**Détection respiratoire – Statut au 2023-03-06**

**Auteur: Vincent Lacasse**

Ce document offre un statut sur le développement d’un système de détection du cycle respiratoire qui utilise le détecteur SLM4X de SensorLogic.

Système

Le système est constitué d’un micro ordinateur Raspberry PI 4 Model B connecté à un senseur SLM4X de SensorLogic via un lien USB à haute vitesse (USB 3).

Logiciel

Un logiciel d’acquisition de données et de traitement est écrit en C++ (<https://github.com/lacasse4/slmx4>). Il roule sur le micro ordinateur Raspberry PI. Il fait l’acquisition des ondes radar émises par le senseur SLM4X et les traite afin d’en extraire le rythme respiratoire d’une personne immobile. La version la plus récente du logiciel porte le nom slm4x/showpsd/src/showfreq.cpp. Une courte description de l’aglorithme est décrite ci-bas.

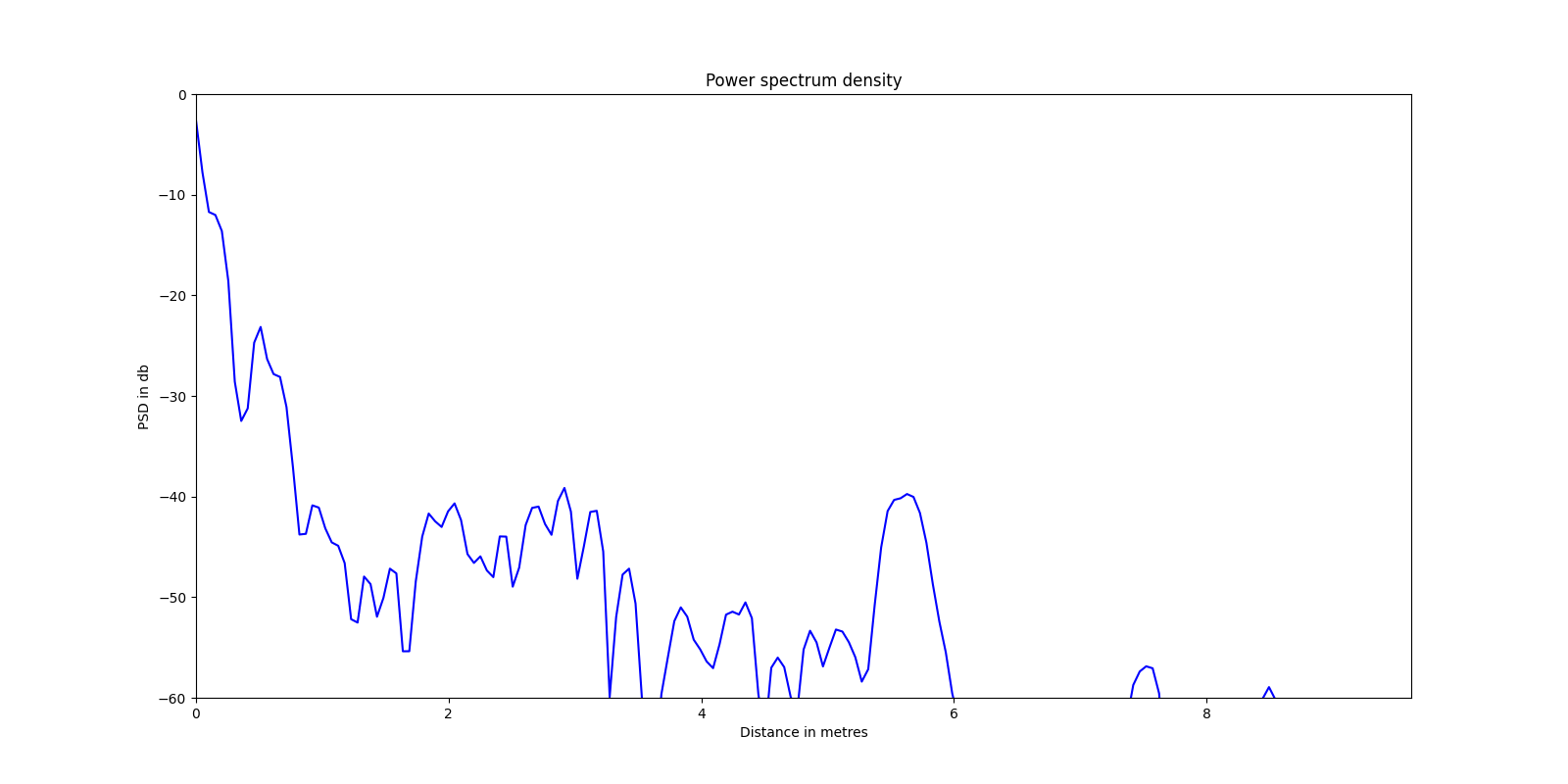
Le repo github contient aussi quelques programmes en Python qui facilitent la visulatisation des données extraites pour le programme showfreq.cpp.

Le senseur SLM4X

Le SLM4X est une radar assez sensible pour détecter les variations créées par une personne immobile et qui respire normalement.

Le radar émet une impulsion et puis détecte le signal réfléchi par un objet qui se situe dans la course de l’impulsion. Cette réflexion est numérisée par le senseur. Les données brutes recueillies par le radar sont constituées de 188 échantillons répartis dans le temps (une trame). Chaque échantillon représente la puissance du signal réfléchi à une distance spécifique. Le senseur peut détecter la présense d’un objet dans un intervalle de 0.25 m à 9.87 m.

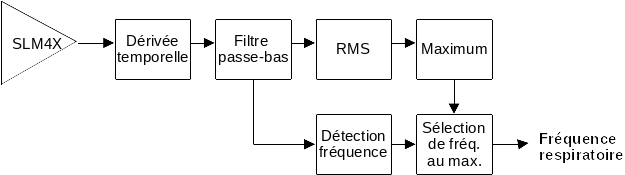
Les données brutes du radar peuvent être exportées vers un ordinateur via une connexion USB. La figure 1 montre un exemple de données brutes numérisées par le radar sur une échelle logarithmique (dB). Dans ce cas, une personne était placée à 50 cm du senseur. On y voit un pic d’environ -22 dB.

Figure 1: Exemple d’une trame du senseur SLM4X (en dB)

Algorithme showfreq.cpp

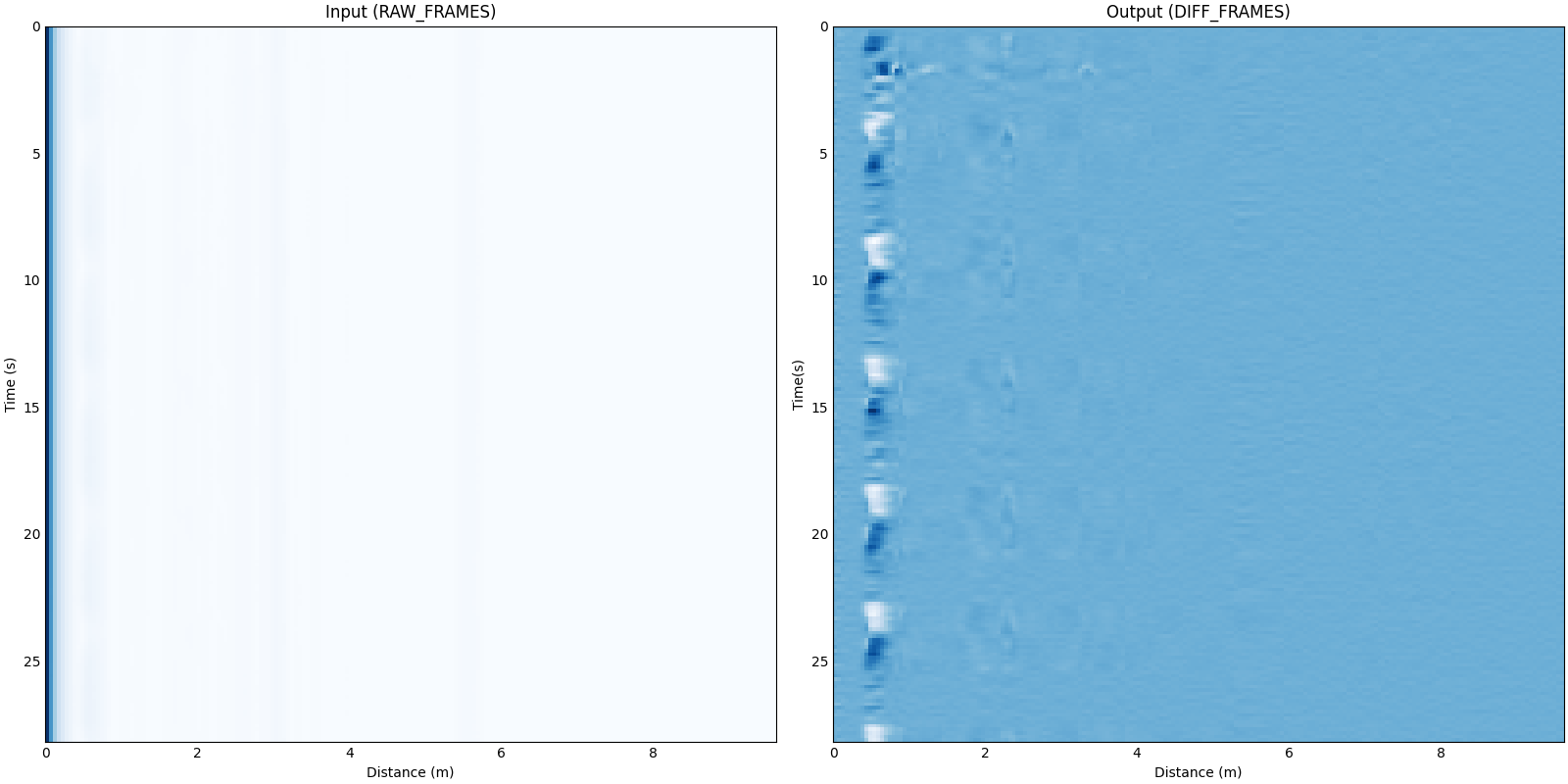
Ce programme showfreq.cpp communique avec le senseur SLM4X, en extrait les données brutes, puis les traite. À ce jour, le traitement effectué permet de mesurer adéquatement la fréquence respiratoire d’une personne immobile.

Le schéma de la figure 2 montre le diagramme des traitements effectués pour trouver la fréquence respiratoire.

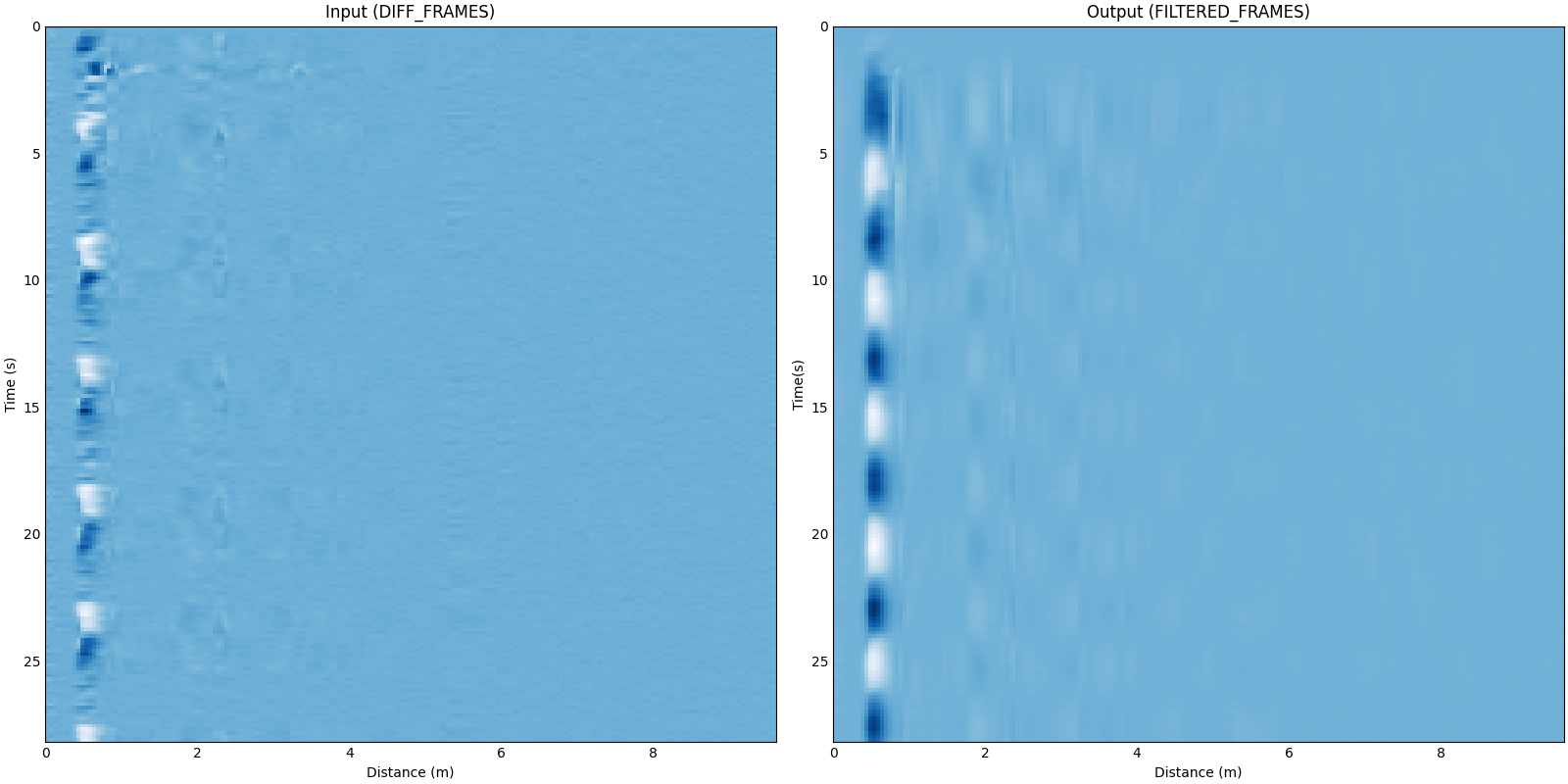
Figure 2: Schéma des traitements pour trouver la fréquence respiratoire

Le senseur SLM4X acquiert continuellement des trames à tout les 141 ms. Dans un premier temps, chaque trame est soustraite de la précédente, ce qui constitue une *dérivée temporelle*. Cette étape permet de mettre en évidence ce qui change entre chaque trame.

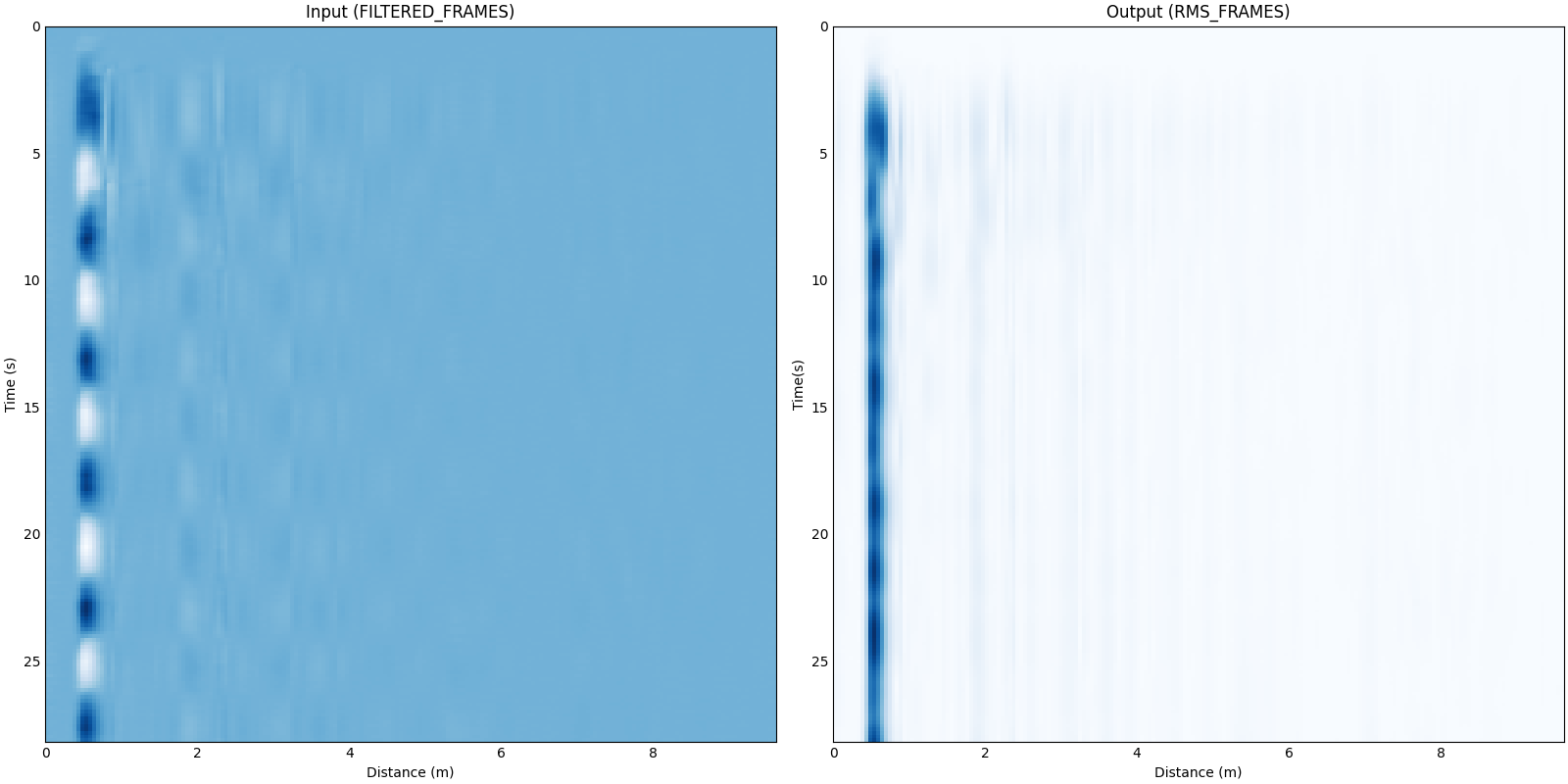
Les images de la figure 3 sont construite en joignant 200 trames successives, ce que représente une période de 28.2 secondes selon l’axe de Y. La période débute en haut de l’image. L’image de gauche (input) est le signal brut du senseur pour un sujet placé à 50 cm du radar. Le sujet respirait à une fréquence approximative de 12 respirations par minute. L’image de droite (output) montre la sortie de la dérivée temporelle. Sur cette dernière image, on remarque déjà les inspirations (bleu foncé) et les expirations (blanc) du sujet. On y voit environ 6 respirations complètes sur une période de 28.2 s, soit un rythme de 12,8 respirations par minutes (RPM).

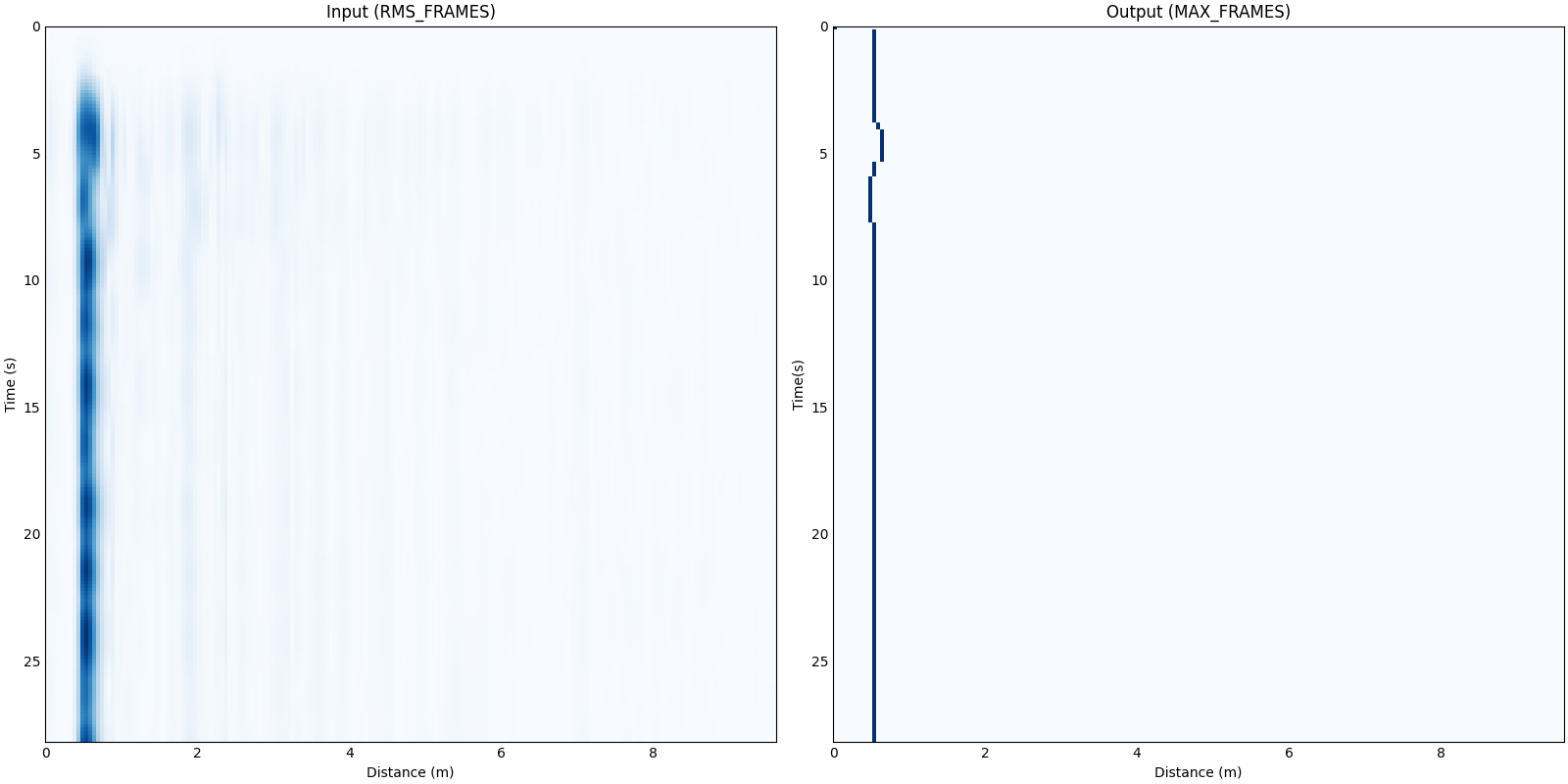
Figure 3: Signal brut (à gauche) et signal dérivé (à droite)

L’étape suivante consiste à appliquer au signal dérivé un filtre passe-bas dans l’axe temporel. Ceci reduit le bruit et offre une meilleure définition au mouvement respiratoire. Cette étape simplifie le traitement des étapes suivantes. La figure 4 montre le signal dérivé à gauche (input) et le signal filtré à droite (output).

Figure 4: Signal dérivé (à gauche) et signal filtré (à droite)

Afin de bien localiser le sujet dans l’espace, nous avons calculé la puissance RMS du signal dans l’axe de la distance. Ce calcul, suivi de la recheche du maximum permet de localiser la portion du signal filtré qui est la plus significative pour le calcul de la fréquence respiratoire. La figures 5 montre le signal filtré à gauche (input) et le signal RMS à droite (output). La figure 6 montre le signal RMS à gauche (input) et la position du maximum de la puissance à droite (output).

Figure 5: Signal filtré (à gauche) et signal RMS (à droite)

Figure 6: Signal RMS (à gauche) et localisation du maximum (à droite)

Finalement, le signal filtré est utilisé pour trouver la fréquence respiratoire. On utilise un algorithme de croisement à 0 (“zero-crossing”) sur le signal filtré pour trouver la fréquence. Cette étape n’est pas illustrée. La frequence est calculée à toutes les distances, car la distance du sujet n’est pas connue à priori. On utilise le point maximum (là où la puissance de la dérivée est à son maxium, figure 6 à droite) comme la distance la plus représentative. La fréquence respiratoire du système est celle qui a été calculée à la distance la plus représentative.

Travaux à faire

Il est maintenant possible de trouver la fréquence respiratoire d’une personne immobile. Cependant, les résultats peuvent grandement varier si la personne bouge un peu (ex: tousser, se déplacer, etc.).

Voici les actions prioritaires pour la suite de ce projet.

* Développer et implémenter un algorithme qui permet de rejeter les situations où le sujet bouge.
* Mettre à jour le README.md pour y ajouter les personnes qui ont contribué au projet (notamment les étudiants de ELE400),
* Réintroduire le code pour communiquer avec Max (les étudiants ont déjà développé et implémenté avec succès ce code)