la suite, nous considérerons ces sous-séquences comme des vecteurs comportant 4 coordonnées.

L'étape suivante consiste à calculer la distance euclidienne entre chacun de ces vecteurs grâce à la formule suivante :

$$D_{xy} = \sqrt{\sum_{i=1}^m (x_i - y_i)^2}$$

Avec x et y désignant les deux vecteurs comparés, x_i et y_i les i -ièmes coordonnées de ces deux vecteurs. i varie de 1 à m , puisque m correspond à la longueur de chacune des sous-séquences, et donc au nombre de coordonnées de ces vecteurs.

Par exemple, pour la distance entre les deux premiers vecteurs, nous aurons :

$$D_{s_1 s_2} = \sqrt{(0^2 + 4^2) + (4^2 + 3^2) + (3^2 + 6^2) + (6^2 + 2^2)}$$

Ces distances sont stockées dans une matrice comme suit :

$$\begin{pmatrix} 0 & D_{s_2 s_1} & D_{s_3 s_1} & \dots & D_{s_{10} s_1} \\ D_{s_1 s_2} & 0 & D_{s_3 s_2} & \dots & D_{s_{10} s_2} \\ D_{s_1 s_3} & D_{s_2 s_3} & 0 & \dots & D_{s_{10} s_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ D_{s_1 s_{10}} & D_{s_2 s_{10}} & D_{s_3 s_{10}} & \dots & 0 \end{pmatrix}.$$

Sur la diagonale de la matrice, on ne trouve que des zéros puisque ces termes diagonaux correspondent au calcul de la distance euclidienne entre un vecteur et lui-même.

Le profil matriciel est défini comme le vecteur, de taille le nombre de sous-séquences, ne regroupant uniquement que les valeurs de distance les plus basses pour chaque ligne de la matrice. Cela signifie que le profil matriciel ne conserve « en mémoire », pour chaque sous-séquence, que la valeur de distance associée à sa plus proche voisine, au sens non pas temporel mais de la distance euclidienne considérée. Dans l'exemple présenté, l'analyse de la matrice pour les 10 sous-séquences donne le vecteur suivant :

$$mp = \begin{pmatrix} 4,5 \\ 2.0 \\ 3.2 \\ 8.5 \\ 8.8 \\ 7.3 \\ 7.0 \\ 2.6 \\ 3.16 \\ 2.0 \end{pmatrix}$$

avec *mp* pour Matrix Profil.

En conservant uniquement « en mémoire » les distances euclidiennes de chaque sous-séquence avec sa plus proche voisine, ainsi que les positions respectives des sous-séquence et de leur plus proche voisine, on peut déjà mener efficacement la détection de motifs présents dans le signal analysé. Ces distances et ces positions forment la base de

données utilisées par la bibliothèque Stumpy pour pratiquer les multiples analyses proposées.

La figure 37 rassemble les tracés de la série temporelle et de son profil matriciel. En étudiant la courbe du profil matriciel, on constate que l'on a des minimums locaux situés aux échantillons 1 et 9, ces minimums correspondant à une distance globalement moins importante que les autres. Cela nous indique une ressemblance importante entre le motif de la seconde sous-séquence et le motif de la dixième sous-séquence. La ressemblance est ici vérifiable directement à partir du tracé ou de la définition des échantillons de la série temporelle.

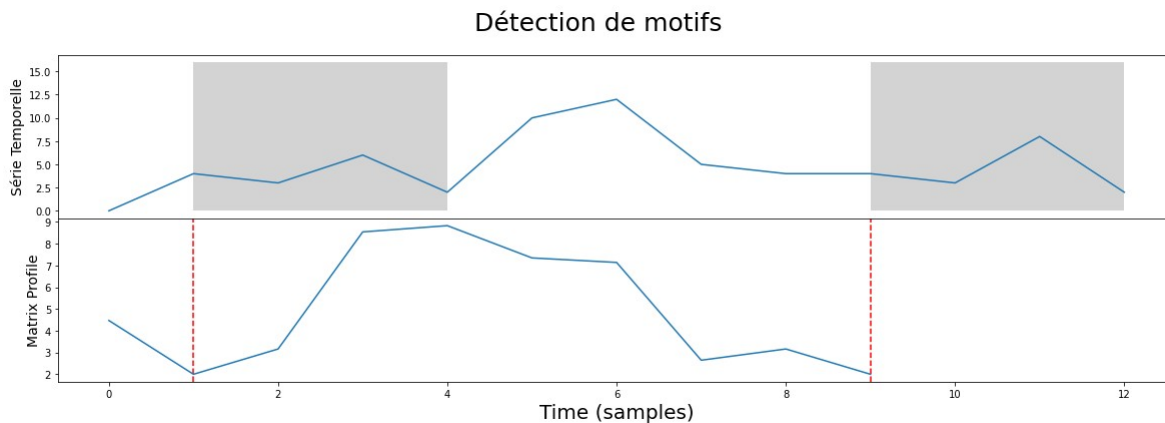


Figure 37: Représentation de la série temporelle (en haut) et son profil matriciel (en bas)

L'exemple précédent illustre la façon dont la bibliothèque Stumpy calcule et utilise ces données pour réaliser les analyses que nous avons effectué pour les sons d'ambiance constituant notre corpus, analyses dont nous rendons compte dans les parties suivantes de ce mémoire.

Il est important de bien noter que les analyses, présentées dans la suite, seront donc uniquement basées sur ces données de distance euclidiennes entre sous-séquences et de position des plus proches voisins.

D'autre part, il semble important de préciser que l'exemple d'obtention du profil matriciel précédent correspond à une version simplifiée du profil matriciel calculé et proposé par la librairie Stumpy. En effet, Stumpy utilise la méthode STOMP (Scalable Time series Ordered-search Matrix Profile), une méthode réduisant la complexité en faisant fi des calculs redondants. Nous n'entrerons pas plus dans les détails de cette méthode dans le cadre de ce mémoire, mais, des informations complémentaires sont exposées dans la documentation de Stumpy. [10]

B – Application au corpus d’ambiances sonores

1. Détection de similarités et/ou d’anomalie

La bibliothèque Stumpy met à notre disposition plusieurs outils pour la détection de similarités et d’anomalies. Nous en retiendrons deux : `stumpy.stump` et `stumpy.scrump`.

Le premier outil, `stumpy.stump`, permet le calcul complet du profil matriciel. Comme on l’a vu précédemment, il permet la détection d’une similarité en comparant l’image du signal avec la distance entre chaque sous séquence et sa plus proche voisine. Les minimas locaux indiqueront ainsi la présence potentielle d’un motif.

Si l’on reprend l’exemple précédent, on observe que l’on arrive au même résultat qu’en calculant nous même le profil matriciel, avec les motifs similaires en gris. (figure 38)

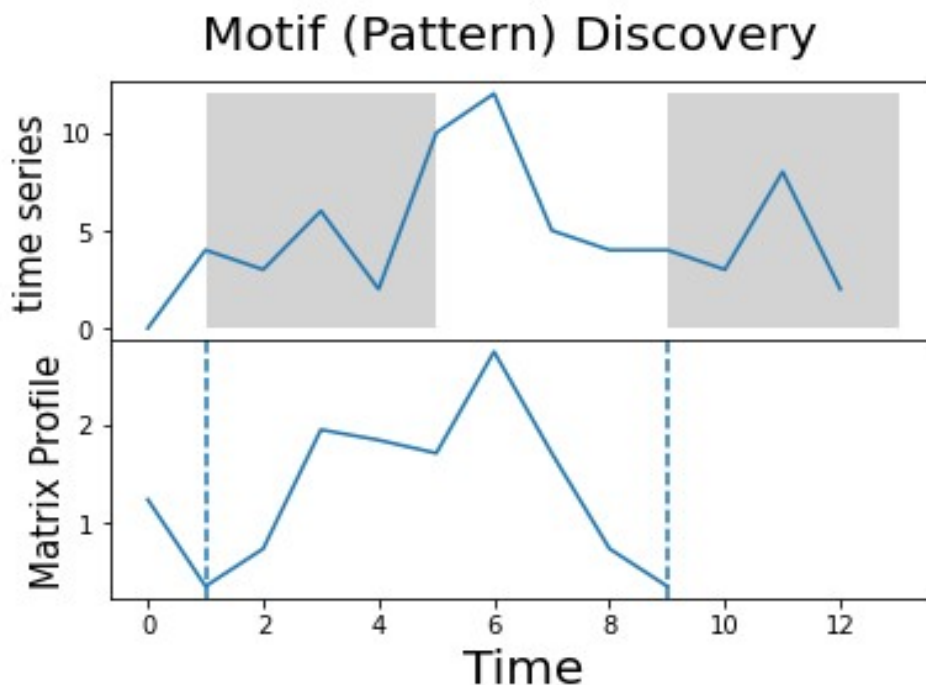


Figure 38: Détection de motifs avec `stumpy.stump`

Il convient de noter qu'avec ces représentations, le motif mis en évidence correspond au motif le plus évident à trouver, à savoir celui dont la distance avec son plus proche voisin est la plus basse. Néanmoins, `stumpy.stump` crée un profil matriciel qui garde en mémoire la distance euclidienne entre chaque sous-séquence avec sa plus proche voisine. En effet, le profil matriciel généré par `stumpy.stump` se présente sous la forme d'un tableau comprenant en première colonne la distance de chaque sous-séquence avec sa plus proche voisine et en deuxième colonne la position de sa plus proche voisine. On peut donc très bien détecter plusieurs motifs d'un seul coup grâce à cette méthode.

Pour la détection d'anomalies, il suffit d'observer quelle sous-séquence présente la plus grande distance euclidienne avec sa plus proche voisine. On doit donc observer les maxima locaux sur le tracé des distances euclidiennes. On a indiqué en rose, pour mieux la distinguer, la séquence la plus éloignée des autres. (cf. figure 39)

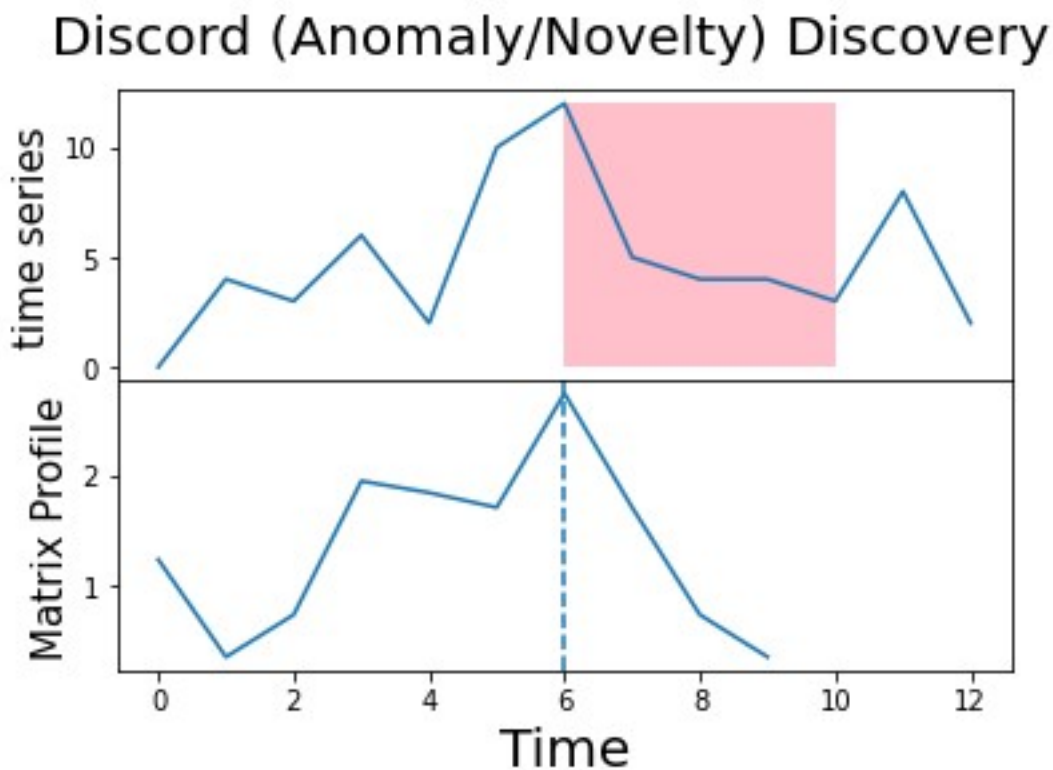


Figure 39: Détection d'anomalie avec `stumpy.stump`

Un problème d'utilisation de la commande `stumpy.stump` demeure néanmoins s'agissant de l'analyse de notre corpus d'ambiances. Si le calcul du profil matriciel par cette commande est plus efficace que l'approche décrite en introduction de cette partie, cette commande demeure trop gourmande en temps de calcul et de ressources mémoire pour l'utiliser sur un son d'une longueur intéressante pour notre étude. En effet, avec la méthode proposée, il faut calculer toutes les distances pour toutes les séquences pour la taille de séquences choisie. Or les essais sur des séquences relativement courtes ont montré une durée de calcul impraticable. Pour un son d'une durée de 13 secondes, et une taille de sous séquence de 10 000 échantillons, le temps de calcul du profil matriciel était d'approximativement 20 minutes. Les essais sur des séquences plus longues se sont parfois conclus par une saturation de la mémoire. Sachant que les sons d'ambiances issus du corpus ont une durée avoisinant une heure, il s'avère que les conditions sont insuffisantes pour effectuer efficacement ces analyses. (La machine utilisée est un PC sous Windows 10 de marque HP, avec un processeur Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz 2.81 GHz et 8,00 Go de mémoire RAM. Les programmes Python ont été effectués via le logiciel Spyder issu du navigateur Anaconda.)

D'autre part, la taille des sous séquences analysées se devait d'être suffisamment courte pour pouvoir offrir une détection efficace des motifs. Cette taille était variable selon les séquences mais toujours dans un ordre de grandeur variant de quelque dizaines à quelques centaines de milliers d'échantillons.

Pour réduire le temps de calcul nécessaire, nous avons eu recours à la commande `stumpy.scrump` qui permet réduire le temps de calcul et fournit une approximation du profil matriciel. Ainsi, nous pourrions observer des profils matriciels pour des signaux plus longs. En effet, comme les sons d'ambiances constituant notre corpus sont des signaux stéréophoniques échantillonnés à 44,1 kHz et les écarts entre les répétitions potentielles sont de l'ordre de la dizaine de secondes, cela signifie que les sous-séquences temporelles que l'on doit considérer comportent beaucoup d'échantillons...

Par commodité et pour réduire le temps de calcul, le profil matriciel a été chaque fois calculé uniquement sur la voie gauche du signal stéréophonique.

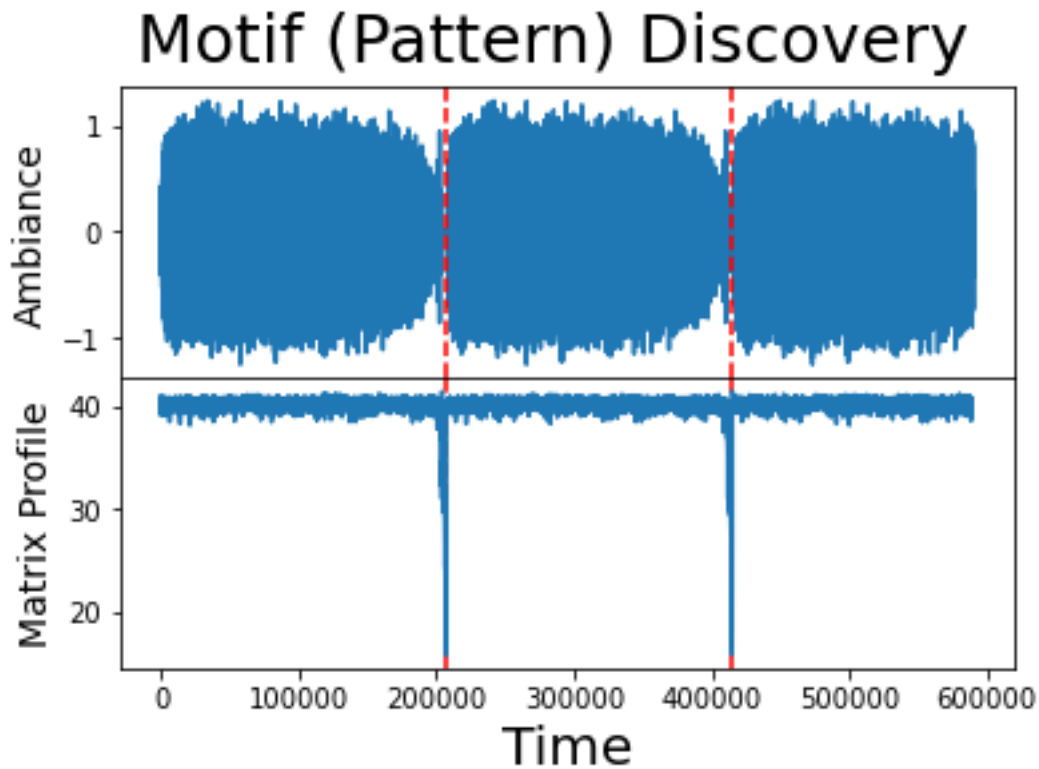


Figure 40: Détection de motifs sur signal sonore plus long (environ 14 secondes)

La figure 40 montre un exemple d'utilisation de `stumpy.scrump` et de l'approximation du profil matriciel qu'il fournit pour un signal caricatural composé d'un seul son similaire de 4,5 secondes (soit un peu moins de 20 000 échantillons) répété 3 fois de suite. On observe bien des minimas locaux aux points de répétition indiquant la présence du même motif, les débuts des répétitions étant indiqués par les lignes pointillées rouges sur le tracé de la forme d'onde de l'ambiance étudiée.

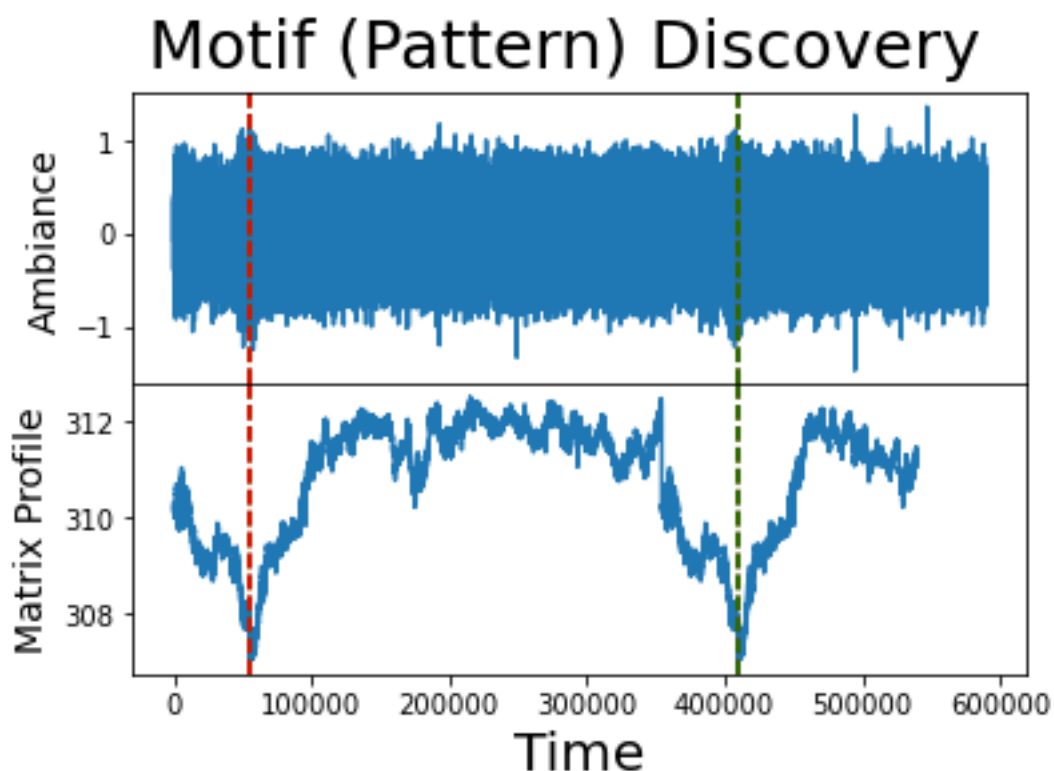


Figure 41: Détection du fond d'air bouclé sur l'ambiance égouts d'Oblivion

La figure 41 illustre l'application de cette analyse au cas de l'ambiance égout issue d'Oblivion. On peut constater la détection d'une similarité pour le fond d'air. Les minimas du profil matriciel indiquent un bouclage du fond d'air, dans le passage étudié, représentant les 12 premières secondes du son. La ligne verticale pointillée rouge indique la position du début du motif détecté et la ligne pointillée verte la position de son plus proche voisin.

L'efficacité de cet outil est donc démontrée. Il a été néanmoins impossible, à ce stade, d'effectuer de telles analyses sur des sons dépassant quelques dizaines de secondes car ces analyses de séquences courtes nécessitent déjà des temps de calcul trop élevés. Il a fallu un temps de calcul de 20 minutes pour calculer un profil matriciel sur une séquence de 35 secondes. On constate donc une amélioration par rapport à la commande `stumpy.stump`, mais cela reste impraticable au regard de la longueur des sons de notre corpus. C'est la raison pour laquelle les tests d'analyse effectués sur les ambiances de The Witcher 3 n'ont

pas permis de révéler la présence significative d'un motif. Comme les sons élémentaires utilisés par The Witcher 3 sont plus longs (en moyenne autour d'une minute), il serait nécessaire pour détecter les motifs d'analyser une séquence d'au moins quelques minutes, ce qui implique un temps de calcul impraticable à notre échelle.

En figure 42, nous présentons un exemple du profil matriciel généré par Stumpy pour l'ambiance village de The Witcher 3. En observant la courbe et la dynamique faible des distances retenues pour le profil matriciel approché (dynamique d'une unité pour des distances moyennes retenues de l'ordre de 311.5), on constate l'absence de minimas très marqués, donc un échec dans la détection de motif de répétition. Cette analyse à été confirmée à l'écoute. Le son analysé dure 35 secondes et a nécessité 20 minutes pour réaliser le calcul du profil matriciel approché.

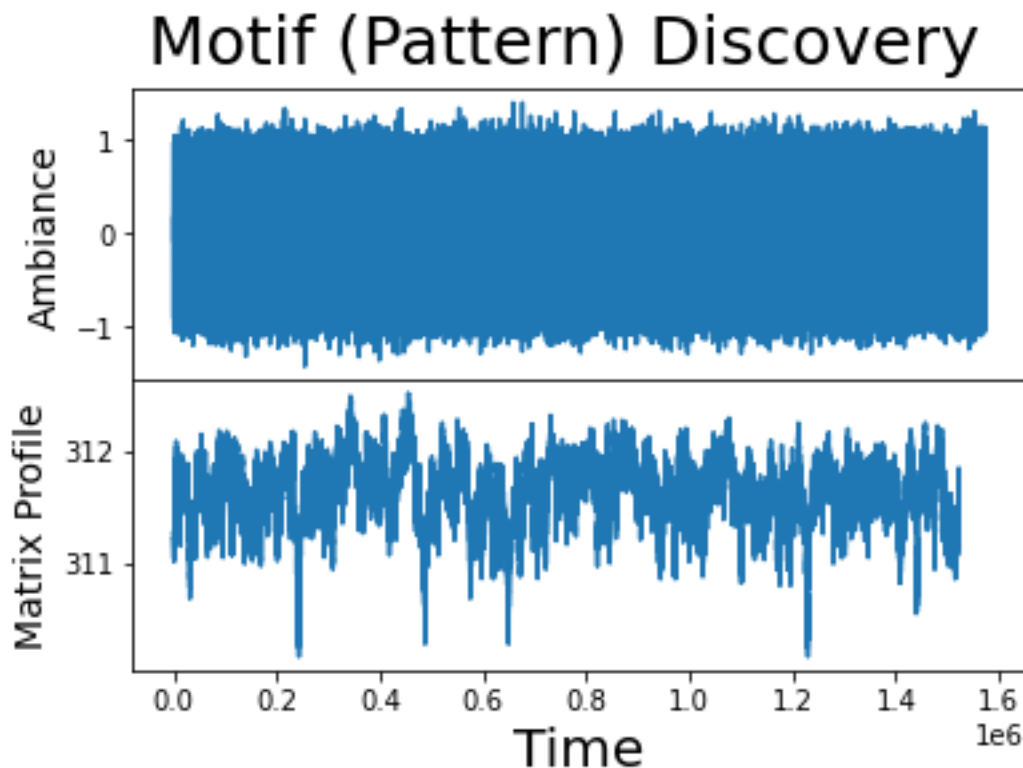


Figure 42: Profil matriciel de l'ambiance village de The Witcher 3

Motif (Pattern) Discovery

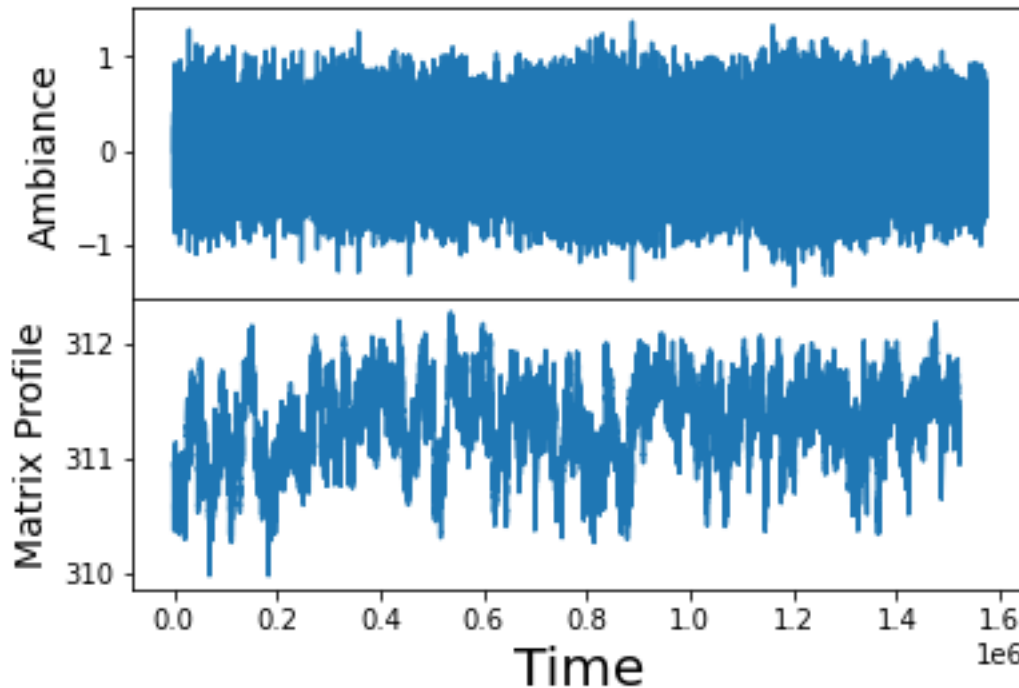


Figure 43: Profil matriciel de l'ambiance grotte de Skyrim

Les mêmes remarques peuvent être formulées pour l'ambiance grotte de Skyrim. Ici (figure 43), les 35 premières secondes du profil matriciel sont insuffisantes pour détecter une réelle similarité dans l'ambiance puisque la dynamique des distances retenues est faible : 2 unités pour une distance moyenne de 311.

Bien que l'approximation du profil matriciel généré par la commande `stumpy.scrump` présente une complexité moins importante que le calcul complet du profil matriciel effectué par la commande `stumpy.stump`, cette commande demeure trop gourmande en termes de temps de calcul et de mémoire vive pour fournir une approximation adéquate et exploitable

pour la détection de motifs. Les essais effectués sur des longueurs plus conséquentes ont plusieurs fois provoqué le plantage du programme. Cet outil demeure néanmoins très prometteur pour notre application, mais il nécessiterait l'intervention d'une phase de programmation plus développée pour permettre une réduction significative des temps de calcul et permettre l'analyse de séquences plus longues. La bibliothèque Stumpy permet néanmoins d'autres applications permettant une analyse de nos ambiances.

Une piste d'amélioration pourrait être la mise à contribution du processeur graphique, permise dans la bibliothèque stumpy par la commande `stumpy.gpu_stump`. Cette commande, impraticable avec notre configuration actuelle, devrait permettre une augmentation significative de la vitesse de calcul.

2. Segmentation Sémantique.

Une autre application intéressante pour notre étude est donnée par la commande `stumpy.fluss`. Cette commande permet, à partir de la donnée du profil matriciel, la détection des ruptures sémantiques au sein d'une série temporelle. Son application à nos ambiances pourrait permettre de distinguer d'un coup d'œil toute modification sémantique modifiant leur continuité, comme un passage jour / nuit, un changement météorologique ou toute autre intervention apportant une modification relativement durable de l'environnement sonore.

Pour ce faire, nous nous concentrerons sur la colonne du profil matriciel nous informant sur la position des plus proches voisins de chaque sous-séquences. En effet, une rupture sémantique notable dans l'ambiance entraîne une variation sensible de la position des plus proches voisines de chaque sous-séquences calculée. Si nous sommes confrontés à une ambiance extérieure où la météo passe subitement de ensoleillée à pluvieuse, l'ambiance sonore sera drastiquement différente entre ces deux phases. Ainsi, la plupart des sous-séquences du côté ensoleillé de l'ambiance trouveront leur plus proche voisine du côté ensoleillé également. De manière similaire, les sous-séquences contenant l'ambiance pluvieuse auront plus de ressemblance avec les autres sous-séquences contenant l'ambiance pluvieuse, et elles auront donc majoritairement leur plus proche voisine dans cette même section.

C'est sur ce principe que se base la commande `stumpy.fluss`. En chaque point du signal, on calcule le nombre de sous-séquences d'un côté de ce point ayant sa plus proche voisine de l'autre côté de ce même point. Ainsi, un point présentant le moins de sous-séquences d'un côté ayant sa plus proche voisine de l'autre aura de grandes chances de correspondre à une rupture dans l'ambiance. Pour pouvoir détecter ces ruptures dans l'ambiance, il faut donc tracer la courbe associée.

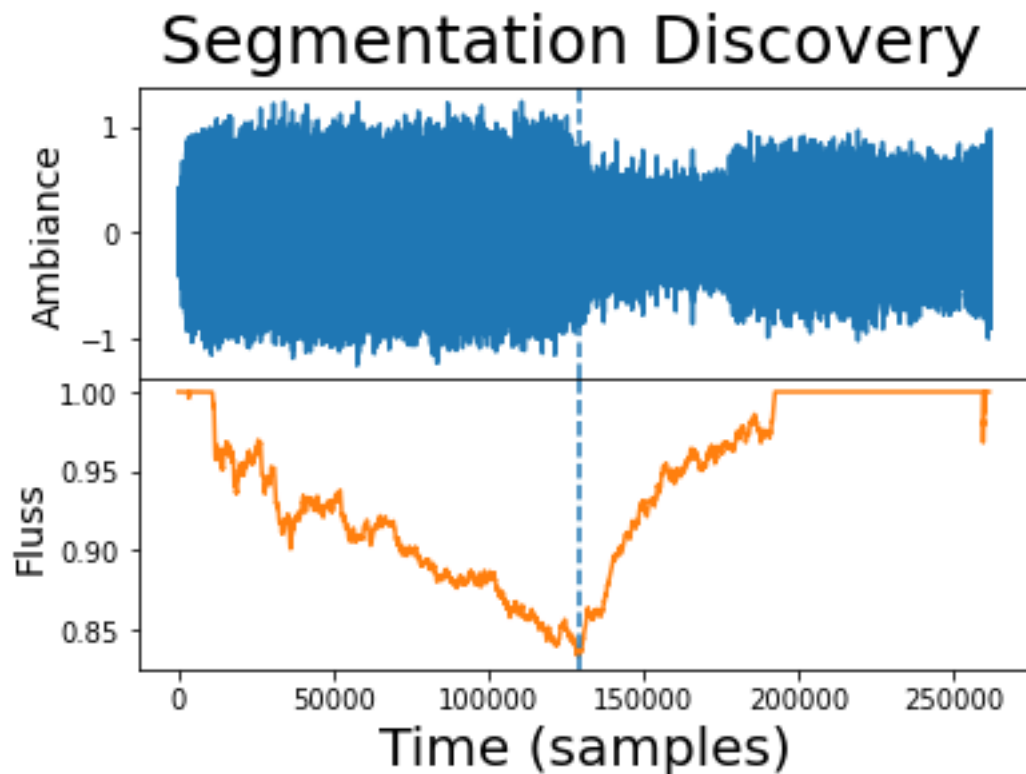


Figure 44: Exemple caricatural de la commande stumpy.fluss

L'exemple de la figure 44 correspond à une ambiance constituée de la succession de deux sons spectralement très différents. Cet exemple, caricatural, permet de mettre en exergue le fonctionnement de la commande `stumpy.fluss`. On observe assez clairement dans la forme d'onde tracée sur la partie supérieure la modification du son qui correspond au minimum de la courbe tracée par la commande `stumpy.fluss`, indiquant qu'on trouve le moins de sous-séquences se ressemblant d'un côté et de l'autre de ce point de rupture.

Cette commande permet de confirmer certaines des observations que nous avons effectuées lors de l'observation des spectrogrammes. On retrouve notamment les modifications météorologiques et les passages du jour à la nuit repérés en analysant les ambiances pour Oblivion et Skyrim (cf. figures 45 et 46).

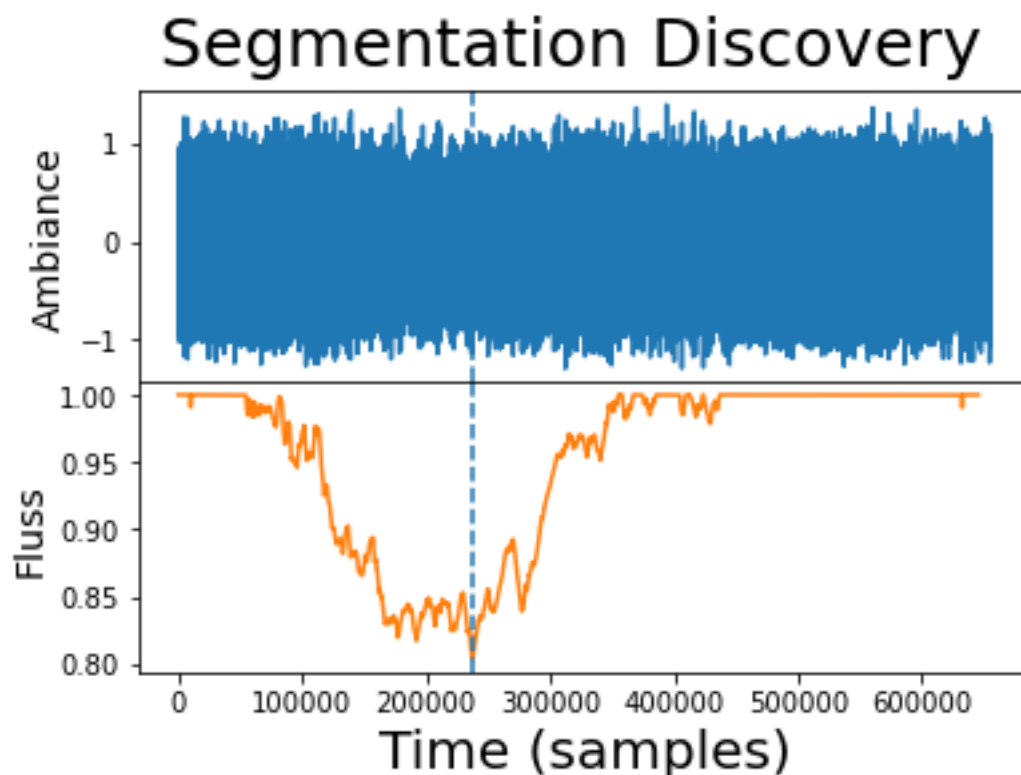


Figure 45: Détection de la rupture sémantique entre l'ambiance jour et nuit pour l'ambiance extérieure d'Oblivion.

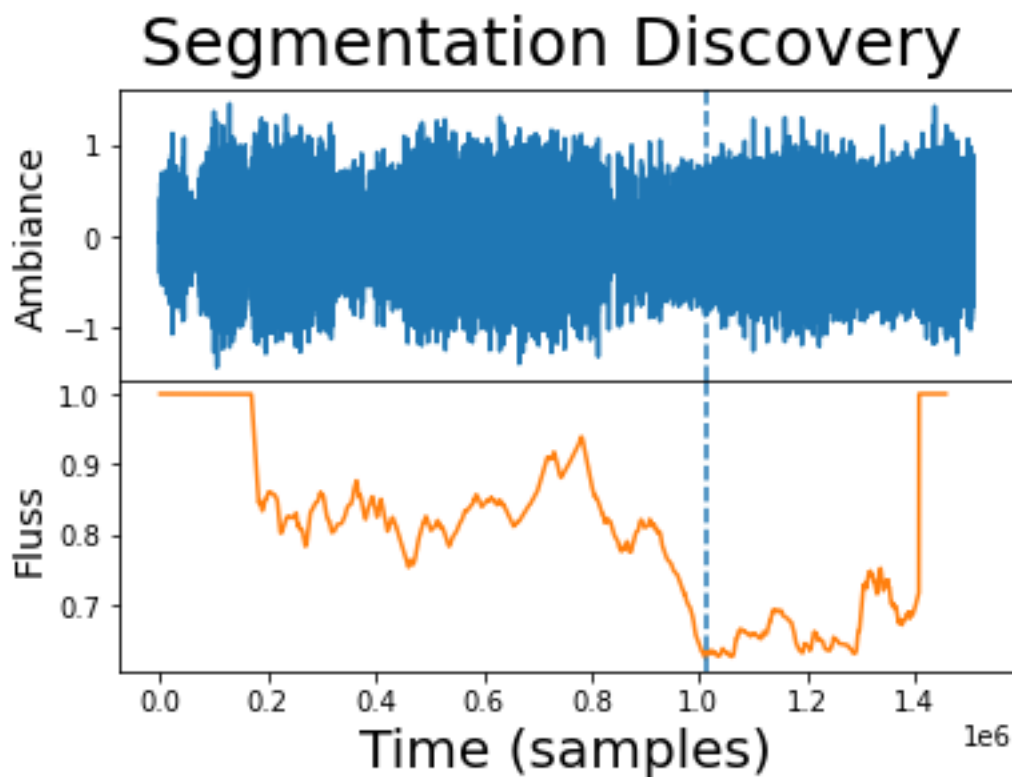


Figure 46 : Détection de la rupture sémantique entre l'ambiance jour et nuit pour l'ambiance extérieure de Skyrim.

3. Détection de motif connu

Le processus d'intercorrélation, étudié dans la partie précédente, nous permettait de révéler la présence d'un son élémentaire issu des banques de données des jeux étudiés. Or, il est possible de réaliser la même recherche à partir du profil matriciel.

La commande `stumpy.match` permet, en effet, de chercher la présence d'un motif précis dans un signal plus long. Ici, plutôt que de calculer les distances de chacune des sous-séquences entre elles, on calcule la distance de chaque sous séquence avec une autre séquence externe prédéfinie, réduisant ainsi drastiquement la complexité et le temps de calcul de l'application.

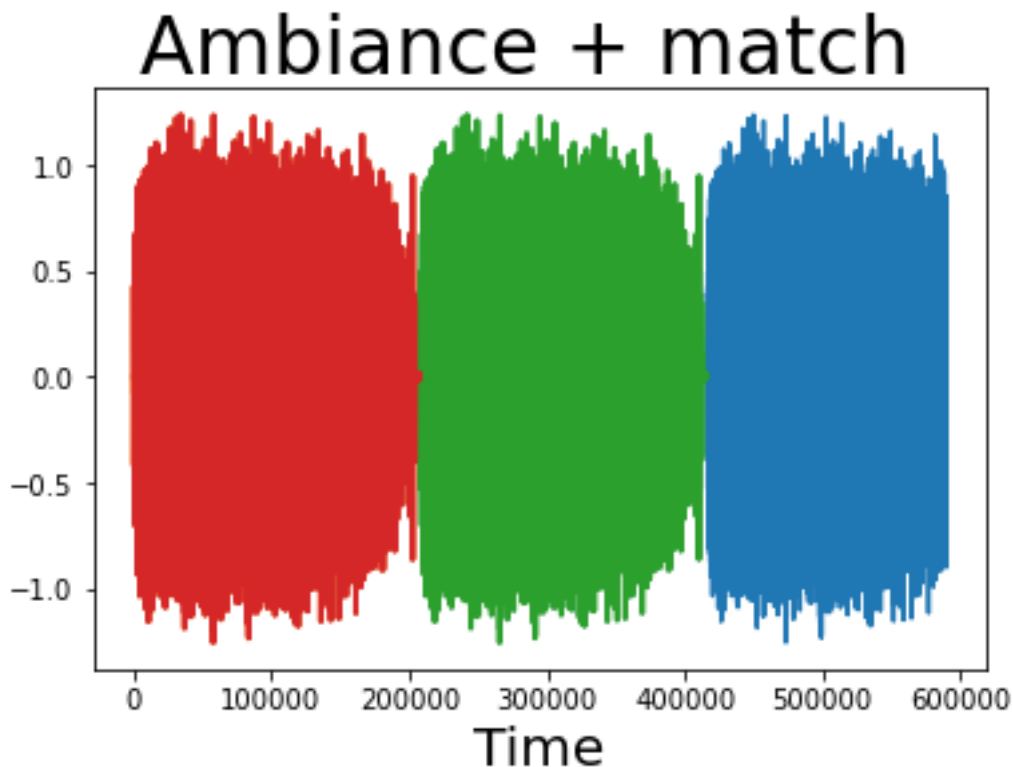


Figure 47: Exemple de la détection de motif avec `stumpy.match`

On peut voir, figure 47, l'exemple sur un signal composé de trois répétitions successives du même son élémentaire. La commande `stumpy.match` a détecté avec efficacité la position des 3 itérations du son recherché (colorées en rouge, vert et bleu).

Cette commande est efficace car elle réduit drastiquement le temps de calcul et nous permet d'effectuer des analyses sur des temps bien plus long de manière plus rapide. On constate néanmoins une imprécision dans la détection des répétitions pour les ambiances diffuses. La figure 48 illustre ce problème potentiel pour le cas d'une ambiance durant environ une minute. On observe un chevauchement du motif détecté (bandes verte et orange) qui n'existe pas dans le signal d'ambiance.

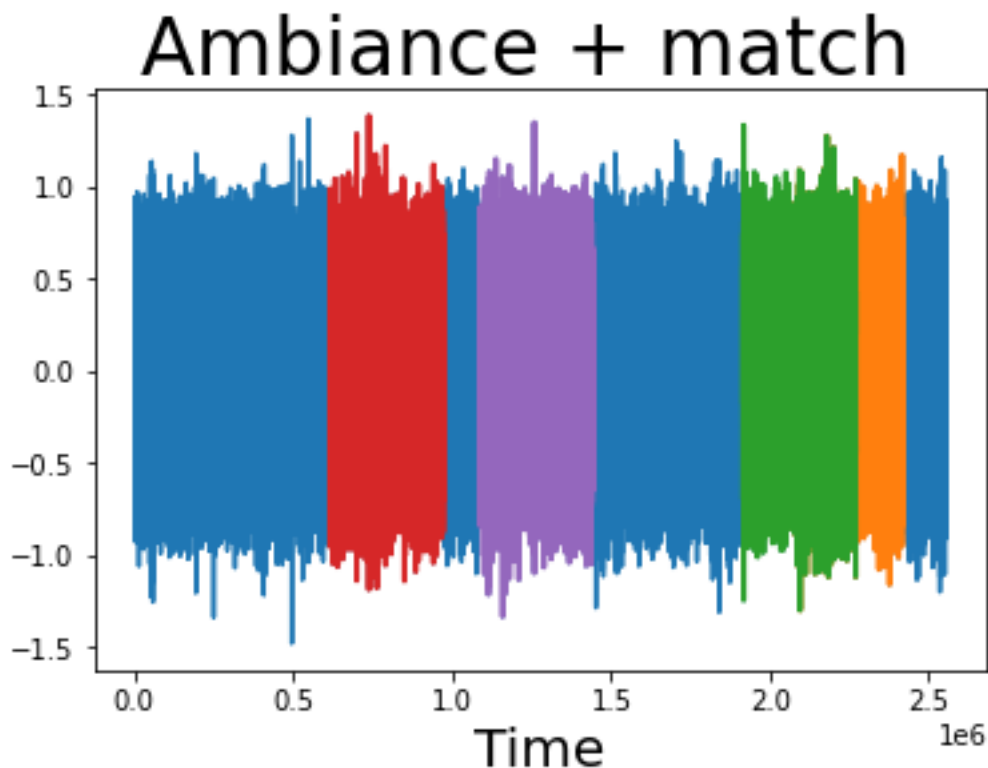


Figure 48: Détection de la boucle d'ambiance égout d'Oblivion

Nous rencontrons ainsi une limite à l'utilisation du profil matriciel pour notre étude. Lors de l'analyse de sons très diffus et bouclés, comme c'est le cas dans l'ambiance égout d'Oblivion, il semble difficile pour la commande `stumpy.match` de détecter avec précision la position exacte du son élémentaire.

Cette commande est aussi limitée par la nécessité de connaître au préalable ce que l'on cherche. Il faut, en effet, indiquer à stumpy.match le motif à détecter dans l'ambiance, ce qui s'avère problématique dans des ambiances très chargées comme celle de The Witcher 3, où beaucoup de sons élémentaires sont utilisés simultanément et adaptés en temps réel à la scène. On risque de rencontrer les mêmes difficultés qu'avec l'utilisation de l'intercorrélation.

4. Perspectives

La bibliothèque stumpy semble très prometteuse mais nous nous heurtons à une trop grande exigence concernant le temps de calcul et la mémoire vive. Néanmoins il est important de souligner ce que cette méthode nous permettrait d'accomplir dans l'hypothèse d'une mise à disposition de moyens plus conséquents.

Le séquençage sémantique issu de l'utilisation de `stumpy.fluss` pourrait être un calcul préalable intéressant en vue d'une automatisation, pour délimiter les zones de recherches. La mise en évidence de rupture sémantique dans une ambiance nous indiquerait les limites des zones à analyser plus finement avec la détection de motifs. Un premier calcul du profil matriciel avec des sous-séquences plus longues nous fournirait alors un séquençage de l'ambiance, permettant ensuite une recherche de motifs uniquement au sein des séquences cohérentes entre elles. Cette détection de motifs pourrait être affinée avec une taille de sous-séquence plus fine sans être parasitée par les séquences sémantiquement trop différentes.

Les commandes `stumpy.stump` et `stumpy.scrump` sont d'une grande efficacité. L'analyse des plus courtes distances euclidiennes repérées dans le profil matriciel permettrait un repérage des sons élémentaires efficaces pour peu que la sous-séquence soit suffisamment petite. Le grand avantage de cette méthode est qu'elle peut repérer plusieurs sons élémentaires différents d'un seul coup et fournir directement leurs positions respectives. L'utilisation de sous-bande fréquentielle par analyse IDS semblerait être un bon moyen, si nécessaire, d'affiner la recherche si l'on désire isoler un son élémentaire précis.

La nécessité de fournir à la commande `stumpy.match` un son élémentaire extérieur pour lui permettre de le détecter dans une ambiance plus longue rend son utilisation comparable à celle de l'« autointercorrélation ». Bien que plus complexe car nécessitant le calcul du profil matriciel, `stumpy.match` présente l'avantage de fournir directement les positions des séquences les plus proches. L'« autointercorrélation » implique une phase de déduction de ces positions dont la commande `stumpy.match` est épargné. Il faudrait cependant définir une distance euclidienne minimale au-delà de laquelle un motif ne sera plus

considéré comme similaire. En effet, `stumpy.match` fournira toujours le nombre demandé de plus proches voisins, fussent-ils très éloignés du son élémentaire.

Le procédé d'automatisation pourrait donc se faire comme suit :

1. Calcul du profil matriciel pour de grandes sous-séquences (quelques secondes) avec `stumpy.fluss`.
2. Découpage de l'ambiance en plusieurs séquences sémantiquement différentes.
3. Calcul du profil matriciel pour de petites sous-séquences (une seconde ou inférieure) avec `stumpy.stump` ou `stumpy.scrump`.
4. Détection de motifs avec le profil matriciel.

Dans le cas où l'on souhaite détecter la présence d'un son élémentaire au sein d'une ambiance plus longue, il semble intéressant pour faciliter la détection d'inclure l'analyse IDS. Le procédé d'automatisation pourrait être celui-ci :

1. Analyse IDS sur le son élémentaire recherché.
2. Détermination de sa ou ses sous-bandes fréquentielles les plus énergétiques.
3. Calcul des signaux de sous-bande nécessaires pour les ambiances à étudier
4. Analyse avec `stumpy.match` et les commandes pertinentes
5. Pour les cas où la détection ne fonctionne pas, essai de la méthode d'« autointercorrélation » pour des sous-séquences comprenant la ou les zones problématiques.

Conclusion

La variabilité des ambiances sonores peut disposer de plusieurs outils permettant sa mesure. Les progrès récents en matière d'exploitation de données et de traitement du signal, matérialisés dans cette étude par la bibliothèque Stumpy, ouvrent un grand champ des possibles pour la mesure et la quantification de cette variabilité. Bien que des difficultés aient été rencontrées dans l'application de ces analyses dues au manque de moyens techniques et structurels, le potentiel usage de machines dédiées plus puissantes et pouvant effectuer des calculs sur un temps long nous permet de croire en une utilisation praticable du profil matriciel.

Il semble que l'établissement d'un quotient de variabilité mesurable nécessiterait la mise en œuvre d'une combinaison des techniques analysées dans ce mémoire. La détection de sons élémentaires peut s'effectuer rapidement et efficacement par la technique d'« autointercorrélation ». Couplé à l'analyse IDS et à la segmentation fréquentielle des sons, nous pouvons envisager une analyse fine de la composition des ambiances les plus complexes. Le profil matriciel tient quant à lui ses promesses dans sa capacité à fournir des informations de séquençage et de détection de motifs. On peut également envisager l'utilisation de la détection de motifs de Stumpy sur les signaux d'intercorrélation découpés fréquentiellement par analyse IDS afin de fournir efficacement les itérations d'un même son au sein d'une ambiance complexe.

Le développement et l'automatisation de l'outil avec des moyens plus efficaces nous mèneront ensuite à l'établissement de la mesure de variabilité. Celle-ci devra impérativement être comparée à des tests d'écoute afin de s'assurer de la robustesse de l'outil et sa cohérence vis-à-vis d'une confrontation usuelle aux ambiances de jeux.

La perspective d'un outil complet de mesure de variabilité sonore peut être envisagée si des études similaires sont effectuées sur les autres éléments sonores composant une bande son de jeu vidéo, à savoir les effets sonores, la musique, voire les dialogues. La

variabilité de ces éléments étant, de par leur nature, soumis à des règles spécifiques, cette étude nécessiterait une adaptation appropriée à chaque type de sons. Bien que les outils étudiés ici soient suffisamment polyvalents pour s'y adapter, des différences d'échelles, de nature sonore, de composition et d'utilisation dans les jeux se devront d'être pris en compte.

Finalement, il est important de souligner que les outils étudiés au sein de ce mémoire pourraient trouver des applications dans d'autres domaines. L'analyse de variabilité étudiée dans ce mémoire est adaptable à tout signal temporel à une dimension. On peut imaginer de nombreuses applications, comme l'analyse de signes vitaux dans le domaine médical, l'analyse de discours en observant les mots utilisés, des analyses géologiques, météorologiques, etc. La mesure de variabilité d'un signal semble avoir des champs d'applications bien plus vastes que l'étude des bandes sonores de jeu vidéo.

Annexes

Annexe 1 : Enregistrement des ambiances des jeux

Il est dans un premier temps indispensable dans le choix préliminaire des jeux de s'assurer que l'on puisse isoler les sons d'ambiance des sons musicaux ou autres effets sonores.

De nombreux jeux ont recours à la musique de manière constante ou parcellaire conjointement à l'utilisation d'ambiance. Si celle-ci n'est pas coupée, il sera alors difficile d'isoler les ambiances pour y réaliser efficacement les mesures et analyses.

Ne seront donc étudiés que les jeux sans musique pendant les ambiances, ou dont la musique peut être coupée. De nombreux jeux, comme ceux de notre corpus, proposent dans les options la possibilité de mixer les différents types de sons (Voix, Musiques, ambiances...). Il suffit alors de couper ce qui ne concerne pas les ambiances.



Figure 49: Écran d'option audio - Oblivion

Il est également conseillé de commencer une nouvelle partie afin de ne pas être dérangé pendant l'enregistrement de l'ambiance par quelques événements scriptés se produisant au bout de x heures de jeux.

L'idéal est de réaliser l'enregistrement grâce à deux ordinateurs. En utilisant la sortie de la carte son de l'ordinateur jouant au jeux en entrée de la carte son de l'ordinateur enregistrant les sons entrant.

N'ayant qu'un ordinateur sous Windows 10 à disposition armé du logiciel Reaper, le processus s'est fait comme suit :

- Définir le mixage stéréo comme périphérique d'entrée.

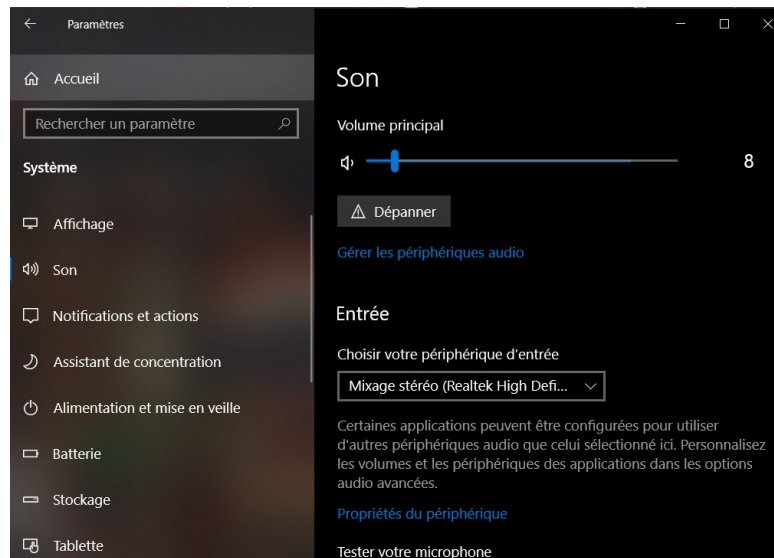


Figure 50: Paramètre son Windows 10, Mixage stéréo en périphérique d'entrée.

- Ouvrir Reaper
- Définir Audio System sur DirectSound, et périphérique d'entrée sur mixage Stéréo
- S'assurer que l'enregistrement soit bien en 44.1 kHz.

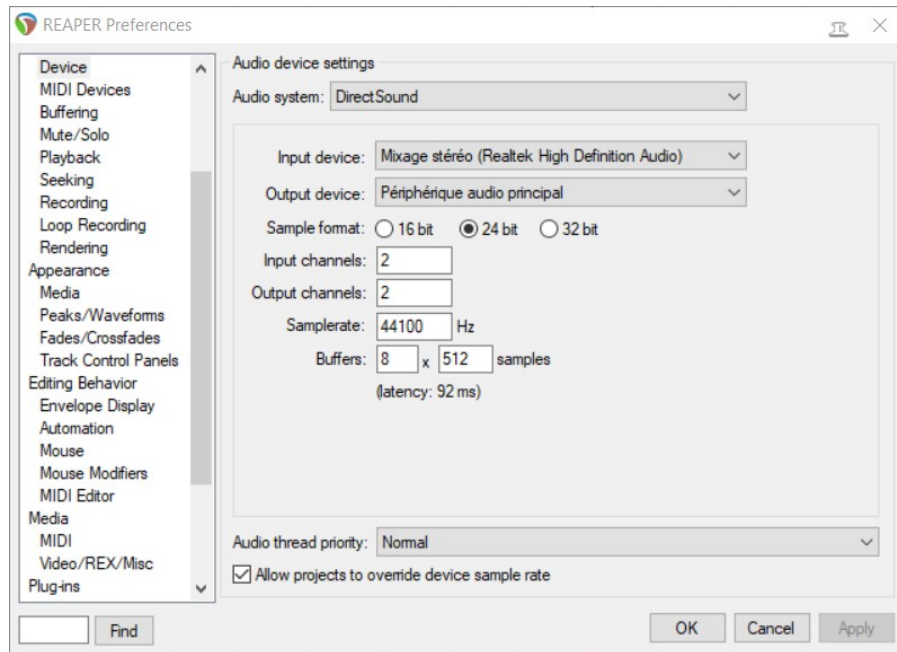


Figure 51: Onglet Preference - Audio - Device de Reaper

- En armant une piste, elle sera par défaut liée au périphérique d'entrée désigné, donc au mixage stéréo de la carte son de l'ordinateur. Sinon, définir l'entrée sur input : Stereo.



Figure 52: Piste armée avec pour entrée le mixage Stéréo

- Une fois la piste armée et l'enregistrement lancé (ctrl + R), on peut lancer le jeu. (S'assurer avant de n'avoir pas d'autre fenêtre ou logiciel ouvert susceptible de générer du son)
- S'assurer dans les options audio du jeu que les sons indésirables sont coupés.
- Placer son avatar à l'endroit désiré.

- Pendant l'enregistrement, limiter au maximum les mouvements du personnage pour ne pas modifier la spatialisation de l'environnement.
- S'assurer de ne pas être interrompu par une veille provenant de l'ordinateur ou du jeu. Il peut être nécessaire pour certains jeux de déplacer légèrement la caméra de manière régulière pour empêcher le jeu de déclencher une animation indésirable suite à une période d'inactivité jugée trop longue.
- Une fois l'enregistrement satisfaisant, quitter le jeu et arrêter l'enregistrement sur Reaper.
- Exporter les sons au format WAV en 44.1kHz. Condition nécessaire à l'utilisation des programmes Python développés lors de ce mémoire.

Annexe 2 : Extractions des sons des fichiers des jeux.

Morrowind :

Les sons sont directement disponibles au format *.wav* dans les fichiers du jeu selon l'arborescence suivante :

Morrowind > Data Files > Sound

Oblivion et Skyrim :

Les sons d'ambiances et effets sonores de Skyrim et d'Oblivion sont compressés dans un seul fichier au format propriétaire *.bsa* (Bethesda Softworks Archive). Pour les extraire, il faut une application tierce comme *bsa unpacker*, disponible en téléchargement sur le site nexusmods.

<https://www.nexusmods.com/skyrim/mods/4804>

Il suffit ensuite d'utiliser cette application pour extraire les fichiers nommés *Oblivion – Sounds.bsa* et *Skyrim – Sounds.bsa*. Ces fichiers sont situés dans le dossier *Data*

The Witcher 3

Comme pour Oblivion et Skyrim, il faut utiliser une application spécifique, à savoir *quickbms*. Les sons du jeu étant compilés en plusieurs fichiers appelés *soundspc.cache* réparti dans les dossiers *content*.

La marche à suivre est décrite dans cette vidéo tutoriel trouvée sur Youtube « *How to extract The Witcher 3 Wild Hunt sound files (WEM)* »

https://www.youtube.com/watch?v=_VwsHAaB3L4

Bibliographie

- [1] **Aletta, F., Axelsson, S., & Kang, J.** Dimensions Underlying the Perceived Similarity of Acoustic Environments. *Frontiers in Psychology*, 8. 2017, <https://doi.org/10.3389/fpsyg.2017.01162>
- [2] **Audiokinetic Inc. (s. d.).** *Wwise Certifications*. <https://www.audiokinetic.com/fr/learn/certifications/>
- [3] **BOITEL, Damien,** *Génération automatisée d'ambiances sonores*. (Mémoire sous la direction de Laurent Millot et Jean-Baptiste Merland) École Nationale Supérieure Louis-Lumière, Section Son, 2007
- [4] **COLLINS, Karen,** *Game Sound, An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*, England, The MIT Press, 2008.
- [5] **GitHub - STUMPY** is a powerful and scalable Python library for modern time series analysis. (s. d.). GitHub. <https://github.com/TDAmeritrade/stumpy>
- [6] **GRIMSHAW, Mark,** *Game Sound Technology and Player Interaction : Concepts and Developments*, England, IGI Global, 2010
- [7] **MERLAND, Jean-Baptiste,** *Son et Jeux vidéos, Comment faire du son un média interactif ?*, CNAM, 2003
- [8] **MILLOT, Laurent,** *Traitement du signal audiovisuel*. Paris: Dunod, 2008.
- [9] **MUEEN Abdullah, KEOGH Eamonn,** *KDD2017 video. (2017, 17 novembre). Time Series data Mining Using the Matrix Profile [Vidéo].* YouTube. https://www.youtube.com/watch?v=1ZHW977t070&list=PLC3JAJ9E3MzefUHzZJVmNns_pbVonIBFc&index=1&t=352s
- [10] **S.M. Law,** *STUMPY: A Powerful and Scalable Python Library for Time Series Data Mining*. Journal of Open Source Software, 4(39), 1504. 2019
- [11] **SENECAUT, Charlie,** *L'apport du découpage IDS dans la compréhension des aspects qualitatifs de la perception auditive et du contrôle de la boucle audio-phonatoire* (Mémoire sous la direction de Laurent Millot et Bernard Auriol), École Nationale Supérieure Louis-Lumière, Section son, 2008

- [12] **SOULIER, Adrien**, *Musique et jeux vidéo. De l'interactivité d'un média singulier*. (Mémoire sous la direction de Jean Rouchouse et Louis Martin), École Nationale Supérieure Louis-Lumière, Section son, 2016
- [13] **Syndicat des Editeurs de Logiciels de Loisirs (SELL)**, *Bilan du marché français 2020*, L'essentiel du jeu vidéo. 2020
- [14] **TRICLOT, Mathieu**, *Philosophie des jeux vidéo*. Paris: La Découverte, 2017.
- [15] **Tutorials — stumpy 1.9.2 documentation. (s. d.)**.
<https://stumpy.readthedocs.io/en/latest/tutorials.html>
- [16] **Wikipedia contributors**. *Roguelike*. Wikipedia. <https://fr.wikipedia.org/wiki/Roguelike> (2021)
- [17] **YEH, C-C. M. ; ZHU, Y. ; Ulanova, L. ; BEGUM, N. ; DING, Y. ; DAU, H. A. ; SILVA D. F. ; MUEEN, A. ; KEOGH, E.** ; *Matrix Profile I : All Pairs Similarity Joins for Time Series : A Unifying View that Includes Motif, Discords and Shapelets*. University of California, Riverside, Universidade de São Paulo, University of New Mexico. 2016
- [18] **ZHU, Y. ; ZIMMERMAN, Z. ; SENOBARI, N.S. ; YEH, C-C. M. ; FUNNING, G. ; MUEEN, A. ; BERISK, P. ; KEOGH, E.** ; *Matrix Profile II: Exploiting a Novel Algorithm and GPUs to break the one Hundred Million Barrier for Time Series Motifs and Joins*. University of California, Riverside, Universidade de São Paulo, University of New Mexico. 2016