A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment : from data quality assessment to the production of relevant noise maps

Ayoub Boumchich

THESE DE DOCTORAT

DE
LE MANS UNIVERSITE
SOUS LE SCEAU DE
LA COMUE ANGERS – LE MANS

ECOLE DOCTORALE N° 602
Sciences de l'Ingénierie et des Systèmes
Spécialité : « *Acoustique*»

Par
Ayoub BOUMCHICH

A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment: from data quality assessment to the production of relevant noise maps

Thèse présentée et soutenue à Nantes, le 19 octobre 2023
Unité de recherche : Unité Mixte de Recherche en Acoustique Environnementale (UMRAE)
Thèse No : 2023LEMA1017

Rapporteurs :

Catherine LAVANDIER	Professeure des Universités, HDR, CY Cergy Paris Université, Cergy-Pontoise, France
Claudio GUARNACCIA	Professeur associé, Université de Salerno, Salerno, Italie

Composition du Jury :

Président :	Adrien PELAT	Professeur des Universités, HDR, Le Mans Université, Le Mans, France
Examinateurs :	Jean-Michel POGGI	Professeur des Universités, HDR, Université Paris Cité, Paris, France
	Matthieu PUIGT	Maître de conférences, HDR, Université du Littoral Côté d'Opale, Calais, France
	François SEPTIER	Professeur des Universités, HDR, Université Bretagne Sud, Vannes, France

Dir. de thèse :	Judicaël PICAUT	Directeur de recherche, HDR, Université Gustave Eiffel, Nantes, France
Co-dir. de thèse :	Erwan BOCHER	Directeur de recherche, HDR, CNRS, Vannes, France

Gratitude

Gratitude fills my heart as I reflect upon my doctoral journey, and I cannot begin to express the depth of my appreciation for the incredible individuals who have played an integral role in shaping my path. Their support, guidance, and unwavering belief in me have touched me deeply, and I am forever grateful.

I would like to express my heartfelt gratitude to my thesis director, Judicael Picaut, for his unwavering support and invaluable guidance throughout my journey. From the very beginning, he has been more than a mentor to me; he has been a constant source of inspiration and knowledge. His expertise in the field of acoustics, coupled with his deep commitment to research, has shaped my understanding and shaped me into the researcher I am today. Furthermore, I am immensely grateful for his belief in me, especially during the challenging times of my first year, when the world was grappling with the impact of the COVID-19 pandemic. His unwavering encouragement and moral support played a pivotal role in helping me navigate through the uncertainties and keep my focus on my work. I am also grateful for his willingness to extend his assistance whenever I sought it, be it in research methodologies, writing, or presenting my work.

I would also like to extend my heartfelt appreciation to my co-director, Erwan Bocher, for his exceptional support throughout my PhD journey. His scientific insights, mentorship, and invaluable advice have significantly contributed to the depth and quality of my research. Additionally, his guidance on future career prospects and his unwavering moral support have been instrumental in shaping my professional growth.

I would like to express my sincere thanks to Nicolas Fortin for his exceptional IT support, which played a crucial role in the success of my research endeavors. His expertise and dedication in providing technical assistance were instrumental in overcoming various challenges I encountered during my work.

I am deeply grateful to Benoit Gauvreau, Gwendal Petit, Arnaud Can, and Pierre Aumond for their insightful contributions in the field of acoustics and their invaluable support in relation to the NoiseCapture application. Their expertise and collaborative spirit have enriched my research and broadened my understanding of the subject matter.

I would like to express my gratitude to Matthieu Puigt and François Septier for their advice and contributions as members of the CSI. Their valuable feedback and constructive criticism have significantly contributed to the refinement and enhancement of my work. And as jury members, alongside Catherine Lavandier, Claudio Garnaccia, Adrien Pelat, and Jean-Michel Poggi.

To my fellow PhD candidates, postdocs, and interns (Bill, Claudia, Lise, Gonzalo, Jonathan, Leonardo, Halyna, Arvind, Charlotte), I am grateful for the assistance and support you have provided, and for always being there when I needed it the most. I extend my heartfelt appreciation for the enriching time we spent together, our discussions on various scientific topics, as well as our conversations about culture, food, and even friendly Mario Kart competitions, have made this journey all the more memorable.

I would like to express my gratitude to the entire lab for their warm hospitality and the meaningful exchanges we have had. The collaborative atmosphere within the lab has been instrumental in fostering a stimulating research environment, where ideas flourish and thrive.

To my parents and siblings, your unwavering support has been the bedrock upon which I have built my dreams. From the moment I made the life-changing decision to pursue my aspirations in France, you stood by me, offering your love and encouragement every step of the way. Your belief in me, even when I doubted myself, has been a constant source of strength. I am profoundly grateful for your sacrifices and for instilling in me the courage to pursue my passions.

Lastly, I want to express my deepest gratitude to Hamza Ben Yahya, Sara Fekkak, Imane Houmine and Oumaima Baidouri. You have been my companions, my confidants, and my pillars of support throughout this journey. Your unwavering belief in me, the countless hours we spent together navigating through doubts and challenges, and the laughter and joy we shared have made this experience all the more meaningful. Thank you for being my rock, for lending me your strength when I needed it most, and for reminding me that together, we can conquer any obstacle.

To each and every person who has been a part of my doctoral journey, whether mentioned here or not, I want to express my deepest gratitude. Your contributions, both big and small, have left an indelible mark on my heart. This journey would not have been the same without you, and I am forever grateful for your presence in my life. Thank you for being a part of this incredible and transformative chapter.

Long summary in French

Introduction

Le bruit est considéré comme l'une des principales sources de pollution en raison de la multiplicité de ses sources, en particulier dans les zones urbaines, et de son impact sur la santé. Pour répondre à ce problème, des réglementations ont été mises en place, comme la directive 2002/49/CE en Europe, qui vise à établir un inventaire des nuisances sonores et à proposer des actions pour les réduire. Les cartes de bruit stratégiques sont le principal outil des décideurs dans ce contexte réglementaire. Ces cartes sont en général produites à l'aide de logiciels spécifiques qui intègrent des modèles d'émission de bruit et de propagation acoustique, couplés à des données géographiques et des informations sur le trafic. Cependant, ces cartes manquent de réalisme, en particulier du point de vue de la dynamique temporelle du bruit, de la nature des sources sonores modélisées (uniquement le bruit routier, ferroviaire, et industriel), des valeurs forfaitaires de trafic imposées, ou encore en raison des hypothèses et limitations des méthodes de calcul considérées par exemple pour la modéliser la propagation acoustique. Certaines villes ont également constitué des observatoires de bruit, afin d'avoir une évaluation plus réaliste des environnements sonores. Néanmoins, notamment en raison du coût des points de mesure, ces observatoires ne permettent pas de construire des cartes de bruit avec la finesse spatiale exigée.

Pour remédier à ces limitations, des alternatives ont été proposées, telles que l'utilisation de réseaux de capteurs bas coût, plus abordables pour densifier les points d'observation. En particulier, l'implication des citoyens en tant que collecteurs de données grâce à leur téléphone mobile (considéré comme un capteur bas coût) offre des perspectives intéressantes pour obtenir des données avec une large distribution spatiale et temporelle. Ce type d'approche, issu des sciences participatives, a été largement développé dans la littérature ces dix dernières années, avec diverses applications pour téléphone portable et plateformes de collecte de données, telles que Ear-Phone, NoiseSPY, NoiseTube, Sound Around You, ou très récemment, Silencio...

Le projet récent NoiseCapture (NC), associé à l'application pour smartphone du même nom, s'inscrit dans cette démarche, et étend le concept de sciences participatives à celui de science ouverte. Tous les codes sources, les données et les productions scientifiques sont ainsi disponibles librement, en particulier auprès de la communauté scientifique ou d'acteurs publiques. A la différence des autres approches connues, l'objectif du projet est également d'assurer une collecte de données sur plusieurs années afin de constituer une base de données de référence pour l'étude des environnements sonores sur le long terme.

L'application NoiseCapture fonctionne de la manière suivante : dans un premier temps, l'utilisateur est invité à spécifier son profil (expert, novice ou aucun) puis à procéder à l'étalonnage de son téléphone en utilisant l'une des méthodes proposées par l'application (*via* l'utilisation d'un appareil de référence ou manuellement, ou grâce à l'utilisation d'une méthode statistique d'étalonnage). Une fois le processus d'étalonnage terminé, l'utilisateur peut commencer sa mesure sur la durée et le trajet de son choix. Pendant le processus de mesure, l'utilisateur parcourt un trajet et, à chaque seconde, l'application recueille des données concernant le niveau de bruit, la position de l'utilisateur (indiquée par un système de coordonnées GPS) et sa précision, sa vitesse de déplacement, ainsi que la date et l'heure précise de la mesure. Une fois le trajet terminé, l'utilisateur peut fournir des informations supplémentaires concernant sa propre perception de l'environnement sonore en utilisant une échelle d'agrément, ainsi qu'un certain nombre d'informations sous forme de *tags* ou balises (conditions de mesure, sources sonores présentes pendant la mesure).

Depuis le lancement de l'application le 1er septembre 2017, au 5 juillet 2023, 102473 contributeurs différents ont collecté 111 450 180 mesures (une mesure correspond à un niveau sonore équivalent sur 1 seconde) à travers 445976 trajets de mesure (un trajet correspond à un ensemble de points de mesure collectés pendant le même enregistrement), soit l'équivalent d'environ 1290 jours de mesure en continu, à l'echelle internationnale (plus de 200 pays). En termes de couverture et de densités spatiales ansi que

de distribution temporelle, cette quantité de données reste bien entendu à relativiser, mais cela ouvre de nombreuses perspectives en matière d'évaluation des environnements sonores. Si l'approche présente de nombreux avantages au regard des méthodes plus classiques, il est important d'en mentionner les limites: une mauvaise mise en œuvre du protocole de mesure, l'absence ou le mauvais étalonnage acoustique de l'application, les incertitudes de mesure, la mauvaise utilisation de l'application, ou encore les limites techniques des smartphones... Or dans une perspective de constructution d'une base de données à long terme mais aussi d'utilisation à des fins publiques de l'application NoiseCapture il est important de fournir des informations factuelles, scientifiques permettant de délimiter la qualité des données mesurées. C'est l'objet de ce cette thèse qui s'interesse à la mise en oeuvre de méthodes pour pouvoir utliser les données produites.

Chapitre 1

Le chapitre 1 propose une analyse détaillée de la base de données NoiseCapture afin d'en cerner les potentiels et les limites. L'ensemble des descripteurs collectés par l'application sont étudiés, dans les dimensions temporelles, spatiales et acoustiques. Cette analyse, détaillée dans le **Chapitre 1** de ce mémoire, appliquée sur les données collectées entre 2017 et 2020 (la plupart des travaux de la thèse porte sur l'exploitation de cette base), montre en particulier les limites de la localisation GPS, puisque 32.5% par exemple des trajets de mesure ne sont pas localisés et que 30% des points de mesure sont géolocalisés avec une précision insuffisante (*i.e.* précision supérieure à 15 m). On observe également que, contrairement à l'hypothèse d'utilisation selon laquelle l'utilisateur marchera sur un chemin pendant la collecte des mesures, un tiers des mesures géolocalisées sont réalisées en position stationnaire, ce qui limite la couverture spatiale, mais à l'inverse, augmente la distribution temporelle des mesures. Concernant l'étalonnage des smartphones, cela concerne à première vue une portion relativement importante des smartphones (34.1%) mais les valeurs d'étalonnage observées paraissent parfois en dehors d'un intervalle acceptable (*i.e.* 66.2% des smartphones étalonnés ont une valeur d'étalonnage supérieure à 10 dB). Enfin, l'utilisation des tags est peu répandue, puisque 47.7% seulement des trajets de mesure disposent d'au moins un tag. D'un point de vue de l'évaluation de la perception et de l'évaluation des environnements sonores, la présence de tags en nombre plus important aurait eu un intérêt.

Globalement, cette analyse a clairement mis en évidence certaines incertitudes, irrégularités et incohérences dans les mesures, qui doivent être prises en compte dans toute exploitation des données. Cela met clairement en évidence l'importance d'un contrôle qualité des données et la nécessité d'associer à chaque indicateur acoustique mesuré, une incertitude. Cette analyse a également montré l'intérêt d'intégrer des améliorations dans l'application, pour une prochaine version, par exemple en collectant des informations supplémentaires pour mieux prendre en compte le contexte de la mesure, telle que la position du smartphone pendant une mesure, le mode de déplacement (à pied, en vélo, en voiture), le type de microphone utilisé (microphone interne ou externe)... Le remplacement de la sélection manuelle des tags par une procédure automatique de reconnaissance des sources et des conditions de mesure, serait également une fonctionnalité intéressante.

Chapitre 2

De nos jours, les techniques d'apprentissage automatique sont de plus en plus utilisées pour analyser les données acoustiques provenant de capteurs. Il est clair que cela constitue une méthode intéressante dans le cadre de l'exploitation des données produites par NoiseCapture. De telles approches nécessitent toutefois de disposer de données de référence (*i.e.* des données étiquetées) sur lesquelles la méthode va pouvoir "apprendre". Dès l'origine du projet NoiseCapture, il avait été imaginé la réalisation d'évènements spécifiquement organisés et encadrés pour la collecte de données, appelés NoiseCapture Party, à l'image des "Carto Party" pour la collecte d'informations géospatiales pour alimenter des bases de données cartographiques collaboratives (en particulier la base OpenStreetMap bien connue). Ce type d'évènement permet de collecter un grand nombre de données sur une faible étendue spatiale (un quartier urbain par exemple), sur une période temporelle assez courte, allant par exemple de 1 heure à 1 journée. L'existence de ces données étiquetées ouvrent donc des perspectives dans l'utilisation de méthodes supervisées et semi-supervisées pour l'évaluation des environnements sonores. Cependant, dans le cas actuel, on se heurte à une quantité insuffisante de données disponibles (*i.e.* les données issues des NoiseCapture Parties représentent actuellement 1.2% des trajets de mesure et 0.6% des points de mesure de la base de référence 2017-2020). Nous savons toutefois, par exemple grâce à des échanges entre l'équipe NoiseCap-

ture et d'autres contributeurs dans le monde, ou *via* l'analyse de la littérature scientifique, que d'autres évènements du même type ont été organisés et ont permis de produire des données qui peuvent être considérés comme "de référence".

Ainsi, dans le **Chapitre 2**, nous proposons de mettre en œuvre une méthode permettant de détecter dans la base de données NoiseCapture, les évènements qui seraient similaires à des NoiseCapture Parties, afin d'augmenter les données de référence. L'approche proposée repose sur le regroupement spatial (*spatial clustering*), à savoir la recherche d'un ensemble de points de mesure collectés avec une forte densité sur une zone géographique donnée, et sur un intervalle de temps limité. A cet effet, nous avons utilisé l'algorithme qui se base sur la recherche d'une forte densité spatiale de données, et donc particulièrement bien adapté à notre objectif. La méthode DBSCAN (*density-based spatial clustering*) commence par la sélection aléatoire d'un point de mesure dans la base de données, puis recherche dans un rayon de taille `Eps` un minimum de points `MinPts`. Si cette condition est satisfaite, la procédure est reproduite pour tous les points obtenus à l'étape précédente. Dans le cas contraire, un nouveau point de départ est sélectionné et la procédure est reproduite. A la fin de l'approche, les données sont soit regroupées au sein de *clusters*, soit non regroupées. Dans notre cas, l'originalité de l'application de la méthode proposée réside dans l'utilisation de certaines variables de filtrage préalable : une variable temporelle, qui peut être fixée à une période spécifique pour préserver la similarité temporelle, et une variable de précision, qui permet d'éliminer les données mal localisées pour éviter un biais dans la recherche des points d'un même *cluster*. Enfin, la zone d'étude peut être définie comme une région spatiale spécifique afin de préserver la similarité spatiale.

A titre d'illustration de la méthode, les paramètres `MinPts` et `Eps` ont été fixés à 5000 points et 3 km, respectivement, ce qui a permis de détecter 2046 *clusters* de données dans 68 pays. Les États-Unis regroupent le plus grand nombre de *clusters* (975), suivis par la France (297) et le Royaume-Uni (111), ce qui est un résultat attendu puisque ces trois pays sont considérés comme faisant partie des trois principaux contributeurs à la base de données NC. L'approche a permis de retrouver 19 des 27 NoiseCapture Parties officielles, les NoiseCapture Party manqués étant caractérisées par un nombre insuffisant de données collectées. L'approche a également permis de détecter des *clusters* liés à des évènements organisés à l'occasion de travaux de recherche publiés. On citera par exemple, deux événements dans la région de Kobe (Japon), avec des données collectées par le même utilisateur en juillet et en août 2020, ou encore un événement à Zagreb (Croatie) qui visait à comparer les performances de plusieurs applications de mesure du bruit. A l'inverse, d'autres données publiées dans la littérature n'ont pas été détectées, par exemple celles associées à un événement organisé à Kobe (Japon) qui visait à comparer les niveaux sonores pendant et après le confinement lié à la période COVID. Cet évènement ne comptabilisait que 3500 points, ce qui était inférieur au seuil fixé. Le choix des paramètres s'avère donc important pour réduire ou augmenter le nombre de *clusters* détectés, et *in fine* pour augumenter la base de données de référence. En augmentant le nombre de *clusters* détectés, on prend le risque d'intégrer des évènements qui ne seraient pas réellement des évènements spécifiquement organisés, mais juste une juxtaposition de mesures réalisées par des contributeurs dans la même étendue spatiale et dans la même période, sans lien direct.

Chapitre 3

La donnée principale d'intérêt, parmi toutes celles collectées avec l'application NoiseCapture, est très clairement celle du niveau sonore, à travers plusieurs indicateurs possibles tels que le spectre par bande de fréquence, le niveau sonore équivalent, les indicateurs percentiles (LA10, LA50...). Des travaux préliminaires présentés en annexe de ce mémoire ont montré que trois variables peuvent être identifiées comme contribuant de manière déterminante sur les indicateurs de niveau sonore : (1) l'information spatiale, (2) l'information temporelle, (3) le gain d'étalonnage (*i.e.* la correction acoustique du niveau sonore après étalonnage du smartphone). Concernant les deux premières variables, un traitement simple à permis de filtrer/corriger les données. Le chapitre 3 s'est quant à lui intéressé à améliorer les valeurs d'étalonnage des smartphones.

En effet, dans toute expérimentation acoustique nécessitant l'utilisation d'une chaîne de mesure, l'étalonnage acoustique du système est un préalable, avec éventuellement la qualification d'un niveau d'incertitude en fonction de la classe de mesure de l'appareil de mesure. En général, pour une chaîne de mesure "de laboratoire" ou pour un sonomètre professionnel, l'incertitude est faible et la valeur mesurée, même en l'absence d'étalonnage, n'est jamais très éloignée de la "vrai" valeur. Le smartphone, par définition, n'est pas un équipement de mesure acoustique classique, et, en fonction du modèle et de la marque, peut être caractérisé par des performances techniques très variables. C'est d'autant plus curieux, que dans

le cas d'Android, des conditions obligatoires sont imposées aux fabricants afin de respecter une qualité de mesure en fréquence et en niveau sonore. En pratique, on observe donc de nombreuses différences en termes de mesure de niveau sonore entre smartphones, même en utilisant la même application. Cette observation impose donc d'étalonner les smartphones au travers de l'application de mesure afin d'assurer la cohérence des mesures collectées. Comme indiqué plus haut, peu de smartphones sont étalonnés, et les valeurs d'étalonnage obtenues sont parfois peu physiques, mettant en doute la mise en œuvre du processus d'étalonnage au sein de l'application. Pour toutes ces raisons et plus globalement pour toute mesure de données environnementales avec des capteurs à bas coût, la solution consiste à étalonner *a posteriori* les données. Parmi les solutions possibles, les méthodes d'étalonnage à l'aveugle, basées sur le croisement de capteurs dans une même zone, au même instant, semblent particulièrement utiles pour les données collectées par des projets de science citoyenne tels que NoiseCapture, en particulier dans les zones urbaines, où plusieurs smartphones peuvent se croiser.

Ainsi, le **Chapitre 3** propose la mise en œuvre d'une méthode d'étalonnage à l'aveugle, basée sur la notion de *rendez-vous* entre smartphones, dans la même zone et sur un même intervalle de temps. Ces notions de "zones" et "intervalle de temps" sont des paramètres importants et sont discutés dans ce chapitre. La méthode est basée sur la modélisation des relations entre les capteurs, qui peuvent être écrites sous forme de matrices et peuvent ensuite être résolues comme un problème d'algèbre linéaire. Le comportement de la méthode a ensuite été testé sur des ensembles de données de référence pour lesquels nous avons déjà des informations sur les valeurs plausibles d'étalonnage utilisées. Les résultats montrent un bon comportement de la méthode mais dépendent avant tout du nombre de liens entre smartphones (*i.e.* le nombre de fois que les mesures prises par le couple smartphone/utilisateur se croisent spatialement) et de l'homogénéité de ces liens (*i.e.* le fait que toutes les mesures aient des relations croisées). Afin d'améliorer la méthode, une approche hybride a été proposée, concentrant la méthode d'étalonnage à l'aveugle sur les smartphones avec le plus de liens (un "seuil" minimum de liens est fixé), puis en utilisant une méthode plus simple (moyennage) pour étalonner les autres smartphones sur la base des smartphones étalonnés à la première étape. La méthode hybride apporte de meilleurs résultats que l'approche initiale, et ce, à mesure que la valeur "seuil" du nombre de liens minimum augmente, jusqu'à ce que finalement le nombre de smartphones concernés deviennent insuffisant (dans ce cas, la méthode hybride décroche). Enfin, à titre expérimental, la méthode hybride a été appliquée sur un jeu de données collectées dans la ville de Rezé en France, sur la période 2017-2023, et a montré la pertinence de l'approche pour produire des cartes de bruit "étalonnées".

Conclusion

L'approche collaborative de la collecte de données acoustiques, en particulier dans un objectif production de cartes de bruit, est une alternative intéressante notamment en comparaison avec des méthodes classiques (simulation numérique et observatoire de bruit). La quantité de données collectées avec NoiseCapture est considérable, mais les résultats de l'analyse de la base de données (chapitre 2) montrent que la qualité est extrêmement diverse, voire majoritairement discutable, ce qui nécessite de poser un regard critique sur les analyses produites. L'idéal serait de pouvoir associer une incertitude à chaque mesure, ce qui nécessiterait sans-doute de revoir l'application pour mieux comprendre le contexte de la mesure, en collectant des informations supplémentaires, par exemple sur le type de mobilité pendant la mesure ou la position du smartphone.

L'exploitation de ce type de base de données pourra passer en particulier par la mise en œuvre de méthodes d'apprentissage, supervisées ou semi-supervisées, mais à condition de disposer de jeux de données de référence. Ce type de données existe déjà, collectées dans le cadre de NoiseCapture Party, mais s'avère en quantité insuffisante. C'est la raison pour laquelle nous avons cherché à compléter ces données de référence avec d'autres données de la base, qui auraient été produites de manière similaire à ce type d'évènement (chapitre 3). La méthode qui a été mise en œuvre (méthodologie de *clustering*, DBSCAN) s'est montrée plutôt efficace, et a notamment permis de détecter des évènements réels, mais la nature des *clusters* obtenus dépend beaucoup des paramètres de la méthode.

Enfin, nous avons également travaillé sur l'application d'une méthode d'étalonnage à l'aveugle des données, qui permet de compenser une absence d'étalonnage des smartphones. La méthode semble plutôt pertinente, mais peut sans-doute être améliorée, notamment en considérant les smartphones déjà étalonnés (issus de la base de données de référence) comme des références. Il pourrait également être intéressant de comparer les valeurs d'étalonnage obtenus pour des mêmes modèles/marques, afin de générer éventuellement une base de données d'étalonnage. La méthode pourrait aussi être appliquée, non pas sur le niveau sonore global, mais sur le niveau sonore en bande de fréquence et pour différents

intervalles de niveaux sonores, pour prendre en compte les problèmes de "linéarité" en fréquence et en niveau sonore (problème de seuil pour les niveaux sonores faibles et de saturation pour les niveaux sonores élevés) de certains smartphones. On peut aussi imaginer que progressivement les nouveaux smartphones nécessiteront de moins en moins une procédure d'étalonnage, en comptant sur une montée en gamme technologique et logicielle.

Plus globalement, l'exploitation des tags et de l'environnement géographique (à travers le croisement de données issues d'autres bases de données géographiques) permettrait également d'améliorer les méthodes développées dans la thèse. Cela faisait partie des options possibles en début de thèse, mais qui n'ont pas été entreprises faute de temps. La détection d'anomalies dans la base de données sera également une étape indispensable de manière à écarter des valeurs *a priori* aberrantes.

Ce travail de thèse a donc permis une avancée significative en matière d'exploitation de la base de données NoiseCapture, en précisant le cadre d'utilisation de données à travers l'identification des limites et autres incertitudes, mais également en proposant des solutions pour les corriger. Cela ouvre des perspectives très intéressantes pour produire, par exemple, des cartes de bruit qui présenteraient, sous une forme cartographique à imaginer, des incertitudes associées à des niveaux sonores ou bien des critères de confiance.

Annexe 1

Des travaux préliminaires ont été réalisés à fin d'évaluer les corrélations entre les variables numériques et catégoriques du jeu de données NoiseCapture. Tout d'abord, concernant les variables numériques, les variables prises en compte étaient la vitesse, la précision du GPS, l'orientation du téléphone et l'étalonnage du gain. En utilisant à la fois la régression, la sélection de caractéristiques ainsi que l'analyse exploratoire des données (EDA), il a été démontré que seule l'étalonnage du gain influençait la variable de niveau sonore. Deuxièmement, pour ce qui est des variables catégoriques, les variables sélectionnées étaient le profil, l'heure, la localisation, la méthode d'étalonnage, les tags/sources et le modèle de l'appareil. En utilisant une analyse par ANOVA ainsi que la visualisation des données, il a été constaté que seules l'heure, la localisation et les balises avaient une influence sur le niveau sonore, tandis que l'effet du modèle de l'appareil était négligeable. Par ailleurs, il a été découvert que le modèle de l'appareil et l'étalonnage du gain étaient liés, de même que le profil et la méthode d'étalonnage. Cependant, en raison du faible nombre de trajets comportant des tags, il a été préféré de ne pas les prendre en compte lors de la création d'un modèle pour contrôler la qualité de la variable de niveau sonore.

En ce qui concerne les informations spatiales, nous avons rencontré deux problèmes majeurs : (1) une grande précision du GPS et (2) une absence totale de géolocalisation. Ces deux problèmes se sont manifestés de quatre manières différentes dans nos trajets : (1) au début, (2) au milieu, (3) à la fin et (4) tout au long de le trajet. Pour résoudre ces problèmes, un processus rapide et efficace a été mis en œuvre pour rectifier et affiner la base de données NC. Les trajets qui souffraient entièrement d'un manque de géolocalisation ou qui présentaient une valeur de localisation de GPS supérieure à 15 mètres ont été entièrement éliminés de l'ensemble des données NC. Cependant, pour les trajets où seules des mesures spécifiques présentaient une précision GPS élevée ou un manque de géolocalisation, le reste de la trace a été préservé tout en nettoyant ces points de données problématiques.

En ce qui concerne les informations temporelles, un processus rapide similaire a été mis en œuvre pour corriger et affiner la base de données NC. Il s'agissait de convertir toutes les informations temporelles du temps universel coordonné (UTC) au fuseau horaire local, afin d'assurer la cohérence et la précision de l'ensemble des données.

Contents

Introduction

Context of the study

Noise pollution is a pervasive environmental problem, particularly in densely populated urban areas, and its detrimental effects on human health are increasingly recognized [1]. In response to the urgency of addressing this issue, public authorities worldwide have implemented regulations and directives to mitigate noise pollution. This growing regulatory context emphasizes the need to measure environmental noise and to develop effective strategies to reduce the noise impacts.

The effects of noise pollution on human health are multifaceted and can be profound [2–8]. Prolonged exposure to high noise levels has been linked to various health issues, including hearing loss, sleep disturbances, stress, cardiovascular problems, and impaired cognitive performance. Noise can disrupt sleep patterns, leading to fatigue, decreased concentration, and diminished productivity. Chronic exposure to noise has also been associated with mental health disorders such as anxiety and depression.

Recognizing the severity of these health effects, local authorities and decision-makers are compelled to take an active interest in the issue of noise pollution. They are driven by the imperative to safeguard public well-being and ensure a high quality of life for citizens (*e.g* 2002 European directive [9]). The regulatory framework governing noise pollution varies across different regions and countries, but its common objective is to establish guidelines and standards for noise levels and encourage measures to reduce excessive noise.

To effectively address noise pollution, it is essential to identify its sources. Noise can originate from a wide range of activities and sources, both indoors and outdoors. Common sources of environmental noise include transportation systems (road traffic, aircraft, trains), industrial activities, construction sites, recreational events, and even everyday activities such as household appliances or loud music. Urbanization and increased transportation have amplified the noise generated by these sources, exacerbating the problem.

Given the complex nature of noise pollution and its diverse sources, it becomes crucial to measure and monitor environmental noise levels. In Europe, for instance, the directive 2002/49/EC was established to create an inventory of noise nuisance, propose actions to reduce noise levels, and inform citizens about their exposure to noise [9]. By quantifying the noise present in a given area, local authorities and decision-makers can accurately assess the extent of the problem and formulate appropriate strategies for mitigation.

The availability of data on environmental noise levels enables authorities to identify noise hotspots, prioritize intervention areas, and implement targeted measures to reduce noise pollution. It also allows for informed decision-making regarding urban planning, transportation management, and the design of noise control measures. Additionally, noise measurement plays a vital role in assessing the effectiveness of noise mitigation strategies and evaluating the impact of policy interventions over time.

To aid decision-makers in combating noise pollution, strategic noise maps have become a crucial tool. These maps are typically created using specialized software that integrates noise emission and acoustic propagation models, along with geo-spatial data and traffic information such as 'CadnaA' [10], 'MithraSIG' [11], 'SoundPLAN' [12] or more recently 'NoiseModelling' [13]. Using software to generate strategic noise maps offers several advantages. It enhances efficiency by automating the process, saving time and resources. The software's advanced algorithms provide more accurate results compared to manual calculations. It can handle large and complex areas, thanks to its scalability. Additionally, the flexibility of software allows customization to specific requirements and integration with other data sources. Software-based noise mapping enables scenario analysis, aiding in decision-making for noise reduction strategies and mitigation measures. Overall, it improves efficiency, accuracy, scalability, and flexibility in noise management and planning processes. However, it can be complex requiring technical expertise and a learning curve, and time consuming for acquiring and preparing the necessary data.

Simplifications and assumptions made by the software, as well as the quality of input parameters may also introduce some level of error or uncertainty; interpreting and validating the results require careful consideration and cross-referencing with real-world measurements. Lastly, costs, both for software and data acquisition, may be involved.

Another approach to evaluate the noise impact, is to use noise observatory such as 'Bruitparif' [14] and 'Acoucité' [15], which are located in Paris and Lyon respectively. The observatory is set-up with professional (class-1) microphones, in order to collect continuous noise data for a period of days or weeks, then analyze the data using signal processing and statistical techniques for example to identify noise patterns, hot-spots, and evaluate noise mitigation measures. Generating data from noise observatories offers advantages such as accurate measurements, representative samples, real-time monitoring, and detailed analysis of noise patterns, but it comes also with potential inconveniences including costs and setup, limited spatial coverage (not enough data to generate a detailed noise map for example), data processing complexity, maintenance requirements, and susceptibility to environmental factors.

In an effort to address the limitations of traditional tools, alternative approaches have been proposed. One approach involves utilizing more affordable sensor networks ('low-cost network') to densify observation points, allowing for a more realistic description of noise environments [16–19]. Among them, the idea of using smartphones as acoustic sensors and citizens as contributors emerged at the end of the 2000s [20–22], with the increasing capabilities of smartphones to perform environmental acoustic measurements [23]. It was followed by several works that have given rise to specific noise and soundscape crowdsourcing type applications and platforms (*e.g.* Ear-Phone, NoiseSPY, NoiseTube applications [24–26]), and particularly the NoiseCapture (NC) approach [16–19], a part of the Noise-Planet project [27]. Lastly, one can also cite projects that use citizen-contributed data from location-based social networks to create maps of sound environments [28, 29].

NoiseCapture: a crowdsourcing approach for the evaluation of the sound environment

The NoiseCapture project, which builds upon the advancements in Information and Communication Technologies (ICT), harnesses the power of smartphones as acoustic sensors and engages citizens as data collectors, fostering a participatory and open science approach. By leveraging smartphones as measuring instruments, citizens can collect noise data along specific paths and share it with the NoiseCapture community. This crowdsourced database contains standardized noise indicators (*i.e.* equivalent noise level LAeq, percentile indicators (LA10, LA50, LA90), min and max values of the sound level...), user perceptions of noise sources and soundscape quality, along with additional information such as measurement time, GPS coordinates, and user speed.

The NoiseCapture application stands out due to its unique combination of advantages derived from classical approaches. It offers scalability, allowing for the efficient handling of large datasets and complex areas. The application also provides flexibility, enabling customization to specific requirements and integration with various data sources. Moreover, it ensures accuracy by leveraging reliable measurement techniques and robust analysis methods. One notable strength of NoiseCapture is its ability to capture a representative sample *e.g.* road traffic, constructions, music, people, animals...) of ambient noise levels, leading to more comprehensive and reliable results. Additionally, the application offers real-time monitoring capabilities, allowing for timely updates and immediate access to noise data for informed decision-making. An intriguing aspect of NoiseCapture is its ability to achieve these advantages without suffering from the inconveniences typically associated with classical approaches. It minimizes challenges related to costs and setup, offers broader coverage by leveraging modern technologies, simplifies data processing through automated workflows, and reduces maintenance requirements. Overall, NoiseCapture stands as a professional and developed solution that combines the best features of classical approaches while mitigating their inherent inconveniences. It provides a robust and efficient platform for noise monitoring and analysis, offering scalability, flexibility, accuracy, representative sampling, and real-time monitoring capabilities.

The PhD thesis topic

Since its launch, the NoiseCapture project has accumulated a substantial amount of data, providing the opportunity for extensive spatial and temporal analysis. With over 445,000 tracks and 111 million

measurement points contributed by more than 102,000 contributors across 200 countries, the database represents a valuable resource for studying sound environments. Nevertheless, NoiseCapture dataset suffers from issues related to its quality. These issues can take form of uncertainties (*e.g.* the quality of smartphone calibration, point geolocalization in case of high GPS accuracy...); anomalies (*e.g.* negative speed, negative noise levels, incorrect time of measurement...) and missing values (*e.g.* geolocalization, gain calibration, soundscape source, perception of soundscape...). The utilization of such a vast dataset requires a comprehensive understanding of its inherent limitations and potential uncertainties. So, what are these issues and limitations? Is the NoiseCapture dataset comprised of a reference dataset that can be utilized to enhance the application, improve data quality, and collect contextual information in the future? What are the key parameters or features in the NoiseCapture dataset that require correction and control in order to generate high-quality data for relevant noise maps? How can these corrections and controls be implemented? Thus, the aim of our thesis was to provide answers to these questions, in order to obtain a 'Qualified' database, to open up the field of data exploitation (figure 1).

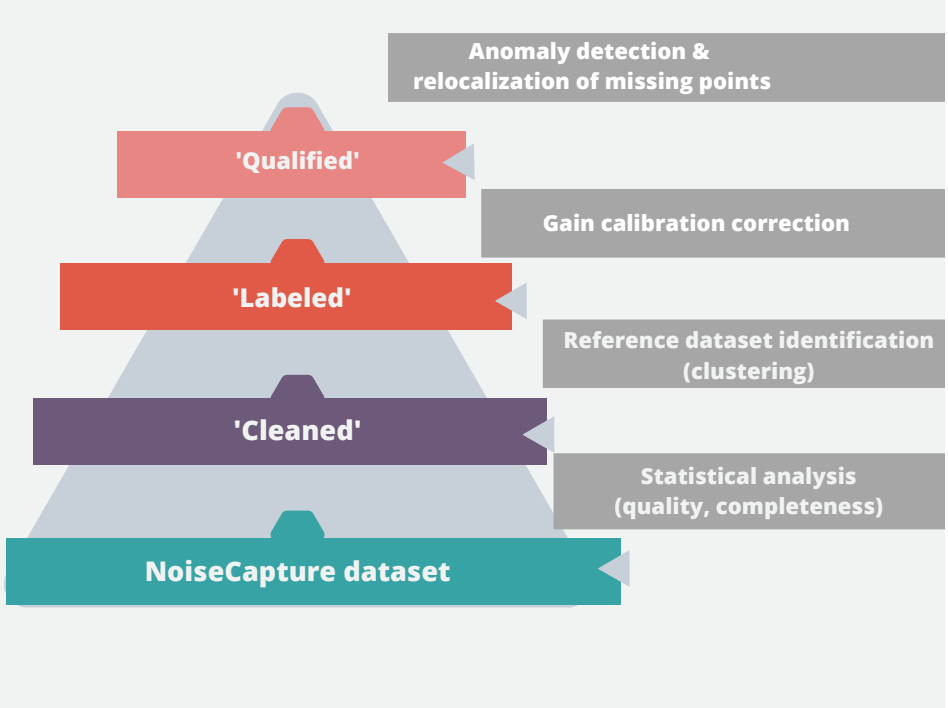


Figure 1: Organization of thesis work.

Manuscript presentation

The **first chapter** details an overall statistical analysis that was performed on NoiseCapture dataset by examining the data gathered over a period of three years. It offers a comprehensive assessment of the data's quality, consistency, and completeness. Furthermore, we delve into the inherent limitations associated with each data point. These limitations can be attributed to factors such as the nature of the data, the measurement protocol, the technical performance of the smartphone, the absence of calibration, and the presence of anomalies in the collected data. The aim of this statistical analysis is to enable everyone to fully utilize the database while maintaining complete control over it. This endeavor yielded a scientific article that has been published in 2021 in the *International Journal of Environmental Research and Public Health* Journal [30]. The chapter reproduces the article as published.

Although this work achieved a preliminary cleaned dataset, it also made the need to control and enhance NoiseCapture dataset quality a must. Many of machine learning approaches require reference (*i.e.* labeled) data to be applied, which makes searching for such a data in NoiseCapture dataset a priority. In the context of NoiseCapture, this reference data can be generated during specific organized events (NoiseCapture Party), where participants undergo specialized training to gather measurements. However, the available data from these events are insufficient in quantity to create a comprehensive reference database, thus requiring supplementation. Considering that other communities worldwide also

utilize NoiseCapture, there is a desire to incorporate the data they have collected into the learning database. To achieve this, it is crucial to identify and extract this data from the huge amount of available information (figure 1). Thus, the **second chapter** proposes utilizing a classical clustering method called DBSCAN, which is well adapted to exhibit higher measurement density in both space and time, as in a NoiseCapture Party. We initially tested this method on the existing NoiseCapture Party data and subsequently applied it on a global scale. By adjusting the DBSCAN parameters, multiple clusters have been detected, each displaying distinct typologies. This work has been published in 2022 in the *Sensors* Journal [31]. The chapter reproduces the article as published.

After succeeding in acquiring more reference data, we could finally tackle the issue of anomalies in NoiseCapture dataset (*i.e.* the final step of the progress figure 1). The **Third chapter** proposes an adaptation of a blind calibration method to the data obtained from the NoiseCapture smartphone application. The method involves modeling the relationships between sensors, which can be expressed in matrix form and solved as a linear algebra problem. To assess the effectiveness of the method, we conduct tests and comparisons using NoiseCapture datasets that already include information about the calibration values of certain smartphones. As an experimental application, we utilize the method on a dataset from a French city, resulting in the creation of a calibrated noise map based on the collected raw data. This work resulted in an article submitted to the *Sensors* Journal, and, here again, which is reproduced in the corresponding chapter as it submitted.

The **last chapter** culminates in a comprehensive global conclusion that summarize the findings from all previous chapters and provides insights into the future prospects of the research.

Although each chapter appears as the insertion of a published or submitted article, the present thesis manuscript follows the logic presented in figure 1.

Chapter 1

A smartphone based crowd-sourced database for environmental noise assessment

International Journal of
Environmental Research and Public Health

MDPI

Article

A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment

Judicaël Picaut [1,*], Ayoub Boumchich [1], Erwan Bocher [2], Nicolas Fortin [1], Gwendall Petit [2] and Pierre Aumond [1]

[1] Centre for Studies on Risks, The Environment, Mobility and Urban Planning (CEREMA), Research Unit in Environmental Acoustics (UMRAE), French Institute of Science and Technology for Transport, Development and Networks (IFSTTAR), University Gustave Eiffel, F-44344 Bouguenais, France; ayoub.boumchich@ifsttar.fr (A.B.); nicolas.fortin@univ-eiffel.fr (N.F.); pierre.aumond@univ-eiffel.fr (P.A.)
[2] Lab-STICC CNRS UMR 6285, IUT de Vannes, 8 Rue Montaigne, BP 561, CEDEX, F-56017 Vannes, France; erwan.bocher@univ-ubs.fr (E.B.); gwendall.petit@univ-ubs.fr (G.P.)
* Correspondence: judicael.picaut@univ-eiffel.fr

Abstract: Noise is a major source of pollution with a strong impact on health. Noise assessment is therefore a very important issue to reduce its impact on humans. To overcome the limitations of the classical method of noise assessment (such as simulation tools or noise observatories), alternative approaches have been developed, among which is collaborative noise measurement via a smartphone. Following this approach, the NoiseCapture application was proposed, in an open science framework, providing free access to a considerable amount of information and offering interesting perspectives of spatial and temporal noise analysis for the scientific community. After more than 3 years of operation, the amount of collected data is considerable. Its exploitation for a sound environment analysis, however, requires one to consider the intrinsic limits of each collected information, defined, for example, by the very nature of the data, the measurement protocol, the technical performance of the smartphone, the absence of calibration, the presence of anomalies in the collected data, etc. The purpose of this article is thus to provide enough information, in terms of quality, consistency, and completeness of the data, so that everyone can exploit the database, in full control.

Keywords: environmental noise; crowd-sourcing; smartphone application; data analysis

Citation: Picaut, J.; Boumchich, A.; Bocher, E.; Fortin, N.; Petit, G.; Aumond, P. A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *Int. J. Environ. Res. Public Health* **2021**, *18*, 7777. https://doi.org/10.3390/ijerph18157777

Academic Editor: Paul B. Tchounwou

Received: 30 April 2021
Accepted: 9 July 2021
Published: 22 July 2021

1. Introduction

Noise is a very significant source of pollution, particularly in urban areas, with significant effects on health. The fight against noise is a fundamental societal and health issue, to which the public authorities are trying to respond by putting regulations in place. In Europe, for example, the directive 2002/49/EC aims to establish an inventory of noise nuisance, to propose actions to reduce nuisance and to communicate to citizens about their exposure to noise [1]. In this regulatory context, the main tool for decision-makers is the production of strategic noise maps.

These maps are generally produced using specific software, integrating noise emission and acoustic propagation models, coupled with geospatial data and traffic information. Although these maps are limited by the calculation assumptions and the quality of the input data, they make it possible to assess the broad outlines of a noise distribution in a city and to evaluate the effect of action plans to reduce noise. However, they generally lack realism, particularly from the point of view of the temporal dynamics of noise. Conversely, noise observatories, consisting of a large number of acoustic sensors, offer a more realistic description of noise environments. However, the limitation of the number of sensors, for technical and cost reasons, does not allow carrying out noise mapping with a sufficient spatial step.

Faced with these observations, alternatives have been proposed. In particular, the use of more affordable sensor networks has been investigated, allowing to densify the observation points [2]. Another way consists in the involvement of citizens as data collectors, in a crowd-sourcing approach. For example, the Smart Citizen System project proposes a low-cost sensor specifically dedicated to collect noise data by citizen action [3]. Considering a soundscape approach, data produced by people on location-based social networks can also be analyzed to produce maps of the sound environment, like with the Chatty maps experiment [4] or more recently by Gasco et al. [5]. The Sound Around You project is another example, by proposing a web interface to collect soundscape recording and opinions [6]. Nowadays, among all the citizen science-oriented approaches, the one based on the use of smartphones is undoubtedly the most developed in the literature. In particular, Santini et al. have demonstrated the capabilities of a smartphone to perform environmental acoustic measurements [7]. It was followed by several works that have given rise to specific noise and soundscape crowd-sourcing type applications and platforms (see Ear-Phone, NoiseSPY, and NoiseTube applications, respectively, in [8–10]). What is interesting in these first works is that, despite the technical limitations of the time (i.e., smartphones with limited technical capabilities and resources), almost all the topics related to this issue had already been discussed: smartphone calibration, data quality, noise maps reconstruction, contextualized data collection (perceptual data), the need for a complementary web interface, the need to know the context of the measurement, the implementation of specific events to organize data collection, contributors privacy, motivation of contributors, etc. Subsequently, other contributions have appeared on this subject; the reader may refer to recent literature reviews [11–15] for more details.

The evolution of Information and Communication Technologies (ICTs) and the experience obtained from the past researches allowed the implementation of very advanced solutions, in the last few years, among them the Sense2Health platform (which led to the Ambiciti plateform) [16], integrating a data assimilation model to produce more realistic noise maps; the Hush City platform [13], for collecting data in quiet areas; the City Soundscape platform [17], with the objective to evaluate action plans for road noise reduction; and the GRCSensing platform [14] with an interesting feature for distributing tasks to users in order to capture noise in specific urban areas and times.

Proposed more recently, the NoiseCapture project is completely in line with the last platforms [18], but extends the concept of participatory science to that of open science. Thus, all source codes, whether for the smartphone application, the spatial data infrastructure, or the web interface, are released as open source. In the same way, the data are available in open data in many ways, and, as far as possible, the scientific productions, in open access. Attention was also paid to the long-term sustainability of the system, the NoiseCapture project being part of an operational framework and not in the form of a short-term experimentation. The objective is to ensure a collection of data over several years, in order to constitute a reference database for the study of sound environments over the long term. The respect for privacy and use of personal data is also a founding element of the NoiseCapture project; in order to respect the national regulation, in particular in Europe, no sound or video recordings are made, nor is any personal information collected; the use of the application does not require the creation of an account. Finally, the developers of the application have paid great attention to the quality of the acoustic data collected, by integrating proven signal processing algorithms, and by proposing several methods for smartphone calibration. After more than 3 years of existence, the amount of data collected worldwide thanks to the application is thus considerable (more than 100,000 downloads, 74,000 contributors, 260,000 tracks that represents around 60 million of one second measurement points), showing the interest of the citizens for this participatory approach and offering very promising operational and research perspectives.

Nevertheless, the exploitation of the database, whether in an operational or research context, requires a good knowledge of the inherent limitations of the methodology, such as the lack of control of the measurement protocol, poor acoustic calibration of the application,

measurements tainted by uncertainties, the misuse of the application, the metrological limitations of smartphones, the context of the measurement, etc. In order to ensure that any user of the database has a perfect knowledge and control of the information contained in the database, a full description and an analysis of the database is performed in this article, in order to highlight the various uncertainties, irregularities, or inconsistencies that need to be considered before any exploitation of the data. This analysis also highlights future evolution that it would be interesting to consider in order to improve the application and to increase the quality of the collected data but also to collect additional information in order to better take into account the context of the measure in its exploitation. This article does not therefore constitute an acoustic study of sound environments, but provides a framework for understanding the NoiseCapture database for its future exploitation. The study of the noise environments using this database will be the subject of further works.

The NoiseCapture platform is first presented in Section 2. The collected data are then described and analyzed in the Section 3, providing sufficient information to a future user, for an exploitation of the database in total control of the nature of the data and their possible limitations. In Section 4, a discussion is provided for improving the application and the methodology to increase the data quality and analysis. Last, Section 5 concludes this work.

2. NoiseCapture Application and Database Description

2.1. NoiseCapture History

The development of the NoiseCapture application was initiated by the french National Center for Scientific Research (CNRS) and the Université Gustave Eiffel (formerly Ifsttar) within the framework of the European ENERGIC-OD project [19], which aimed at producing and redistributing geospatial information in open data to user communities. The development continued thereafter, as a part of the Noise-Planet project [20], with the objective to combine geomatic and acoustic sciences for the evaluation of outdoor sound environments. In line with the general goal of the Noise-Planet project, it was decided to develop the NoiseCapture application in the framework of an Open Science approach, with the dissemination of source codes in Open Source, data in Open Data, and as far as possible, scientific dissemination through publications in Open Access journals.

The initial objective of the NoiseCapture application was to propose a smartphone application to a community of specialists (technical staff within a local authority for example), in order to assess the outdoor sound environments in their territory, by using a collaborative mapping tool. The target audience was therefore initially people with technical and, possibly, acoustic knowledge, allowing them to understand a rather professional smartphone application.

The NoiseCapture application was designed in order to carry out acoustic measurements over a shorter period of time, if possible while walking, in order to collect data on a large spatial area. The user was expected to keep the smartphone in hand throughout the measurement, especially to control the measurement. Thus, the measurements are user-initiated and not background. Each user can then decide to upload data to a remote server that collects all the data in a database, performs further analysis, and represents the results collected by a set of users in the form of a noise map.

The choice of the development environment was oriented towards the most widespread platform, namely, Android, whose market share has been above 80% for many years (around 15% for iOS (a mobile operating system created and developed by Apple Inc.)) [21]. The porting of the application to iOS has not been achieved, although a gain in terms of metrological quality may be possible due to a lower variability in devices [22]. In order to promote the diffusion of the application worldwide, the application has been translated thanks to volunteers, in several languages (English (en), Chinese (China, zh_CN), French (fr), Greek (el), Polish (pl), Portuguese (Brazil, pt_BR), Spanish (es)). In the rest of this article, the terms and features of the application refer to the 1.2.15 version (release 51) of NoiseCapture with the default language (i.e., "en" for English). The last public NoiseCapture release is available on Google Play ("Google Play" brand is property of Google LLC) [23].

2.2. NoiseCapture Description

2.2.1. NoiseCapture Android App

From a functional point of view, the NoiseCapture application uses the principles of a "pocket" sound level meter. The main screen (Figure 1, "Measurement") presents the results of an acoustic measurement through several classical acoustic indicators: an instantaneous sound level (calculated on a sliding window), as well as the minimum (Min), maximum (Max), and average (Mean) instantaneous sound levels over the duration of a measurement. The instantaneous spectrum by third octave band between 100 Hz and 16 kHz is also proposed on a specific tab, as well as a spectrogram. The duration of the measurement is also indicated: the user can start, pause/resume, and stop the measurement at their convenience, and the measurement duration can also be automatically be fixed in the application settings (the user starts the measurement, but it stops by itself after a certain duration).

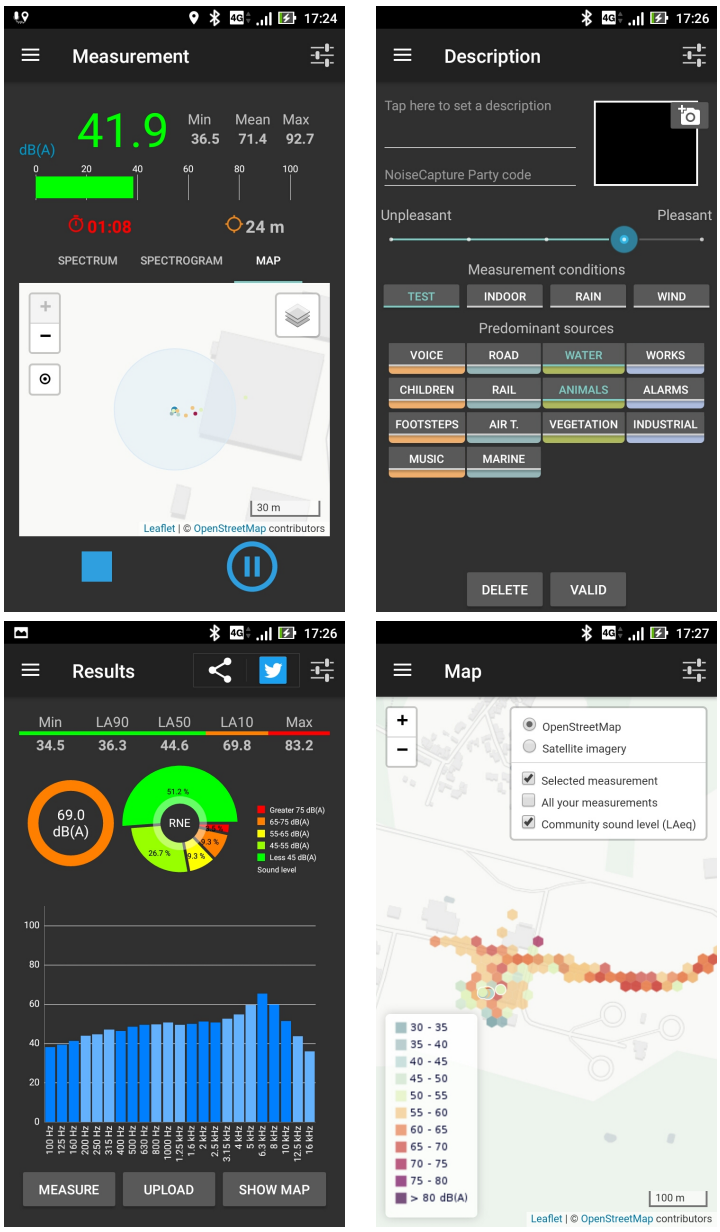


Figure 1. NoiseCapture Android application main screens. From **top**/**down** and **left**/**right**: Measurement, Description, Results, Map.

In accordance with the initial objective of the application, i.e., the production of a noise map, each measurement is geolocalized with the last known GPS location. The measurement screen also indicates the position of each measurement, every second (i.e., the "Measurement Point"), and more globally the trace of a measurement (i.e., the "Measurement Track") according to the user displacement. The accuracy of the location is also indicated both numerically and graphically on the map. At this point, note that the acoustic indicators that are calculated and displayed on this screen, as well as those that are presented on the "Results" screen, do not result in any audio recording; these indicators are calculated on the fly.

After a measurement has been performed, the user accesses a second screen (Figure 1, "Description"), which allows the user to give additional information to the measurement. Filling this form is entirely optional. Some information, such as "Description" and "Pictures", are only stored on the smartphone, while other data may be collected and transmitted to the NoiseCapture remote data server. The choice of whether or not to transmit the measurements and information can be configured by the user. On this screen, 3 types of information can be provided: (1) information on the perceived quality of the sound environment ("Pleasantness"); (2) information on the measurement conditions using 4 tags ("Test", "Indoor", "Rain", and "Wind"); (3) information on the nature of the sound sources perceived during the measurement, using 14 tags ("Footsteps", "Voice", "Natural", "Mechanical", "Human", "Works", "Air t."). (i.e., "Air Traffic"), "Entertainment", "Children", "Music", "Road", "Rail", "Marine", "Alarms", "Industrial", "Water", "Animals", "Vegetation").

Once this optional information has been validated, the user has access to a summary of the measurement in the "Results" screen (Figure 1). Acoustic indicators are specific to the evaluation of outdoor sound environments, based on 1 s average sound level [24], such as noise levels in percentiles (L_{A10}, L_{A50} and L_{A90}), maximum (Max) and minimum (Min) values as well as the average sound level in dB(A) and the average spectrum over the measurement time. In addition, a graphical representation, noted RNE, shows the distribution of the 1-second noise levels.

A "Map" can also be displayed on a specific screen (Figure 1) in order to locate the measurement points and to represent the average values shared by the user community and aggregated by the NoiseCapture remote server. Other functionalities are also offered by the application, such as smartphone calibration and data archiving, but are not detailed in the present paper. More details are given in the following reference [18].

2.2.2. NoiseCapture Web Interface

While the NoiseCapture application can be used to meet the need of a user (i.e., to assess a noise level in his own environment), the overall interest of the approach lies in the sharing of data within a community, which requires the data to be centralized on a remote server. To this end, a Spatial Data Infrastructure (SDI), called OnoMap, has been specifically implemented to propose 3 functionalities: to (1) collect, (2) display, and (3) share all the data produced by the contributors [20,25,26].

The second functionality is the most visible part of this SDI, as it allows to display the collected data to any visitor of the website, in an aggregated and understandable form. Figure 2 illustrates an example of a graphical representation, centered on the city of Lyon in France. Depending on the zoom scale of the map, the main window presents the collected data either in a numerical form, in terms of number of points per geographical area (represented by hexagons of different sizes depending on the zoom level), or in the form of a 'classical' noise map. In the latter representation, only certain acoustic indicators are presented, by aggregating all the values collected over a fixed spatial extent (i.e., average of an acoustic indicator in a hexagon). The left-hand side of the web page gives access to additional contents, such as the history of the last 30 series of measurements (almost in real-time), general statistics on all the data collected (most contributing countries, number of measurements, most used tags, etc.). By clicking on a hexagon on the noise map (at the highest zoom levels), it is also possible to access to more detailed information, such

as the number of points and the total duration of measurements in the corresponding hexagon, the average equivalent sound level ($L_{A,eq}$ and LA50), the tags used (in the form of a tag cloud), as well as the hourly distribution of sound levels on different days of the week. All the information presented in this web page results from a direct exploitation of the NoiseCapture database, and illustrates some relatively simple analysis. Downloading the collected data (the third functionality of the SDI) offers many more perspectives of analysis and representation of the data. The upper screenshot of Figure 2, which displays the position of the measurement points, underlines again the interest of the method and the very rich perspectives of analysis of the sound environments, with regard to the quantity of data that can be collected on a given spatial extent. The purpose of this article is precisely to propose a first analysis of these raw data, in Section 3, so that they can be exploited, in a second step, to perform a relevant sound environment analysis.

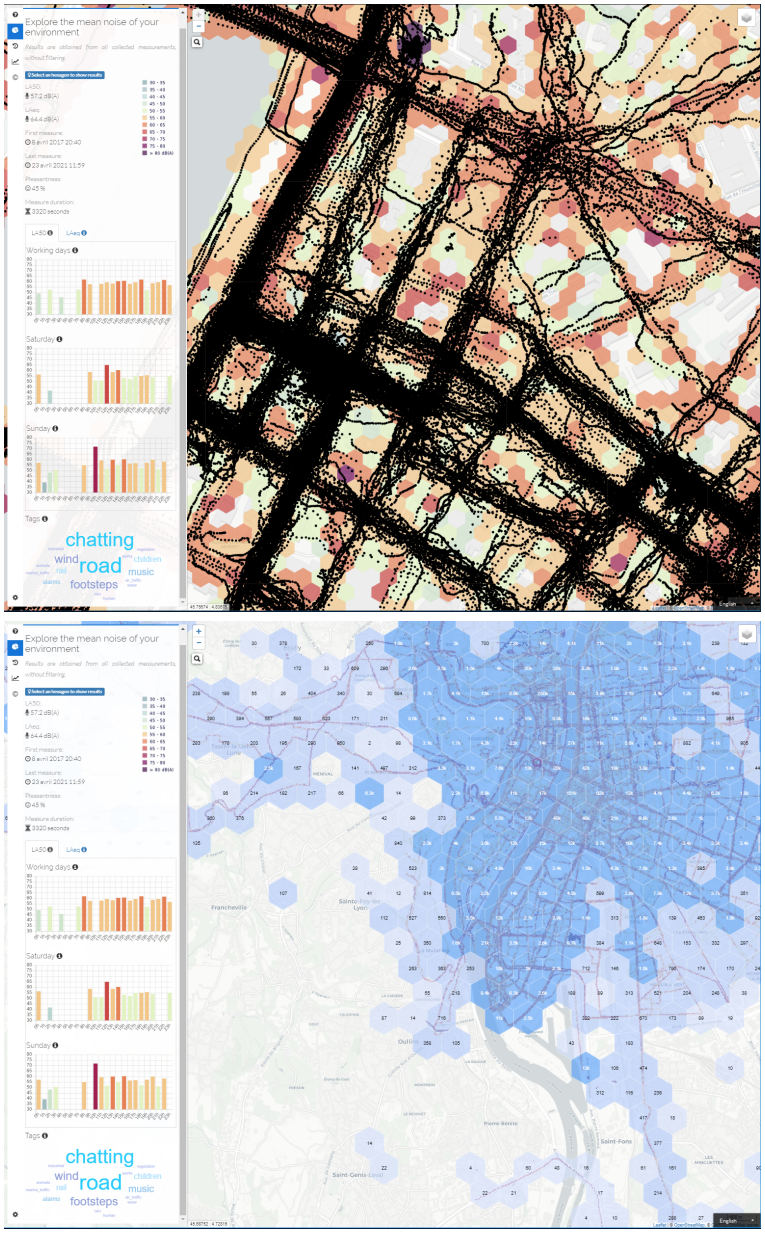


Figure 2. Screenshots of the NoiseCapture map website. Example of collected data representation, on the Lyon French city (from https://noise-planet.org/map_noisecapture/index.html#18/45.75387/4.84052/ (accessed on 15 July 2021)), at a higher (**up**) and lower (**down**) zoom levels.

2.3. NoiseCapture Raw Database

The analysis that is carried out thereafter covers the data collected since the official publication of the application on 29 August 2017 until 28 August 2020 (3 years). These collected data cover several releases of the application (from 28 to 51, see Table 1), some of which make changes to the nature of the data collected and the features. Previous release (before release 28) and intermediate pre-releases correspond to beta versions, published on Google Play to a specific panel of testers. The database available for download may include data from beta and pre-release versions; it may be useful to filter these data both on a period (from the launch of the application) and on the version (since release 28), for a relevant analysis.

Table 1. NoiseCapture application releases. Each new version, defined by a version release (for example '51') and a version number (for example '1.2.15') proposes changes (bug corrections, user interface enhancement, etc.). The reader can refer to the detailed list of fixes in each (pre-)release from the GitHub source code management platform [27]. The changes made on the data export, from the smartphone to the data server (adding new data, patches, etc.), are detailed in the history of the source code file `MeasurementExport.java` [28]. The date of publication of the application on Google Play is also provided for information (corresponding official public release are indicated in bold with symbol *).

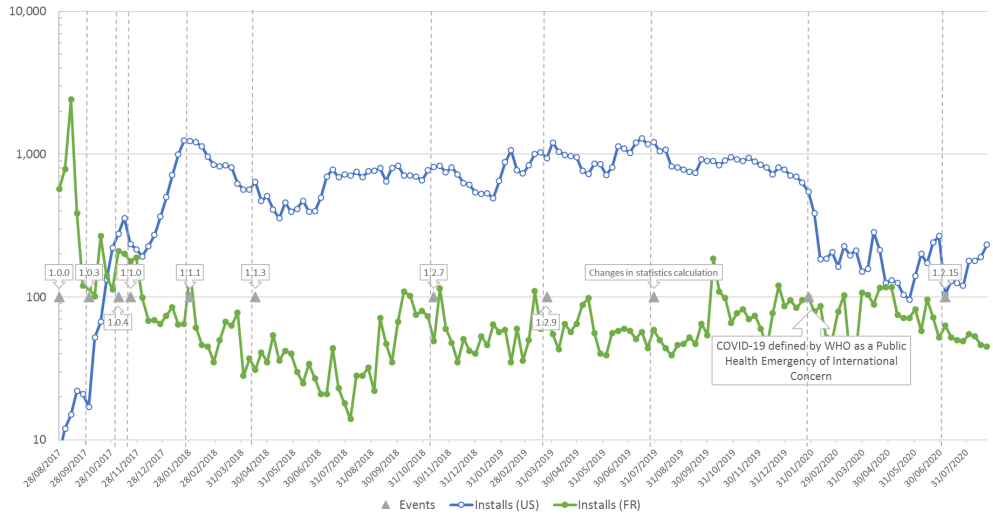
Release (Number)	Source Code Publication	STATUS	Application Publication	Comments
51 * (1.2.15)	3 July 2020	Release	7 July 2020	Fix automated measurement upload
49 (1.2.13)	17 February 2020	Pre-release		Add calibration method using road traffic Add `'calibration_method'`
45 * (1.2.9)	27 March 2019	Release	26 March 2019	Calibration in $L_{A,eq}$ instead of L_{eq}
43 * (1.2.7)	16 November 2018	Release	16 November 2018	Minor changes
35 * (1.1.3)		Release	20 April 2018	Minor changes
34 * (1.1.2)		Release	29 January 2018	Minor changes
33 * (1.1.0)	23 November 2017	Pre-release	24 November 2017	Ability to use a calibrated sMarchtphone to automatically calibrate other sMarchtphone(s)
32 * (1.0.4)		Release	6 November 2017	Minor changes
31 * (1.0.3)		Release	6 October 2017	Minor changes
30 (1.0.2)	18 September 2017	Pre-release		Add NoiseCapture Party functionalities
29 * (1.0.1)		Release	31 August 2017	Minor changes
28 * (1.0.0)	23 August 2017	Release	29 August 2017	Official first release Add `'user_profile'`

2.4. NoiseCapture Installs and Uninstalls

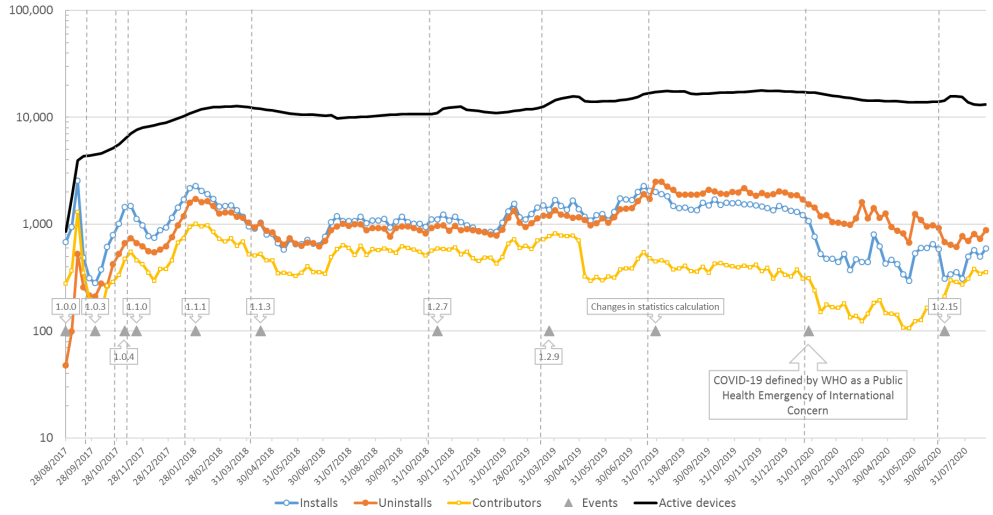
As mentioned above, the initial audience targeted during the development of the application was primarily technical staff, with sufficient expertise to be able to use the application in satisfactory conditions (compliance with a measurement protocol, acoustic calibration of the smartphone, critical analysis of the measurements, etc.). The production of data was therefore initially part of a supervised activity with a professional purpose. In practice, the publication of the application on Google Play, combined with an institutional communication, was relayed by the national and then European media, generating the interest of a wider public than initially foreseen. Very quickly, the application was then downloaded in other countries, notably the United States, by a large audience. This confirms once again the interest of citizens and communities in the issue of noise environments and reaffirms the major societal challenge of research on this subject.

Figure 3a illustrates the number of installs of the application for the two countries (US and FR) that contribute the most to the data collection today; these data are obtained from the application dashboard on Google Play. This figure clearly shows the impact of the launch of the application in France, with a high number of installs in the first few weeks, followed by a decrease to an average level of about 60 installs per week; conversely, there is a gradual increase in the number of installs in the US, to an average level of 800 installs per week. From a global point of view, Figure 3b shows a certain stability around the 1000 weekly installs worldwide, over most of the period concerned. Unsurprisingly, the uninstalls rate follows the rate of installations, but the number of uninstalls tends to exceed the number of installations since the end of 2019, which leads to a decrease in the number

of active devices (i.e., devices having installed the application and being turned on over a 30-day period), which has gone from about 17,000 at the end of 2019 to 13,000 at the end of 2020.



(**a**)



(**b**)

Figure 3. Weekly evolution of user installs and uninstalls of the NoiseCapture application (data from Google Play dashboard): (**a**) global data and (**b**) data for France and United states of America. 'Installs': number of users who have installed the application at least on one device; 'Uninstalls': users who have uninstalled the application from all their devices; 'Active devices': number of active devices that contains application, and which was turned on at least once in the previous 30 days; 'Contributors': Users who have upload data to the NoiseCapture remote server; 'Events': Events that may have a particular impact on the users behavior. (**a**) Application installs for France (FR) and United-States of America (US); (**b**) Application installs/uninstalls/contributors.

Even if it is difficult to make a direct link between the number of installations and the number of different contributors, we can see that on average, about 50% of new installations give rise to at least one contribution on the NoiseCapture server over the period from 2017 to mid 2019 (Figure 4). The 'break' that is visible on this figure in mid-2019, due to a very sharp drop in the number of contributors (Figure 3b), is at this stage undetermined. Conversely, Figure 3b shows the interaction between certain events quite well, in particular the impact of the publication of a new release on the number of new installs (also visible

on the number of active devices). For example, the decrease in number of installs and contributors observed from the beginning of 2020 coincides with the beginning of the COVID-19 pandemic, particularly visible in the US community (Figure 3a); this is not visible with the French data, but we can quite imagine that there is a link between these two events. A detailed analysis of COVID-19 lockdown and user behavior in each country would undoubtedly lead to some hypotheses. This shows again the interest of such alternative way for collecting data for the study of the noise environment.

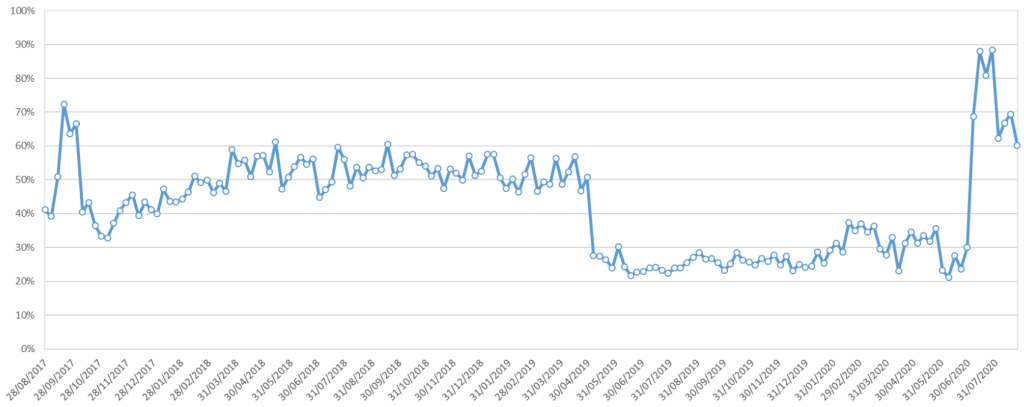


Figure 4. Weekly ratio between the number of contributors to the NoiseCapture database and NoiseCapture installs.

3. Analysis of the Collected Data

3.1. Collected Data

As mentioned above, the statistical analysis on NoiseCapture data presented in this section involves data collected from 29 August 2017 to 28 August 2020 (3 years of data). During this period, the NoiseCapture application has proven to be successful to perform and gather acoustics measurements. NoiseCapture has been downloaded more than 160,000 times on Google Play [23], with 76,229 contributors to the database all over the world. Approximately 91.7% of the users present in the database (69,898 of the 76,229 contributors in the present database) have contributed within this period. Table 2 shows that 260,422 tracks (59,685,328 points) have been collected, with an average of 229.2 points (i.e., seconds) per track (a median value of 28 points per track).

Table 2. Distribution of collected data per release, from 29 August 2017 to 28 August 2020. The number of collected data during NoiseCapture Parties (see Section 3.6) is also indicated, as well as in terms of percentage of the total number of data. Releases in bold with symbol * correspond to public releases on Google Play. Note that the total number of contributors in this table (74,082) does not correspond to the total of unique contributors, as a contributor may have use several release of the application.

Release	Contributors		Tracks		Points	
	Total	Party	Total	Party	Total	Party
28 *	26	–	354	–	46,268	–
29 *	2705	–	8991	–	1,588,156	–
30	35	7 (20.0%)	416	133 (32.0%)	140,627	11,523 (8.2%)
31 *	1432	2 (0.1%)	4426	6 (0.1%)	957,920	770 (0.1%)
32 *	1553	3 (0.1%)	4746	9 (0.2%)	847,093	1556 (0.2%)
33 *	5442	4 (0.1%)	19,225	52 (0.3%)	3,530,349	8819 (0.2%)
34 *	9053	13 (0.1%)	28,607	67 (0.2%)	6,121,154	18,793 (0.3%)
35 *	15,734	67 (0.4%)	68,911	921 (1.3%)	12,465,115	117,797 (0.9%)

Table 2. *Cont.*

Release	Contributors		Tracks		Points	
	Total	Party	Total	Party	Total	Party
36	4	–	6	–	1861	–
37	11	–	55	–	20,732	–
38	3	–	7	–	1774	–
39	2	–	2	–	108	–
40	1	–	1	–	2	–
41	1	–	1	–	97	–
42	1	–	1	–	10	–
43 *	11,169	82 (0.7%)	37,960	643 (1.7%)	9,309,934	89,794 (1.0%)
44	6	–	33	–	4276	–
45 *	23,331	183 (0.8%)	67,765	1306 (1.9%)	18,629,005	142,168 (0.8%)
46	3	–	7	–	5030	–
47	4	1 (25.0%)	6	1 (16.6%)	21,101	134 (0.6%)
48	3	–	12	–	8597	–
49	25	–	233	–	235,882	–
50	4	–	4	0 0.0%)	146	–
51 *	3534	2 (0.1%)	18,653	4 (0.02%)	5,750,091	221 (0.003%)
Total	74,082	364 (0.5%)	260,422	3142 (1.2%)	59,685,328	391,575 (0.6%)

All the collected data during the corresponding period of analysis have been integrated into a spatial relational PostGIS database [29] (i.e., a spatial database extender for PostgreSQL object-relational database [30], adding a support for geographic objects). The database is fully available for download [31] and can be used according to the OdBL license [32]. It is important to specify that all the data integrated in this database fully respects the privacy of users as no personal data is collected.

The data collected from smartphones are organized into several tables (Figure 5):

- For each measurement 'Point' (i.e., a measurement performed every second during a 'Track'), the global `'noise_level'` value measured at the measurement date `'time'` is given in the `'noisecapture_point'` table. In addition, the `'speed'` at the measurement point, the geolocalization (`'the_geom'`), the date of the localization (`'time_location'`), the `'accuracy'` of the geolocalization as well as the smartphone `'orientation'`, all obtained by the smartphone GPS, are given. In this table, the measurement point is defined by a primary key `'pk_point'` (generated by the database) allowing to make the relation with two other tables `'noisecapture_freq'` and `'noise_capture_track'` (via the primary key `'pk_track'`);
- The `'noisecapture_freq'` table contains for the measurement point defined by the primary key `'pk_point'`, the `'noise_level'` spectrum by third octave band `'frequency'` between 100 Hz and 16 kHz;
- The `'noisecapture_track'` table contains all the information associated with a measurement corresponding to a set of measurement points. Each measure is defined by a primary key `'pk_track'` (generated by the database) and a unique identifier `'track_uuid'` (generated by the application). Each measurement contains the following information: the user primary key `'pk_user'`, the release number of the application `'version_number'`, the characteristics of the smartphone (the reference `'device_product'`, the model `'device_model'` and the manufacturer `'device_manufacturer'`), the date of the start of the measurement `'record_utc'`, the duration `'time_length'` of the measurement, the average sound level over the duration of the measurement `'noise_level'` and the perception of the sound environment `'pleasantness'`. Information on the acoustic calibration of the smartphone is also associated with the measurement: the choice of the calibration method `'calibration method'` and the corresponding calibration value `'gain calibration'`. Finally, if the measurement was performed during a NoiseCapture Party (see Section 3.6), the corresponding code is indicated in the value `'pk_party'`.

- The 'noisecapture_user' table gives for each primary key 'pk_user', the user identifier 'user_uuid' (this unique identifier is randomly created each time the application is installed on a smartphone), the user creation date 'date_creation' (created by the remote server when uploading the data, not at the application installation), as well as the user 'profile' defined by the choice of a value in a list, as 'EXPERT', 'NOVICE', and 'NONE'. The value 'pseudo' in the table has been created for future functionalities and is currently not used.
- The 'noisecapture_track_tag' table contains for each measure defined by the primary key 'pk_track', the list of tags selected by the user to describe the sound environment. The identifiers of the corresponding tags are defined in the value 'pk_tag'. The correspondence between the identifier of the 'pk_tag' tag and the name of the tag ('tag_name') is defined in the 'noisecapture_tag' table.
- The 'noisecapture_party' table contains information about the realization of the NoiseCapture Party events [33] (see Section 3.6 for details). In principle, such event is supervised by an expert, over a limited duration and spatial extent, allowing to generate a series of measurements. It can for example be an action carried out by a Community in order to carry out a series of measures concentrated in a particular district. A NoiseCapture Party has much the same objectives as an OpenStreetMap (OSM) Mapping Party to feed the OSM global database [34]. This table gives for each NoiseCapture Party, a specific primary key 'pk_party' (generated by the database) returning the code of the NoiseCapture Party ('tag'), the title 'title' and a description 'description', the spatial extent defined by a geometry 'the_geom', the start and end dates of the event 'start_time' and 'end_time'. The boolean values 'filter_time' and 'filter_area' are used to define whether the collected data are integrated into the NoiseCapture Party set, whether or not the measurements have been made with the right NoiseCapture Party code, but outside of the temporal and and spatial limits. The value 'layer_name' is only used to give a name to the corresponding map layer in the web page displaying the data on the corresponding website [26]. It is important to specify that the NoiseCapture Party is technically created by the people in charge of the development of NoiseCapture. If an invalid value is used for the NoiseCapture Party code field in the 'Description' screen of the application (Figure 1), the code is removed, but the corresponding data are still included in the database.

3.2. User Information

3.2.1. User Profile

The use of the application according to the respect of technical procedures in acoustics is an important issue for the quality of the produced data. In order to have information on the user experience, at the installation step of the application, the user is asked to define his expertise using a 3 levels scale: 'EXPERT', 'NOVICE', or 'NONE'. Analyzing the 76,229 different contributors in the database, over the period from 29 August 2017 to 28 August 2020 (using the field 'date_creation', which corresponds to the date of creation of the user in the NoiseCapture database on the remote server), 10.19% defined themselves as 'EXPERT', 24.78% as 'NOVICE', and 64.17% as 'NONE'. A very large majority of contributors therefore have no experience in the field, which can necessarily lead to a bias in the quality of the data collected. This is an expected behavior for a citizen science project.

Note that for 653 contributors (0.86% of the total number of contributors), the profile field is empty, meaning that the information is not available during this period. This is due to an update of the application from a version prior to version 28 (the field 'user_profile' has been integrated from version 28), as the user is not asked to modify this field during an update. In detail, the analysis of these cases shows that most of the concerned contributors (641) were declared in the database in the first 2 months after the launch of the application, while the other 12 contributors were declared during the rest of the period.

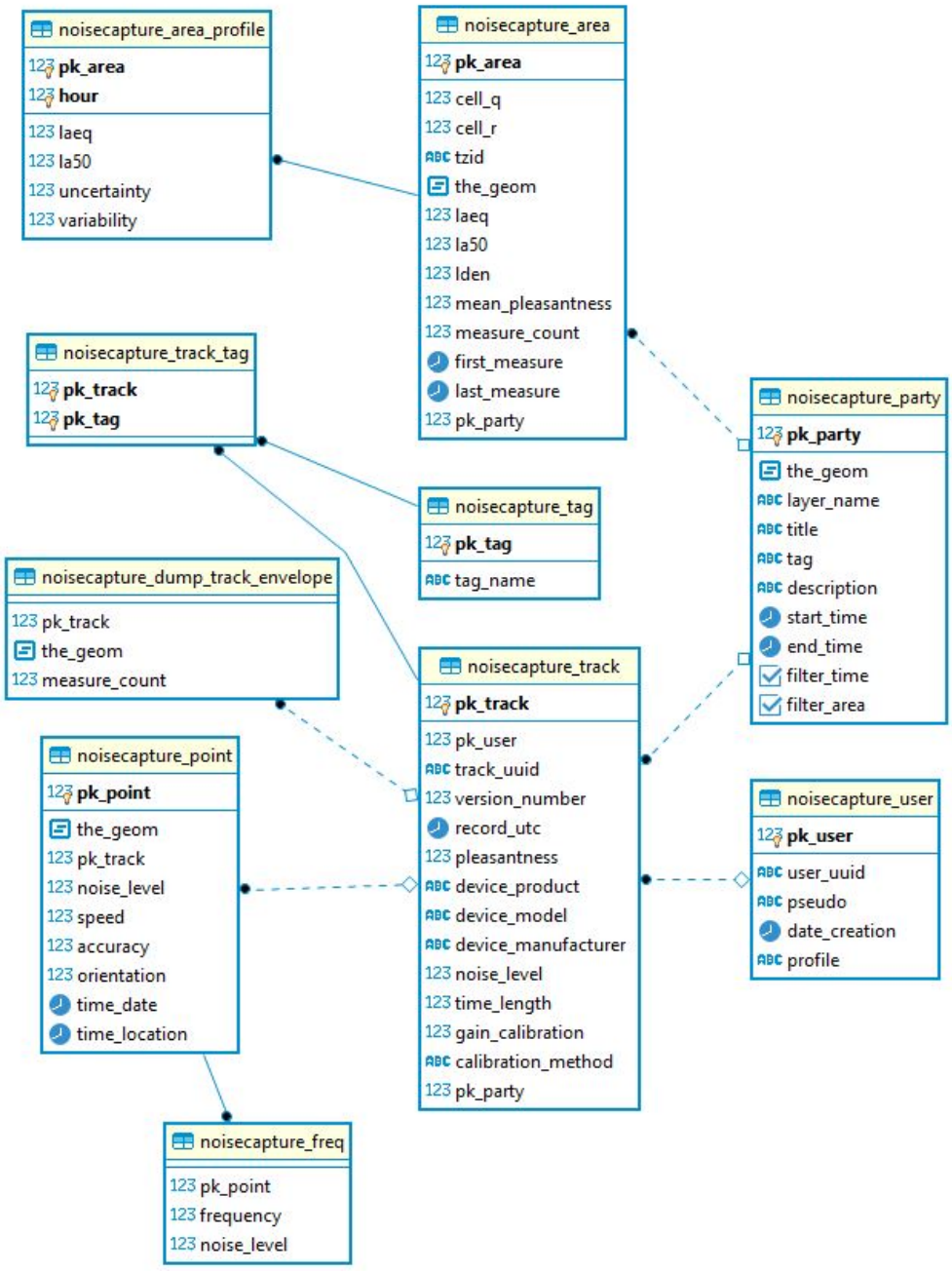


Figure 5. Entity relation diagrams (ERD) of the NoiseCapture PostGIS database. The type of each field in the tables is mentioned: '123' for 'float' values, 'ABC' for 'text' chains, 'timestamptz' for time stamp date, '☑' for 'boolean' value. The 'key' yellow symbol is used to display primary keys of the table, whose corresponding names are displayed in bold.

3.2.2. User Devices

For this type of measurement application, the metrological quality of the device, whether for acoustic measurement or for other data (GPS and other sensors), is an essential aspect. On this point, the identification of the smartphone can provide useful information for a later analysis of the collected data, in postprocessing. Among the possible treatments, an *a posteriori* calibration of the acoustic data, for example, based on a smartphone knowledge base, offers interesting prospects for improving the quality of the acoustic indicators produced by the application [35,36]. Some works have also shown that the knowledge of the manufacturer can provide a useful information on the accuracy of the measurement [37]. This justifies the need to collect hardware-related information, namely,

the 'device_product', the 'device_model', and the 'device_manufacturer', defined by Android documentation as the name of the overall product, the end-user-visible name for the end product and the manufacturer of the product/hardware respectively [38].

As an example, considering the Samsung Galaxy A10, which is one of the best selling Android phone, the device field will give the data of Table 3. This table shows that the commercial name of the corresponding smartphone can be declined in several device models that most of time refer to distinct version ('A10E' for 'SM-A102', 'A10' for 'A105', 'A10S', for 'A107') or to the international region where they were deployed.

Table 3. Collected device information for the Samsung Galaxy A10 Android phone, as well as, the count of corresponding phones in the NoiseCapture database.

'device_model'	'device_product'	'device_manufacturer'	Count
SM-A102N	a10ekx	samsung	1
SM-A102U	a10esq	samsung	880
SM-A102U1	a10eue	samsung	1
SM-A102W	a10ecs	samsung	4
SM-A105F	a10dd	samsung	53
SM-A105FN	a10eea	samsung	176
SM-A105G	a10dx	samsung	31
SM-A105M	a10ub	samsung	97
SM-A107F	a10sxx	samsung	508
SM-A107M	a10sub	samsung	23

Over the period in question, the database references 646 distinct manufacturer names. However, the same manufacturer can appear under a different spelling; this is the case, for example, for Samsung, appearing with the following names: 'samsung', 'Samsung', and 'SAMSUNG'. By grouping the manufacturers without taking into account the sensitivity to upper and lower case, one can identified finally 520 manufacturers (Table 4) with 5300 different smartphone models. Nevertheless, three manufacturers alone (Samsung, LGE and HUAWEI) account for about 35.2% of the models, and cumulate nearly two thirds of the tracks (65.1% or 66.3% in number of points). The top 15 manufacturers account for 90.3% of the tracks (91.1% of the points).

Table 4. Top 15 of smartphone manufacturers ('device_manufacturer', case insensitive) in the NoiseCapture database. The number of corresponding distinct device models ('device_model'), the number of tracks, as well as the cumulative number of tracks are also given. Note that this table do not regroup data from the same manufacturer but with a different writing (upper/lower case, as for 'Samsung' and 'samsung').

Rank	Device_MANUFACTURER	Nb of Models	Nb of Tracks	%	Cumul. Nb of Tracks	%
1	samsung	1032	101,420	38.9%	101,420	38.9%
2	LGE	383	36,288	13.9%	137,708	52.9%
3	HUAWEI	454	31,937	12.2%	169,645	65.1%
4	motorola	126	17,822	6.8%	187,467	71.9%
5	ZTE	171	12,840	4.9%	200,307	76.9%
6	Xiaomi	106	6334	2.4%	206,641	79.3%
7	TCL	191	5180	2.0%	211,821	81.3%
8	Sony	167	5116	2.0%	216,937	83.3%
9	OPPO	91	3742	1.4%	220,679	84.7%
10	WIKO	73	3223	1.2%	223,902	86.0%
11	asus	115	2870	1.1%	226,772	87.1%
12	HTC	130	2406	0.9%	229,178	88.0%
13	HMD Global	41	2208	0.8%	231,386	88.8%
14	LENOVO	108	1881	0.7%	235,065	89.6%
15	OnePlus	32	1798	0.7%	235,065	90.3%

The distribution of measurements is more important in number of models (Table 5), as the top 15 models only have 15.9% of tracks (16.7% of points), each model accounts to only between 1.8% and 0.8% of the whole measurements. To reach half of the tracks, we have to consider 130 different models, and 1077 models to exceed 90%. In addition, Figure 6 shows that most of devices appears only few times in the database; for example, 3407 different devices are used 10 times or less; conversely, only 775 device models appear more than 50 times in the database. In detail, there are 1228 smartphones that are used only once and 677 twice.

Table 5. Top 5 of smartphone model (`'device_model'`) in the NoiseCapture database.

Rank	`Device_MODEL`	`'Device_MANUFACTURER'`	Nb of Tracks	%	Cumulative Nb of Tracks	%
1	ANE-LX3	samsung	4729	1.8%	4729	1.8%
2	SM-G930F	samsung	3722	1.4%	8451	3.2%
3	LM-X210(G)	LGE	3479	1.3%	11,930	4.6%
4	SM-A520F	samsung	3205	1.2%	15,135	5.8%
5	Z982	ZTE	2890	1.1%	18,025	6.9%
6	SM-G935F	samsung	2854	1.1%	20,879	8.0%
7	Moto E (4)	motorola	2748	1.0%	23,627	9.1%
8	SM-N950U	samsung	2384	0.9%	26,011	9.9%
9	moto e5 play	motorola	2361	0.9%	28,372	10.9%
10	SM-G950F	samsung	2301	0.9%	30,673	11.8%
11	SM-J327T1	samsung	2297	0.9%	32,970	12.6%
12	LGMP260	LGE	2213	0.8%	35,183	13.5%
13	SM-S327VL	samsung	2200	0.8%	37,383	14.3%
14	VTR-L09	samsung	2095	0.8%	39,478	15.1%
15	SM-J727T1	HUAWEI	2048	0.8%	41,526	15.9%

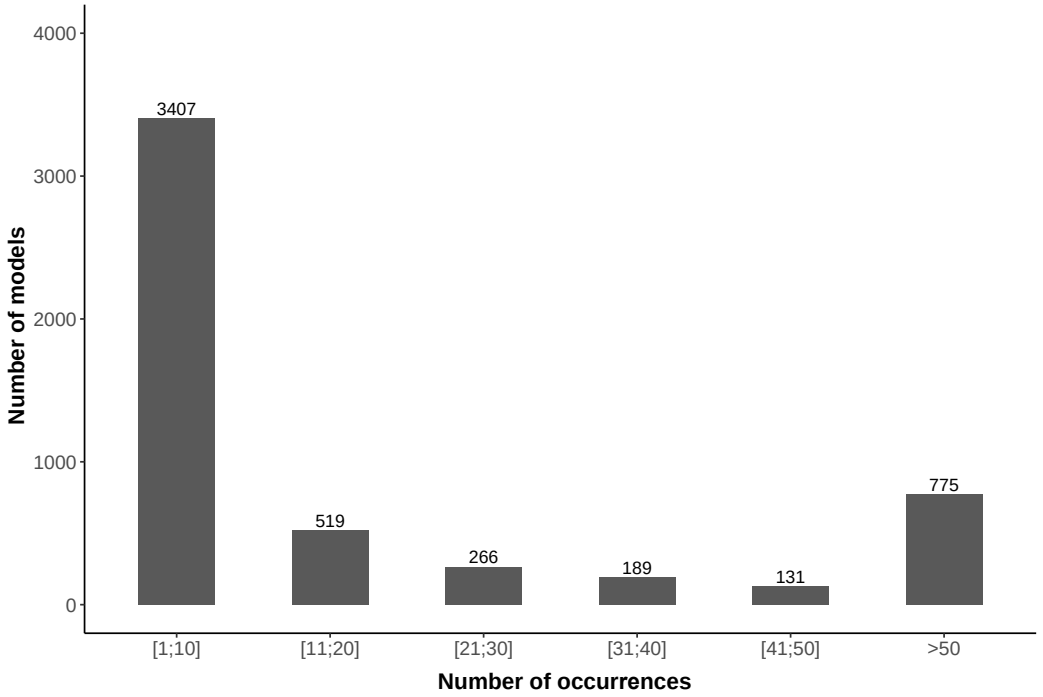


Figure 6. Distribution of the number of occurrences of a smartphone model in the database. As an example, 775 device models appear more than 50 times in the database.

By focusing on the two most contributing countries (US and FR, Table 6), we find a consistency between the manufacturers market share and the brands most represented in the database. This table also shows that Apple with the iPhone model (Apple and iPhone are trademarks of Apple Inc.) is a very important manufacturer in the US and in

France; it suggests that the current NoiseCapture database excludes a very large number of users, that in the case of iOS users, represents a specific segment of the population, considered with higher income and education levels, in-app engagement [39]. The initial choice to select Android as the only development platform, as it represents a global market share of 80%, can thus be questioned. It would seem wise to consider an additional iOS version of the application in the future, considering the user audience, but also metrological considerations.

Table 6. Top 6 smartphone manufacturers (`'device_manufacturer'`, case-insensitive) for USA and France between August 2017 and August 2020. Top 6 manufacturers data are from Statcounter Global Stats website (licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License) [40].

Country	Device_MANUFACTURER	Number of Tracks	%	Top 6 Manufacturer in Country
United States	samsung	32,341	36.6%	IPhone
	LGE	23,668	26.8%	Samsung
	motorola	9257	10.5%	LGE
	ZTE	8154	9.2%	Motorola
	TCL	2117	2.4%	Google
	Alcatel	1289	1.4%	ZTE
France	samsung	12,899	46.2%	Samsung
	HUAWEI	4864	17.4%	IPhone
	WIKO	1731	6.2%	HUAWEI
	Xiaomi	1294	4.6%	Sony
	Sony	1254	4.5%	Xiaomi
	motorola	1093	3.9%	WIKO

3.2.3. User Contribution

Table 7 illustrates the use of the application in terms of number of contributions. Slightly more than half of the contributors have contributed to the database only by 1 track, and nearly 95% by less than 10 tracks. It is likely that most of the contributors concerned by only few contributions were just interested by testing the application, before either uninstalling it or putting it aside. This table also shows that there is a small proportion of contributors who have collected a very large number of tracks, up to several thousand for some. It seems obvious that these contributors have integrated themselves into an active approach to collect measurements and that this type of user is the most interesting part of the community, *a priori* motivated by the collaborative approach. The animation of this specific community must be a priority in the future. This last point will be discussed in Section 4.

Table 7. Distribution of the number of contributors in function of the number of track measurements. The number of corresponding points is also given.

Number of Tracks	Number of Contributors	%	Number of Points	%
1	36,405	52.0%	8,709,872	14.6%
2–10	30,043	43.0%	24,106,578	40.1%
11–50	3063	4.4%	14,779,033	24.7%
51–100	236	0.3%	3,915,310	6.5%
101–1000	143	0.2%	8,016,517	13.4%
>1000	8	0.1%	158,018	0.7%
Total	69,898		59,685,328	

Considering the contributors with only one contribution, Figure 7 shows that 6155 of them (16.9%) have used the "test" tag, meaning that they were just testing the application. In addition, Table 8 shows that 14,034 (38.5%) of these "one-shot" use of the application have duration less than 20 s. These two observations may partially support our hypothesis

that these one-shot contributors just want to test the application, and probably do not plan to use the application again.

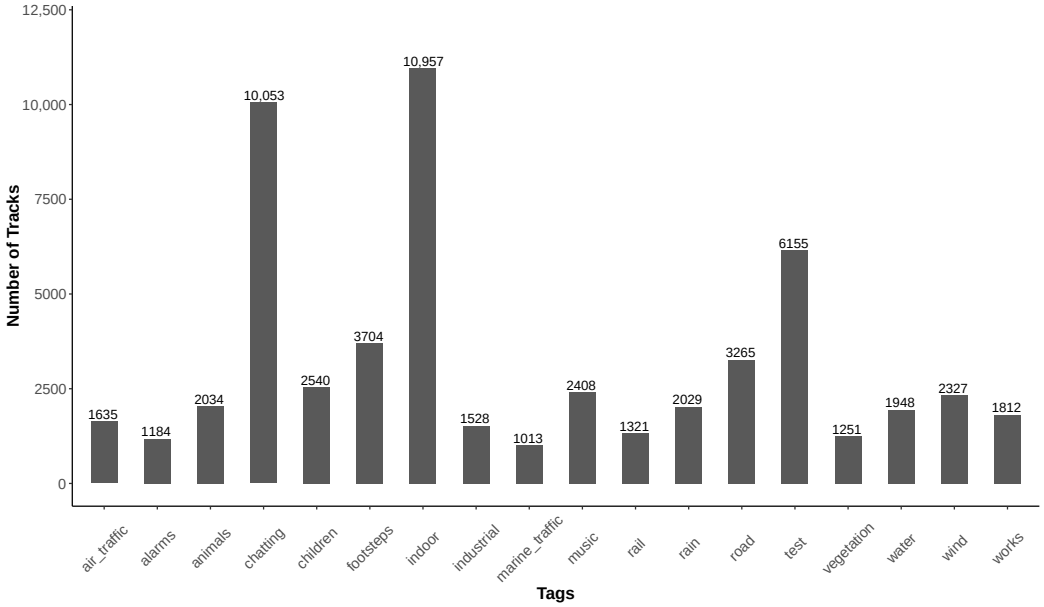


Figure 7. Distribution of the tags used for tracks collected by user who have 1 only contribution to the database.

Table 8. Distribution of the time length (in second) in function of the number of track measurements for tracks collected by user who had 1 contribution to the database.

Time Length	Number of Tracks	%
1–20	14,034	38.5%
21–60	9452	26.0%
61–300	9242	25.4%
301–600	1537	4.2%
601–900	534	1.5%
901–1200	315	0.8%
1201–1800	368	1.0%
1801–2400	213	0.6%
2401–3000	111	0.3%
3001–3600	103	0.3%
>3600	496	1.4%
Total	36,405	

Table 9 shows that for users that realize more than one contribution, the second contribution comes in the next 4.4 days, on average. However, for the major part of the contributors (158,631, 83.3%), the second contribution is realized in the same day, and in the same week for 9.9% (18,971).

Table 9. Duration between two successive measurements for users who have more than one contribution.

Duration between 2 Successive Measurements (Day)	Number of Tracks	%
0	158,631	83.3%
1	7140	3.7%
2–7	11,831	6.2%
8–14	3911	2.0%
14–21	1923	1.0%
21–30	1450	0.8%
31–60	2256	1.2%

Table 9. *Cont.*

Duration between 2 Successive Measurements (Day)	Number of Tracks	%
61–90	1063	0.6%
91–180	1269	0.7%
181–365	768	0.4%
>365	282	0.1%
Total	188,794	100%

3.3. Measurement Geolocalization

3.3.1. Geolocalization

In this paragraph, we present statistics and information related to the geolocalization of the NoiseCapture data. The variable `'the_geom'`, which gives the coordinates of the measurement point in the WGS 84 (EPSG:4326) map projection, has been used to perform this study.

As the country of the measurement is not in the data set, the following 2-step process has been carried out in order to define the country of origin of each measurement. First, a table called `'noisecapture_track_frame'` was created using the PostGIS/PostgreSQL function 'ST_EXTENT()' [41] that returns a box that bound each track. Second, a table called `'gadm'`, mapping the administrative areas of all countries [42], has been used to create a table called `'noisecapture_country_track'` by associating each track to the first country that contains the track bounding box.

Table 10 shows that the United States contributes more than third of NoiseCapture database, while France contributes approximately 10% of track data (8.3% point data). A strong French contribution was obviously expected, the application having been developed by French research institutes, and also because of a strong relay by national media. Conversely, it is difficult to explain the large amount of data produced by the US, except to consider that this country has a high population (3rd in the world in 2020, [43]), compared to France (22nd in the world). In addition, because some countries do not have access to Google Play or use alternative app stores, and since the NoiseCapture application is only available on Google Play store, it is not surprising that they are not found as a data producer. This is the case of China (Google Play not available in China) and Russia (an alternative app store is mainly used), for example, while they represent an important part of the world's population (1st and 9th in the world, respectively).

Table 10. Distribution of the collected data per country and ranking (first ranks are displayed in bold). Population per country (percentage of world population) data are from in [43].

Country	Population (Rank)	Contributors (Rank)	Tracks (Rank)	Points (Rank)	Points/Track
China	**17.9% (1)**	58 (54)	354 (43)	158,797 (26)	448.6
India	17.5% (2)	894 (4)	2241 (16)	243,778 (22)	108.9
United States	4.2% (3)	**29,108 (1)**	**88,341 (1)**	**22,676,833 (1)**	256.7
Indonesia	3.4% (4)	91 (45)	199 (55)	20,244 (65)	101.7
Pakistan	2.8% (5)	124 (32)	321 (45)	27,035 (57)	84.2
Brazil	2.7% (6)	448 (14)	1503 (20)	244,482 (21)	162.6
Nigeria	2.7% (7)	33 (67)	66 (78)	9319 (75)	141.2
Bangladesh	2.1% (8)	160 (26)	572 (28)	332,316 (19)	581
Russia	1.86% (9)	174 (24)	850 (24)	94,277 (33)	110.2
Germany	1.1% (19)	790 (7)	3093 (7)	1,216,164 (5)	393.2
France	0.9% (20)	5516 (2)	27,911 (2)	4,972,054 (2)	178.1
United Kingdom	0.8% (21)	1164 (3)	4693 (4)	2,067,182 (3)	440.5
Canada	0.5% (37)	792 (6)	2512 (15)	1,551,808 (4)	**617.7**
Peru	0.4% (42)	77 (48)	11,231 (3)	138,716 (27)	12.3
Netherlands	0.2% (67)	435 (15)	3409 (5)	413,897 (16)	121.4

Although the ranking of Peru and the Netherlands in terms of population is low, these two countries are in the top 5 in terms of number of tracks (Table 10). For these countries, the number of tracks compared to the number of contributors is very high (especially for Peru), which highlights an intensive measurement activity, which is perhaps part of a voluntary and organized action (like a NoiseCapture Party for example). A spatio-temporal analysis of the data produced in these two countries, as well as a detailed analysis of the behavior of the corresponding contributors, could eventually provide some answers. More globally, the implementation of cluster detection techniques could be an interesting way to identify organized events.

While conducting the study, it was observed that 32.5% of the tracks contain points without geolocalization (i.e., the field `the_geom` is empty), which represents a total of 10,783,609 points (18% of the total number of points), distributed over 141 countries. For 75% of these tracks (63,538 tracks), all the corresponding points are concerned by a lack of geolocalization (i.e., the whole track can not be geolocalized). The main reason is that the geolocalization has not enabled on the smartphone. By further analyzing, it was also observed that a large part of these tracks correspond to indoor measurements (20,045 (7.7%) of the corresponding tracks are defined with the 'indoor' tag). Tracks with a partial lack of geolocalized points may be due to a local loss of GPS localization, for example, when passing through a tunnel. A spatial analysis crossed with other geographical data can possibly bring elements of answer in this case.

Even when the measurement points are localized (i.e., the GPS actually transmits a position), this measurement can have a poor accuracy. Putting aside the technical quality of the hardware used in the smartphone for GPS location, this poor accuracy may be obtained when the measurement is made in an environment that is not clear enough (in or near a building, overcast sky) making it difficult to connect to a sufficient number of GPS satellites.

3.3.2. Accuracy

The `'accuracy'` data collected by the NoiseCapture application allow one to associate a location accuracy (in meters) to each measurement. This value is obtained using the `getAccuracy()` function in Android [44], meaning that there is a 68% probability that the true location is inside the circle (with a radius equal to the value of the 'accuracy') centered at the corresponding location. The analysis of this parameter shows that the median value of `'accuracy'` is around 8 m. It should also be mentioned that it is possible to find some measurement points with non-realistic accuracy values (such as 1.1×10^5 m) that may due to a wrong technical implementation of the GPS algorithm in the smartphone. For points with geolocalization, Table 11 shows that most of accuracy are under 25 m (42,313,601 points, 86.5%), which can be considered as a relevant accuracy for noise studies [45], and 35,184,828 (71.9%), 18,069,523 (36.9%), and 420,150 (0.8%) under 15 m, 5 m, and 1 m, respectively. Finally, one can observe from Figure 8 that the accuracy tends to increase (i.e., the accuracy value decrease) when the measurement duration increases. This is due to the fact that a sufficient duration may be required for the GPS receiver within the smartphone to detect GPS satellites, and then to obtain the best accuracy of location. It suggests that a NoiseCapture user should wait few seconds after starting the application (for example, ref. [14] mentions a duration of 4 seconds), before performing a measurement, in order to obtain the best geolocalization.

Last, it must be mentioned that the function `getAccuracy()` returns the value 0.0 when the smartphone is not able to obtain a value for the accuracy. This should not be consider as an accuracy value of 0 m.

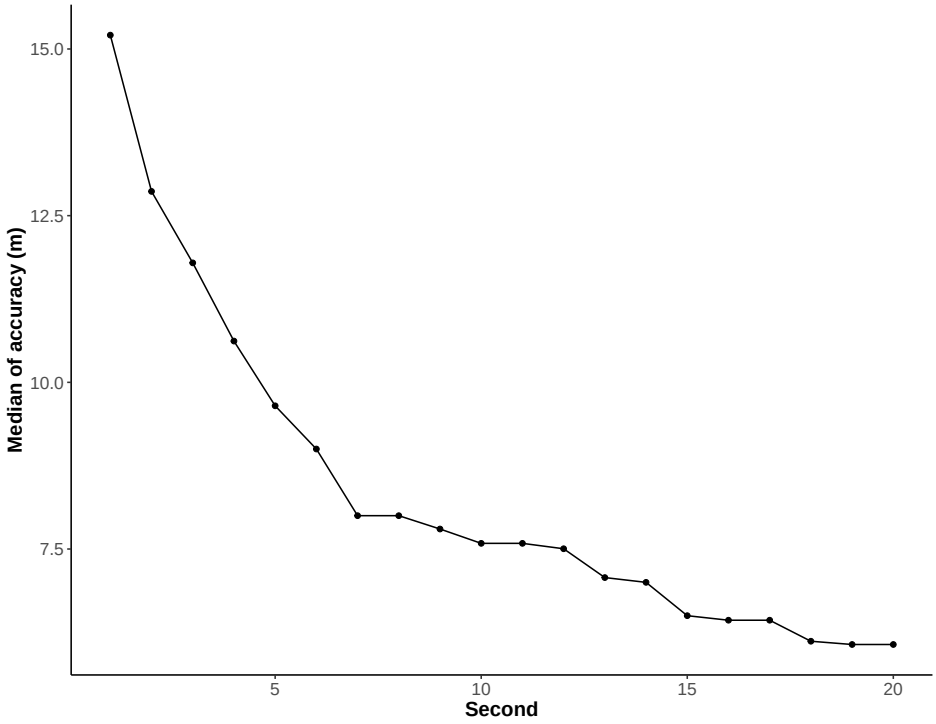


Figure 8. Evolution of the median value of the accuracy of geolocalization (for all measurement points with geolocalization) in function of time, since the first second of measurement.

Table 11. Distribution of `accuracy` for the point with geolocalization.

'accuracy'	Number of Points	%
0	0	0
[0, 1]	420,150	0.9
[1, 2]	1,400,466	2.8
[2, 3]	3,574,004	7.3
[3, 4]	7,406,060	15.1
[4, 5]	5,268,843	10.8
[5, 10]	11,142,305	22.8
[10, 15]	5,973,000	12.2
[15, 25]	7,128,773	14.6
[25, 35]	1,725,169	3.5
[35, 50]	1,192,908	2.4
[50, 100]	1,426,475	2.9
>100	2,243,566	4.6
Total	48,901,719	

3.3.3. Speed

The NoiseCapture data set contains information about the `'speed'` value, which represents the speed (in meter per second) measured by the smartphone GPS at the time of the measurement point. Table 12 shows that 38.5% of tracks (65.4% of the measurement points) have a speed equal to 0. According to the Android documentation for the `getSpeed()` function [46], a null value is returned when the location does not have a speed; it does not mean that the speed is equal to zero, but that it is not possible to evaluate its value, even when the measurement is geolocalized. Note also that it could be interesting to also collect the estimated speed accuracy using the Android function `getSpeedAccuracyMetersPerSecond()` in a future release of the application.

Table 12. Distribution of 'speed' for points with geolocalization.

'speed'	Number of Points	%
0	31,986,102	65.4%
[0, 1.4]	837,662	18.2%
[1.4, 4.2]	2,058,585	4.2%
>4.2	5,919,370	12.2%
Total	48,901,719	

When the speed value is different greater than 0, it means that the measurement point is moving, but it may be difficult to determine in a simple way the transportation mode that is used (walking/running, bicycle, light vehicle, public transportation, etc.), with the only knowledge of the speed value, as the speed ranges corresponding to each transportation mode may overlap [47]. Assuming people walking at a speed ranging from 0.5 km/h to 5 km/h (0.14 to 1.4 m/s, respectively), one can consider that around 14.4% of the measurements are realized during walking. One can also find speed values that correspond very clearly to measurements carried out in fast mode of transportation, including air transportation of the order of 280 m/s.

Additional analysis of the 'speed' information also shows several anomalies, such as negative values (26,304 measurement points, 0.04%), with the '−2' (26,257 meas.) or '−1' (14 meas.) values; such values may probably have a signification, but this information is missing in the Android documentation. Other negative values (33 meas) are in the range $[-1, 0]$ and may be due to numerical accuracy.

3.4. Temporal Characteristics of Measurements

3.4.1. Measurement Timestamp

As already mentioned, the analysis developed here concerns an extraction of the database, as the official launch of the application over a period of 3 years, from 29 August 2017 to 28 August 2020 (considering all versions of the application since number 28). At the time of a measurement, the beginning of a 'track' is defined by the field `record_UTC` (given by the smartphone) and each 'point' of a 'track' is defined by the field `time_location` (given by the GPS).

The analysis of the entire database (i.e., between the date of the track `record_UTC` and the date of the first point `time_location` in the corresponding track), shows some measurements that are visibly incorrectly time-stamped; this corresponds to points without geolocalization (defined with the `time_location`='1970-01-01' by default). One can also observe measurements (21,897 tracks, 849,128 points) with a time shift of several hours (Table 13), but it represents less than 2% of the total number of tracks. Last, it is also possible that some users use date and location metadata scrambling tools on their smartphone to avoid tracking. The number of tracks/points concerned being however very low, one can imagine that the database analyzed here is little or not at all concerned by this type of error.

Table 13. Time shift (in hour) between `record_utc` and `time_location`, for the tracks are are 100% geolocalized.

Time Shift (h)	Number of Tracks	%
<−24	797	0.4
−24	151	0.07
−23	35	0.02%
−22	4	0.002%

Table 13. *Cont.*

Time Shift (h)	Number of Tracks	%
−20	3	0.002%
−19	8	0.004%
−18	8	0.004%
−15	1	0.0005%
−14	1	0.0005%
−13	3	0.002%
−12	25	0.012%
−11	17	0.008%
−10	8	0.004%
−9	6	0.003%
−8	3	0.002%
−7	10	0.005%
−6	7	0.004%
−5	6	0.003%
−4	10	0.005%
−3	67	0.034%
−2	47	0.023%
−1	258	0.13%
0	194,690	98.54%
1	479	0.242
2	54	0.027%
3	28	0.013%
4	15	0.008
5	10	0.005%
6	6	0.003%
7	6	0.003%
8	10	0.005%
9	3	0.002%
10	10	0.005%
11	4	0.002%
12	1	0.0005%
13	3	0.002%
14	2	0.001%
15	8	0.004%
16	11	0.006%
18	8	0.004%
19	4	0.002%
20	5	0.002%
21	2	0.001%
22	3	0.002%
23	18	0.009%
24	11	0.006%
>24	702	0.355%
Total	197,568	

Figure 9 illustrates the distribution of the tracks in function of the hour of a day. For the entire database (Figure 9a), one can observe a moderate variation from one hour to another, which can be explained by the fact that measurements are collected in all the time zones simultaneously (assuming that measurements are done all over the word simultaneously). When focusing on the data collected in France only, Figure 9b shows, as expected, a small number of measurements during night and early morning and more measurements during day and afternoon.

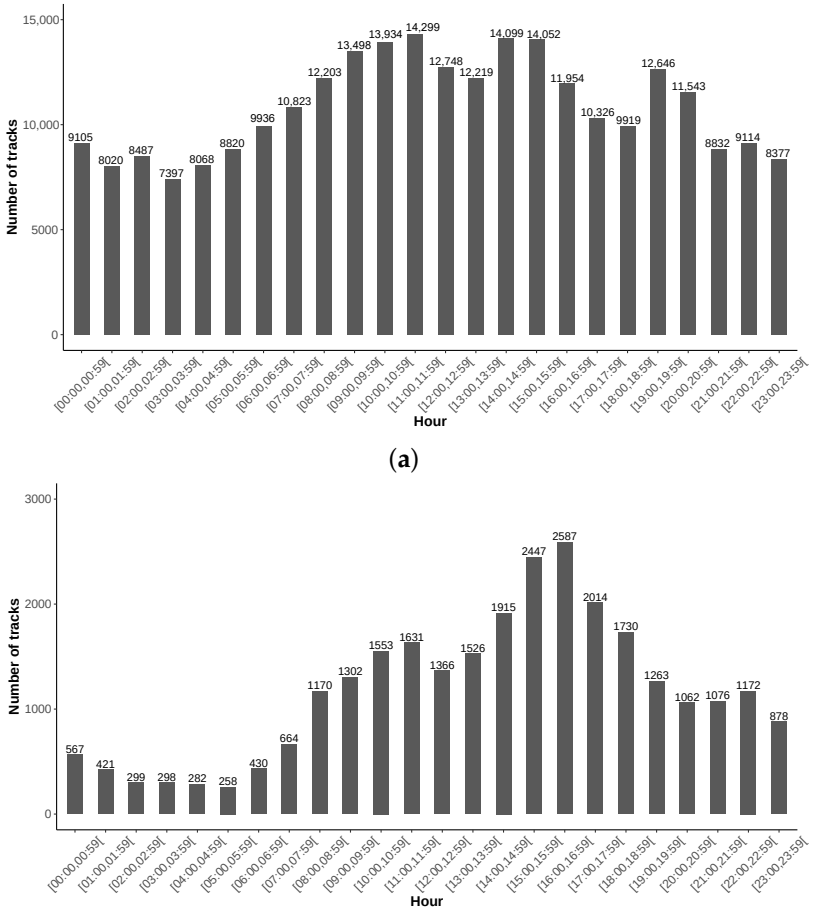


Figure 9. Distribution of tracks per hour of the day, for the entire database and for the data collected in France only. (**a**) Distribution of tracks per hour of the day, for the whole database. (**b**) Distribution of tracks per hour of the day for data collected in France only.

In addition, Figure 10 illustrates a small variation from one day/month to another, except for `'October'` with more tracks for the year 2018, due to an unusual and large amount of data (8470 tracks) collected on 8–9 October 2018, by few users only, localized in Peru (6326 tracks, 78,789 points). This can be due to a specific event.

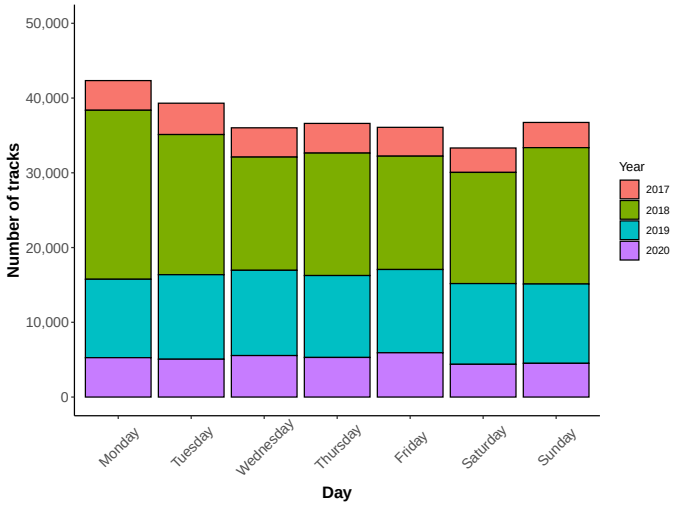
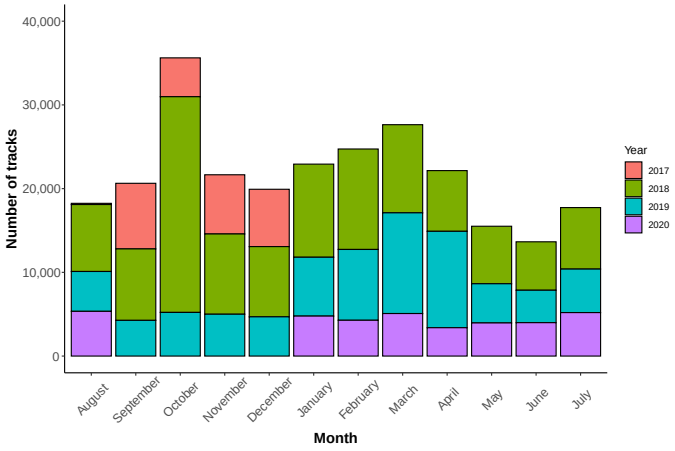


Figure 10. *Cont.*



(**b**)

Figure 10. Distribution of track measurements in function of the day and the month, over the 3 years of the collected data. (**a**) Distribution of tracks per day of the week. (**b**) Distribution of tracks per month.

3.4.2. Measurement Duration

The ‘time_length’ data (in second) are defined from the start of the measurement during a track, until the user ends the measurement. Figure 11 shows that most measurements are done with a track duration around 1–20 s (44.6%) and 1–3 min (17.3%). Only 6.6% of tracks have duration greater than 10 min. The 10 s duration corresponds to a large part of the measurement (51,098 tracks 19.6%); this is due to the fact that user can used a predefined duration, which is fixed to 10 s by default. One can note that measurement duration between 1 and 3 min has also as strong presence. Table 14 shows that only small percentage of user collecting tracks with long period move along the track ($\leq$6.45%, when considering a minimum speed value of 0.5 m/s).

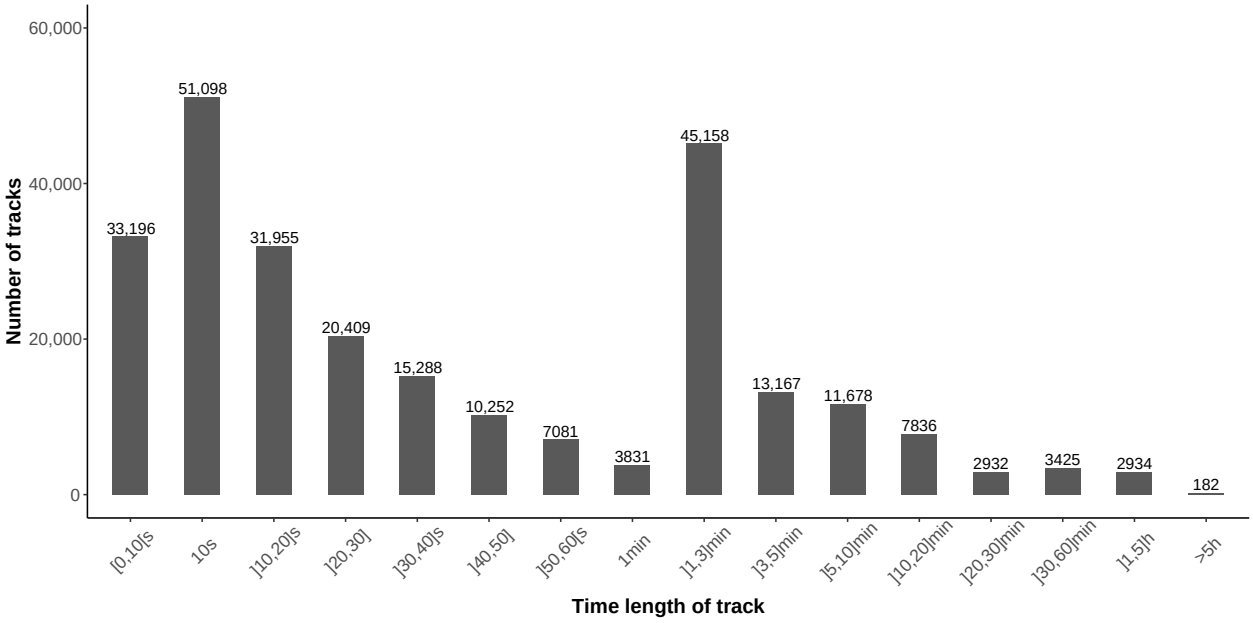


Figure 11. Distribution of tracks in function of the duration of the measurement (‘time_length’).

Table 14. Total number of collected data and distribution per time length and part of the measurement tracks and points that have been collected in motion (only data with geolocalization are considered, with a speed greater than 0.5 m/s).

'time_length'	Tracks		Points	
	Total	Moving	Total	Moving
[1, 3] min	21,963	4477 (20.4%)	679,675	437,640 (64.4%)
[3, 5] min	7190	1575 (21.9%)	516,389	343,035 (66.4%)
[5, 10] min	7204	2029 (28.16%)	1,080,596	769,976 (71.2%)
[10, 20] min	5231	1678 (32.1%)	1,661,753	1,189,641 (71.6%)
[20, 30] min	2083	752 (36.1%)	1,237,893	925,499 (74.7%%)
[30, 60] min	2362	741 (31.4)	2,094,625	1,446,091 (69.0%)
[1, 5] h	2026	466 (22.9%)	3,694,683	1,912,894 (51.7%)
>5 h	142	19 (13.4%)	752,485	189,075 (25.1%)

An issue can be observed regarding the total number of measuring points (59,685,328), which is greater than the sum of all the measurement track duration ($\sum$ time_length = 59,684,657), with a difference of 671 s (i.e., points). Analyzing this issue, one can observe that 82 measurement tracks (made by different users, with different devices, using different application release) have a number of points (i.e., seconds) that are not equal to `time_length`. In addition, one can mentioned that few points have been removed from their corresponding tracks. For now, the exact reason of such anomalies is still not defined, but may be due to an unusual behavior of the application or to numerical inaccuracies.

3.5. Smartphone Acoustic Calibration

The relevance of the collected acoustic measurements is largely based on the measurement protocol applied by the contributor as well as on the metrological quality of the smartphone. At this stage, concerning the first point, it is expected that the contributor follows the recommendations available in the application. No other information is included in the collected data in order to analyze if this measurement protocol is well followed (excepted for the calibration); this point will be discussed in Section 4. The second point has given rise to numerous studies in the literature, as it is a critical element.

It is indeed hoped that the user calibrates his/her smartphone before collecting measurements. Numerous studies, among them recent ones [48–50], have shown the need to make a correction on the values measured by smartphones, in order to get closer to those that would have been measured with a reference device, such as a sound level meter. However, from a statistical point of view, this condition is not as critical, since it can be expected that due to a large number of measurements collected by different smartphones, the results may statistically converge to the expected values. This hypothesis seems to be confirmed by Murphy and King, showing, that in the absence of calibration, the sound levels measured on average by a wide variety of smartphones are very close to the expected value, but at the expense of a large standard deviation. Acoustic calibration can however reduce the standard deviation.

In the NoiseCapture application, the calibration procedure consists in evaluating a correction (or a calibration gain, i.e., the `'calibration_gain'` value) that will be applied on the input temporal signal before the postprocessing of all noise indicators, assuming a linear relationship both in frequency and in amplitude, which is of course questionable for some smartphones. Within NoiseCapture, several calibration methods are proposed and defined by the field `'calibration_method'` in the database:

- The most relevant solution ('Calibrator' method) consists in using an acoustic calibrator, according to the classical rules for acoustic measurement. This solution requires an external microphone, connected to the smartphone, with a diameter compatible with the use of an acoustic calibrator. Note that using an external microphone can also improve the measurement accuracy in comparison with the internal microphone [51]; thus, this solution must be promoted to contributors. The `'calibration_gain'` is

then determined for a reference frequency and for a reference level (for example 94 dB@1 kHz) and applied, during measurements, to the entire temporal signal before processing;

- Another method ('Reference' method) is used to correct the sound level measured (overall or for a given frequency) by the smartphone using another measuring device (i.e., using a visual comparison), considered as a reference. The value of the gain `'calibration_gain'` is obtained from an ambient noise measurement.
- A third method ('CalibratedSmartPhone' method) is used to calibrate one or more smartphones simultaneously, using an already calibrated smartphone as a reference. The procedure is fully automatic, controlled by the reference smartphone, and is based either on the measurement of the ambient noise or a pink noise generated by the reference smartphone.
- Finally, a more recent method ('Traffic' method) is based on the measurement of several pass-by of light road vehicles, which, by comparison with a statistical model of noise emission, makes it possible to estimate the correction to be made to make the measurement coincide with the expected statistical value [52].

The user can also directly change the value of the calibration gain in the application settings at any time. The default value of `'calibration_gain'` is set to 0 dB as long as no calibration method has been applied, or as long as the user has not directly changed this setting. A change of this parameter will be considered as a 'ManualSetting' for the `'calibration_method'` field.

The choice of the method is defined by the field `'calibration_method'`, but only since version N°49 (17 February 2020). If, since the launch of the application, several calibration methods were already available, the information on the choice of the applied method was not known and only the value of `'calibration_gain'` was actually uploaded to the remote server. For database consistency reasons, all data collected using versions prior to version 49 of the application, the choice 'None' is affected to `'calibration_method'`, although a calibration method may have been used. However, since version 49, the choice 'None' is only affected when no calibration is performed.

Table 15 shows the distribution of the collected tracks according to the calibration method. As indicated above, the field that defines the choice of the calibration method, is only available since release 49. By analyzing the data collected before release 49, one can observe that 62,731 of the 241,532 collected tracks, i.e., ~26%, have a calibration value different from 0, meaning that the corresponding users have probably applied either a calibration method or a manual change of the calibration gain in the settings of the application.

Table 15. Distribution of tracks in function of the calibration method, before and after release 49 ('n.a.' for 'not available').

`'calibration_method'`	Since Release	Nb of Tracks before R49 (%)	Nb of Tracks Since R49 (%)
CalibratedSmartPhone	33	n.a.	277 (1.4%)
Calibrator	28	n.a.	139 (0.7%)
ManualSetting	28	n.a.	838 (4.4%)
None	28	241,532	17,395 (92.1%)
Reference	28	n.a.	167 (0.9%)
Traffic	49	n.a.	74 (0.5%)
Total		241,532	18,890

From release 49, we can see that 7.9% of the tracks have been made by smartphones that have been calibrated (92.1% are defined by 'None' for the `'calibration_method'`), but for about half of them, the manual method has been applied. For these tracks, it is therefore difficult to determine how the value of the calibration gain has been evaluated. Among the other calibration methods, the one using the automatic procedure between smartphones is the most used, followed by methods using a reference and an acoustic calibrator. The traffic calibration method is the most recent and has generated little data

so far. In the future, it will be important to highlight this last method, which is the only one that is able to calibrate a smartphone without the need for an external device, while offering sufficient accuracy.

The application of a calibration method is not enough to justify the quality of the measurements. The obtained value of the calibration gain (field 'gain_calibration') is also a very important information. Figure 12 illustrates the distribution of tracks according to this value and brings some comments. The presence of abnormally high (in absolute value), even extreme and aberrant values shows either a bad use of the calibration methods or a technical problem. The number of tracks collected with a calibration gain of zero (default value) globally reflects a lack of calibration, as it is unlikely that a smartphone is calibrated by default. Finally, we can see that 86.8% of the collected tracks have a calibration value between −10 and +10 dB, which seems rather realistic, but does not bring any certainty on the quality of the measurement.

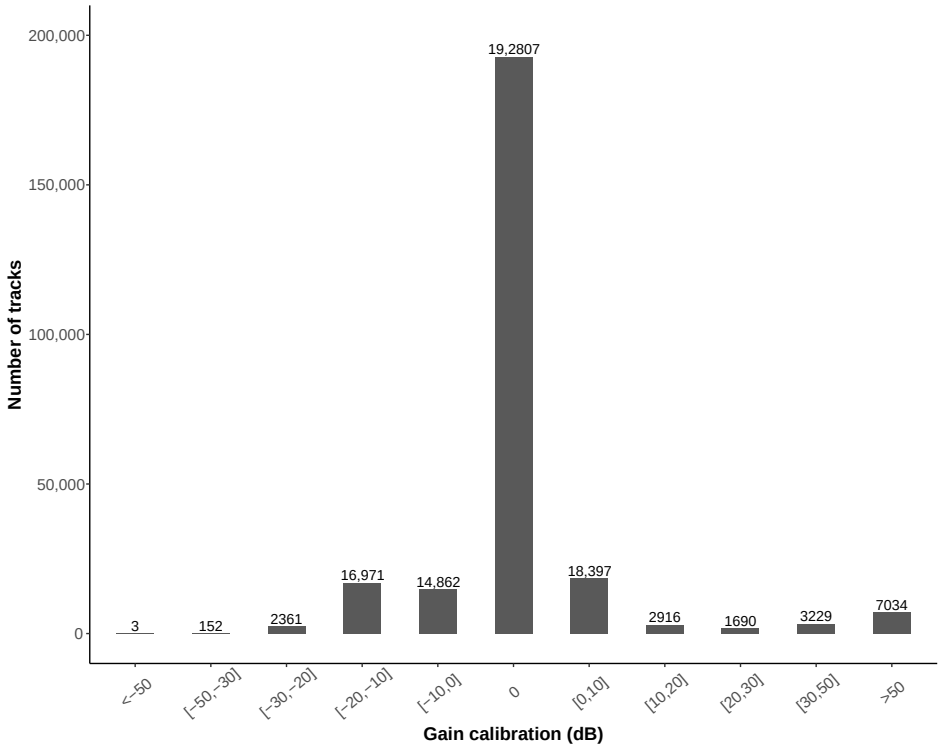


Figure 12. Number of tracks in function of 'gain_calibration' value.

Table 16 completes this first analysis by showing the distribution of calibration gain values according to the method (data collected since release 49). As expected, it can be observed that when a calibration method considered as 'robust' is applied ('CalibratedSmartPhone', 'Calibrator' and 'Traffic'), the gain is different from 0 dB, except for the 'Reference' method. There is also slightly less disparity in the gain values when a calibration method is applied. One can also observe that for the 'Calibrator' method, 90.7% of the calibration gain values are greater than 10 dB, which shows a different behavior than other methods. This may be a misunderstanding of the method, with some users attempting to calibrate their smartphone without a reference device.

The analysis of these calibration data, for example, in relation to the type of device and user profile, is in itself a separate study. The creation of a 'validated' database of calibration values for each smartphone model, for example, is an interesting prospect. However, this perspective study is beyond the scope of the present article, which, at this stage, only aims to present the data collected and their limits of use.

Table 16. Cross table of the number of collected tracks in function of the calibration method and the calibration gain (data from release 49).

Calibration Method/Gain	<−10 dB	[−10, −5] dB	[−5, 0] dB	0 dB	[0, 5] dB	[5, 10] dB	>10 dB	Total
CalibratedSmartPhone	-	61 (22.0%)	136 (49.1%)	-	75 (27.1%)	3 (1.1%)	2 (0.7%)	277
Calibrator	1 (0.7%)	2 (1.4%)	3 (2.2%)	-	7 (5.0%)	-	126 (90.7%)	139
ManualSetting	56 (6.7%)	13 (1.6%)	150 (17.9%)	126 (15.%)	156 (18.6%)	54 (6.4%)	283 (33.8%)	838
None	559 (2.4%)	911 (5.3%)	865 (5.1%)	13,857 (79.9%)	485 (2.9%)	227 (1.5%)	491 (2.9%)	17,395
Reference	9 (5.4%)	21 (12.6%)	38 (22.7%)	23 (13.8%)	57 (34.1%)	7 (4.2%)	12 (7.2%)	167
Traffic	20 (27.0%)	14 (18.9%)	28 (37.8%)	-	6 (8.1%)	5 (6.8%)	1 (1.4%)	74
Total	645	1022	1220	14,006	786	296	915	18,890

3.6. NoiseCapture Parties

As mentioned above, a NoiseCapture Party is a special event organized by a given organization, aiming to carry out measurements, generally over a limited time and a spatial area, for educational, scientific dissemination, or research purposes. The advantage of these events lies in the fact that the measurements are generally well 'controlled', and most of the smartphones have been previously calibrated. It can thus be considered that the collected measurements have a better quality compared to the other measurements in the database.

The list of all NoiseCapture Parties are given in Table 17, with the number of considered smartphones and the total of collected tracks and points. As expected, the ratio of calibrated smartphones is greater for NoiseCapture Parties. The total number of collected tracks and points represents 1.2% and 0.6% of all data in the database.

Table 17. List of NoiseCapture Parties. More information are located in the `'noisecapture_party'` table of the database. The number of contributors, as well as the total of collected tracks and points are given. The organization in charge of the NoiseCapture Party is also mentioned (Noise-Planet is the organization in charge of the development of NoiseCapture application). While the CICAM NoiseCapture Party has been planned, it was canceled due to the pandemic situation (i.e., there are no corresponding tracks).

'pk_party'	'tag'	Organization	'filter_area'	'filter_time'	Contributors	Tracks	Points
1	SNDIGITALWEEK	Noise-Planet	TRUE	FALSE	7	133	11,523
2	ANQES	Noise-Planet	TRUE	TRUE	4	29	4479
3	FDS2017	Noise-Planet	TRUE	TRUE	2	6	1239
5	IMS2018	Noise-Planet	TRUE	FALSE	13	67	18,793
6	UDC	Universidade da Coruña	TRUE	TRUE	8	56	6879
9	TEST44	Noise-Planet	TRUE	TRUE	1	3	91
10	UNISA	University of Salerno, Italy	TRUE	TRUE	13	149	15,912
11	PNRGM	Noise-Planet	TRUE	TRUE	2	13	6089
12	AMSOUNDS	Waag Technology & Society	TRUE	TRUE	2	18	693
13	PNRGM	Parc Naturel du Morbihan	TRUE	TRUE	14	100	21,470
14	FDSSTRAS	Noise-Planet	TRUE	TRUE	5	31	2967
15	AGGLOBASTIA	Noise-Planet	FALSE	TRUE	19	507	59,838
17	FDSNTS	Noise-Planet	TRUE	TRUE	7	66	5916
18	H2020	Noise-Planet	TRUE	TRUE	11	89	22,060
19	UDC	Universidade da Coruña, Spain	FALSE	TRUE	20	138	5866
20	MSA	Noise-Planet	TRUE	TRUE	9	9	1885
21	GEO2019	Noise-Planet	TRUE	TRUE	43	420	63,521
22	IMS2019	Noise-Planet	TRUE	TRUE	23	192	17,309
23	FPSLYO	Noise-Planet	TRUE	TRUE	11	34	10,285
24	SSSOROLL2019	Generalitat de Catalunya	FALSE	TRUE	68	372	36,272
26	UNISA	University of Salerno, Italy	TRUE	TRUE	20	332	23,220
27	FDSSTRAS	Noise-Planet	TRUE	TRUE	3	7	1771
28	H2020	Noise-Planet	TRUE	TRUE	9	39	32,948
29	UDC	Universidade da Coruña, Spain	FALSE	TRUE	9	73	2099
30	MSA	Noise-Planet	TRUE	TRUE	10	10	3665
31	CICAM	EPN, Quito, Ecuador	FALSE	TRUE	-	-	-
32	UDC_COVID	Universidade da Coruña, Spain	TRUE	TRUE	33	249	14,785

It must be specified that other similar events could have been organized, without having given rise to the creation of a specific tag, and without having informed the developers

of the application. For example, this is the case for several recently published research works [53–57].

3.7. Soundscape Description

NoiseCapture allows users to complete the acoustic measurement with information about his/her own perception of the sound environment, using 'tags' (field `'noisecapture_track_tag'`) for describing the noise environment and the noise source along the track. In addition, they can give an information of 'pleasantness' (field `'pleasantness'`) by selecting a value (0 for 'unpleasant', 25, 50, 75, 100 for 'pleasant'); this field may be empty if no value is selecting (default value).

The analysis of the tags can be particularly interesting to distinguish the measurement conditions; indeed in certain conditions, such as rainy or windy weather, the acoustic measurement may be distorted, and it is therefore interesting to have such information before analyzing the acoustic indicators. The information about the indoor/outdoor measurement is also interesting for people who would like to use the data to characterize indoor or outdoor sound environment, specifically. Last, the knowledge of the sound sources that are perceived and the evaluation of the pleasantness are also interesting data for researchers that study the notion of soundscape.

Both `'Pleasantness'` and `'Tags'` are supplementary info. Their use add beneficial information about the sound environment. Ideally, this information should be systematically provided by the contributors. However, Table 18 shows that only 17.5% of the tracks have both information, while 48.7% do not have any and 33.8% have either one of them (mainly the pleasantness with 30.2%). An independence test showed that there is a dependency between using both `'Pleasantness'` and `'Tags'`: a participant using tag will use pleasantness more often.

Table 18. Cross table between pleasantness and tags.

Tag/Pleasantness	Used	Not Used
Used	45,549 (17.5%)	78,814 (30.2%)
Not used	9457 (3.6%)	126,602 (48.7%)

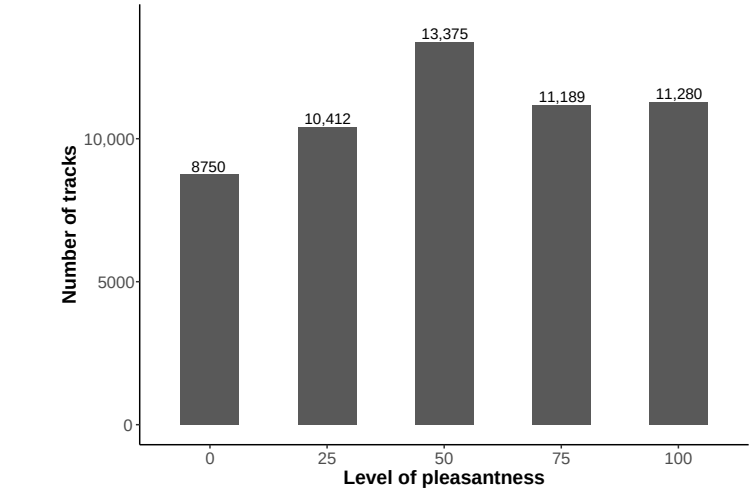
3.7.1. Pleasantness

Figure 13a shows that most of tracks (205,416 (78.9%), equivalent to 46,623,131 points (78.1%)) are not associated with a value of pleasantness, meaning that the default empty value is not modified. Excepted for the level '50', all other values are used quite uniformly. The over-representation of the level value '50' can be explained by three possible reasons:

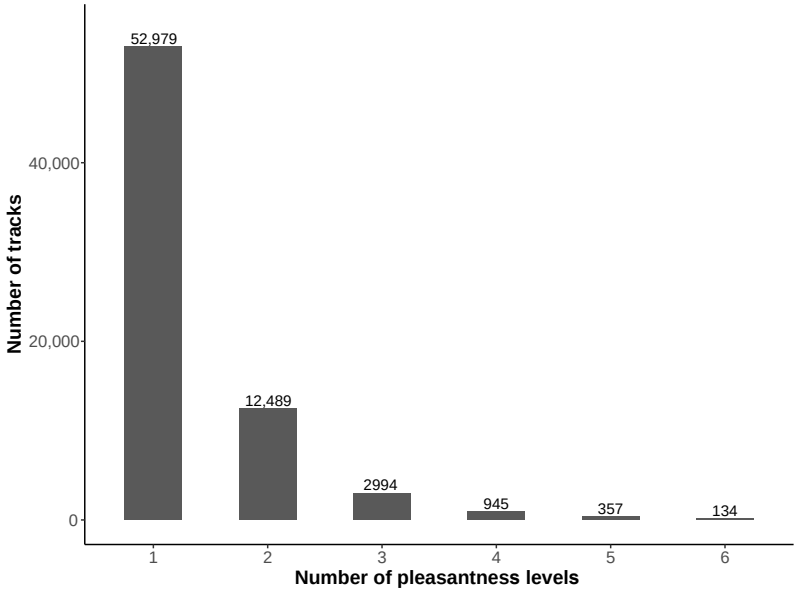
1. for the most part, users cannot judge the quality of the sound environment in a clear-cut way;
2. by default, the selection cursor is positioned on the value '50' that can influence the user;
3. the user may be tempted to select the cursor, without however wanting to make a decision. Once the cursor is activated, it is no longer possible to go back and a value will be automatically validated.

The two last hypothesis can introduce a bias, suggesting that a more suitable selection mode should be proposed in a future release of the application.

The behavior of a contributor can also be analyzed in terms of his propensity to use all the possible values of the pleasantness scale (Figure 13b). This analysis shows that for 52,979 users, only 1 level is used; however, in detail, for 43,764 of them (i.e., 82.6% of 52,979), the 'NULL' value is used; 12,489 used 2 levels, etc. Few users therefore use the pleasantness scale, and even fewer use these different levels of the scale. In a future evolution of the application, it could be interesting to 'motivate' users to provide information, for example by 'forcing' them to give an answer, including a 'don't say' answer.



(**a**)



(**b**)

Figure 13. User evaluation of pleasantness on a track: (**a**) Distribution of pleasantness values on tracks. (**b**) Distribution of number of pleasantness levels used by contributors.

3.7.2. Tags

In addition to the perception of their sound environment, users also have the possibility to specify the measurement conditions (4 tags) and the nature of the perceived sound sources (14 tags in four categories: human activity, transportation, natural, and mechanical activity). Table 19 gives a description of the tag fields in the database (`'pk_tag'` and `'tag_name'`) as well as the corresponding English description within the NoiseCapture application (see Figure 1). The list of `'pk_tag'` is not continuous, some missing numbers correspond to tags that are no longer used since the first official release of the application.

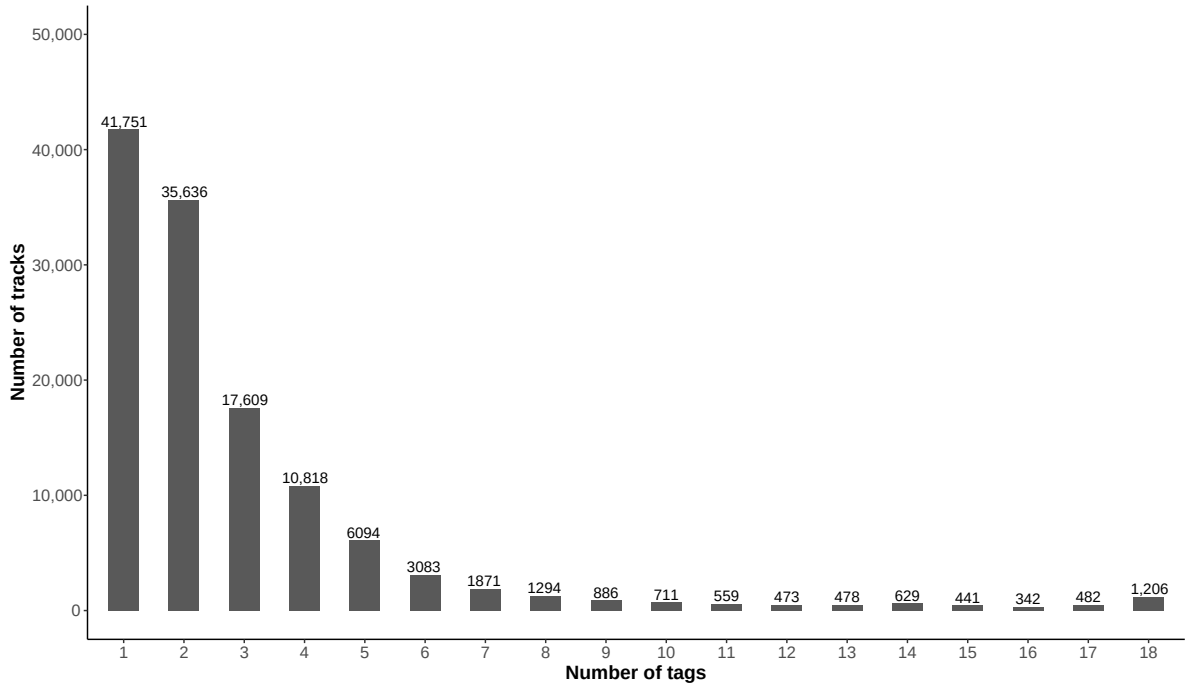
Figure 14a shows the number of tags that are simultaneously used to describe a track. In about half of the collected tracks (124,363, 47.7%), the contributors do not use any tag to describe the measure. This is better than for the pleasantness evaluation.

Nearly 30% of the tracks contain 1 or 2 tags: when considering 1 tag only, 17,094 tracks (40.9%) are defined by an environment tag (`'test'` or `'Indoor'`); when considering 2 tags, 2061 tracks are defined by 2 environment tags, 9260 tracks by 2 source tags and 24,315 tracks by a combination of two types of tag. One can also note that a number of tracks simultaneously contains a large number of tags, even the 18 possible tags, which is not realistic. We

can assume that the corresponding tracks are test measurements, but they do not necessary mention the 'test' tag. Figure 14b shows that this 'test' tag is used in 30,077 of the collected tracks, which is important. An analysis of the database, for the purpose of studying sound environments, will necessarily exclude the collected data with this tag.

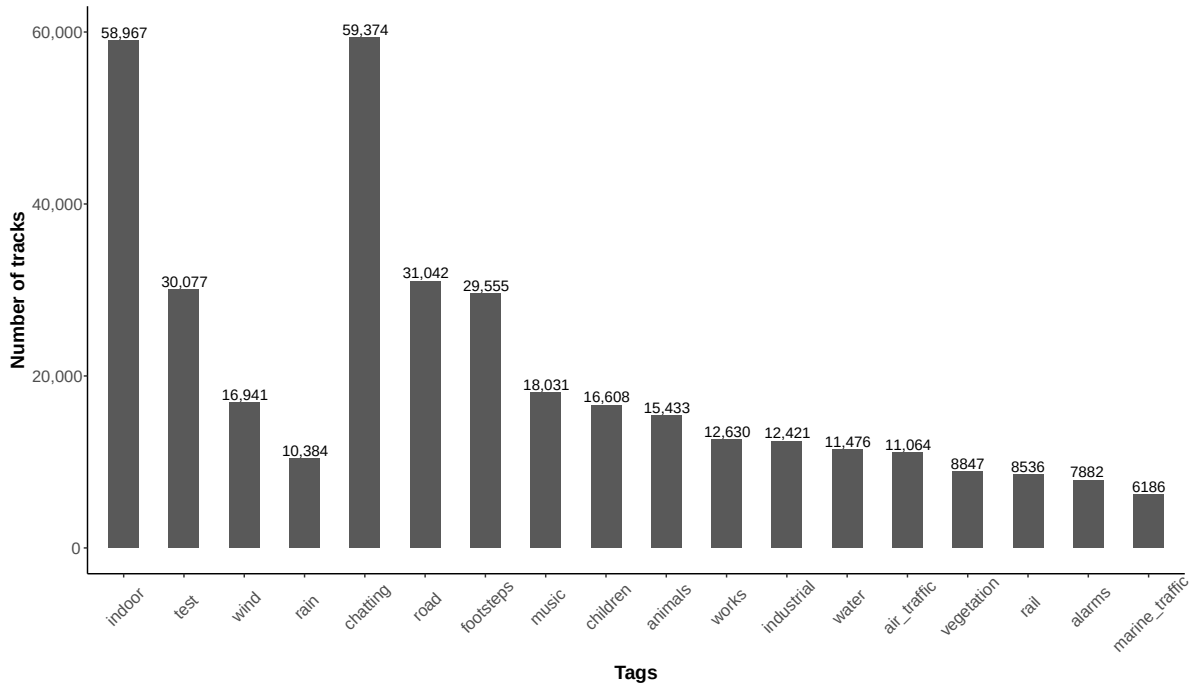
Table 19. Tags description: `'pk_tag'` and `'tag_name'` are the primary key and the name of the tags. The 'Description' correspond to the name of the tag in the corresponding NoiseCapture screen.

Category		**Measurement Conditions**		
`'pk_tag'`	1	6	13	23
`'tag_name'`	test	indoor	rain	wind
Description	Test	Indoor	Rain	Wind
		Human Activity Sources		
`'pk_tag'`	18	30	20	28
`'tag_name'`	chatting	children	footsteps	music
Description	Voice	Children	Footsteps	Music
Category		**Transportation Sources**		
`'pk_tag'`	27	32	26	35
`'tag_name'`	road	rail	air_traffic	marine_traffic
Description	Road	Rail	Air T.	Marine
		Natural Sources		
`'pk_tag'`	34	33	29	
`'tag_name'`	water	animals	vegetation	
Description	Water	Animals	Vegetation	
		Mechanical Activity Sources		
`'pk_tag'`	24	36	31	
`'tag_name'`	works	alarms	industrial	
Description	Works	Alarms	Industrial	



(**a**)

Figure 14. *Cont.*



(**b**)

Figure 14. Use of soundscape tags by contributors. (**a**) Number of tags simultaneously used in a track. (**b**) Tags name.

The other interesting aspect is that the 'Indoor' tag is present in 58,967 of the tracks collected, which represents an interesting quantity for the study of sound environments in closed spaces (building, transportation), even though the initial objective of the application was to study outdoor environments. Note also, that the tags 'Indoor' and 'Test' are not independent, meaning that both tags can be used together.

Among the sound sources mentioned by the contributors, 'voice', 'footsteps', and 'road' are present, which is consistent with a contributor who collects measurements closed to road infrastructures, while walking and talking. Again, it is obvious that the analysis of these tags and their occurrence can provide interesting information on the perception of sound environments. However, this is beyond the scope of this article.

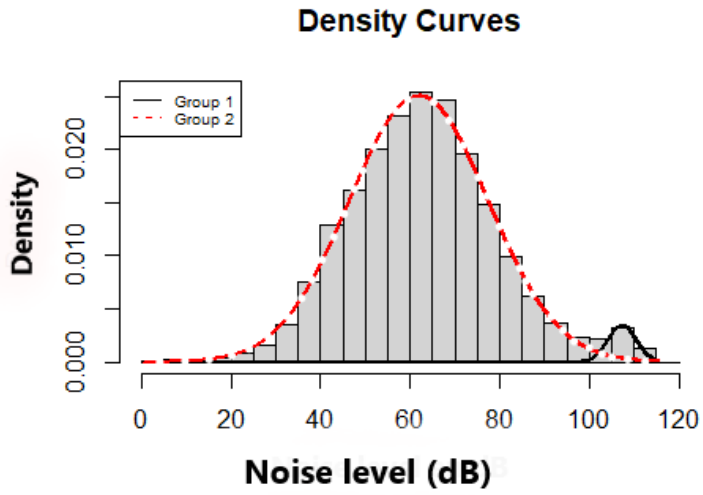
3.8. Noise Indicators

The purpose of the NoiseCapture application is based on the measurement of acoustic indicators for the analysis of sound environments. The data that are present in the NoiseCapture database concern the equivalent sound level $L_{A,eq}$ on a track, as well as the spectrum and sound level at each point of the track, measured every second. The postprocessing of these data can, in a second step, give access to percentile indicators (such as L_{A10} or L_{A50}) or to sound level distributions, for example. In the following, the analysis is restricted to the data as such, and not to the sound environments.

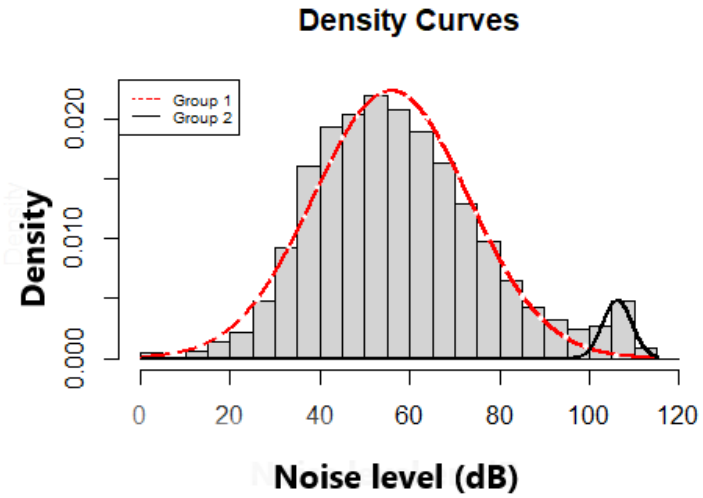
First of all, it should be remembered that in terms of acoustic measurement, smartphone manufacturers under the Android OS must respect a number of recommendations defined in the Android Compatibility Definition [38]. In particular, they should offer (1) an audio capture with approximately flat amplitude and frequency characteristics of ± 3 dB from 100 Hz to 4000 Hz, (2) an input sensitivity such that a 90 dB Sound Power Level (SPL) at 1000 Hz gives an RMS value of 2500 for 16-bit samples, and (3) a linear change of the amplitude over a range of at least 30 dB from -18 dB to $+12$ dB relatively to 90 dB SPL at the microphone. Some smartphones may offer superior features, but it is expected that all smartphones meet the minimum requirements.

Figure 15 shows the range of $L_{A,eq}$ values measured along the tracks and at each point of a track. While not visible in this figure, one can observe data with very low sound levels

(a few decibels, even negative ones), which seems physically both unrealistic in a real environment, but also *a priori* outside the measurement capabilities of a smartphone. On the other hand, the highest levels are of the order of 125 dB, which is not unrealistic but, nevertheless, unlikely in a normal environment.



(**a**)



(**b**)

Figure 15. Distribution of noise levels (**a**) on the 260,422 tracks and (**b**) on the 59,685,328 points.

In details, Figure 15a shows that the noise levels measured on tracks can be represented as a mixture of 2 normal distributions (noted $\mathcal{N}(\text{mean}, \text{standard deviation})$), a first group X_1 defined as $\sim \mathcal{N}(107.4, 3.1)$ and a second group X_2 defined as $\sim \mathcal{N}(62.0, 15.5)$. These normal distribution are also respectively defined by a `'gain_calibration'` of mean 73 and 0 dB (median 80 and 0 dB). Figure 15b shows similar results for the noise levels at the measurement points, as a mixture of 2 normal distributions, $\sim \mathcal{N}(106.3, 3.2)$ and $\sim \mathcal{N}(50.9, 17.2)$ with similar statistical values for the `'gain_calibration'`. For highest sound levels, it is quite evident that the calibration was not performed correctly.

This simple study shows that the range of variation of the measured sound levels is abnormally wide, the absence of calibration or a bad calibration being a probable cause.

4. Discussion and Future Developments

4.1. Synthesis

The analysis of the data collected during the first 3 years of operation since the launch of the application clearly shows that the information may be made of anomalies and uncertainties. Quantifying and reducing some of these biases is possible, either by a better knowledge and control of the user's behavior and of the context of measurement or by improving the smartphone application.

Table 20 already proposes at this stage some simple modifications to implement within the application, mainly by checking some settings (verification of the smartphone date/time, activation of the geolocalization, user profile update, change of the 'Pleasantness' selection mode). Most critical aspects concern the lack of a good geolocalization and bias in the noise level measurements mainly due to a wrong or a lack of smartphone calibration. These two subjects are specifically discussed in Sections 4.2 and 4.3. In addition, a better knowledge of the context of the measurement could also judiciously complete the collected data, or even replace certain user actions, such as the use of tags. Some suggestions will be proposed in Section 4.4. Last, increasing trust in the data also means increasing trust in the users. The animation of the community of contributors is another essential challenge. This is discussed in Section 4.5.

4.2. Increasing Localization Accuracy

In the current release of the application, localization is performed in an elementary way, and it was realized afterwards that it may not be sufficient depending of the objective of the use of data. With a view to improve the performance of the application (and therefore the quality of the data produced), the quality of GPS localization is a point on which the user must be made aware. In the application, this can for example take the form of recommendations to improve the quality of GPS location, such as activating the 'High accuracy' mode in the Android settings, re-calibrating the GPS *via* the use of a third party application, or activating additional localization functions *via* WiFi, Bluetooth and mobile networks.

4.3. Building a Smartphone Calibration Database

Some authors have rightly proposed to provide contributors with a database to calibrate smartphones, in order to limit bias during an acoustic measurement. Building such a database can be tedious because of the large number of smartphone models present on the market simultaneously, as well as their very rapid evolution. However, the NoiseCapture experimentation has opened new perspectives. Indeed, the analysis carried out in the present article shows that a large part of the contributions come from a limited number of manufacturers and models (three manufacturers account for about 35.2% of the models and nearly two thirds of the tracks; 15 models only have 15.9% of tracks). This information would limit the number of calibrations to be performed in the laboratory to build a calibration database. The other perspective would be to use the calibration data proposed by the contributors for their smartphone. The quality of this calibration can however be discussed, except for contributors performing their calibration during a NoiseCapture Party type event.

Table 20. Possible enhancements of the NoiseCapture application and database.

Data	Uncertainties/Bias	Possible Sources	Possible Solutions
User profile	Profile information is empty	Cannot evaluate the expertise of the contributor	In the app: update the field during an app update if the field is empty.
Geolocalization	No geolocalization of a track	The geolocalization is turned off	In the app: add a message for turning on the geolocalization
		Indoor measurements	In the app: wait for future methodologies (Indoor positioning System) and high sensitivity GPS for indoor localization
	No geolocalization of a point in a track	Local loss of geolocalization	Use GIS methodologies to re-locate the point within the track
	Inhomogeneous worldwide coverage	No access to Google Play	Use alternative app stores
Accuracy	Value equal to '0.0'	No geolocalization	In the app: add a message for turning on the geolocalization
	Extreme (not realistic) values	Unknown	No known solutions
	Large (but realistic) values		In the app: ask contributors to wait for a better localization before starting the measurement
Speed	Value equal to '0.0'	No geolocalization	In the app: add a message for turning on the geolocalization
	Negative values	Unknown	No known solutions
		No evaluation of the accuracy of the speed value	In the app: use the Android function `getSpeedAccuracyMetersPerSecond()` to store this missing information.
Timestamp	Wrong date	The geolocalization is turned off	In the app: add a message for turning on the geolocalization
		Wrong phone setting	In the app: check that the date is correct and add a message if not
Calibration	The calibration method is not known	The information about the selected calibration method is collected since the version 49 only	No solution
	Extreme (not realistic) values	No calibration method used	In the app: send a notification to calibrate the smartphone In the app: check the calibration value and send a notication if the value seems incorrect In the app/remote server: create a smartphone model calibration database
Pleasantness	Possible bias at level 50%	The default value is fixed at a pleasantness of 50%	In the app: change the selection mode for the pleasantness without default value
Noise levels	Extreme (not realistic) values	Calibration is not correct	Improve the calibration of the smartphone

4.4. Collecting Information about the Context Awareness

From the very beginning of the application creation, the kind of the information sent back to the remote server was deliberately restricted to what was strictly necessary, so that it would not be considered as invasive. The study of the data collected over 3 years nevertheless shows that their use in a better controlled scientific approach would require additional information.

In particular, information on the context of the measurement, such as wind detection, activity recognition, transportation mode detection, how the smartphone is used during measurements, and place recognition [47,58–62] could be useful. The use of information provided by other smartphone sensors (accelerometer, orientation, brightness sensor, and proximity sensor) could also provide information on the process of the measurement [14,17]. Note also that, as mentioned in [47], specific functions are already available in Android API to identify some user activities [63], which could be a first attempt to obtain new information. To a lesser extent, it may also be interesting to collect the speed accuracy (adding a new data `'speed_accuracy'`), since this value is also available in Android API.

Providing that smartphones have sufficient resources, the integration of sound source identification algorithms can also give interesting additional information, and can advantageously replace the use of tags [64]. Otherwise, it should also be possible to include such identification as a postprocessing on the remote server, for example, by using the collected 1 s spectra. All these development perspectives must nevertheless be integrated in the total respect of the privacy of the contributors [65], in particular, in the respect of laws in specific Regions/Countries, such as the General Data Protection Regulation (GDPR) in Europe [66].

4.5. Increasing and Animating the Community of Contributors

The participatory approach is of course the main originality of the application, allowing one to considerably multiply the number of measurement points, with a large variability in time and space. Like any participatory approach, the main challenge is to maintain the initial interest of the contributor to support a research project or make their individual contribution a major social issue (i.e., noise pollution) [67], beyond a time of discovery and a few measurements. The analysis of installs/uninstalls detailed in Section 2.4 shows indeed a tendency to a negative imbalance between installs and uninstalls of the application, which suggests that it is important to propose a solution to better retain users. Moreover, the analysis of the contributors behavior in Section 3.2.3, shows that there are finally few active contributors (half of the contributors made only one measurement, mostly to test the application), and that almost half of the contributions do not exceed 20 s (38.5%, Table 8). It is therefore require to develop strategies that allow for the development of a community of very active contributors.

This must be achieved by enhancing the application in order to motivate users to regularly produce measures, for example by adding reminder notifications to contributors or by developing a more playful aspect (creation of pseudonyms, setting up a challenge or a serious game based application such as noise battle or noise quest [68], creating badges…). If the target is more oriented towards a community of professionals (i.e., the initial target), the animation of the community can be more distributed, by calling upon 'ambassadors' (teachers, student researchers, technical agents of communities, government services…) who will see a particular interest in organizing, for example, NoiseCapture Parties. As also mentioned by others authors [48,67,69,70], the advantage of organizing 'controlled' events lies in the possibility of training users to carry out measurements using a validated protocol, particularly from the point of view of the calibration of smartphones, which would increase 'confidence' in the measurements. Once trained, users could in turn train other users, increasing the 'trusted' community.

Implementing a serious game type application, or increasing interactions with contributors, should also encourage the contributor to take measurements in specific spaces and at specific times. As considered by the authors of [14], this would make it possible to compensate for a lack of measurements in certain places or at certain times.

The analyzing of the geolocalization of the measurement points also showed a inhomogeneity in the diffusion of the application throughout the world, mainly due to the initial choice of the development platform (Android) and the associated application stores (on Google Play only). If one can observed the very wide use of the application throughout the world offers (currently, the application has been used in 204 different countries), this analysis shows the need to disseminate the application even more widely in order to acquire data in some countries with large populations. This could offer a wealth of data that is particularly interesting from the point of view of evaluating sound environments in countries with very different cultures and environments.

5. Conclusions

The use of a crowd-sourcing type approach offers interesting perspectives in the analysis of sound environments, in particular because of the spatial extent and the temporal dynamics that the data collected can provide. The involvement of citizens in a collaborative approach also brings another dimension to scientific research on the subject. The initial and legitimate fears about the relevance of using such data in an environmental approach (evaluation of public policies to reduce noise nuisance, effects of noise on health, perception of noise environments) are being allayed. Studies have indeed shown the relevance of this type of approach [67,69], while underlining some important points, for example, users proactivity, critical mass of contributors, increasing of measurement accuracy or the need of organizing collective sessions of noise sensing, etc.

The development of the NoiseCapture application is fully in line with this alternative approach. Compared to similar approaches, however, the NoiseCapture approach offers a completely open source platform, ensuring total transparency on the methods of collecting and processing data, and giving the possibility to everyone to freely use the data. The sustainability of the approach was also considered, by making effort to ensure the functioning of the project over time. These specificities are certainly the reasons for the success of the approach, whether it be with many communities.

Since the launch of the application on 29 August 2017, the amount of data collected is considerable. After 3 years of operation, thanks to the participation of 74,082 contributors, the database has accumulated 260,422 tracks and 59,685,328 one-second measurement points, spread over 204 different countries (Figure 16). To our knowledge, there is no other similar experimentation.

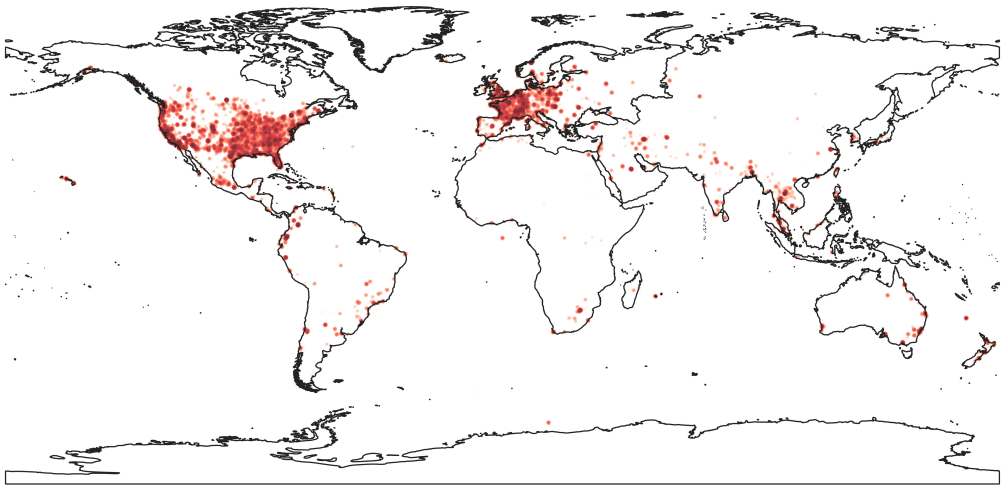


Figure 16. 'Heatmap' representation of the NoiseCapture data collected around the world.

Although the amount of data collected is considerable, any exploitation of the database for applications related to the study of sound environment requires a perfect understanding of the data, in order to limit bias in the analysis. The objective of this article was therefore to review all the data collected (nature, content, limits, etc.) and to identify specific behavior

linked to the use of the application (Section 3). This analysis now provides a precise framework for the further exploitation of the data. In view of the very large amount of data collected, it is however clear that depending on the nature of the expected analysis, a large part of the data cannot be used, either because it does not present any interest for the corresponding analysis, or due to a lack of completeness and accuracy. As discussed in Section 4, in our opinion, enhancing/controlling the quality of the data and of the measurement conditions constitute two major developments for improving the database. The other major perspective consists in the animation of the contributors community to increase confidence in the data.

Thus, as soon as attention is paid to the inherent limits of the collected data, the exploitation of this database offers very interesting perspectives on the characterization of sound environments. Any relevant analysis could be useful for communities to assess the noise environment of their territory, and usefully complement regulatory requirements, such as the 2002/49/CE Directive, in Europe, relating to the assessment and management of environmental noise. As an example, a very simple analysis of the sound levels collected in France shows, without any particular treatment, an overall decrease in sound levels during the periods of lockdown related to COVID (Figure 17).

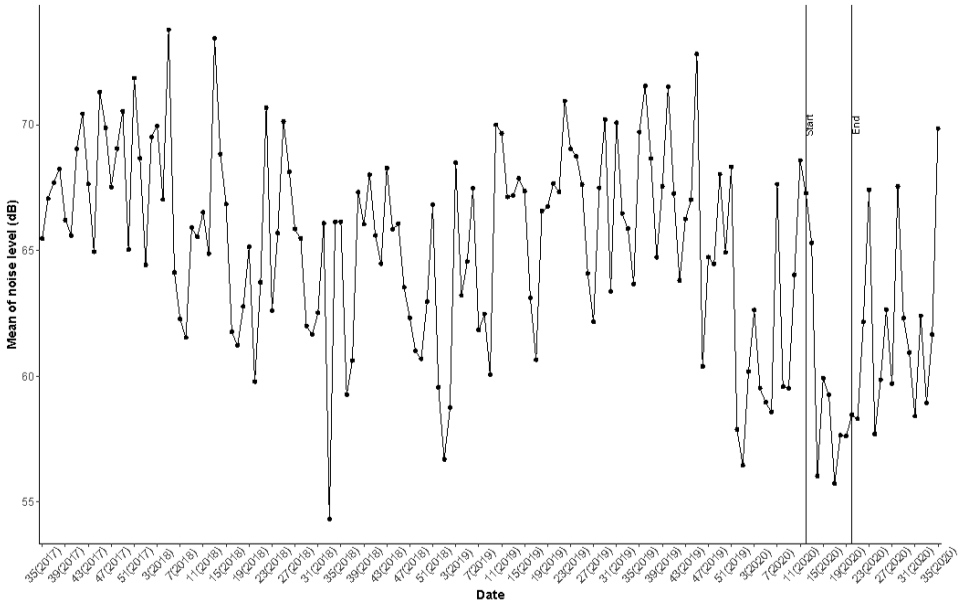


Figure 17. Distribution of mean `'noise_level'` in France by week accompanied with 2 verticals lines that represent the start and end of the first lockdown.

Beyond the exploitation of the database, one can also mention that the use of the NoiseCapture application with a dedicated use of the collected data (i.e., without using the NoiseCapture database, but using only the data export capabilities of the application) can be an interesting tool for scientific purposes [53–57].

Author Contributions: Conceptualization, J.P., E.B., G.P. and N.F.; methodology, J.P. and A.B.; software, N.F. and G.P.; formal analysis, J.P. and A.B.; investigation, J.P. and A.B.; writing—original draft preparation, J.P. and A.B.; writing—review and editing, J.P., A.B., E.B., G.P. and P.A.; visualization, A.B. and N.F.; supervision, J.P. and E.B.; project administration, J.P. and E.B.; funding acquisition, E.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was initially funded in the framework of the ENERGIC-OD Project (European Network for Redistributing Geospatial Information to user Communities—Open Data), under the ICT Policy Support Programme (ICT PSP) (CIP-ICT-PSP-2013-7) as part of the Competitiveness and Innovation Framework Programme by the European Community.

Data Availability Statement: The data presented in this study are openly available from Université Gustave Eiffel Dataverse Repository at https://doi.org/10.25578/J5DG3W.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

API	Application Programming Interface
CNRS	Centre national de la recherche scientifique (French National Centre for Scientific Research)
ERD	Entity relation diagrams
GPS	Global positioning system
ICT	Information and communication Technologies
Ifsttar	Institut Français des sciences et technologies des transports, de l'aménagement et des réseaux
OSM	OpenStreetMap
RNE	Repartition of the noise exposure
SDI	Spatial data infrastructure
SPL	Sound Power Level

References

1. Directive 2002/49/EC of the European Parliament and of the Council of 25 June 2002 Relating to the Assessment and Management of Environmental Noise—Declaration by the Commission in the Conciliation Committee on the Directive. 2002. Available online: http://data.europa.eu/eli/dir/2002/49/oj/eng (accessed on 26 April 2021).
2. Picaut, J.; Can, A.; Fortin, N.; Ardouin, J.; Lagrange, M. Low-Cost Sensors for Urban Noise Monitoring Networks—A Literature Review. *Sensors* **2020**, *20*, 2256. [CrossRef]
3. Camprodon, G.; González, O.; Barberán, V.; Pérez, M.; Smári, V.; de Heras, M.A.; Bizzotto, A. Smart Citizen Kit and Station: An open environmental monitoring system for citizen participation and scientific experimentation. *HardwareX* **2019**, *6*, e00070. [CrossRef]
4. Aiello, L.M.; Schifanella, R.; Quercia, D.; Aletta, F. Chatty maps: Constructing sound maps of urban areas from social media data. *R. Soc. Open Sci.* **2016**, *3*, 150690. [CrossRef] [PubMed]
5. Gasco, L.; Clavel, C.; Asensio, C.; de Arcas, G. Beyond sound level monitoring: Exploitation of social media to gather citizens subjective response to noise. *Sci. Total. Environ.* **2019**, *658*, 69–79. [CrossRef] [PubMed]
6. Mydlarz, C.; Drumm, I.; Cox, T. Application of novel techniques for the investigation of human relationships with soundscapes. In Proceedings of the INTER-NOISE and NOISE-CON Congress and Conference Proceedings, Osaka, Japan, 4–7 September 2011; Volume 2011, pp. 738–744.
7. Santini, S.; Ostermaier, B.; Adelmann, R. On the use of sensor nodes and mobile phones for the assessment of noise pollution levels in urban environments. In Proceedings of the 2009 Sixth International Conference on Networked Sensing Systems, Pittsburgh, PA, USA, 17–19 June 2009; pp. 1–8. [CrossRef]
8. Rana, R.K.; Chou, C.T.; Kanhere, S.S.; Bulusu, N.; Hu, W. Ear-phone: An End-to-end Participatory Urban Noise Mapping System. In Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, Stockholm, Sweden, 12–16 April 2010; pp. 105–116. [CrossRef]
9. Kanjo, E. NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping. *Mob. Netw. Appl.* **2010**, *15*, 562–574. [CrossRef]
10. Maisonneuve, N.; Stevens, M.; Ochab, B. Participatory Noise Pollution Monitoring Using Mobile Phones. *Info. Pol.* **2010**, *15*, 51–71. [CrossRef]
11. Duda, P. Processing and Unification of Environmental Noise Data from Road Traffic with Spatial Dimension Collected through Mobile Phones. *J. Geosci. Environ. Prot.* **2016**, *4*, 1. [CrossRef]
12. Guillaume, G.; Can, A.; Petit, G.; Fortin, N.; Palominos, S.; Gauvreau, B.; Bocher, E.; Picaut, J. Noise mapping based on participative measurements. *Noise Mapp.* **2016**, *3*, 140–156. [CrossRef]
13. Radicchi, A.; Henckel, D.; Memmel, M. Citizens as smart, active sensors for a quiet and just city. The case of the "open source soundscapes" approach to identify, assess and plan "everyday quiet areas" in cities. *Noise Mapp.* **2016**, *5*, 1–20. [CrossRef]

14. Zamora, W.; Vera, E.; Calafate, C.T.; Cano, J.C.; Manzoni, P. GRC-Sensing: An Architecture to Measure Acoustic Pollution Based on Crowdsensing. *Sensors* **2018**, *18*, 2596. [CrossRef]
15. Brambilla, G.; Pedrielli, F. Smartphone-Based Participatory Soundscape Mapping for a More Sustainable Acoustic Environment. *Sustainability* **2020**, *12*, 7899. [CrossRef]
16. Hachem, S.; Mallet, V.; Ventura, R.; Pathak, A.; Issarny, V.; Raverdy, P.; Bhatia, R. Monitoring Noise Pollution Using the Urban Civics Middleware. In Proceedings of the 2015 IEEE First International Conference on Big Data Computing Service and Applications, Redwood City, CA, USA, 30 March–2 April 2015; pp. 52–61. [CrossRef]
17. Zappatore, M.; Longo, A.; Bochicchio, M.A. Crowd-sensing our Smart Cities: A Platform for Noise Monitoring and Acoustic Urban Planning. *J. Commun. Softw. Syst.* **2017**, *13*, 53–67. [CrossRef]
18. Picaut, J.; Fortin, N.; Bocher, E.; Petit, G.; Aumond, P.; Guillaume, G. An open-science crowdsourcing approach for producing community noise maps using smartphones. *Build. Environ.* **2019**, *148*, 20–33. [CrossRef]
19. ENERGIC OD—Official Website. Available online: https://www.energic-od.eu/ (accessed on 18 March 2021).
20. Noise-Planet Website. Noise-Planet—Data. 2021. Available online: https://data.noise-planet.org/index.html (accessed on 13 January 2021).
21. Wikipedia. Mobile Operating System. 2021. Available online: https://en.wikipedia.org/w/index.php?title=Mobile_operating_system&oldid=998546997 (accessed on 13 January 2021).
22. Kardous, C.A.; Shaw, P.B. Evaluation of smartphone sound measurement applications. *J. Acoust. Soc. Am.* **2014**, *135*, EL186–EL192. [CrossRef] [PubMed]
23. NoiseCapture—Applications sur Google Play. Available online: https://play.google.com/store/apps/details?id=org.noise_planet.noisecapture (accessed on 18 January 2021).
24. Can, A.; Gauvreau, B. Describing and classifying urban sound environments with a relevant set of physical indicators. *J. Acoust. Soc. Am.* **2015**, *137*, 208–218. [CrossRef] [PubMed]
25. Bocher, E.; Petit, G.; Picaut, J.; Fortin, N.; Guillaume, G. Collaborative noise data collected from smartphones. *Data Brief* *14*, 498–503. [CrossRef]
26. Noise-Planet Website. NoiseCapture Interactive Community Map. 2021. Available online: https://noise-planet.org/map_noisecapture/index.html (accessed on 13 January 2021).
27. Releases Ifsttar/NoiseCapture. Available online: https://github.com/Ifsttar/NoiseCapture/releases (accessed on 18 January 2021).
28. Ifsttar/NoiseCapture: Measurement Export File, from Smartphone to Database. Available online: https://github.com/Ifsttar/NoiseCapture/blame/master/app/src/main/java/org/noise_planet/noisecapture/MeasurementExport.java#L136 (accessed on 18 January 2021).
29. PostGIS—Spatial and Geographic Objects for PostgreSQL. Available online: https://postgis.net/ (accessed on 14 January 2021).
30. Group, P.G.D. PostgreSQL. 2021. Available online: https://www.postgresql.org/ (accessed on 14 January 2021).
31. Picaut, J.; Fortin, N.; Bocher, E.; Petit, G. Université Gustave Eiffel, NoiseCapture Dataverse Repository, NoiseCapture Data Extraction from 29 August 2017 until 28 August 2020 (3 Years), V1. 2021. Available online: https://doi.org/10.25578/J5DG3W (accessed on 18 July 2021).
32. Open Data Commons Open Database License (ODbL) v1.0—Open Data Commons: Legal Tools for Open Data. Available online: https://opendatacommons.org/licenses/odbl/1-0/ (accessed on 18 March 2021).
33. Noise-Planet—Scientific tools for environmental noise assessment. Available online: https://noise-planet.org/ (accessed on 18 March 2021).
34. Mapping Parties—OpenStreetMap Wiki. Available online: https://wiki.openstreetmap.org/wiki/Mapping_parties (accessed on 15 January 2021).
35. Zhu, Y.; Li, J.; Liu, L.; Tham, C.K. iCal: Intervention-free Calibration for Measuring Noise with Smartphones. In Proceedings of the 2015 IEEE 21st International Conference on Parallel and Distributed Systems, Melbourne, Australia, 14–17 December 2015; pp. 85–91. [CrossRef]
36. Ventura, R.; Mallet, V.; Issarny, V.; Raverdy, P.G.; Rebhi, F. Evaluation and calibration of mobile phones for noise monitoring application. *J. Acoust. Soc. Am.* **2017**, *142*, 3084–3093. [CrossRef]
37. Murphy, E.; King, E.A. Testing the accuracy of smartphones and sound level meter applications for measuring environmental noise. *Appl. Acoust.* **2016**, *106*, 16–22. [CrossRef]
38. Android 11 Compatibility Definition. Available online: https://source.android.com/compatibility/android-cdd (accessed on 10 February 2021).
39. Android vs. iPhone Users: The Difference between Behavior. 2017. Available online: https://buildfire.com/ios-android-users/ (accessed on 24 February 2021).
40. Global Stats. Mobil Vendor Market Share Dec2019-Dec2020. 2021. Available online: https://gs.statcounter.com/vendor-market-share/mobile/ (accessed on 27 January 2021).
41. The PostGIS Development Group. 'ST Extent' Function. Available online: https://postgis.net/docs/ST_Extent.html (accessed on 9 February 2021).
42. GADM Data. Available online: https://gadm.org/data.html (accessed on 9 February 2021).
43. List of Countries and Dependencies by Population. 2021. Available online: https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population (accessed on 18 January 2021).

44. Android Developers. 'getAccuracy' Function. Available online: https://developer.android.com/reference/android/location/Location#getAccuracy() (accessed on 17 February 2021).
45. Aumond, P.; Can, A.; Mallet, V.; De Coensel, B.; Ribeiro, C.; Botteldooren, D.; Lavandier, C. Kriging-based spatial interpolation from measurements for sound level mapping in urban areas. *J. Acoust. Soc. Am.* **2018**, *143*, 2847–2857. [CrossRef]
46. Android Developers. 'getSpeed' Function. Available online: https://developer.android.com/reference/android/location/Location#getSpeed() (accessed on 17 February 2021).
47. Bedogni, L.; DiźFelice, M.; Bononi, L. Context-aware Android applications through transportation mode detection techniques. *Wirel. Commun. Mob. Comput.* **2016**, *16*, 2523–2541. [CrossRef]
48. Aumond, P.; Lavandier, C.; Ribeiro, C.; Boix, E.G.; Kambona, K.; D'Hondt, E.; Delaitre, P. A study of the accuracy of mobile technology for measuring urban noise pollution in large scale participatory sensing campaigns. *Appl. Acoust.* **2017**, *117*, 219–226. [CrossRef]
49. Zamora, W.; Calafate, C.T.; Cano, J.C.; Manzoni, P. Accurate Ambient Noise Assessment Using Smartphones. *Sensors* **2017**, *17*, 917. [CrossRef]
50. Garg, S.; Lim, K.M.; Lee, H.P. An averaging method for accurately calibrating smartphone microphones for environmental noise measurement. *Appl. Acoust.* **2019**, *143*, 222–228. [CrossRef]
51. Kardous, C.A.; Shaw, P.B. Evaluation of smartphone sound measurement applications (apps) using external microphones—A follow-up study. *J. Acoust. Soc. Am.* **2016**, *140*, EL327–EL333. [CrossRef] [PubMed]
52. Aumond, P.; Can, A.; Rey Gozalo, G.; Fortin, N.; Suárez, E. Method for in situ acoustic calibration of smartphone-based sound measurement applications. *Appl. Acoust.* **2020**, *166*, 107337. [CrossRef]
53. Dubey, R.; Bharadwaj, S.; Zafar, M.; Sharma, V.; Biswas, S. Collaborative noise mapping using smartphone. *ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *43*, 253–260. [CrossRef]
54. Graziuso, G.; Grimaldi, M.; Mancini, S.; Quartieri, J.; Guarnaccia, C. Crowdsourcing Data for the Elaboration of Noise Maps: A Methodological Proposal. *J. Phys. Conf. Ser.* **2020**, *1603*, 012030. [CrossRef]
55. Mohammed, H.M.E.H.S.; Badawy, S.S.I.; Hussien, A.I.H.; Gorgy, A.A.F. Assessment of noise pollution and its effect on patients undergoing surgeries under regional anesthesia, is it time to incorporate noise monitoring to anesthesia monitors: an observational cohort study. *Ain Shams J. Anesthesiol.* **2020**, *12*, 20. [CrossRef]
56. Sakagami, K. How did "state of emergency" declaration in Japan due to the COVID-19 pandemic affect the acoustic environment in a rather quiet residential area? *UCL Open Environ. Prepr.* **2020**. [CrossRef]
57. Nourmohammadi, Z.; Lilasathapornkit, T.; Ashfaq, M.; Gu, Z.; Saberi, M. Mapping Urban Environmental Performance with Emerging Data Sources: A Case of Urban Greenery and Traffic Noise in Sydney, Australia. *Sustainability* **2021**, *13*, 605. [CrossRef]
58. Chon, Y.; Lane, N.D.; Li, F.; Cha, H.; Zhao, F. Automatically characterizing places with opportunistic crowdsensing using smartphones. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, Pittsburgh, PA, USA, 5–8 September 2012; pp. 481–490. [CrossRef]
59. Hwang, K.; Lee, S. Environmental audio scene and activity recognition through mobile-based crowdsourcing. *IEEE Trans. Consum. Electron.* **2012**, *58*, 700–705. [CrossRef]
60. Kendrick, P.; Cox, T.; Li, F.; Fazenda, B.; Jackson, I. Wind-induced microphone noise detection—Automatically monitoring the audio quality of field recordings. In Proceedings of the 2013 IEEE International Conference on Multimedia and Expo (ICME), San Jose, CA, USA, 15–19 July 2013; pp. 1–6. [CrossRef]
61. Shoaib, M.; Bosch, S.; Incel, O.D.; Scholten, H.; Havinga, P.J.M. A Survey of Online Activity Recognition Using Mobile Phones. *Sensors* **2015**, *15*, 2059–2085. [CrossRef] [PubMed]
62. Wang, W.; Chang, Q.; Li, Q.; Shi, Z.; Chen, W. Indoor-Outdoor Detection Using a Smart Phone Sensor. *Sensors* **2016**, *16*, 1563. [CrossRef] [PubMed]
63. DetectedActivity | Google APIs for Android. Available online: https://developers.google.com/android/reference/com/google/android/gms/location/DetectedActivity (accessed on 20 January 2021).
64. Gontier, F.; Lavandier, C.; Aumond, P.; Lagrange, M.; Petiot, J.F. Estimation of the Perceived Time of Presence of Sources in Urban Acoustic Environments Using Deep Learning Techniques. *Acta Acust. United Acust.* **2019**, *105*. [CrossRef]
65. Silva, M.D.; Viterbo, J.; Bernardini, F.; Maciel, C. Identifying Privacy Functional Requirements for Crowdsourcing Applications in Smart Cities. In Proceedings of the 2018 IEEE International Conference on Intelligence and Security Informatics (ISI), Miami, FL, USA, 9–11 November 2018; pp. 106–111. [CrossRef]
66. EUR-Lex—32016R0679-EN-EUR-Lex: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation). Available online: https://eur-lex.europa.eu/eli/reg/2016/679/oj (accessed on 18 February 2021).
67. D'Hondt, E.; Stevens, M.; Jacobs, A. Participatory noise mapping works! An evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring. *Pervasive Mob. Comput.* **2013**, *9*, 681–694. [CrossRef]
68. Martí, I.G.; Rodríguez, L.E.; Benedito, M.; Trilles, S.; Beltrán, A.; Díaz, L.; Huerta, J. Mobile Application for Noise Pollution Monitoring through Gamification Techniques. *Entertainment Computing—ICEC 2012*; Herrlich, M., Malaka, R., Masuch, M., Eds.; Springer: Berlin, Heidelberg, 2012; pp. 562–571. [CrossRef]

69. Lefevre, B.; Agarwal, R.; Issarny, V.; Mallet, V. Mobile crowd-sensing as a resource for contextualized urban public policies: A study using three use cases on noise and soundscape monitoring. *Cities Health* **2019**, 1–19. [CrossRef]
70. Lee, H.P.; Garg, S.; Lim, K.M. Crowdsourcing of environmental noise map using calibrated smartphones. *Appl. Acoust.* **2020**, *160*, 107130. [CrossRef]

Chapter 2

Using a clustering method to detect spatial events in a smartphone-based crowd-sourced database for environmental noise assessment

sensors

MDPI

Article

Using a Clustering Method to Detect Spatial Events in a Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment

Ayoub Boumchich [1], Judicaël Picaut [1,*] and Erwan Bocher [2]

[1] UMRAE, CEREMA, Univ Gustave Eiffel, F-44344 Bouguenais, France
[2] Lab-STICC CNRS UMR 6285, IUT de Vannes, F-56017 Vannes, France
* Correspondence: judicael.picaut@univ-eiffel.fr

Abstract: Noise has become a very notable source of pollution with major impacts on health, especially in urban areas. To reduce these impacts, proper evaluation of noise is very important, for example by using noise mapping tools. The Noise-Planet project seeks to develop such tools in an open science platform, with a key open-source smartphone tool "NoiseCapture" that allows users to measure and share the noise environment as an alternative to classical methods, such as simulation tools and noise observatories, which have limitations. As an alternative solution, smartphones can be used to create a low-cost network of sensors to collect the necessary data to generate a noise map. Nevertheless, this data may suffer from problems, such as a lack of calibration or a bad location, which lowers its quality. Therefore, quality control is very crucial to enhance the data analysis and the relevance of the noise maps. Most quality control methods require a reference database to train the models. In the context of NC, this reference data can be produced during specifically organized events (NC party), during which contributors are specifically trained to collect measurements. Nevertheless, these data are not sufficient in number to create a big enough reference database, and it is still necessary to complete them. Other communities around the world use NC, and one may want to integrate the data they collected into the learning database. In order to achieve this, one must detect these data within the mass of available data. As these events are generally characterized by a higher density of measurements in space and time, in this paper we propose to apply a classical clustering method, called DBSCAN, to identify them in the NC database. We first tested this method on the existing NC party, then applied it on a global scale. Depending on the DBSCAN parameters, many clusters are thus detected, with different typologies.

Keywords: environmental noise; noise mapping; smartphone application; spatial clustering; DBSCAN

Citation: Boumchich, A.; Picaut, J.; Bocher, E. Using a Clustering Method to Detect Spatial Events in a Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *Sensors* **2022**, *22*, 8832. https://doi.org/10.3390/s22228832

Academic Editor: Thomas P Karnowski

Received: 13 July 2022
Accepted: 26 October 2022
Published: 15 November 2022

1. Introduction

1.1. Noise Mapping Using Data Collected with Smartphones

Noise has become a very notable source of pollution with major impacts on health, especially in urban areas [1]. The public authorities are trying to solve this essential societal and health issue by setting up new regulations. In Europe, for example, the directive 2002/49/EC seeks to evaluate noise annoyance, to propose actions to reduce this annoyance, and to communicate with citizens about their noise exposure. In this regulatory context, the key tool for decision makers is the use of strategic noise maps.

Instead of using noise prediction software, with its inherent limits, an alternative method may be to use more affordable sensor networks, allowing us to densify the observation points [2] and, in particular, to consider the participation of citizens as data collectors in a crowd-sourcing approach using smartphones as sensors (i.e., a measuring instruments). This idea of using smartphones as acoustic sensors and citizens as contributors emerged at the end of the 2000s with the increasing capabilities of smartphones to perform environmental acoustic measurements [3]. It was followed by several works that have given

rise to specific noise and soundscape crowd-sourcing-type applications and platforms (e.g., Ear-Phone, NoiseSPY, NoiseTube applications [4–6]) and, particularly, the NoiseCapture (NC) approach [7,8], a part of the Noise-Planet project [9].

The NC approach consists of measuring noise and additional information along a path and then sharing data with the community (Figure 1). The noise data acquired by volunteers (i.e., the "user" or the "contributor") from all over the world are then stored in a community database. This database contains the measurement path (a "track" in the NC vocabulary, which is made of "measuring points"), standardized noise indicators, a description of the user perception of noise sources (using "Tags") and of the soundscape quality (using a pleasantness scale), and other useful information, such as the date and time of the measurements, the GPS localization and accuracy, and the speed of the user during measurement...

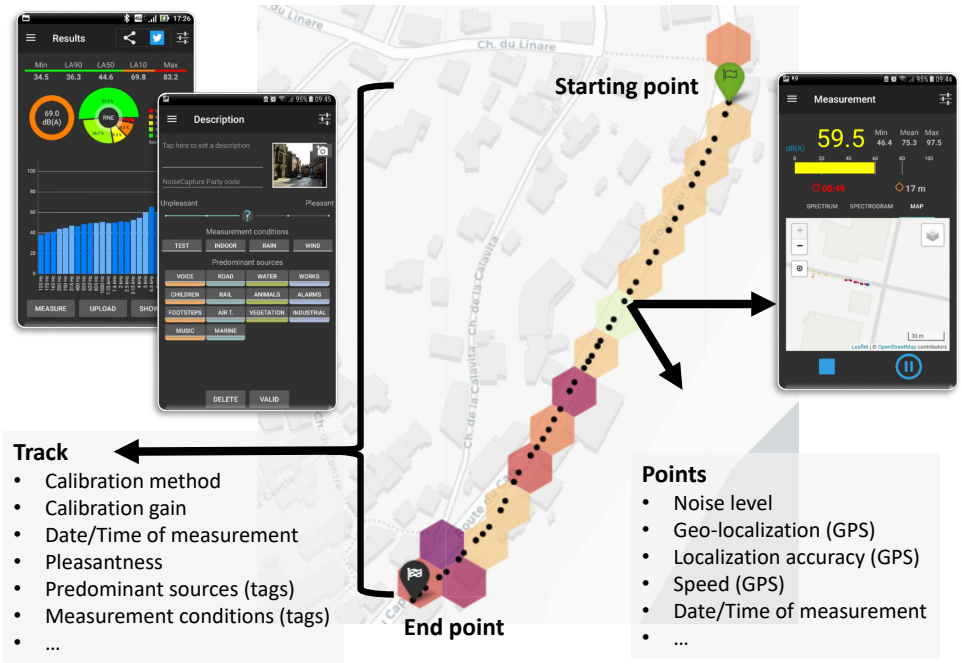


Figure 1. Representation of a NoiseCapture (NC) measurement. After manually activating the measurement from a starting point, the user can move along a path of their choice or stay at the same location. An acoustic measurement takes place every second, which allows us to calculate noise indicators, such as 1 s equivalent sound level and spectrum. Other information, such as the date and time of the measurements, the GPS location of the measurement point, its accuracy, and its speed are recorded at the same time. At the end of a track, the user stops the measurement and is invited to provide additional information about the presence of specific sound sources (tags) during the measurement, the measurement conditions, his own perception of the quality of the sound environment (i.e., the pleasantness), and other additional information. If the user has given his consent, the anonymous data are transferred to a remote server, controlled and then integrated into an open database, which is available for free. In addition, the raw data from the entire community are aggregated into a hexagonal based noise map and displayed on a public web page [10], as shown in this figure. Graphical contents of this figure are part of the Noise-Planet project website [11], licensed under CC BY-NC-SA 4.0.

This data can be relevant for later evaluation of the ambient noise through specific processing, for example, to generate noise maps that could be useful for communities to implement action plans in order to reduce the noise exposure of citizens or to protect quiet environments. In this way, the use of NC maps would be consistent with the European directive 2002/49/EC except that they simultaneously integrate the multiplicity of noise

sources encountered in the environment. In comparison with conventional noise maps, for example, focusing on traffic noise only, those produced with NC data could be closer to reality.

Since the launch of the application on 29 August 2017, a large amount of data has been made available, offering the possibility to analyze the data over a large spatial area and a long period of time. This database is distributed under the Open Data License [9] and updated every day at 3:30 am (local time in Paris, France). At this date, the database represents the equivalent of more than 1103 days of 1 s measurements (more than 382,000 tracks and 93 million measurement points) spread over more than 200 countries and collected by over 92,000 different contributors.

1.2. Data Quality Control and the Need for a Reference Database

This amount of data, however, needs to be put into perspective, firstly with regard to the surface of the planet (150 million km^2 for the terrestrial surface) and the time elapsed from the launch of the application (1737 days by 31 May 2022); finally, the density of points in space and over time remains globally very low (less than 0.4 measurement points per 1000 km^2 by day) except in urban areas where the density of measurement may be more important. In addition, the quality of the collected data may sometimes be questioned. A recent analysis [8] showed that data may suffer from several problems due to the technical performance of the smartphone, respect for a relevant measurement protocol, the lack of smartphone calibration, the misuse of the application, and insufficient GPS accuracy…

The evaluation of the quality of the collected data is therefore essential to characterize the relevance of the noise maps and the analysis of the sound environment that will be produced. For example, it is important to identify wrong or incomplete data, as well as anomalies in the NC database…which is more generally called data Quality Control (QC). For this purpose, among all the possible approaches, those based on machine learning techniques have been widely used in many fields of interest [12], in particular, when considering crowd-sourced data [13–16]. Three main methodologies can thus be used in machine learning: the supervised, the semi-supervised, and the unsupervised ones, depending on whether they use a large amount of labelled data, a small amount, or none, respectively.

In the NC context, the supervised and semi-supervised machine learning methods are well-suited because of the existence of labelled data characterized by a maximum level of confidence. Indeed, in the NC approach, such data are provided during NC parties, i.e., an event that is organized by acousticians or other experts in a limited space and in a relatively short time to simultaneously collect a large number of contributions. In this case, the contributors are coached, the measurement protocols are correctly applied, and the smartphones are most of time calibrated; thus, the data collected during such an event are generally of better quality and may be considered as reference data. However, the performance of (semi-)supervised methods requires a sufficient amount of data, which is undoubtedly not the case on the basis of data produced only by the NC parties.

If most of the time NC parties are organized by close collaborators of the NC project, the recent scientific literature shows that similar events are also organized for research purposes by people who are not directly related to the NC project. More globally, such events can also be organized by communities, citizens associations, or schools during awareness-raising events on noise issues. The data collected during these events can also be considered as labelled data and should therefore be identified in the NC database.

1.3. Objective of the Paper

The objective of the present paper is therefore to propose a method to identify in the NC database events similar to the NC party, i.e., a set of data collected by multiple users, produced over limited spatial extent and temporal period, and most of the time with a higher spatial and temporal density of measurement points. More explicitly, regrouping data with similar spatial and temporal characteristics is known as spatial clustering and temporal clustering [17]. Note that in the following, the paper focuses only on spatial

clustering, with the temporal parameters being considered as a filter for the data (this will be presented later in the document).

Many clustering methods have been proposed in the literature. Among them, the DBSCAN approach seems well-suited to our problem and is considered in the present paper. Thus, the originality of the proposed work lies mainly in the use and validation of this method for the needs of our problem and not in the development of a specific clustering method.

Some generalities on spatial clustering methods are presented in Section 2. Then, in Section 3, the DBSCAN method is detailed and tested on known NC parties in order to evaluate its performance and to identify the most appropriate processing parameters. Finally, the DBSCAN method is applied in Section 4, firstly to a few countries in order to identify the typology of the obtained clusters and secondly to the entire NC database. Lastly, Section 5 concludes this study, showing the interest of clustering methods to identify data that could be integrated into a reference database.

2. Spatial Clustering Related Work

Clustering is a useful method in data science [18,19], allowing us to identify similar groups of data in a data set (i.e., objects in each group are comparatively more similar to objects in other groups), which are called *cluster*. It should be noted that the term "close" is sometimes used instead of "similar". For example, let us consider two houses ("A" and "B") and one studio ("C"), where houses "A" and studio "C" are in the same neighborhood and "B" in a different city; if the clusters were formed to regroup housing with similar characteristics for sale, then houses "A" and "B" can be clustered together, but if we are talking about spatial clustering, then house "A" and studio "C" could be regrouped together because they are close to each other. Therefore, it is important to pay attention to the characteristics that are considered when looking for similarity or closeness.

It must be noted that classification is another technique that has certain similarities with clustering and which could have been envisaged for the present study. Classification refers to the act of categorizing or predicting the class of any given data. Classification is supervised and demands a training data set with class labels, while clustering is non-supervised, aiming to find underlying unknown groups or clusters). In the present study, the lack of criteria, such as spatial/temporal limit, number of users, and number of tracks/-points to define classes, made the clustering approach more relevant than classification.

Clustering has a large number of applications stretching out across various domains, such as recommendation engines [20], market segmentation [21], social network analysis [22], search result grouping [23], and anomaly detection [24]. Clustering analysis can also be applied to the environmental measurement data (e.g., temperature [25,26] and humidity [25,27]). Clustering is also widely used in geospatial analysis [28].

In the environmental acoustic field, clustering has already been applied to analyze the results of the "Think About Sound" smartphone application, using the experience sampling methods (ESM) [29]. Clustering was also used to group similar urban soundscapes using the fuzzy ants rule and K-means method [30,31] or to monitor road traffic noise by using temporal clustering methods [32,33]. Nevertheless, the last clustering methods were applied on the basis of the information contained in the collected data, and, to our knowledge, no methodology has been applied to spatially clustering the data as we plan to do with the NC data. The data collected by NC can be analysed by path (i.e., track) or by point. The applicable spatial clustering methodologies can therefore cover *Points Spatial Clustering* or *Line/Trajectory Spatial Clustering*. In the following, we will focus on *Points Spatial Clustering*.

Points Spatial Clustering methods that may be applicable to the present study can follow several approaches that are inspired by generic clustering methods [34,35]:

- *Partition Clustering* allows grouping data in K non-overlapping sub-groups (i.e., K clusters), one datum being in only one subgroup. One can consider several methodologies, for example, the K-means, K-medians, or K-medoids methods, depending on

the choice of the cluster center, at the average point, the median point, or the point in the data set closest to the median point, respectively.

- *Hierarchical Clustering*, using the Agglomerative or the Divisive Hierarchical Clustering methods, tries to build a hierarchy of clusters using a bottom-up approach (each datum starts in its own cluster, then the two closest clusters according to a chosen distance are merged until all clusters are merged, creating a tree that one has to cut according to the relevant number of clusters) or a top-down approach (at the opposite of the bottom-up approach, all data are initially in the same cluster and, then, the cluster is split according to the hierarchy level), respectively.
- *Fuzzy Clustering* is another form of data clustering in which each datum can be included into several clusters. As a possible approach, the Fuzzy C-means method, which is the most widely used, is quite similar to the K-means clustering method.
- *Density-Based Clustering* methods propose to group data by considering the density of data. Then, each cluster is built by considering regions with a high density of data.
- Lastly, the *Model-Based Clustering* method groups data by considering that they are generated by probability distributions and that each cluster represents one given distribution.

In the framework of the NC data, it is clear that the fuzzy clustering method cannot be used in the present study since each measurement point can be associated with only one event. Considering the model-based clustering methods, since the data are not associated, a priori, with any distribution model, the method seems inapplicable. The partition clustering method, which imposes fixing the number of clusters to identify in advance, is obviously also inapplicable since it is impossible to know the number of NC events that will be found. Lastly, the use of a hierarchical clustering method is very empirical since it requires the retention of a number of clusters that should be coherent with the data, but with, a priori, the possibility of omitting clusters, in particular if the amount of data is very large. Moreover, the processing time can become considerable because of a quadratic increase of the complexity in $o(n^2 \log n)$ (n being the number of data), compared, for example, with an analysis using the K-means method that is characterized by a linear increase in $o(n)$. Finally, the density-based clustering approach seems an interesting way, since one can consider that an NC event generates an increase in the density of measuring points locally, over a short period of time. Among the possible approaches, the density-based spatial clustering of applications with noise approach (DBSCAN) method is the most commonly used [36]. It must be noted that the term "noise" that is employed in the name of the method does not refer to the environmental "noise", but to "noise" in the data, meaning that some points may not be part of a given cluster.

Considering automatic learning, such as clustering, most techniques will either fall into (1) supervised, (2) semi-supervised, or (3) unsupervised learning. Supervised learning provides the model with both the input and the output of the data (also called labelled data), while unsupervised learning provides only the input of the data. On the other hand, semi-supervised learning provides the model with a small amount of data that contains both the input and the output and a large amount of data that contains only the input. In the NC context, supervised learning can be hard to use due to the lack of labeled data (i.e., NC party data). So either unsupervised learning or semi-supervised could be performed, but due to the lack of similarities in NC party events, it is feared that a direct semi-supervised approach will not work. Therefore, an indirect semi-supervised approach has been considered in the present work. This approach consists firstly of using the labelled data (i.e., NC party data) to perform supervised learning in order to find the optimal parameters of the DBSCAN method, and secondly, of performing an unsupervised learning task, using DBSCAN with these parameters to detect events that are similar to NC parties.

3. Spatial Clustering of the NoiseCapture Data with the DBSCAN Method

3.1. Implementation of the DBSCAN Method

The principle of the method is trivial and based on very pragmatic considerations (i.e., a cluster represents a set of points very close to each other), without any theoretical basis. In the present study, the approach is reproduced as it was proposed by its authors [36]. It consists of searching a minimal set of points (`MinPts`) around a given point (in a circle of radius `Eps`) to start forming a cluster. This cluster evolves progressively by searching for new points to integrate from each of its points that are already members of the cluster. The method is thus defined by these two parameters `Eps` and `MinPts` only (Figure 2).

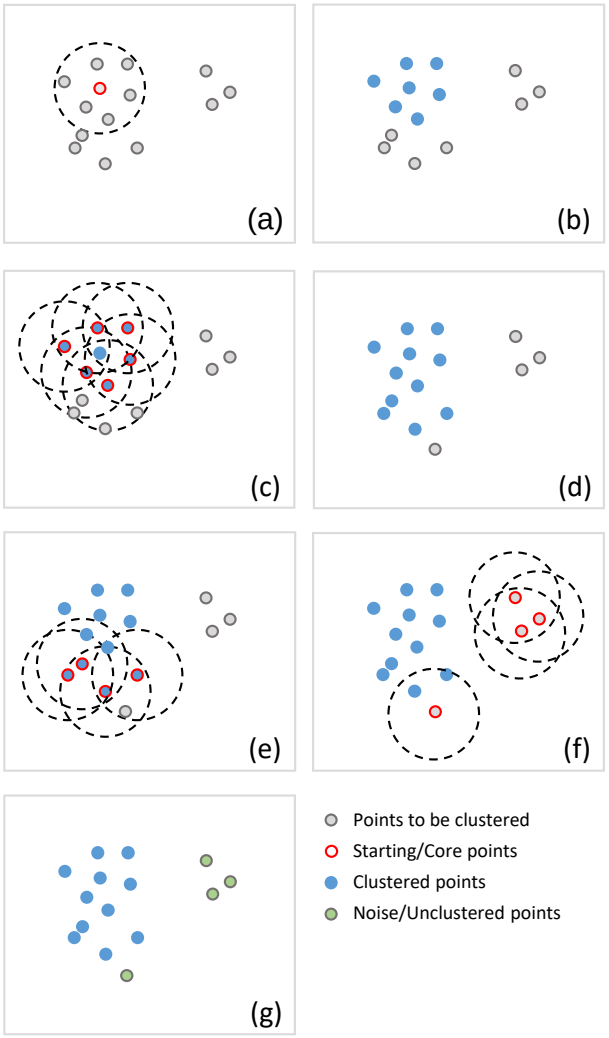


Figure 2. Graphical representation of the DBSCAN approach, with `MinPts`=4 as an example: (**a**) a starting point is first selected in a set of points to be clustered; if the number of points in a circle with a radius of `Eps` centered on the starting point is greater or equal to `MinPts`, all of the points are included in the same cluster: in this example, there are 6 points in the circle, thus the condition is validated; (**b**) a first cluster is created with the staring point and the 6 new members; (**c**) each new member of the cluster becomes a starting point; the condition presented in (**b**) is applied again: in this example, only one starting point respects the conditions, then (**d**) 4 points are included in the cluster; (**e**) the condition is applied again on the new members; in this example, the condition of a minimum of points in the circle is not verified: the process stops and a cluster is then complete; (**f**) a new starting point is selected outside the existing clusters and the process is repeated until all points have been proceeded; (**g**) in this example, 11 points have been grouped together in one cluster and 4 points are not clustered.

The method first randomly selects a point in the database. Then, it selects all the points within a circle of radius *Eps*. If the number of those points is greater than or equal to *MinPts*, then they are integrated into this cluster. At the next step, each integrated point becomes the center of a circle of radius *Eps* of a new iteration. When the number of points within one of these circles is less than *MinPts*, then the procedure stops for the corresponding center. When the search becomes unsuccessful for all the current points in the cluster, the cluster stops growing. The procedure then starts again by randomly selecting a new point in the database outside of any cluster already formed as the new center of a possible cluster. When all the points in the database have been processed, the procedure stops and eventually leads to the creation of a finite number of clusters, or none. As mentioned before, all the points that are not part of a cluster are considered "noise".

In agreement with the NC database model that is used in the present study (cf. Section 3.2), the DBSCAN method has been implemented using the `ST_ClusterDBSCAN()` function available in PostGIS, a spatial database extender for the PostgreSQL object–relational database [37]. This function is a 2D implementation of the DBSCAN algorithm (the spatial clustering of the NC database is carried out on the 2D coordinates without the elevation). This function takes three inputs, `geom` as a geometry data type, `Eps` as a float data type, and `MinPts` as an integer data type, and returns one identifier per found cluster, as an output. The results are saved in a table called `NoiseCapture_cluster`, which contains the measurement point identification `pk_point` of coordinates `the_geom` and cluster identification `c_id`. Another variable, `pk_party`, is also added to verify if the corresponding point is already part of a known NC party event and, if so, the corresponding NC party identification. Finally, the data and the clusters are plotted in GIS software. Note that all the tools used in this study are open source.

3.2. NoiseCapture Database

The NC measurement locations are stored in WGS 84 (World Geodetic System 1984) format [38], which is a horizontal component of a 3D system used, for example, by the GPS satellite navigation system. In other words, the type of the coordinates is geographic. This encoding uses "degree" as a measure distance, while the `ST_ClusterDBSCAN()` function only takes coordinates as geometry data type. Therefore, a transformation to the metric projection EPSG:3857 (Pseudo-Mercator) [39] is required before applying the algorithm. This projection can only be used for data between 85.06° S and 85.06° N, which is in agreement with the NC database (NC data are bounded between 74.49° S and 78.73° N). The transformation was performed using the PostGIS function `ST_Transform` [40].

The DBSCAN algorithm was used on a 3-year extraction of the NC database, from 29 August 2017 to 28 August 2020, starting from the official release version "28" of the NC smartphone application [41]. Any measurement point in the NC database without geo-localization (may be a full track or only a part of a track) was removed before processing by the DBSCAN method. Using this NC database extraction, all the data can be organized into a relational database, which can be manipulated using GIS tools.

In order to find the optimal parameters for DBSCAN, we use official NC party events as reference data. The objective of the first part of the study is to find the parameters of the method that allow us to find a maximum number of NC parties in the form of clusters, or at least, the best detection rate. During the corresponding 3-year period, 27 NC parties were planned [8] (Table 1), but one of them (i.e., NC party number 31 with code name `CICAM`) was not performed. The training data set has 26 NC parties, spread over 3 years. These NC events took place in France (18 NC parties, one including an event also shared with Germany), in Spain (5 NC parties), in Italy (2 NC parties), and in The Netherlands (1 NC party). During these events, 3142 tracks were collected, for a total of 391,575 measurement points. Around 377,206 (96.33%) points were geo-located, and 320,891 (81.95%) had accuracy less than or equal to 15 m; 3012 (95.86%) tracks were completely geo-located (i.e., all the points of a track are geo-located), and 2641 (84.05%) tracks had accuracy less or equal to 15 m for all corresponding points.

Table 1. NC parties description.

NoiceCapture Party				All Data		Geo-Localized Data		Not Geo-Localized Data	
ID	**NC party Code**	**Contributors**	**Date/Duration**	**Points**	**Tracks**	**Points**	**Tracks**	**Points**	**Tracks**
1	SNDIGITALWEEK	7	20 September 2017	11,523	133	11,192	118	331	15
2	ANQES	4	16–27 October 2017	4479	29	4458	28	21	1
3	FDS2017	2	15 October 2017	1239	6	1209	5	30	1
5	IMS2018	13	28 March 2018	18,793	0	18,758	67	35	0
6	UDC	8	17 April–27 June 2018	6879	56	4509	44	2370	12
9	TEST44	1	2 May 2018	91	3	91	3	0	0
10	UNISA	13	17, 24 May 2018 and 6 June 2018	15,912	149	15,479	141	433	8
11	PNRGM	2	9 June 2018	6089	13	5957	12	132	2
12	AMSOUNDS	2	20, 21 June 2018	693	18	660	17	33	1
13	PNRGM	14	18, 20 July 2018	21,470	100	19,812	92	1658	8
14	FDSSTRAS	5	12, 13 October 2018	2967	31	2909	25	58	6
15	AGGLOBASTIA	19	4 October–26 November 2018	59,838	507	58,771	506	1067	1
17	FDSNTS	7	12–14 October 2018	5916	66	5840	61	76	5
18	H2020	11	6–9 December 2018	22,060	89	19,869	88	2191	1
19	UDC	20	25 February–5 April 2019	5866	138	4946	108	920	30
20	MSA	9	10 January 2019	1885	9	1883	9	2	0
21	GEO2019	43	12–14 March 2019	63,521	420	62,199	409	1322	11
22	IMS2019	23	28, 29 March 2019	17,309	192	14,161	189	148	3
23	FPSLYO	11	6, 17, 18 May 2019	10,285	34	9548	31	737	3
24	SSSOROLL2019	68	16 April–19 May 2019	36,272	372	35,253	361	979	11
26	UNISA	20	24 May 2019	23,220	332	22,937	328	283	4
27	FDSSTRAS	3	12, 13 October 2019	1771	7	1730	6	41	1
28	H2020	9	4–8 December 2019	32,948	39	31,659	35	1289	4
29	UDC	9	3, 5, 6 March 2020	2099	73	2036	71	63	2
30	MSA	10	23 January 2020	3665	10	3659	9	9	1
31	CICAM	–	–	–	–	–	–	–	–
32	UDC_COVID	33	5–20 May 2020	14,785	249	14,691	249	94	0

NC party events may be defined by the organisers with two parameters: the `filter_area` and `filter_time`, which stand for the spatial and temporal limits that were decided for each NC party. Four NC parties (15.38%) did not have a spatial limit, while only two (7.69%) did not have a temporal limit. Statistics showed that ten (38.64%) of NC parties had points outside their limit area and that three (11.54%) NC parties had measurements outside the time limit; in this last case, measurements were all dated in the past (in 1994, 1999, and 2000), expressing a problem with time synchronization of some smartphones. In terms of measuring points, it means that 87.28% points (85.13% tracks) were collected within the area limit and 91.06% points (93.03% tracks) within the time limit. When both the area and time filters were used, 86.28% of the points (84.3% of the tracks) of NC parties were collected within the time and space constraints. The duration is specific to each NC party and is between 2 and 10 days, with an average of around 5 days (less than a week).

The number of events (26) and their amount of tracks/points represent only 1.2% of the NC database and 0.6% of the points. It may not be enough to assess the optimal parameters for DBSCAN.

3.3. Filtering Variables

In addition to the two parameters `Eps` and `MinPts`, one can also consider supplementary parameters in the DBSCAN methodology in order to reduce the amount of data to consider by pre-filtering the NC database. The corresponding filtering variables may have an impact on the computational duration of the process or can contribute to increasing the quality of the clustering results:

Time window The temporal dimension is not considered in the classical DBSCAN method, which implies that in the context of NC all data are considered simultaneously, just spatially, without any consideration of date. It can then be difficult to identify NC events of relatively short duration if the corresponding measurement points are ”drowned“ in the mass of data that can be collected progressively (i.e., out of a particular event), to the same spatial extent but over a long time. As mentioned above, the past NC parties have durations of a few days. Thus, it seems interesting to test the DBSCAN methodology, filtering the NC database to focus only on data collected over a ”day“, a ”week“ or a ”month“.

GPS accuracy The accuracy of the DBSCAN method is necessarily based on the accuracy of the location of the measurement points; if the points are poorly located, then their membership in a cluster may be questioned. In the NC application, the localization of the measurement is based on the GPS system of the smartphone. In some cases, the measurement points may not be located at all; in this case, as mentioned before, the corresponding measurement points were removed from the database. For the remaining points, the associated location uncertainty can reach several tens of meters. The variable related to the accuracy of localization can also be an important element in the quality of the clusters obtained by the DBSCAN method. The method will therefore also be tested by filtering the data on the GPS accuracy values.

Zone of study The search for clusters depends on the number of points in the database and thus, in particular, on the size of the study area. The larger the study area is, the longer the processing time will be. Reducing the study area to territories in which NC events are potentially expected will reduce the computational time.

3.4. Validation of DBSCAN

3.4.1. Methodology

As mentioned before, the objective of the preliminary study is to find the parameters of the DBSCAN method that allow for the best detection rate of NC parties in the form of clusters. Thus, a series of experiments were performed on the NC data by varying the values of the two DBSCAN parameters `Eps` and `MinPts` and for different filtering conditions:

- `Eps` was started from 50 m as the initial distance and then gradually increased (by 50 m between 50 m and 500 m, then by 100 m from 500 m to 2000 m and finally by 500 m from 2000 m forward) until a maximum of NC party events were detected as clusters.
- `MinPts` was started from 20 points as the initial value and then gradually increased (firstly 50, then 100, and finally by 100 until the maximum number of points for each respective NC party event) until a maximum of NC party events were detected as clusters.
- Time window: due to the typology of current NC parties, the clusters analysis was performed by filtering the NC database by day, by week, and by month, on a total duration that includes all the NC points (i.e., if an NC party took place over a period between 2 months, the 2 months concerned were fully considered).
- GPS accuracy: the DBSCAN process was performed with two settings for the GPS accuracy: "Off", meaning that the GPS accuracy is not considered; "On", meaning that measurements with a GPS accuracy strictly greater than 15 m are removed from the data.
- Zone of study: in order to reduce the computational time, the zone of study was reduced to the spatial areas that contained the current NC parties. However, the area must be large enough to avoid edge effects, especially if the points of a cluster are too close to the spatial boundaries.

The quality of the clustering analysis, i.e., the success in the detection of NC parties, was evaluated, on the one hand, qualitatively on the basis of maps by comparison between the obtained clusters and the points of the NC parties, and on the other hand, quantitatively on the basis of the detection rates of tracks and points. Since the cluster analysis was performed on the points and not on the tracks, the detection rate of the tracks must be evaluated on the basis of the percentage of the corresponding points belonging to the cluster. In the present case, an entire track was considered to be part of a cluster if at least one point of the corresponding track was part of the cluster. However, if only one point is concerned, and this point is integrated into a cluster by mistake (for example, due to a high GPS accuracy value), this may introduce a bias.

3.4.2. Results

All the results have been summarised in a table, with the processing parameters and the detection rates of the NC parties, both in tracks and in points. More than 600 trials were performed. Results of the 600 trials are summarized in Table 2. Some of the results for these 600 trials are presented in Table 2.

For example, line #1 refers to the analysis carried out on the NC party N°1 (Figure 3), inside the spatial area "Pays de la Loire" (PdL, i.e., an administrative zone) in France (FR), made of 10,700 points within 113 tracks, using the DBSCAN parameters `Eps` = 50 and `MinPts` = 20, after filtering the database in order to retain the data collected during 1 month only (i.e., the month in which the NC party measurements were taken). Using these parameters, 7 clusters are found (Figure 4a): 93.5% of the NC party points (10,008/10,700 points) are clustered in one main cluster, which corresponds to 91.2% of the associated tracks (103/113); the remaining 692 points (6.5%, which corresponds to 10 tracks, 8.8%) are clustered in 6 "secondary" clusters. In addition, the main clusters contain 12,212 extra points (76 extra tracks), i.e., points/tracks that are not part of the NC party. When applying the same DBSCAN parameters but changing the time window to 1 week (line #8 of Table 2), the results show that the main cluster contains the same number of tracks and points, while the number of extra data decreases slightly (12,212 extra points and 76 extra tracks). Such results may be expected when the event duration is less than a week, which is the case in this example. Lastly, when the time window is fixed to 1 day (line #10 of Table 2, which is the official day of the event), all the NC party points are clustered into only one cluster.

Table 2. Each line corresponds to a simulation with specific DBSCAN parameters (`Eps` and `MinPts`) and filtering variables (time window, accuracy, and zone), in order to determine if the clustering analysis is able to detect an NC party. The number and percentage of the detected NC points/tracks in the main cluster and possible secondary clusters are given, as well as the missing NC points/tracks (non-clustered NC data), and the extra points/tracks in the main cluster (i.e., points/tracks that are not linked to the corresponding NC party). Only 22 out of more than 600 results are displayed in this table. Notations: "Time window": "M" month; "W" week; "D" day; Country: "FR" France; "SP" Spain; "IT" Italy; "NL" The Netherlands.

NoiceCapture Party				DBSCAN Parameters and Filtering Variables					Main Cluster		Secondary Clusters		Non Clustered Data		Nb Of	Extra Data in the Main Cluster	
#	ID	Points	Tracks	Eps	MinPts	Time Window	Acc.	Zone (Country)	Points	Tracks	Points	Tracks	Points	Tracks	Clusters	Points	Tracks
1	1	10,700	113	50	20	1 M (September 2017)	Off	PdL (FR)	93.5%	91.2%	6.5%	8.8%	0%	0%	7	12,212	76
2	1	10,700	113	50	100	1 M (September 2017)	Off	PdL (FR)	93.5%	91.2%	5.9%	8%	0.6%	0.8%	2	12,212	76
3	1	10,700	113	50	700	1 M (September 2017)	Off	PdL (FR)	93.5%	91.2%	0%	0%	6.5%	8.8%	1	12,212	76
4	1	10,700	113	100	20	1 M (September 2017)	Off	PdL (FR)	97.2%	96.5%	2.8%	3.5%	0%	0%	2	17,147	119
5	1	10,700	113	500	20	1 M (September 2017)	Off	PdL (FR)	99.8%	100%	0.2%	0%	0%	0%	2	17,148	119
6	1	10,700	113	4000	20	1 M (September 2017)	Off	PdL (FR)	100%	100%	0%	0%	0%	0%	1	17,244	122
7	1	9380	89	50	20	1 M (September 2017)	On	PdL (FR)	100%	100%	0%	0%	0%	0%	7	11,828	70
8	1	10,700	113	50	20	1 W (18–24 September 2017)	Off	PdL (FR)	93.5%	91.2%	6.5%	8.8%	0%	0%	7	11,315	74
9	1	9380	89	50	20	1 W (18–24 September 2017)	On	PdL (FR)	100%	100%	0%	0%	0%	0%	1	10,956	68
10	1	10,700	113	3000	20	1 D (20 September 2017)	Off	PdL (FR)	100%	100%	0%	0%	0%	0%	1	16,231	117
11	3	1209	5	3000	20	1 D (15 October 2017)	Off	PdL (FR)	100%	100%	0%	0%	0%	0%	1	19	1
12	5	18,758	67	3000	20	1 D (28 March 2018)	Off	Quimper (FR)	97.3%	100%	2.7%	0%	0%	0%	2	15,044	114
13	6	4509	44	3000	20	3 M (April–June 2018)	Off	Coruña (SP)	99.4%	95.5%	0.6%	4.5%	0%	0%	4	718	35
14	9	91	3	3000	20	1 M (May 2017)	Off	PdL (FR)	100%	100%	0%	0%	0%	0%	1	0	0
15	10	15,479	141	3000	20	2 M (May–June 2018)	Off	Fisciano (IT)	100%	100%	0%	0%	0%	0%	1	9691	77
16	11	5957	12	3000	20	1 D (9 June2018)	Off	Elven (FR)	100%	100%	0%	0%	0%	0%	1	0	0
17	12	660	17	3000	20	1 W (18–24 June 2018)	Off	Amsterdam (NL)	100%	100%	0%	0%	0%	0%	1	495	12
18	13	19,812	92	3000	20	1 W (16–22 July 2018)	Off	Morbihan (FR)	99.8%	100%	0%	0%	0.2%	0%	1	1021	21
19	13	16,374	84	3000	20	1 W (16–22 July 2018)	On	Morbihan (FR)	100%	100%	0%	0%	0%	0%	1	961	21
20	14	2909	25	3000	20	1 W (8–14 October 2018)	Off	Strasbourg (FR)	100%	100%	0%	0%	0%	0%	1	1028	21
21	15	58,771	506	3000	20	2 M (October–November 2018)	Off	Corse (FR)	100%	100%	0%	0%	0%	0%	1	9654	124
22	17	5840	61	3000	20	1 W (8–14 October 2018)	Off	PdL (FR)	100%	100%	0%	0%	0%	0%	1	1669	35
23	24	35,253	361	5000	20	2 M (April–May 2019)	Off	Catalonia (SP)	50.7%	79.9%	48.8%	19.5%	0.5%	0.6%	9	25,578	246

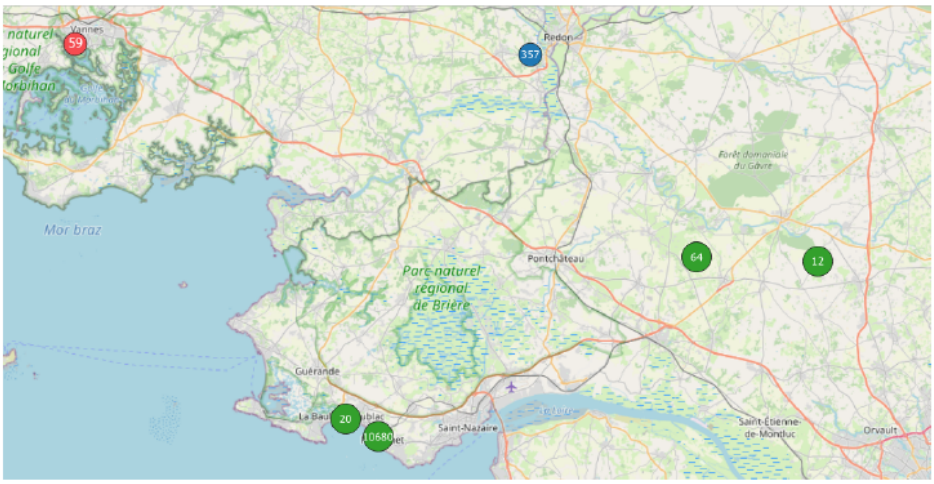
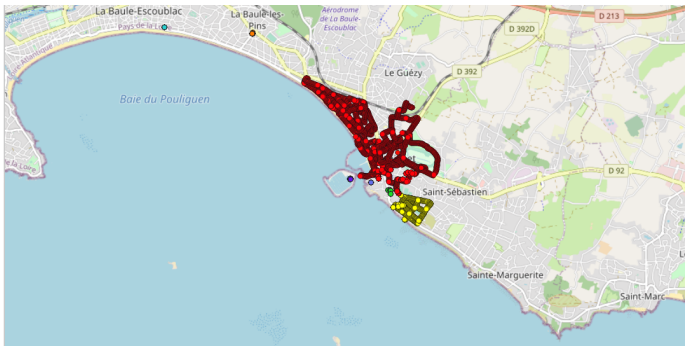
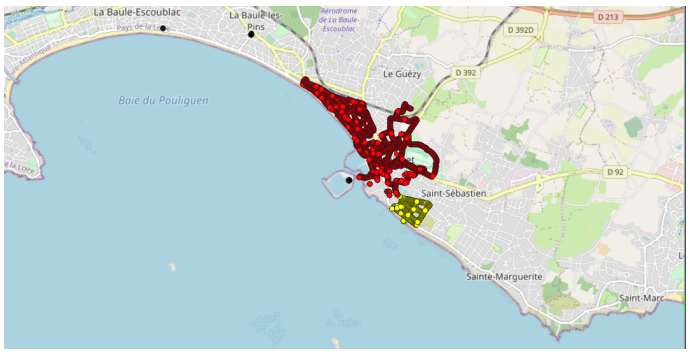


Figure 3. Measurement location for the NC party event Id N°1 (cf. Table 1). To simplify the representation, the points are grouped into 5 main zones. For each zone, the number of measurement points is indicated in the circle. In total, 11,192 geo-localized points have been collected during the official day of the event on 20 September 2017 (green circles), with additional points before (red circle, on 19 September 2017) and after (blue circle, on 23 September 2017) the official day. The collection points cover 2 administrative regions, "Pays de Loire" (20 + 10,680 = 10,700 points), which corresponds to the official spatial zone of the NC party, and "Britain" (59 + 357 + 64 + 12 = 492 points); represented area of approximately 44.0 km × 102.0 km.

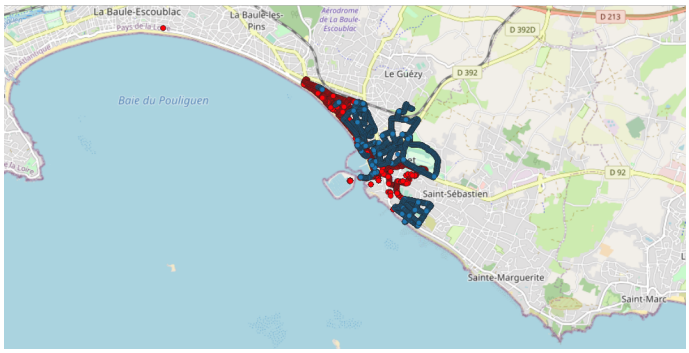


(**a**)



(**b**)

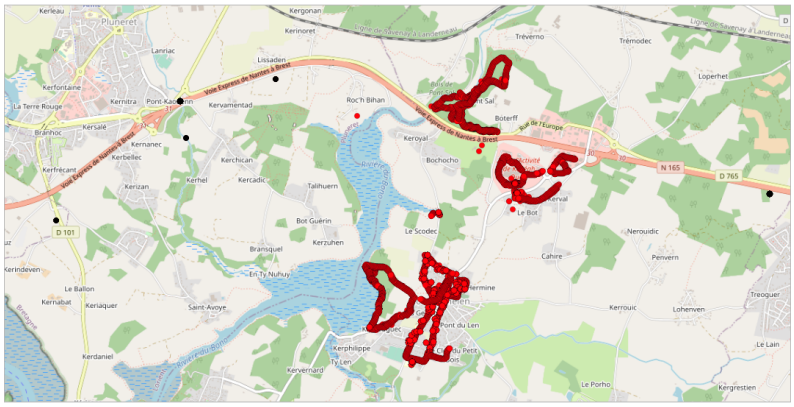
Figure 4. *Cont.*



(**c**)

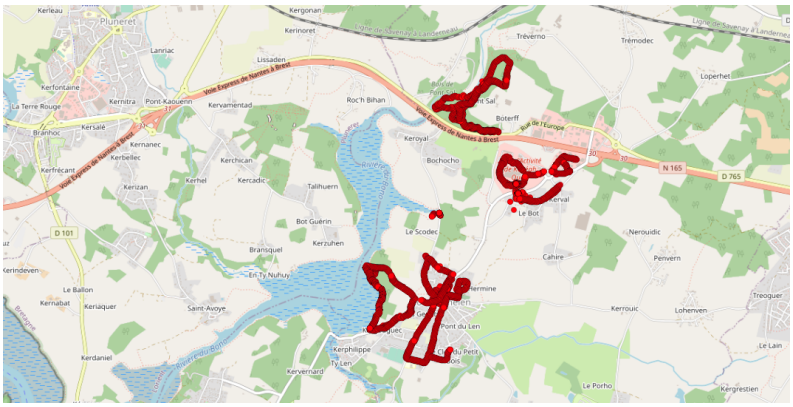
Figure 4. DBSCAN clustering approach, applied to the NC party Id N°1. Due to the scale of the map and because the measurement points are close together, most of the points are stacked, which makes it harder to see all of them individually. Represented area of approximately 6.3 km × 11.5 km: (**a**) representation of line #1 of Table 2 (`Eps` = 50 and `MinPts` = 20): red points represent the main cluster; the other colors represent the secondary clusters; all NC points are grouped in 7 clusters but mainly in one main cluster (red) and one secondary cluster (yellow); (**b**) representation of line #2 of Table 2 (`Eps` = 50 and `MinPts` = 100): red points represent the main cluster; yellow points represent the secondary cluster; black points represent the non-clustered data; all NC points are included in 2 clusters, the main and a secondary one; (**c**) representation of line #10 of Table 2 (`Eps` = 3000 and `MinPts` = 20); all data are located in only one cluster: red points represent all the points of the NC party Id N°1 that are part of the cluster; blue ones represent the extra data that are part of the same cluster, but not from the NC party.

When the GPS accuracy filter is considered (i.e., "Acc. = On"), as in line #7 in Table 2, the number of tracks and points in the NC party decreases, since some of them are removed from the analysis; in this specific case, one can observe that 100% of the NC points/tracks are found in only one cluster, which confirms that the GPS accuracy may have an impact on the quality of the clustering. This is also illustrated in Figure 5 showing the effect of the accuracy filter for the clustering of the NC party Id N°13. In this example, several measurement points with bad accuracy are removed when the accuracy filter is enabled; all the remaining points are found in the same cluster.



(**a**)

Figure 5. *Cont.*



(**b**)

Figure 5. Representation of the clustering results for the NC party Id N°13 in the "Morbihan" department in France (lines 18 and 19 of Table 2). It compares the two options "Off" and "On" for the "Accuracy" filter. Only one cluster is found: red points correspond to the points of the cluster (NC party + Extra points); black points correspond to the points of the NC party that are not clustered (5 "isolated" black points: 4 on the left and 1 on the right of Figure 5a); represented area of approximately 3.4 km × 5.6 km; (**a**) "Accuracy" = "Off"; (**b**) "Accuracy" = "On".

Considering NC party N°1, the number of extra points is very important, the obtained cluster being about twice that of the original NC party. Looking more precisely at the time stamp and the measurement area of these extra points, one can observe that they are very consistent with the specific data of the NC party (Figure 4c). The experience of the NC party organizers shows that in some cases, the participants in the event may forget to mention the NC party code when collecting the measurements. The corresponding data are therefore not counted in the NC party but are nevertheless well associated with the event. In the present case, this hypothesis seems more than likely. An analysis of the extra points shows that 100% of the points would be well associated with the NC party. In absolute terms, this shows that the clustering method worked rather well in this case since it would have allowed the integration of unexpected but relevant data.

Lines # 1, 4, 5, and 6 of Table 2 for NC party N°1 show that increasing the `Eps` parameter increases the ratio of NC point/track detection (more quickly for NC tracks). This is of course an expected and obvious result, since by increasing the search radius of the points, without increasing the `MinPts`, it is easier to find new members for the cluster. This may suggest that the two parameters `Eps` and `MinPts` cannot be chosen independently.

This interrelation of the two parameters, `Eps` and `MinPts`, can be analyzed through a contingency-table-based analysis. The combinations of DBSCAN parameters that succeed the most in clustering 100% of the NC party tracks/points are `MinPts` = 20 points with `Eps` = 3000 m (21 NC parties are found out of the 26, i.e., 80.8%) and 2000 m (16 NC parties are found out of the 26, i.e., 61.5%). Nevertheless, these combinations worked for the NC parties in France, Italy, and The Netherlands only. Regarding the "Zone" and "Accuracy" filters, the analysis of variance (ANOVA) does not allow us to conclude on their effect on the clustering due to lack of ANOVA assumptions (i.e., homogeneity and normality). In addition, a Principal Component Analysis (PCA) was also performed, but here again, it was ineffective to describe the relationship between variables. Lastly, considering a linear regression analysis, it is observed that not only the `Eps` and `MinPts` parameters, but also the "Time window" filter, contribute explaining information regarding "Clustered data" and "Extra data". Nevertheless, the information that is explained is rather weak; it can help when choosing optimal DBSCAN parameters and filtering variables, but they cannot be expected to be the only factors to consider. Overall, due to the small sample size (i.e., only 26 NC parties with a few points/tracks), the result is quite hard to interpret, and the statistical analysis cannot be performed under the best conditions.

Therefore, it is clear that the choice of DBSCAN parameters and filtering variables will need the experience of an "expert" to achieve the optimal clustering analysis. There are probably no optimal values, and the choice will mostly depend on the nature of the clusters to obtain. To be very selective, one should decrease the value of `Eps` and increase that of `MinPts`. On the other hand, if we simultaneously choose a value of `Eps` that is too high and a value of `MinPts` that is too low, the number of clusters may be considerable, with the risk that they are not at all representative of a specific event. Nevertheless, it is well shown that this DBSCAN approach may be useful to find clusters of interest for NC data analysis.

4. Application of DBSCAN on the NoiseCapture Database

4.1. Preliminary Results of DBSCAN in Some Countries

At this stage, one can already apply the DBSCAN method to the NC database. One of the objectives of this first application is to determine if the method is able to detect clusters and then to identify the typology of the clusters that are obtained by this approach and their possible interest.

As mentioned above, the values for `Eps` and `MinPts` parameters will condition the relevance of the detected clusters. The role of the expert is therefore important in the choice of these parameters. In the present case, the parameters are fixed to `Eps` = 3000 m and `MinPts` = 200 points, in order to limit the number of detected clusters and focus on larger ones. Indeed, the use of the previously identified set of parameters (`Eps` = 3000 m and `MinPts` = 20 points) on a large spatial area leads to a number of clusters that is too large to allow a relevant analysis.

In this preliminary study, the DBSCAN method is applied to whole countries (i.e., the "spatial zone" is equal to a country), with all the geo-located data (i.e., "Accuracy" fixed to "Off") and considering the data collected each month (i.e., the "Time window"). In this application, four countries were selected in regard to past observations [8]:

- The first application was carried out in Peru since an unusual and large amount of collected data was observed on 8–9 October 2018 in a past study [8]. Applying the DBSCAN method leads to eight clusters (Figure 6 and Table 3). Among them, one of the most important clusters effectively took place in October 2018 in the City of Cajamarca (Figure 7, green points, 10,740 tracks, 108,785 points). Another cluster was also identified in November 2018 (Figure 7, pink points, 248 tracks and 2334 points) at the same place. The tracks for the green cluster were collected by 23 contributors in 18 days, with a high concentration of measurements on a few days. Moreover, the highest number of points were collected between 09:00 and 09:59 (18.6% of points) and between 13:00 and 13:59 (17.2% of points). For the pink cluster, data were collected in 2 days by 3 users, who have also participated in collecting tracks for the green cluster. Moreover, most points were collected between 07:00 and 07:59 (90.8%). Considering the distribution of measurements over time and the total number of measurements, it is likely that these two clusters are the result of specifically coordinated events.
- The next application was realized in the United Kingdom, one of the top contributors to the NC database with 4693 tracks (4th in tracks contribution) and 2,067,182 points (3rd in points contribution). Applying the DBSCAN method leads to 440 clusters. The cluster with the highest number of points (126,533 points with 33 tracks) took place in October 2019, close to the city of Stevenston in Scotland (Figure 8). This cluster was collected by one user only during 12 days, with a few tracks per days; the highest number of the data collection was performed between 22:00 and 01:59 (25.8% points) and between 06:00 and 06:59 (8.5% points), which could suggest that an objective of the measurements was to evaluate the noise distribution during late night and early morning. The metadata show that the smartphone was calibrated for all the measurements, but the calibration value (40 dB) seems excessive in relation to what can normally be expected. The cluster with the highest number of tracks (248 tracks for 38,104 points) took place on November 2018 in the city of Strood in England. This cluster was gathered by two users, and all of the tracks were calibrated

to the same value (0 dB). This cluster was collected in 21 days, with 10 to 20 tracks per day, mostly between 16:00 to 17:59 (12.1% points).

- Italy was also considered, since it is also one of the major contributors to the NC database with 2654 tracks (ranked 11th) and 364,613 points (ranked 18th) and because few NC parties have been organized. Applying the DBSCAN method gives 151 clusters, among them, the two known NC party events, which took place in Fisciano in May 2018 and May 2019 [42,43] (Figure 9).
- Lastly, the DBSCAN method was also applied to France, where most of the NC parties were carried out and potentially, some non-official events. A total of 1852 clusters were found, among them 211 during the month of September 2017, which corresponds to the first month of the application's existence. The cluster with the most tracks and points (1358 tracks/32,0850 points by 429 contributors) is observed in Paris in September 2017. This cluster was collected during the entire month with a majority of points collected on 11, 12, and 13 of the month, with most points collected between 12:00 and 12:59, 10:00 and 10:59, and 19:00 and 19:59. Using corresponding DBSCAN parameters in the case of France, the method returns too many clusters to make a detailed and individual analysis. It is also unlikely that all these clusters are associated with events. This suggests that the number of clusters should perhaps be limited to be sure of their interest.

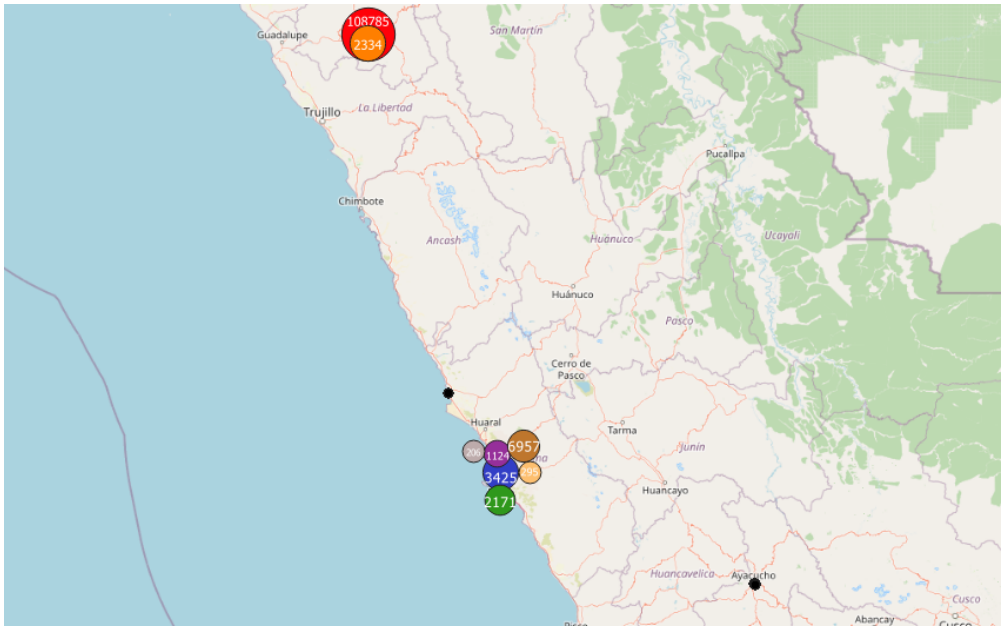


Figure 6. Representation of the 8 clusters found in Peru (the black points represent the non-clustered data), using the DBSCAN approach with `Eps` = 3000 m and `MinPts` = 200 points; represented area of approximately 750 km × 1200 km.

Table 3. DBSCAN approach for the data collected in Peru: 8 clusters are found (see also Figure 6).

Cluster Nb	Tracks	Points	Contributors	Month (Year)	Nearest City	Comments
1	10,740	108,785	23	October (2018)	Cajamarca	
2	248	2334	3	November (2018)	Cajamarca	3 users were part of Cluster 1
3	23	3425	2	March (2019)	Lima	
4	1124	1	1	March (2019)	Lima	
5	16	2171	1	May (2019)	Lima	Same users as for Cluster 3
6	1	295	1	May (2019)	Lima	Same users as for Cluster 4
7	20	6957	2	November (2019)	Lima	Same users as for Clusters 3 and 4
8	4	206	1	November (2019)	Lima	Same users as for Cluster 3

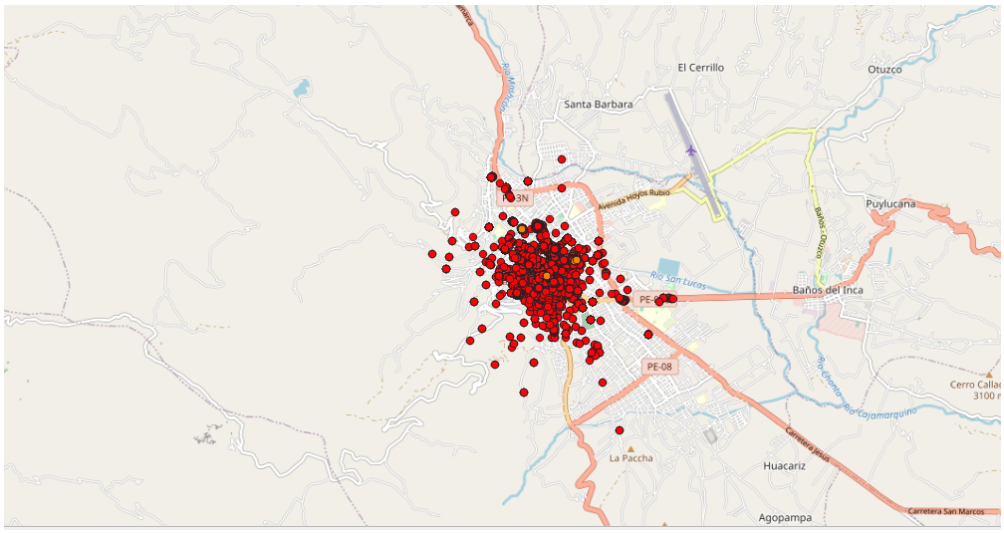
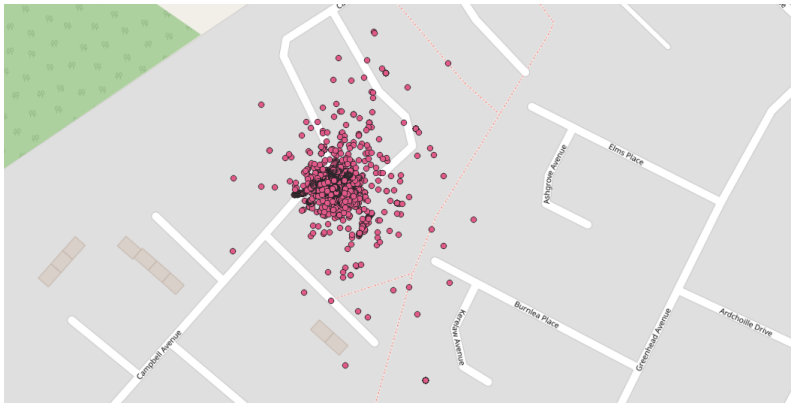
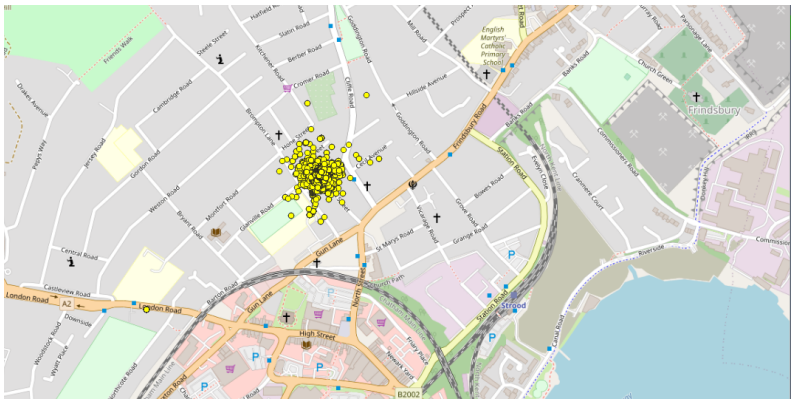


Figure 7. Representation of the clusters found in Cajamarca: red cluster: 10,740 tracks/108,785 points in October 2018; orange cluster: 248 tracks/2334 points in November 2018); represented area of approximately 10 km × 20 km.

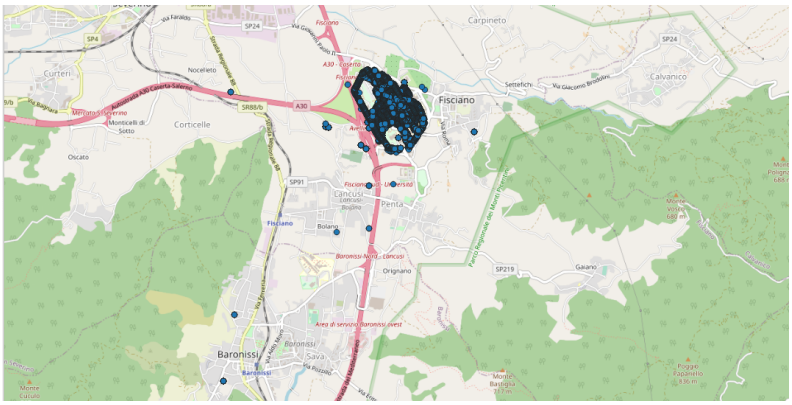


(**a**)

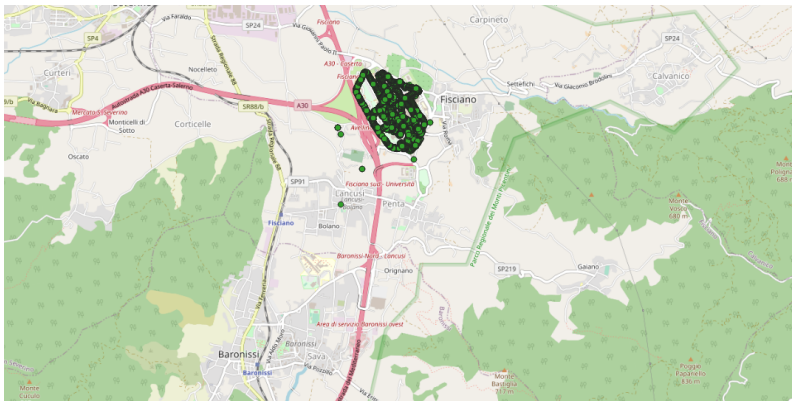


(**b**)

Figure 8. Representation of the clusters found in Stevenston (Scotland, October 2019) and Strood (England, November 2018): (**a**) Stevenston cluster (in pink); represented area of approximately 195 m × 300 m; (**b**) Strood cluster (in yellow); represented area of approximately 0.86 km × 1.70 km.



(**a**) Cluster at Fisciano (Italy) in May 2018



(**b**) Cluster at Fisciano (Italy) in May 2019

Figure 9. Representation of the 2 clusters found in Italy, at Fisciano. Both clusters correspond to NC party events: (**a**) NC party Id N°10 in May 2018; (**b**) NC party Id N°26 in May 2019; represented area of approximately 4.5 km × 8.9 km.

4.2. Cluster Typology

The last application shows the ability of the DBSCAN method to identify clusters of interest, for which the observed behaviors suggest that they are specifically organized events. Among the large number of clusters detected, not all can give rise to the same attention. The analysis of the clusters obtained in this first step allows us to identify the following cluster typology:

- Type A (38.0%): Clusters composed of a large number of tracks/points and with data collected by multiple users during a period of a month or less. This is the most expected cluster type, since it typically corresponds to an NC party type event.
- Type B (40.7%): Clusters with data that are collected by one or a few users express the involvement of one or a few people in the collection of a large number of measurements. It is a priori an individual behavior, which can illustrate the involvement of some people in the "crowd-sourced" spirit of the NC project. This type of cluster can be interesting, especially if the user is considered an expert.
- Type C (4.4%): Clusters composed of a lot of tracks but with a small number of points in total.
- Type D (13.5%): Clusters that are composed of only one track and contain a few points (between 200 and 500 points).
- Type E (3.4%): Clusters with a regular daily collection for more than several days.

Other behaviors were also observed, which would suggest that it is possible to combine several clusters into the same cluster:

- Clusters of data collected during the same time period (day or hour) but in different locations.
- Clusters that are close together and may be related to the same event.
- Clusters of data collected by the same users in the same location but at different times.

Based on the previous parameters for DBSCAN, it is clear that the number of detected clusters can be very important once applied to the whole NC database. Moreover, based on this typology of clusters, and depending on the final goal of the clustering, not all clusters have the same importance. Coming back to the initial objective of the present study, i.e., the constitution of a reference database, type A and B clusters are certainly the most relevant due to a much higher density of measurement points, expressing a willingness to collect data over a given spatial extent. On this basis, the higher the value of the parameter `MinPts` will be, the larger the number of measurement points that will be important and the more the number of concerned clusters that will decrease (as shown by Table 4). As also expected, regrouping data by month decreases the number of clusters, since a cluster associated with the same event over two weeks will be separated into two distinct clusters if the search is performed by week instead of by month.

4.3. Applying the DBSCAN Method to the Full NC Database

As a second application of the DBSCAN approach, the full NC database is considered (197,568 tracks, 48,901,719 points, 50,868 contributors, 195 countries), using the parameters `Eps` = 3000 m and `MinPts` = 5000 points, filtering the data weekly. On a laptop (Intel(R) Core(TM) i5-10210U CPU 64-bit Processor), the method takes about 16 h to process without any special optimization.

Overall, 2046 clusters were found in 68 countries. The United States showed the most clusters (975 clusters), followed by France (297 clusters) and the United Kingdom (111 clusters), which is an expected result since these three countries are considered among the top three contributors to the NC database [8]. Among these 2046 clusters, 1567 clusters (76.59%) were collected by one contributor each (found in 40 countries, by 1155 different contributors). In addition, 252 clusters (12.32%) were collected by two contributors each, in 31 countries. This leaves 227 clusters (24,280 tracks and 4,548,638 points) with at least three users contributing to the data collection for each cluster (Figure 10): 95 clusters in France, 36 clusters in United States, and 9 clusters in Switzerland. Moreover, 19 of these clusters were NC party events.

The analysis of the literature shows that scientific studies have also been carried out on the basis of the collection of measurements using the NC application by teams without any link with the NC project team. At this stage of the study, it seems interesting to check if corresponding clusters have been found by the DBSCAN approach for these specific experiments.

Table 4. Number of clusters found by the DBSCAN approach in function of the `MinPts` parameters by filtering the data by month or by week. The number of clusters with one track or one or two contributors is also given, as well as the clusters with a regular daily collection (i.e., between 2 and 5 tracks per day). The number in parenthesis is the number of NC party events in the country. "Failed NC party" means that all the points of the event were not clustered, while "Partly failed NC" means that at least 70% of the event was clustered. When "Failed NC party" and "Partly failed NC" are both equal to zero, the NC parties are fully clustered.

Country	Eps	MinPts	Period	Number of Clusters						
				Total	1 Track	1 Contributor	2 Contributors	Regular Daily Collection	Failed NC party	Partly Failed NC party
France	3000	20	Month	5204	2416	4370	471	635	0 (18)	0 (18)
	3000	200	Month	1852	429	1278	260	143	1 (18)	0 (18)
	3000	5000	Month	224	19	98	28	29	3 (18)	0 (18)
	3000	10,000	Month	125	8	42	13	27	7 (18)	0 (18)
	3000	5000	Week	297	32	164	41	0	4 (18)	1 (18)
	3000	10,000	Week	125	12	61	14	0	8 (18)	1 (18)
Italy	3000	20	Month	564	270	525	27	5	0 (2)	0 (2)
	3000	200	Month	155	34	129	14	4	0 (2)	0 (2)
	3000	5000	Month	16	3	11	1	0	0 (2)	0 (2)
	3000	10,000	Month	9	1	6	0	0	0 (2)	0 (2)
	3000	5000	Week	15	2	10	1	0	0 (2)	0 (2)
	3000	10,000	Week	7	0	4	1	0	0 (2)	0 (2)
United Kingdom	3000	20	Month	1094	509	1024	51	22	–	–
	3000	200	Month	440	122	389	32	24	–	–
	3000	5000	Month	77	19	62	4	5	–	–
	3000	10,000	Month	48	7	37	4	1	–	–
	3000	5000	Week	111	23	99	9	0	–	–
	3000	10,000	Week	57	10	49	6	0	–	–
Peru	3000	20	Month	83	40	72	9	2	–	–
	3000	200	Month	17	3	11	4	0	–	–
	3000	5000	Month	2	0	0	1	0	–	–
	3000	10,000	Month	1	0	0	0	0	–	–
	3000	5000	Week	2	0	0	1	0	–	–
	3000	10,000	Week	1	0	0	0	0	–	–

A first study, published in 2020, was carried out in Japan [44] in order to evaluate the effect of COVID-19 pandemic lockdowns in the eastern edge of the city of Kobe. An NC measurement campaign was set up over a period of one hour (10:00–11:00 am), with an average time of 30 s, at six locations in an urban area (as well as at a fixed position in front of a building), on two different days in May 2020, during and after the lockdown period. Using the clustering approach, the measurements related to the study were found as non-clustered data (Figure 11a). This is probably due to the number of measurement points (3850 points) that was a priori lower than the number of measurement points for detecting clusters (`MinPts` = 5000 points). Nevertheless, two clusters were detected in the same area in July 2020 (Figure 11b) and in August 2020 (Figure 11c). These measurements campaigns were carried out by the same two users that carried out the study in May 2020 [44], which may indicate additional measurements to the initial experimentation.

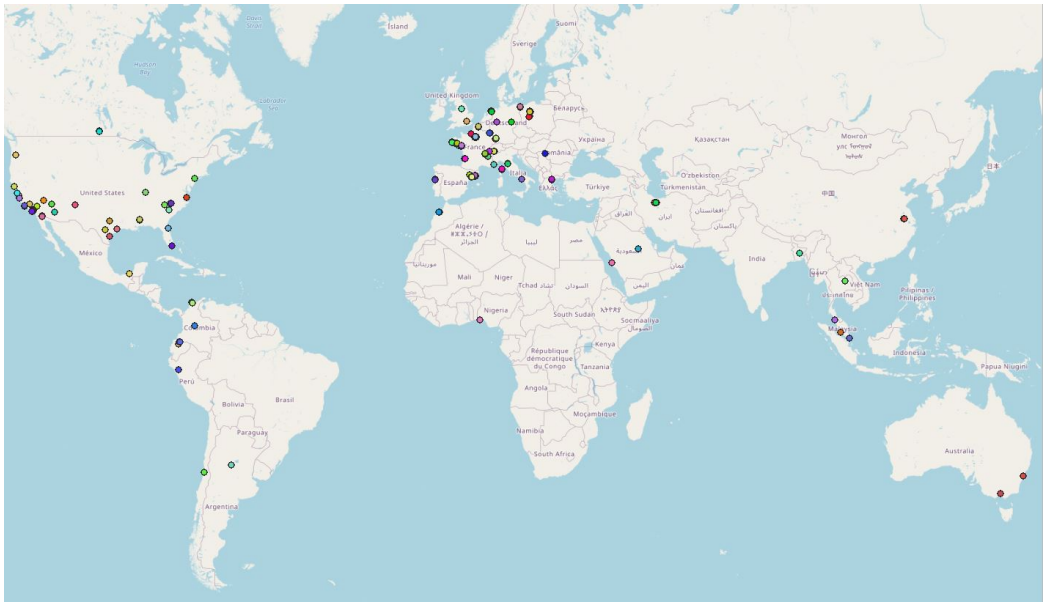


Figure 10. Representation of the 227 clusters with at least 3 contributors found around the globe using the DBSCAN approach (1 point is a cluster; cluster points may overlay; the color of each cluster is arbitrary).

An NC measurement campaign was also carried out in India [45] in three urban zones corresponding to specific noise ambiances of Lucknow City: "Polytechnic chauraha", "Hazrat ganj chauraha", and "Haniman chauraha". The measurements were collected at three time periods of the day (morning, afternoon, and evening), for 10 min each. However the number of measurement points was a priori not sufficient for the clustering methodology to detect this event as a cluster.

Another experiment, involving the comparison of several smartphone noise measurement applications was conducted in 2018 [46]. Measurements were performed over three periods of one day (7:00–9:00, 15:00–17:00 and 19:00–21:00), twice, in an area of the city of Zagreb, Croatia. The reference [46], which is a student report, does not give enough indication about the sampling of the measurements and the exact date of the measurements. Nevertheless, applying our methodology, a cluster was identified, which corresponds to a part of this experimentation (Figure 12). This cluster is located in the proximity of Zagreb train station, and the measurements were collected during 12–18 March 2018.

(**a**) (**b**)

(**c**)

Figure 11. Representation of the clusters found in Kobe (Japan), in 2020; represented area of approximately 1.3 km × 1.05 km: (**a**) representation of the measurements carried out during the event [44] in May 2020 (190 tracks, 3850 points); (**b**) [representation of the measurements carried out by the same users of the event [44]. July 2020 (28 tracks, 7778 points); (**c**) representation of the measurements carried out by the same users of the event [44] in August 2020 (12 tracks, 15,928 points). All the data are extracted from the NoiseCapture database [41].

The last event that can be found in the scientific literature took place in Cairo, Egypt in August 2018, inside the "Kasr Al Ainy Hospitals" building. These experiments were conducted in order to study the effect of noise pollution on patients undergoing surgery [47]. The experiment appears to have been correctly discovered by the clustering approach, consisting of five tracks with a total of 9418 points and collected by a single user (Figure 13).

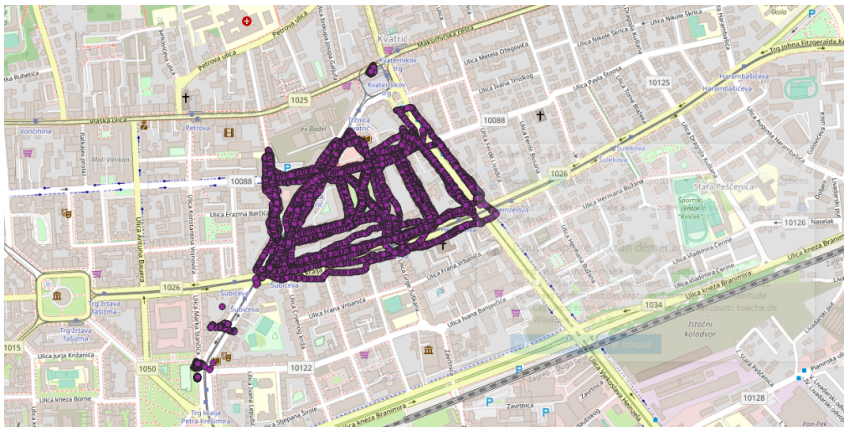


Figure 12. Representation of the cluster found as part of the event (3 tracks/6417 points) in Zagreb (Croatia), in March 2018 [46] ; represented area of approximately 1 km × 2 km. All the data are extracted from the NoiseCapture database [41].

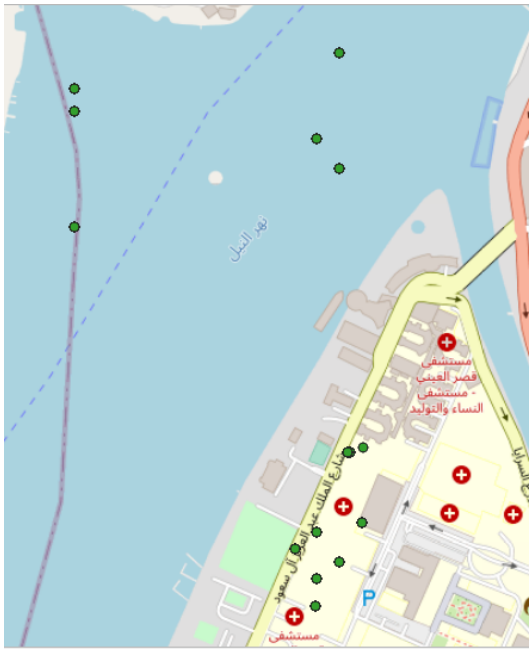


Figure 13. Representation of the cluster found as part of the event (5 tracks, 9418 points) in Cairo (Egypt) in August 2018 [47]. Note that 6309 points are affected by low GPS accuracy due to the measurement inside a building, which makes them poorly positioned on the Nile River; represented area of approximately 950 m × 820 m. All the data are extracted from the NoiseCapture database [41].

5. Conclusions

Since 2017, the NC project has collected a large amount of information around the world, which suggests that data may now be useful to evaluate the quality of sound environments. However, the preliminary analysis of the data showed that it is important to quantify the quality of the data in order to control its use [8]. Most of the quality control methods that have been identified in the framework of the NC project for future developments are based on machine learning methods; among them, (semi-)supervised ones seem well-suited. This, however, implies the existence of labelled data to train the models.

Such labelled data can, for example, be obtained during the organization of specific events, called NC parties, for which a supervision of the contributors allows us to ensure

that the measurement protocol has been followed and that the smartphones have been calibrated. The data collected during NC parties can thus be considered as reference data and can be used to learn models. However, these reference data are insufficient in number to ensure the quality of learning; additional data are then required in the reference database. The spread of the NC application around the world has also allowed other participants to organize NC party-like events, which would complete the reference database as soon as this data can be identified in the database. In general, the realization of these events generates an increase in the spatial and temporal density of the measurement points, which suggests that clustering-type methods would be well-suited to detect them. Among them, the DBSCAN method was retained in the present paper and tested in several configurations.

The method was first applied in order to verify that the known NC parties could be detected. By judiciously choosing the two main DBSCAN parameters, the search radius of the measurement points (`Eps`) and the minimum number of points (`MinPts`), it is thus possible to find 100% of the known NC parties in the form of clusters. In a second step, the method was applied to a selection of countries in order to analyze the typology of the detected clusters. Several events similar to the NC party were indeed detected, but, depending on the value of the processing parameters, a greater number of additional clusters can also be detected without being able to associate them in an obvious way with a particular event. Finally, the method was applied to the whole NC database, with a set of parameters aimed at detecting the most important clusters. More than 2000 clusters were detected in the world, some of which could be associated with events organized in the framework of research published in the literature.

It is clear that the method is very dependent on the parameters `Eps` and `MinPts`, and their choice therefore requires the help of an expert, depending on the typology and number of clusters that are expected. It seems, however, possible in the future to modify the DBSCAN method to automatically find the appropriate values for the parameters `Eps` and `MinPts` [48,49].

It is also clear that the classical DBSCAN method may return clusters with a lower level of relevance, and other clustering methods would deserve to be compared in terms of performance [50]. One can, for example, cite the ordering points to identify the clustering structure (OPTICS) [51], which sets out to solve one of DBSCAN weaknesses (i.e., detecting meaningful clusters in data of varying density), by performing the clustering after ordering the points in a linear order (e.g., date/time of measurements or spatial directions, for example). The OPTICS method can give better results than DBSCAN, but it increases the computational time due to the initial ordering of data. Instead of focusing on points to build clusters, it could also be interesting to consider trajectories (since a set of measurement points comes from the same track), using line/trajectory spatial clustering methods. Among them, the TRACLUS method [52] would seem to be well adapted in this case since it allows us to group trajectories with common sub-trajectories in the form of clusters. This is usually the case in practice when an NC party is organized locally, since all contributors start collecting data from the same location (from the same street, for example).

Considering the initial objective (i.e., to build a reference database), post-processing may be carried out after the clustering, for example by merging some detected clusters. Indeed, some measurements can belong to the same event but may be grouped into different clusters because they are performed in different places or different periods. This can be solved by investigating the contributors of each cluster to see whether they are the same, or by detecting if the area of the clusters is overlapping (this is the case, for example, when the measurements are collected in a same space but at a different period). In addition, the reference database may be increased by selecting the most important/relevant clusters and then associating the data that are produced independently by the participants to these events. Finally, this would allow the construction of a larger reference database that could be suitable for the use of supervised machine learning methods to develop quality control protocols for the NC data.

Author Contributions: Conceptualization, J.P. and E.B.; methodology, A.B. and J.P.; formal analysis, A.B. and J.P.; investigation, A.B., J.P. and E.B.; writing—original draft preparation, A.B. and J.P.; writing—review and editing, J.P., A.B., and E.B.; visualization, A.B.; supervision, J.P. and E.B.; project administration, J.P. and E.B.; funding acquisition, E.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was initially funded in the framework of the ENERGIC-OD Project (European Network for Redistributing Geospatial Information to user Communities - Open Data), under the ICT Policy Support Programme (ICT PSP) (CIP-ICT-PSP-2013-7) as part of the Competitiveness and Innovation Framework Programme by the European Community. A part of this research is funding by the Région Pays de La Loire grand number 2020_10361.

Data Availability Statement: The data presented in this study are openly available in the NoiseCapture database extraction from 29 August 2017 until 28 August 2020 (3 years) at https://doi.org/10.25578/J5DG3W.

Acknowledgments: Map data are from OpenStreetMap licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF) (see https://www.openstreetmap.org/copyright/).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Peris, E. *Environmental noise in Europe: 2020. EEA Report No 22/2019*; European Environment Agency, Publications Office: Luxembourg, 2020. [CrossRef]
2. Picaut, J.; Can, A.; Fortin, N.; Ardouin, J.; Lagrange, M. Low-Cost Sensors for Urban Noise Monitoring Networks—A Literature Review. *Sensors* **2020**, *20*, 2256. [CrossRef] [PubMed]
3. Santini, S.; Ostermaier, B.; Adelmann, R. On the use of sensor nodes and mobile phones for the assessment of noise pollution levels in urban environments. In Proceedings of the 2009 Sixth International Conference on Networked Sensing Systems (INSS), Pittsburgh, PA, USA, 17–19 June 2009; pp. 1–8. [CrossRef]
4. Rana, R.K.; Chou, C.T.; Kanhere, S.S.; Bulusu, N.; Hu, W. Ear-phone: An End-to-end Participatory Urban Noise Mapping System. In Proceedings of the 9th International Conference on Information Processing in Sensor Networks, IPSN 2010, Stockholm, Sweden, 12–16 April 2010; ACM: New York, NY, USA, 2010; pp. 105–116. [CrossRef]
5. Kanjo, E. NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping. *Mob. Netw. App.* **2010**, *15*, 562–574. [CrossRef]
6. Maisonneuve, N.; Stevens, M.; Ochab, B. Participatory Noise Pollution Monitoring Using Mobile Phones. *Info. Pol.* **2010**, *15*, 51–71. [CrossRef]
7. Picaut, J.; Fortin, N.; Bocher, E.; Petit, G.; Aumond, P.; Guillaume, G. An open-science crowdsourcing approach for producing community noise maps using smartphones. *Build. Environ.* **2019**, *148*, 20–33. [CrossRef]
8. Picaut, J.; Boumchich, A.; Bocher, E.; Fortin, N.; Petit, G.; Aumond, P. A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *Int. J. Environ. Res. Public Health* **2021**, *18*, 7777. [CrossRef] [PubMed]
9. Noise-Planet Website. Noise-Planet-Data. 2021. Available online: https://data.noise-planet.org/index.html (accessed on 16 September 2022).
10. NoiseCapture Map Public Webpage. Available online: https://noise-planet.org/map_noisecapture (accessed on 16 September 2022).
11. Noise-Planet Website. Available online: https://noise-planet.org (accessed on 3 October 2022).
12. Jhaveri, R.H.; Revathi, A.; Ramana, K.; Raut, R.; Dhanaraj, R.K. A Review on Machine Learning Strategies for Real-World Engineering Applications. *Mob. Inform. Syst.* **2022**, *2022*, 1833507. [CrossRef]
13. Lease, M. On quality control and machine learning in crowdsourcing. In Proceedings of the Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 8 August 2011.
14. McNicholas, C.; Mass, C.F. Smartphone Pressure Collection and Bias Correction Using Machine Learning. *J. Atmos. Ocean. Technol.* **2018**, *35*, 523–540. [CrossRef]
15. Sheng, V.S.; Zhang, J. Machine Learning with Crowdsourcing: A Brief Summary of the Past Research and Future Directions. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 9837–9843. [CrossRef]
16. Niu, G.; Yang, P.; Zheng, Y.; Cai, X.; Qin, H. Automatic Quality Control of Crowdsourced Rainfall Data With Multiple Noises: A Machine Learning Approach. *Water Resour. Res.* **2021**, *57*. [CrossRef]
17. Kisilevich, S.; Mansmann, F.; Nanni, M.; Rinzivillo, S. Spatio-temporal clustering. In *Data Mining and Knowledge Discovery Handbook*; Maimon, O., Rokach, L., Eds.; Springer: New York, NY, USA, 2010; pp. 855–874. [CrossRef]
18. Jobson, J.D. *Applied Multivariate Data Analysis: Categorical and Multivariate Methods/Book and Disk*, 1st ed.; 1992, corr. 2nd printing 1994 édition ed.; Springer: New York, NY, USA, 1994.

19. Aggarwal, C.C.; Reddy, C.K. *Data Clustering: Algorithms and Applications*, 1st ed.; Chapman and Hall/CRC Data Mining and Knowledge Discovery Series: Boca Raton, FL, USA, 2013.
20. Shahzad, A.; Coenen, F. Efficient Distributed MST Based Clustering for Recommender Systems. In Proceedings of the 20th IEEE International Conference on Data Mining Workshops (ICDMW 2020), Sorrento, Italy, 17–20 November 2020; pp. 206–210. [CrossRef]
21. Li, C.-L.; Lian, B.; Lu, H.-S. The Application of Factor Cluster Composite Analysis in Market Segmentation Research. In Proceedings of the 2011 International Conference on Management Science & Engineering 18th Annual Conference Proceedings, Rome, Italy, 13–15 September 2011; pp. 563–568.
22. Ayanegui-Santiago, H.; Reyes-Galaviz, O.F.; Chavez-Aragon, A.; Ramirez-Cruz, F.; Portilla, A.; Garcia-Banuelos, L. Mining Social Networks on the Mexican Computer Science Community. In Proceedings of the MICAI 2009: Advances in Artificial Intelligence, Guanajuato, Mexico, 9–13 November 2009; Aguirre, A.H., Borja, R.M., Garcia, C.A.R., Eds.; Springer: Berlin, Germany, 2009; Volume 5845, pp. 213–224.
23. Hsieh, L.C.; Wu, G.L.; Hsu, Y.M.; Hsu, W. Online image search result grouping with MapReduce-based image clustering and graph construction for large-scale photos. *J. Vis. Commun. Image Represent.* **2014**, *25*, 384–395. [CrossRef]
24. Zhao, M.; Chen, J. A Review of Methods for Detecting Point Anomalies on Numerical Dataset. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; pp. 559–565.
25. Akdemir, S.; Tagarakis, A. Investigation of Spatial Variability of Air Temperature, Humidity and Velocity in Cold Stores by Using Management Zone Analysis. *J. Agric. Sci.-Tarim Bilim. Derg.* **2014**, *20*, 175–186. [CrossRef]
26. Dupuis, D.J.; Trapin, L. Structural change to the persistence of the urban heat island. *Environ. Res. Lett.* **2020**, *15*, 104076. [CrossRef]
27. Fakhruddin, M.; Putra, P.S.; Wijaya, K.P.; Sopaheluwakan, A.; Satyaningsih, R.; Komalasari, K.E.; Mamenun; Sumiati; Indratno, S.W.; Nuraini, N.; et al. Assessing the interplay between dengue incidence and weather in Jakarta via a clustering integrated multiple regression model. *Ecol. Complex.* **2019**, *39*, 100768. [CrossRef]
28. Smith, M.J.d.; Goodchild, M.F.; Longley, P.A. *Geospatial Analysis: A Comprehensive Guide*, hardback ed.; The Winchelsea Press: London, UK, 2018.
29. Craig, A.; Moore, D.; Knox, D. Experience sampling: assessing urban soundscapes using in-situ participatory methods. *Appl. Acoust.* **2017**, *117*, 227–235. [CrossRef]
30. De Coensel, B.; Botteldooren, D.; Debacq, K.; Nilsson, M.E.; Berglund, B. Clustering outdoor soundscapes using fuzzy ants. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 1556–1562. [CrossRef]
31. Pita, A.; Rodriguez, F.J.; Navarro, J.M. Cluster Analysis of Urban Acoustic Environments on Barcelona Sensor Network Data. *Int. J. Environ. Res. Public Health* **2021**, *18*, 8271. [CrossRef] [PubMed]
32. Zambon, G.; Benocci, R.; Brambilla, G. Cluster categorization of urban roads to optimize their noise monitoring. *Environ. Monit. Assess.* **2016**, *188*, 26. [CrossRef]
33. Socoró, J.C.; Alías, F.; Alsina-Pagès, R.M. WASN-Based Spectro-Temporal Analysis and Clustering of Road Traffic Noise in Urban and Suburban Areas. *Appl. Sci.* **2022**, *12*, 981. [CrossRef]
34. Saxena, A.; Prasad, M.; Gupta, A.; Bharill, N.; Patel, O.P.; Tiwari, A.; Er, M.J.; Ding, W.; Lin, C.T. A review of clustering techniques and developments. *Neurocomputing* **2017**, *267*, 664–681. [CrossRef]
35. Lim, Z.Y.; Ong, L.Y.; Leow, M.C. A Review on Clustering Techniques: Creating Better User Experience for Online Roadshow. *Future Int.* **2021**, *13*, 233. [CrossRef]
36. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996; The AAAI Press: Menlo Park, CA, USA, 1996; pp. 226–231.
37. ST_ClusterDBSCAN. Available online: https://postgis.net/docs/ST_ClusterDBSCAN.html (accessed on 20 April 2022).
38. WGS 84-WGS84-World Geodetic System 1984. Available online: https://epsg.org/crs_4326/WGS-84.html (accessed on 25 September 2022).
39. WGS 84/Pseudo-Mercator. Available online: https://epsg.org/crs_3857/WGS-84-Pseudo-Mercator.html (accessed on 25 September 2022).
40. ST_Transform. Available online: https://postgis.net/docs/ST_Transform.html (accessed on 3 October 2022).
41. Picaut, J.; Fortin, N.; Bocher, E.; Petit, G. NoiseCapture Data Extraction from August 29, 2017 until August 28, 2020 (3 Years). 2021. Available online: https://research-data.ifsttar.fr/dataset.xhtml?persistentId=doi:10.25578/J5DG3W (accessed on 3 October 2022).
42. Graziuso, G.; Grimaldi, M.; Mancini, S.; Quartieri, J.; Guarnaccia, C. Crowdsourcing Data for the Elaboration of Noise Maps: A Methodological Proposal. *J. Phys. Conf. Ser.* **2020**, *1603*, 012030. [CrossRef]
43. Graziuso, G.; Mancini, S.; Francavilla, A.B.; Grimaldi, M.; Guarnaccia, C. Geo-Crowdsourced Sound Level Data in Support of the Community Facilities Planning. A Methodological Proposal. *Sustainability* **2021**, *13*, 5486. [CrossRef]
44. Sakagami, K. How did the 'state of emergency' declaration in Japan due to the COVID-19 pandemic affect the acoustic environment in a rather quiet residential area? *UCL Open Environ.* **2020**, *2*, e009. [CrossRef]

45. Dubey, R.; Bharadwaj, S.; Zafar, M.I.; Bhushan Sharma, V.; Biswas, S. Collaborative noise mapping using smartphone. *Int. Arch. Photogramm. Remote Sens. Spat. Inform. Sci.* **2020**, *XLIII-B4-2020*, 253–260. [CrossRef]
46. Njegovan, A. *Analiza Slobodnih Aplikacija za mjerenje Buke (Analysis of Free Applications for Noise Measurement)*; Technical Report; Geodetski Fakultet, Zagreb: Zagreb, Croatia, 2018.
47. Mohammed, H.M.E.H.S.; Badawy, S.S.I.; Hussien, A.I.H.; Gorgy, A.A.F. Assessment of noise pollution and its effect on patients undergoing surgeries under regional anesthesia, is it time to incorporate noise monitoring to anesthesia monitors: An observational cohort study. *Ain-Shams J. Anesthesiol.* **2020**, *12*, 20. [CrossRef]
48. Chowdhury, A.R.; Mollah, M.E.; Rahman, M.A. An efficient method for subjectively choosing parameter 'k' automatically in VDBSCAN (Varied Density Based Spatial Clustering of Applications with Noise) algorithm. In Proceedings of the 2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE), Singapore, 26–28 February 2010; Volume 1, pp. 38–41. [CrossRef]
49. Wang, W.T.; Wu, Y.L.; Tang, C.Y.; Hor, M.K. Adaptive density-based spatial clustering of applications with noise (DBSCAN) according to data. In Proceedings of the 2015 International Conference on Machine Learning and Cybernetics (ICMLC), Guangzhou, China, 12–15 July 2015; Volume 1, pp. 445–451. [CrossRef]
50. Bushra, A.A.; Yi, G. Comparative Analysis Review of Pioneering DBSCAN and Successive Density-Based Clustering Algorithms. *IEEE Acc.* **2021**, *9*, 87918–87935. [CrossRef]
51. Ankerst, M.; Breunig, M.M.; Kriegel, H.-P.; Sander, J. OPTICS: Ordering Points To Identify the Clustering Structure. *ACM SIGMOD Rec.* **1999**, *28*, 49–60. [CrossRef]
52. Lee, J.G.; Han, J.; Whang, K.Y. Trajectory Clustering: A Partition-and-Group Framework. In Proceedings of the SIGMOD '07: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, Beijing, China, 11–14 June 2007; pp. 593–604. [CrossRef]

Chapter 3

Blind calibration of environmental acoustics measurements using smartphones

MDPI

Article

Blind calibration of environmental acoustics measurements using smartphones

Ayoub Boumchich [1], Judicaël Picaut [1], Pierre Aumond[1], Arnaud Can [1], Erwan Bocher [2]

[1] Univ Gustave Eiffel, CEREMA, UMRAE, F-44344 Bouguenais, France
[2] Lab-STICC CNRS UMR 6285, IUT de Vannes, F-56017, Vannes, France
* Correspondence: judicael.picaut@univ-eiffel.fr

Abstract: Environmental noise control is a major health and social issue, particularly in urban areas. Numerous environmental policies require local authorities to draw up noise maps of their territory with the aim of establishing an inventory of the noise environment and then proposing action plans to improve its quality. In general, these maps are produced with the help of numerical simulations, which may not be sufficiently representative, for example, with regard to the temporal dynamics of noise levels. Acoustic sensor measurements are also insufficient in terms of spatial coverage. More recently, an alternative approach has been proposed, consisting of using citizens as data producers by using smartphones as tools of geo-localized acoustic measurement. However, a lack of calibration of smartphones can generate a significant bias in the results obtained. Against the classical metrological principle that would aim to calibrate any sensor beforehand for physical measurement, some have proposed mass calibration procedures, called "blind calibration". The method is based on the crossing of sensors in the same area, at the same time, which are therefore supposed to observe the same phenomenon (*i.e.*, measuring the same value). The multiple crossings of a large number of sensors at the scale of a territory, and the analysis of the relations between sensors allow to calibrate the set of sensors. In this article, we propose to adapt a blind calibration method to data from the NoiseCapture smartphone application. The method is based on the modeling of the relationships between sensors, which can be written in matrix form and can then be solved as a linear algebra problem. The behavior of the method is then tested and compared on NoiseCapture datasets, for which information on the calibration values of some smartphones is already available.

Keywords: Environmental noise, noise mapping, smartphone application, calibration

Citation: Boumchich, A.; Picaut, J.; Aumond P.; Can. A.; Bocher, E. Blind calibration of smartphone acoustics measurements. *Sensors* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

Managing environmental noise, particularly in urban areas, is a major health and social issue. Numerous environmental policies encourage local authorities to produce noise maps of their territory with the aim of establishing an inventory of the noise environment and then proposing action plans to improve its quality. This is the case, for example, with the European directive 2002/49/EC relating to the assessment and management of environmental noise.

The production of noise maps remains the most widely used tool when considering environmental policies. In general, these maps are produced using simulations, based on calculation models requiring traffic data for the calculation of acoustic emission and spatial data for the modeling of acoustic propagation. Because access to these data is sometimes complicated, and their quality is sometimes questionable, the result of the simulations only partially reflects the existing state of the sound environment. Conversely, the use of acoustic sensors arranged within noise observatories gives a more detailed and realistic image of the noise environment of an area, but the insufficient number of sensors available does not allow for covering the whole territory and producing a detailed noise map [1].

The densification of sensors through the deployment of low-cost sensor networks is an interesting alternative, but the network thus produced may prove difficult to maintain in

the long term. Although several experiments have already taken place, to our knowledge, there is no functional network of this type that can produce noise maps.

Another alternative is for citizens to become data producers themselves, using smartphones as measuring instruments, as part of a participative or crowdsourcing approach. On this subject, since the pioneering work in the early 2010s [2–4], many studies have been conducted [5,6], notably on the quality of acoustic measurements produced with a smartphone, as well as on the implementation of a participatory approach to collect data on a large scale and over the long term. Among these approaches, the one based on the NoiseCapture application is the most advanced today [7]. Since the application was released in 2017 (for Android smartphones only), a considerable amount of data has been collected, worldwide [8]. Analysis of the data revealed a wide range in the quality of the noise indicators collected, due to the measurement protocol and, in particular, the lack of acoustic calibration of the smartphones in most cases. A lack of calibration, or even a bad calibration, can indeed generate a significant bias in the measurement results. The realization of a calibration in the state of the art, from a reference device (for example, an acoustic calibrator), would normally constitute a pre-requisite for the realization of measurements, but the access to such reference devices by any citizen makes this procedure difficult to apply in practice. The proportion of calibrated smartphones in the totality of collected data is then very low, making its use for the production of noise maps more difficult.

In contrast to the classical metrological principle of calibrating any sensor for physical measurement, others have proposed so-called "blind" mass calibration procedures. The method is based on the crossing of sensors in the same area, at the same time, which are therefore supposed to observe the same phenomenon (*i.e.*, to measure the same value). The repetition of these crossings of a large number of sensors at the scale of a territory, and the analysis of the relations between sensors allow, in theory, to calibrate all the sensors. This type of blind calibration seems particularly interesting for data such as those collected by NoiseCapture, especially in urban areas, where several sensors can cross each other in the same area at equivalent time periods.

In this paper, we propose to implement a blind calibration method for uncalibrated mobile noise measurements. It is applied on NoiseCapture data, but could be generalized for any equivalent dataset. The method, described in Section 2, is based on modeling the relationships between sensors, which can be written in matrix form, and which can then be solved as a linear algebra problem. The behavior of the method, as well as a modified model, is then tested on NoiseCapture datasets for which information on the calibration values of some smartphones is available (Section 3). Finally, as an experiment, the method is applied to the dataset of the City of Rezé in France, allowing the production a "calibrated" noise map based on the collected raw data (Section 3.5). Section 4 concludes on the next challenges to deploy this method on a large variety of territories.

2. Methodology

2.1. The problem of the acoustic calibration of smartphones on a large scale

The principle of involving citizens in a participative science approach in the acoustical context is to collect massively geo-localized objective and subjective acoustic data. These data can then be used to produce noise maps for the benefit of local authorities, for example, in the context of establishing action plans to reduce noise pollution. This project can also be part of an educational [9,10] or citizen approach to raising awareness and co-construction of public policies [11,12]. Whatever the purpose of the collected data, the calibration of smartphones is an issue that is often discussed.

Several works have shown that different acoustic measurement applications installed on the same smartphone or the same application installed on different smartphones can generate differences in the measured acoustic indicators [13–15] that can reach up to nearly 30 dB compared to a reference device [16]. This can be explained in particular by the different coding of the applications as well as by hardware differences between the

smartphones. In this context, particular attention was paid to the development of the NoiseCapture application, to ensure compliance with the acoustic acquisition protocol on Android smartphones. One can expect that the dispersion of measured noise values within the NoiseCapture application are lower. However, the calibration of the application/smartphone pairs is still required to obtain acoustic results with a minimum of bias [14,16,17].

In this paper, acoustic calibration is seen as the correction of a measured sound pressure signal so that this measurement coincides with a reference signal (*i.e.*, an acoustic calibrator most of the time). This correction allows for a systematic error between the device to be calibrated and the reference device. In the simplest case, if X is the temporal sound pressure signal measured by the smartphone, then, the true value Y of the observable is related to the measured measurement X, *via* a calibration coefficient k such that:

$$Y = k \times X. \tag{1}$$

Within a smartphone application for noise measurement, the calibration consists of estimating this coefficient k, which normally takes into account all the elements of the analog-digital conversion chain, such as the correction linked to the sensitivity of the microphone and the effects of the digital discretization of the signal. Considering sound level in decibels (dB) instead of acoustic pressure, the estimated sound level L_Y can be calculated using the measured sound level L_X by the smartphone with the following relation:

$$L_Y = L_X + 20 \log k = L_X + \Delta. \tag{2}$$

Without the correction, the smartphone will produce a systematic offset (in dB) of a value equal to Δ.

In most experiments, the calibration procedure consists of evaluating the difference Δ in measurement between a smartphone and a reference device (*e.g.*, a class 1 sound level meter) and then proceeding to a correction in overall sound level, possibly A-weighted, by using an acoustic correction factor [18]. Most of the time, this correction is assumed to be a constant compared to the reference device; however, linearity problems can occur at low and high levels and in frequencies, which could justify a more adapted calibration [16,17], such as proposed by [19] for example. Instead of using reference devices, some alternative calibration methods have also been proposed, based, for example, on the measurement of a quiet sound level [14] or on the *in situ* measurement of road traffic noise [20]. In addition, if the calibration corrections are collected for different smartphone models and integrated in a reference database, the calibration of a smartphone can also be performed indirectly by searching for the corresponding calibration value in this database [14]. Nevertheless, some works have also shown possible differences between two identical models of smartphones, depending on different versions of the operating system or due to hardware changes on two generations of the same model [16]. Note also that the use of an external microphone instead of the smartphone's internal microphone can improve the accuracy of the measurement but it still requires microphone calibration [21–24].

Considering NoiseCapture, in the 2017–2020 period, around 24% of the measurement points (26% of tracks, 34% of the smartphones) have been collected after calibration. However, even though it represents a very large mass of data, the observed calibration values may call into question the quality of the calibration: 61.12% of the calibrated smartphones, for example, have calibration values higher than $\pm$15 dB, which does not seem realistic, even considering the low metrological quality of some smartphones. Finally, only specific events organized by specialists, for example with the objective of raising awareness among citizens or for research purposes, can ensure a high quality of data by considering a state-of-the-art calibration and a training of the users [18,25,26]. This is particularly the case for *NoiseCapture Party* events, which aim at collecting data during a specific event, supervised by qualified persons, generally over a short period of time and a limited spatial extent. However, such data represent only 0.6% of the data collected over the 2017–2020 period [8].

Relevant exploitation of mobile data at a large scale is therefore hampered by the heterogeneity of the collected data, mainly due to the lack or misapplication of a calibration protocol. To solve this problem, a relevant solution consists in simultaneously calibrating *a posteriori* all the collected data, including those that would have given rise to a calibration, in order to ensure total coherence between the data. In the literature, this mass calibration of data measured with mobile sensors, instead of considering the individual calibration of sensors, has led to the development of specific methodologies referred to as blind calibration, self-calibration, or re-calibration. In [27], the authors propose, for example, to take advantage of the multiple *rendez-vous* between an uncalibrated smartphone and several calibrated smartphones to estimate its bias; a consensus is then found to calibrate all the smartphones simultaneously by solving a discrete average consensus problem. Here again, the fact of having only a few reference data points limits the use of the method. On the contrary, in [28,29], the Moments Based Calibration approach does not require reference data but considers that all mobile sensors move in the same way in the whole study domain, with the same probability. The ergodicity property then simplifies the mathematical analysis of the problem; in practice, as in our case, it is however not verified since at the scale of a large territory, it is admitted that two smartphones will never meet. In [30], the calibration method does not rely on any such assumption and formulates the mutual calibration problem as a linear algebra problem whose solution relies on the resolution of a Laplacian matrix.

2.2. NoiseCapture application and database

The principle of mobile noise measurements is to collect geo-referenced acoustic data in a spatial area (figure 1). A given user starts a measurement, moves along a path, then stops the measurement. At each time step of 1 second, several acoustic indicators are calculated on the fly, recorded on the smartphone, and sent anonymously to a remote server. The transmitted data are verified and archived, and then processed in a simplified way in order to represent them in a cartographic representation. This representation takes the form of a noise map, where some acoustic indicators are aggregated on a hexagonal elementary spatial extent, the network of hexagons covering the entire globe.

Figure 1. NoiseCapture approach. Using the NoiseCapture application, a user moves along a path; each second, several noise indicators (sound level, spectrum) and other information (date/time, localization, speed, etc) are calculated. When the user stops the measurements, the data are stored within the smartphone, and, if authorized by the user, uploaded to the NoiseCapture remote server. Raw data collected by the entire NoiseCapture community is pre-processed and displayed in the form of noise maps.

2.3. Blind calibration model

2.3.1. Natural Graph Model

Among the solutions proposed in the literature for blind calibration, as a first attempt, the Natural Graph Model (NGM)-based blind calibration scheme proposed in [30] seems adapted to the mobile noise measurements, such as collected using the NoiseCapture application. This method consists of exploiting the multiple appointments of sensors at positions close in time and space (*i.e.*, in the same hexagon at a nearby time period) in order to establish mutual calibrations between sensors (figure 2)). In other words, if two smartphones simultaneously measure the same acoustic phenomenon, they should produce the same indicators (in the next development, we will say that there is a *link* between the two smartphones). Due to differences in calibration for both smartphones, this *rendez-vous* leads to the establishment of a correction factor between the two smartphones, *i.e.*, a relative calibration, which can be generalized to the scale of a network of smartphones to establish relative calibrations between devices. For a very dense sensor network, the multiple appointments create redundancy of information, which can also be exploited to improve the quality of the calibration.

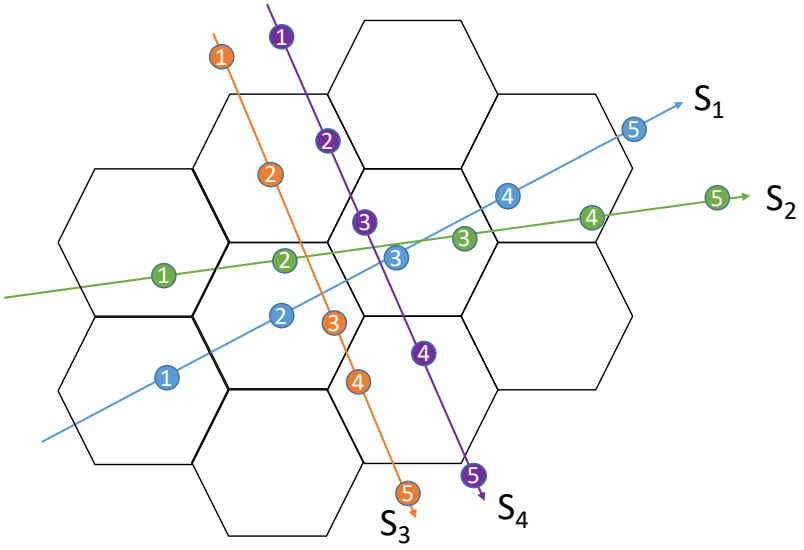


Figure 2. Principle of the blind calibration methodology applied to mobile noise measurements. During the procedure, several sensors noted that S_1, S_2, S_3, and S_4, crossed the same spatial area at the same time (t_1, t_2, t_3, t_4, t_5). In theory, these sensors should measure exactly the same acoustic event and therefore produce the same noise indicators. The path of a user is symbolized by a colored arrow; at each time step, the user is localized at a given position, symbolized by the colored circle with the time increment inside.

In the following, the original NGM methodology [30] is detailed and applied to the mobile noise measurement, using the same notations. However, we do not repeat all of the original developments so as not to make this article too long. Readers are invited to consult the original article.

Let us consider, for example, four sensors (S1, S2, S3, S4) traveling a path passing indifferently several hexagons covering a spatial extent at different times t. All the users numbered i present at the same time t in the same area define a zone Z of sensors that measure the value x of the same observable y of the event. Using the relation (2), we have [30]:

$$y = x_i + \Delta_i = x_i + d_i + n_i. \tag{3}$$

Table 1. Co-location sensor measurements based on the scenario of figure 2.

Smartphone\Zone	Z^1	Z^2	Z^3	Z^4	Z^5
S_1		x_1^2	x_1^3	x_1^4	
S_2		x_2^2	x_2^3	x_2^4	
S_3	x_3^1		x_3^3		x_3^5
S_4	x_4^1	x_4^2			x_4^5

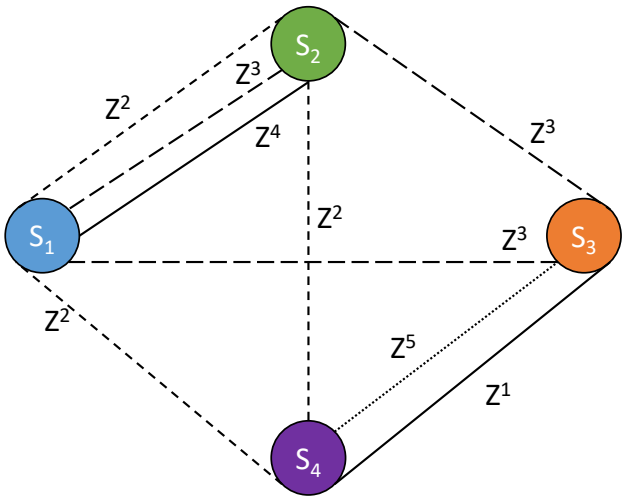


Figure 3. Network graph based on the scenario of figure 2 and Table 1.

In the relation, it is assumed that the offset Δ that is estimated for a sensor is the sum of the exact drift d related to the calibration, assumed to be systematic and stationary over time, and an error n associated with a non-predictable external effect and non-systematic, assumed to be white noise.

Thus, we can define the zone Z^α containing N^α co-located sensors (Table 1), performing the measurement x^α of the same observable y^α (*i.e.*, the true value) such that [30]:

$$\{y^\alpha = x_i^\alpha + d_i + n_i^\alpha\}_{S_i \in Z^\alpha}. \tag{4}$$

For each sensor $S_i \in Z^\alpha$, the corresponding drift d_i can thus be expressed by the other smartphone drifts d_j ($S_j \in Z^\alpha$, $S_j \neq S_i$) using the following relation [30]:

$$d_i = \frac{1}{N^\alpha - 1} \sum_{(S_j \in Z^\alpha, S_j \neq S_i)} \left(d_j + \Delta x_{ji}^\alpha + \Delta n_{ji}^\alpha \right), \tag{5}$$

with $\Delta x_{ji}^\alpha = x_j^\alpha - x_i^\alpha$ and $\Delta n_{ji}^\alpha = n_j^\alpha - n_i^\alpha$.

Since the sensor i moves along other zones and the drift d_i is stationary over time, one can derive a set of linear equations. Considering the whole set of sensors, the linear equations can be written following a matrix form [30]:

$$\mathbf{L}\vec{d} = \Delta\vec{x} + \Delta\vec{n}, \tag{6}$$

where $\mathbf{L}$ is the calibration matrix, $\vec{d}$ is the drift vector, $\Delta\vec{x}$ is the differential vector, and $\Delta\vec{n}$ is the differential white noise vector. Due to the properties of $\mathbf{L}$, the calibration matrix is the Laplacian matrix. Lastly, the authors consider two more hypotheses: (1) the differential white noise vector $\Delta\vec{n}$ is negligible when considering a large number of sensors, meaning that $\mathbf{L}\vec{d} \approx \Delta\vec{x}$; (2) the mean value of all smartphone drifts is nearly zero, which leads to the equivalent constraint $\mathbf{M_1}\vec{d} = 0$ where the elements of $\mathbf{M_1}$ are all equal to 1. Finally, the authors show that the drift vector can be obtained by resolving the following matrix inversion [30]:

$$\vec{d} = (\mathbf{L} + \mathbf{M_1})^{-1}\Delta\vec{x}. \tag{7}$$

Once the drift vector is obtained, the estimated true value in a zone can be calculated using relation (4).

2.3.2. Simple Mean Model

Instead of using the NGM methodology, one can consider a very simplified approach, the Simple Mean Model [31], also considered in the reference works for predicting the

gain calibration value for each smartphone. First, we take the average of the measurement values in each column of Table 1 to estimate the true input value of a zone. The SMM assumes a large number of sensors and estimates the true input value y^α using:

$$\hat{y}^\alpha = \frac{1}{N^\alpha} \sum_{(S_i \in Z^\alpha)} (x_i^\alpha) = \frac{1}{N^\alpha} \sum_{(S_i \in Z^\alpha)} (y^\alpha - d_i - n_i^\alpha). \tag{8}$$

Next, the drift value of a sensor can be estimated by calculating:

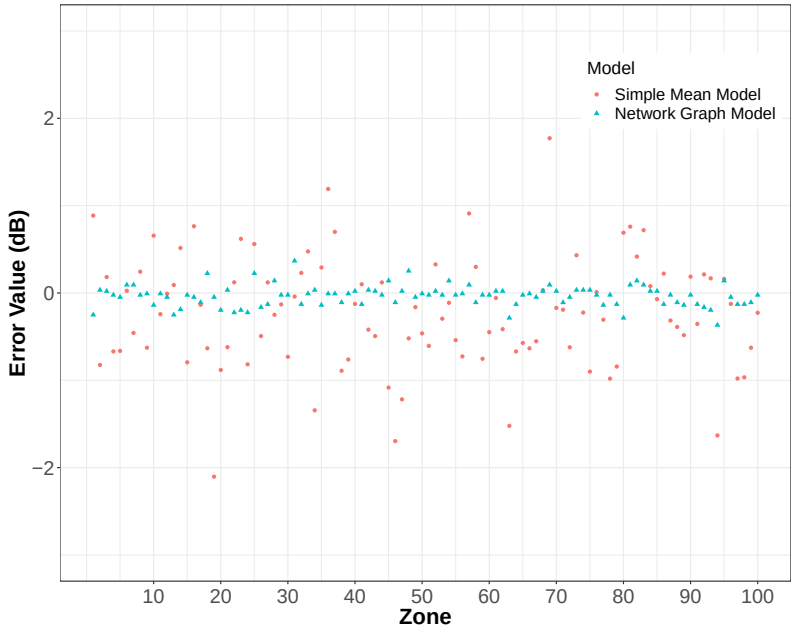
$$d_i = \hat{y}^\alpha - x_i^\alpha. \tag{9}$$

The linear equation (5) for the NGM model, plus the constraint $\sum_i \left(d_i + n_i^\alpha\right) \approx 0$ is then equivalent to the SMM. In other words, the NGM is a generalized extension of the SMM.

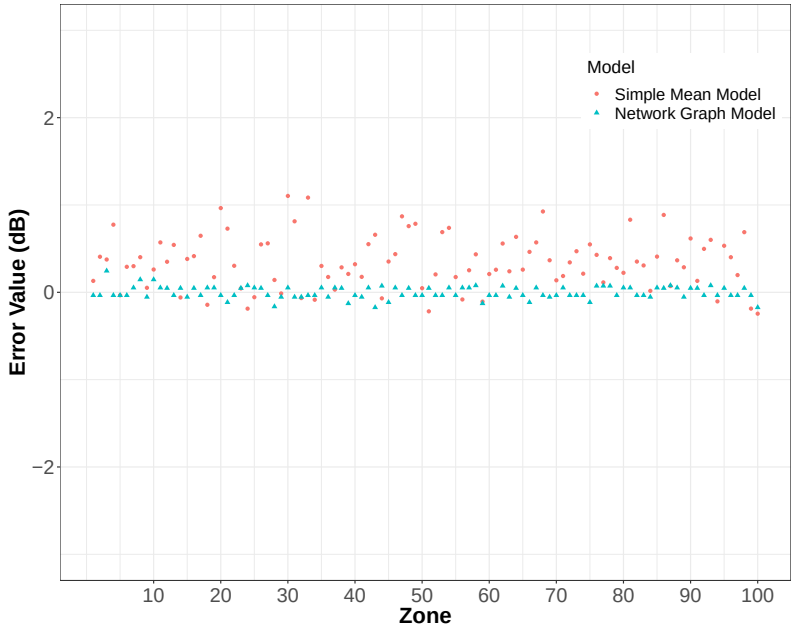
2.3.3. Validation of the NGM implementation

The NGM implementation was validated by direct comparison with the results published in the reference article [30] for a test dataset. This dataset is based on $S = 100$ simulated measurements located in $G = 100$ zones. Each measurement is simulated as the sum of the true value of the measurement y (a random number between 0 and 100 according to a uniform distribution), of a drift d (a random number according to a Gaussian distribution of variance Δ_{drift}) and of a noise n (a random number according to a Gaussian distribution of variance Δ_{noise}). The membership of a measurement in a zone is obtained randomly. Note that, at this step, this dataset has no relation to sound levels and is only used for evaluating the NGM behavior.

On the basis of this dataset, a network graph can be generated. The system (6) is then solved in order to determine the estimated value of the drift according to the relation (7) as well as the estimated value of the measurement in the corresponding zone according to the relation (4). The Mean Square Error (MSE) between the true value y and the estimated value $\hat{y}$ can then be computed in order to evaluate the model efficiency. In the reference article, the authors choose to represent the results through the link density metric $l_d \equiv 2L/[S(S-1)]$, which represents, on average, the number of times a given smartphone encounters other smartphones, with L designating the number of links. In addition to the application of the present NGM, the results obtained by the simple mean model defined at Section 2.3.2 are also represented. The results are presented in the two following figures 4 and 5, and are very similar to figures 3 and 4 of the reference article [30]. This simple comparison validates our implementation of the NGM.



(a) $l_d = 1.0$



(b) $l_d = 10.0$

Figure 4. Comparison between the NGM and the MSM: error between the true value and the estimated value, for a link density $l_d = 1.0$ and $l_d = 10.0$, with $N = 100$ smartphones in G zones.

Figure 4 illustrates the error between the estimated value and the true value of the measurement for two values of the link density ($l_d = 1.0$ and $l_d = 10.0$). As expected, when the number of links between sensors increases (when l_d increases), the estimation error decreases. Moreover, this figure shows very clearly that the NGM gives a better estimation than the SMM.

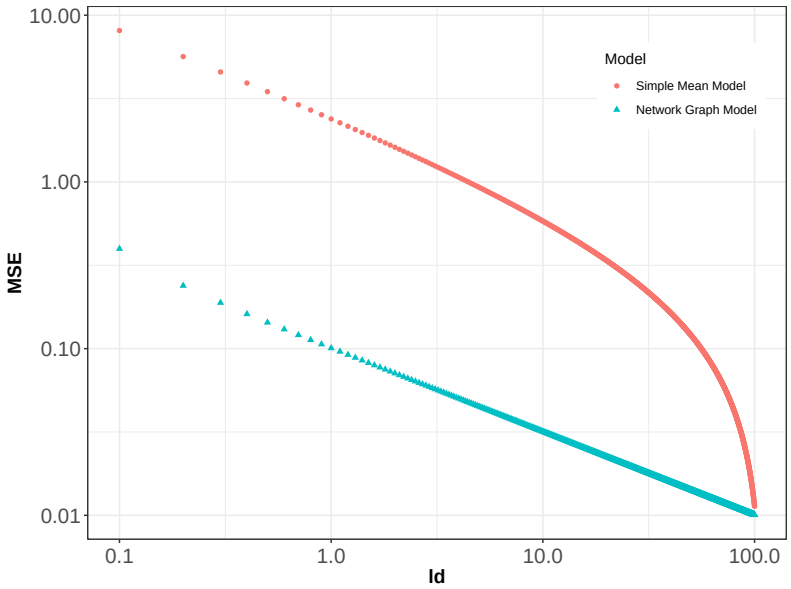


Figure 5. Comparison between the NGM and the MSM: mean square error in function of the link density l_d, with $N = 100$ sensors in G zones.

Figure 5 generalizes this conclusion by summarizing the results for several values of the link density l_d. The NGM converges quickly to the true values, even for low link densities, while the SMM requires a larger number of links to reach an equivalent level of performance.

From a practical point of view, the optimization of the results of the model requires both an increase in the number of sensors and in the link density. Understandably, the more links there are between different sensors and the higher the number of sensors, the better the results.

3. Application of the NGM to a mobile acoustic dataset

3.1. Discussion of NGM application assumptions

The development of the NGM is based on several assumptions that need to be discussed regarding its applicability to mobile acoustic data dataset. Overall, the reliability of all these assumptions, although questionable, is also supported by the results that will presented later.

3.1.1. NGM mathematical assumptions

Regarding the mathematical assumptions of the model, one can considered the following discussion:

- *The drift d of a given sensor is stationary over time.* In principle, the variation of drift over time of a professional microphone is small, especially with respect to its impact on measured noise indicators. A smartphone microphone, on the other hand, is exposed to numerous constraints that may partially modify its acoustic characteristics over time. To our knowledge, there is no published study on the acoustic monitoring of smartphones over time, at least for environmental acoustics applications, but our experience within the NoiseCapture project has not revealed any anomalies on this subject. Moreover, considering the rapid change in the smartphone fleet, the assumption of stationarity over a short or medium time period seems quite acceptable. In the event of a full deterioration of the smartphone microphone, following an accident, for example, the smartphone will become unusable for its primary function, and it is likely that it will no longer be used to collect data.
- *The average value of drifts d on all sensors is null.* The average value of all known calibration values in the NoiseCapture database, if we exclude calibration values at

zero (default value in the absence of calibration), is of the order of -0.43 dB, *i.e.*, close to zero. This hypothesis, therefore, seems globally acceptable. It is important to note first of all that this assumption is introduced by the authors to ensure the uniqueness condition of the solution of the equation (6) [30]. The assumption can therefore be discussed but is, in any case, required in the approach.

- *The noise vector $\vec{n}$ is small in front of $\vec{x}$ for a large number of sensors*. It is difficult to quantify the error introduced by external conditions or insufficient control of the measurement protocol (noise generated by the operator, bad holding of the smartphone, effect of the wind on the microphone, etc). However, on can consider that this noise is negligible in comparison with the measurement, and that it can be assimilated to a white noise.

3.1.2. Sensor definition in the context of a mobile acoustic measurement

It is also important to consider the definition of a sensor in the context of a mobile acoustic measurement. Indeed, in the present application we consider a sensor as a (smartphone model, NoiseCapture user) pair (noted later as a (smartphone,user) pair), even if several users can use the same smartphone model. This allows to consider a specific calibration for each pair: it enables to take into account the fact that two users can, for example, use the same smartphone model with a different measurement protocol, or that the same smartphone model can give rise to several technically different generations, then different calibration corrections. In the NoiseCapture approach, a given user is defined by an Universally Unique Identifier (UUID), that is associated to the corresponding smartphone.

3.1.3. Assumption of simultaneous measurements between two sensors

The major assumption of the NGM model, which requires matching data that were measured at the same time and at the same place, is very crucial and raises the question of the choice of "homogeneous" time periods for the collected data in the context of mobile acoustic dataset. In reference [32], authors consider, for example, that a measurement of 10 minutes duration can be sufficient to characterize the sound environment equivalent to a period of one hour and that "homogeneous" periods of the same day can be discriminated by measurements of 10 to 20 minutes. For the moment, the temporal distribution of the collected data with NoiseCapture is not controllable, and only the accumulation of a large number of data with time will be able to ensure, in the future, a sufficient number of data for all temporal and homogeneous reference periods of a day. At this stage, within the framework of the present work, we will consider larger time period of 1 hour or more, with the hypothesis of homogeneous sound environments.

3.2. Comparison with reference datasets: NoiseCapture Parties

In the NoiseCapture approach, specific events can be specifically organized in order to collect acoustic data over a defined spatial extent and over a given period. These events, called NoiseCapture Parties, are supervised by experts and give rise to both the acoustic calibration of smartphones and the respect of a measurement protocol. Therefore, on these reference datasets, some calibration data is available for a large number of smartphones (*i.e.*, the initial calibration value).

In this section, and as a preliminary step, we propose to apply the NGM to several reference datasets (Table 2). Each dataset is defined by an identifier `pk_party` that identifies the corresponding data in the reference database [8]. The total number of 1 second measurement points, the number of tracks (consisting of all 1 second measurement points during the same track), the measurement time period, as well as the total number of (calibrated) smartphones, are also indicated. In addition, in the framework of the application of the NGM model to these datasets, the number of links and the value of the link density l_d are also given.

Table 2. Application of the NGM on NoiseCapture Parties datasets (reference data).

pk_party	Country	Tracks	Points	Time period	Nb of Sensors	Nb of cal. Sensors	Zones	Links	l_d
10	Italy	149	15,912	11h00-12h00	12	11	479	357	5.4
13	France	100	21,470	10h00-11h00	11	11	817	508	9.2
22	France	192	17,309	12h00-19h00	23	23	403	1902	7.5
26	Italy	332	23,220	10h00-12h00	20	20	619	2526	13.3

Each event allows for the collection of data on a spatial extent defined by a set of contiguous hexagonal areas, as illustrated for example in Figure 1. The rayon of the hexagons is set to 15 m by default in the NoiseCapture approach, but the influence of this size on the behavior of the model will be discussed later in Section 3.4.

By construction, it is expected that the NGM performance will increase as the number of links between sensors increases and, therefore, as link density increases too. In view of the l_d values in the table 2 and by looking at figure 6, this hypothesis does not appear so clearly, even if the trend is globally respected.

Beyond a high l_d value, it is important that all smartphones are linked together. For example, in the case of NoiseCapture Parties N°13 and 26, one can observe that there are several groups of smartphones, with many links within each of these groups but not between smartphones from different groups (see, for example, the figure 7b for the NoiseCapture Party N°26). Conversely, the NoiseCapture Party event N°22 yields satisfactory results because most of the sensors are linked together (see figure 7a for the NoiseCapture Party N°22).

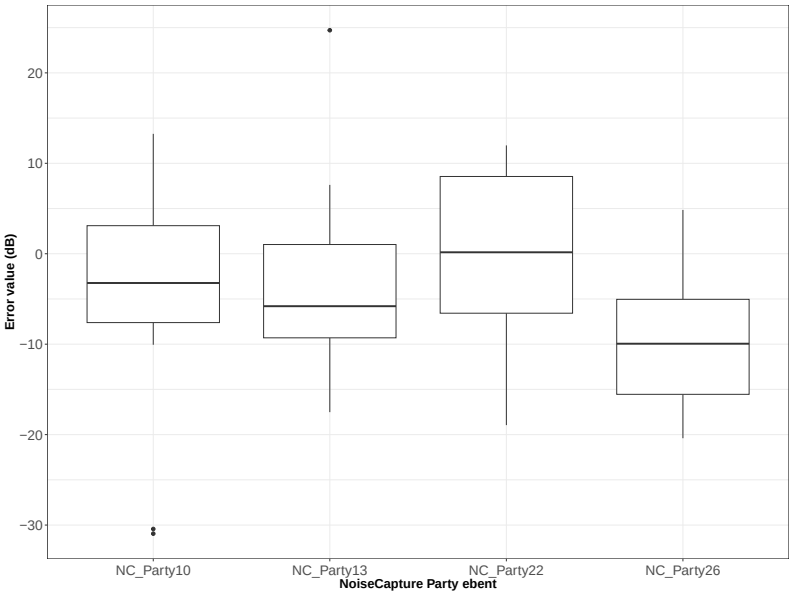
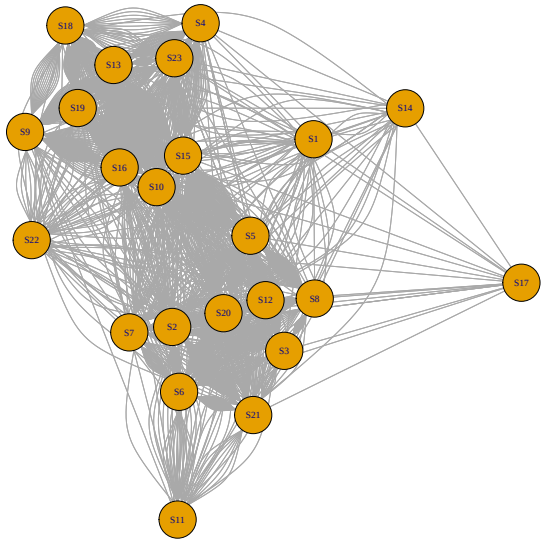
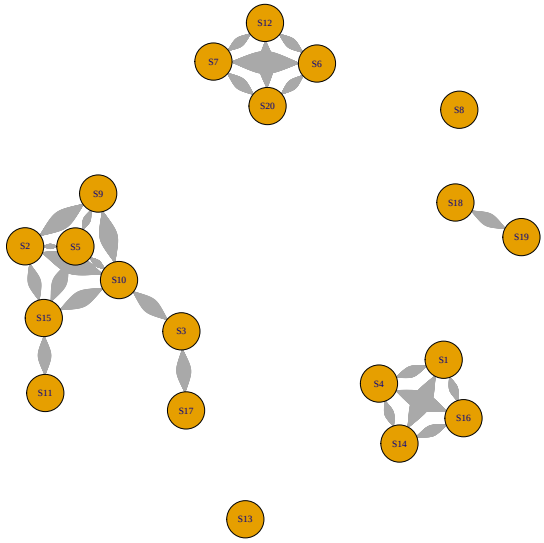


Figure 6. Error value between the estimated drift value and the initial calibration value for each calibrated smartphone used in the NoiseCapture Parties N°10, 13, 22 and 26.



(a) NoiseCapture Party N°22



(b) NoiseCapture Party N°26

Figure 7. Smartphone network graph for the NoiseCapture Party (a) N°22 with 23 linked smartphones within the same subset of data and (b) N°26 with 20 linked smartphones within 6 distinct subsets of data.

3.3. Hybrid NGM-SMM

As discussed in the last paragraph, the improvement of the NGM method relies on the increase in the number of links between smartphones and, thus, the increase in link density. Obviously, if there are too many smartphones with few links with other smartphones, then the link density will decrease and the model efficiency will also decrease. An alternative to the original approach consists in applying the NGM to the pairs (smartphone and user)

with the most links, then using the corresponding calibrated pairs to determine the drift of the other pairs by using SMM. This methodology, which can be qualified as a hybrid NGM-SMM method, makes it possible to "focus" the NGM efficiency on the most relevant pairs by optimizing the link density and to determine the calibration values for the other pairs more easily with the SMM.

This methodology has been first tested on the dataset of the NoiseCapture Party N°22. Several values of the minimal number of links per pair (smartphone,user) to be considered as a cut-off between NGM and SMM in the hybrid method were tested: from more than 1 link (this corresponds to the full NGM, with 23 (smartphone,user) pairs) to more than 140 links (12 remaining pairs), in order to evaluate the hybrid model efficiency. As expected, when the minimum number of links increases, the number of remaining (smartphone,user) pairs naturally decreases.

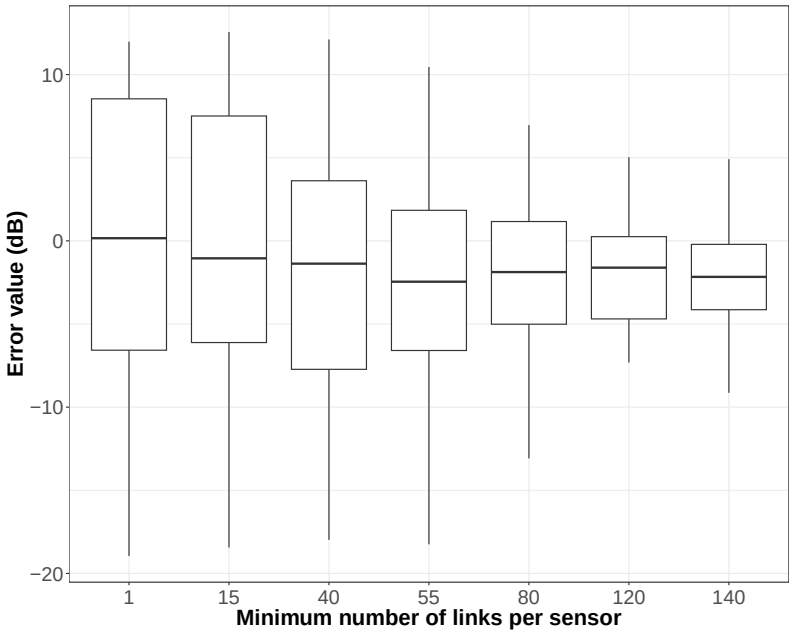


Figure 8. Application of hybrid NGM-SMM methodology on the NoiseCapture Party N°22 dataset. Error (in dB) between the estimated drift and the initial calibration of smartphones in function of the number of links between (smartphone,user) pairs from 1 (this corresponds to the full NGM, *i.e.*, the reference using the initial 23 smartphones) to 140 (12 remaining smartphones).

Figure 8 illustrates the results of this hybrid method, through the mean error between the estimated drift values and the initial smartphone calibration values. In these results, all smartphones are concerned, whether they are calibrated by the NGM method or by the SMM method. Compared to the NGM reference, we observe a better behavior of the hybrid approach (the variance decreases), and this is more so as the minimum number of links increases. This result clearly shows the contribution of the hybrid NGM-SMM method compared to the NGM method alone.

3.4. Effect of the size of the spatial area on the hybrid method

As mentioned below, the size of the spatial area may have an effect on the method's efficiency. In this paragraph, we compare the effect of the size of the hexagon on the result of the hybrid model using the NoiseCapture Party N°22 dataset. Results are detailed in table 3, in terms of mean error (in dB) between the estimated drift and the initial calibration value and in terms of uncertainty (*i.e.*, the interval between the 75 and 25 quantiles after correcting with the bias value). It should be noted that the larger the area, the fewer the links between smartphones; this explains why some of the rows in the Table 3 do not give

any results. Whatever for the mean error or for the uncertainties, the results in Table 3 shows that, for the corresponding dataset, the best compromise is obtained for a hexagonal size of 15 m. This results confirms the initial hypothesis of the NoiseCapture approach, suggesting that the sound environment may be considered as homogeneous in an area of 15 m size.

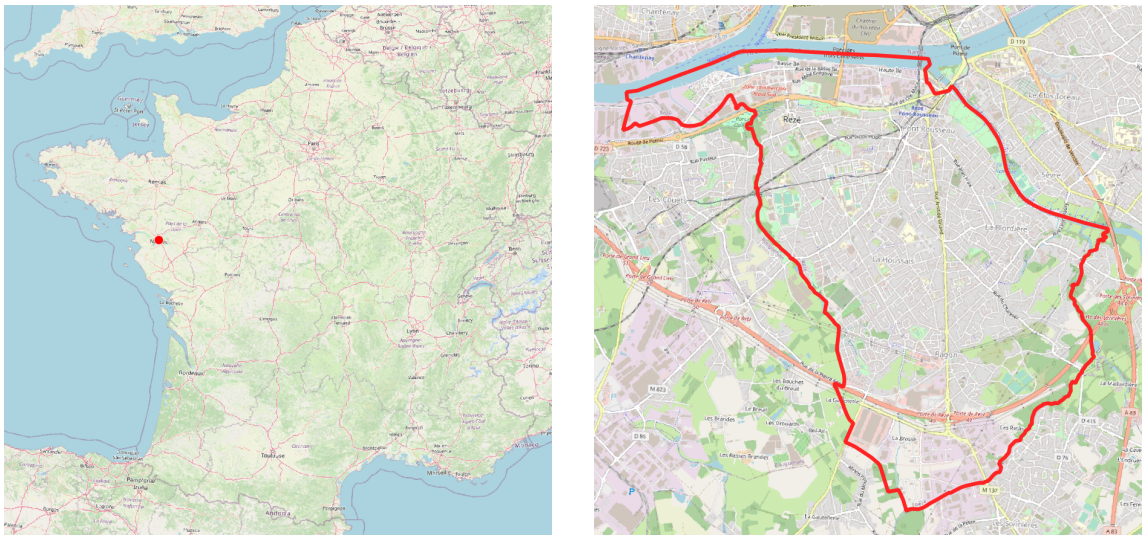
Table 3. Effect of the size of the hexagon on the hybrid method as a function of the minimum number of links per smartphone. When the minimum number of links is equal to 1, it corresponds to the reference NGM.

		Hexagon size			
Minimum number of links		10 m	15 m	30 m	50 m
1 (NGM)	Mean error	-2.33	0.36	-0.97	-1.28
	Uncertainty	±7	±8	±7	±6.5
15	Mean error	-2.33	-0.04	-0.97	-1.21
	Uncertainty	±7	±7.5	±7	±6.5
40	Mean error	-2.77	-2.13	-3.12	-3.69
	Uncertainty	±6.5	±5.5	±7.5	±7
55	Mean error	-3.86	-2.77	-3.71	-2.54
	Uncertainty	±6	±5	±5	±5
80	Mean error	-3.37	-2.34	-3.72	
	Uncertainty	±5	±4	±4.5	
120	Mean error	-3.54	-1.88		
	Uncertainty	±4	±3.2		
140	Mean error	-3.48	-1.92		
	Uncertainty	±4	±2.5		
190	Mean error	-4.76			
	Uncertainty	±3.5			

3.5. Comparison with large realistic dataset: City of Rezé (France)

3.5.1. Description of the dataset

The previous analysis is now extended to the City of Rezé, part of the Nantes metropolitan area, in France (figure 9), for which a very large amount of data has been collected, both in the context of NoiseCapture Party events (NoiseCapture Party N°2, N°9, and N°52) and by "independent" contributors. In this area, additional data have also been collected similarly to a NoiseCapture event, in the framework of the Sonorezé research project [12], but are not a part of NoiseCapture Parties. The involved area represents a surface of 13,780,000 m^2, gathering a total of 450,335 of 1 second measurement points and 2,336 tracks on 10,365 hexagons (figure 10), collected by 331 (smartphone,user) pairs with 163 different smartphone models. Reference data (NoiseCapture Parties) represents 1,877 of 1 second measurement points (0.4% of the whole dataset) and 16 tracks (0.7%), collected by 4 (smartphone,user) pairs (1.2%) and 3 different smartphone models (1.8%). Of the 331 pairs, only 134 smartphones were calibrated by users, which corresponds to 278,561 (61.9%) of 1 second calibrated measurement points and 1,529 (65.5%) calibrated tracks. The map shown in Figure 10 is obtained by averaging the sound levels at all the measurement points in each hexagon over the entire data collection period [33].



(a) Localisation of the City of Rezé (France)

(b) Boundaries of the City of Rezé

Figure 9. Localisation of the City of Rezé in France.

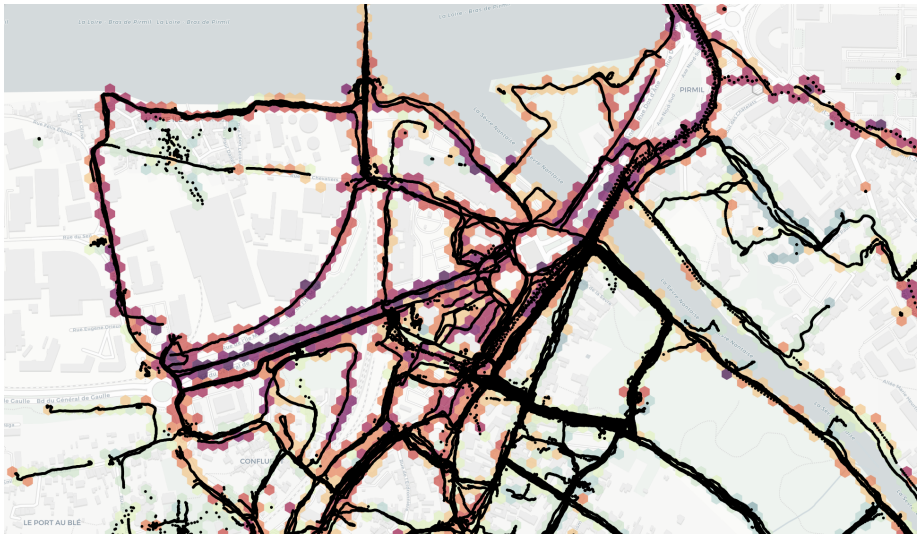


Figure 10. NoiseCapture data collected on a small part of the City of Rezé in France: measurement points and noise map (in dBA) built with raw data.

This dataset was collected over 6 years (2017–2023) at different times of the day and on different weekdays and weekends. In the present work, we have chosen to limit the application of the hybrid method to 08:00-20:00 (as a unique time period), for which a large number of data sets are available, considering that the long-term sound environment would be homogeneous during these periods. It corresponds to 315,598 of 1 second measurement points (*i.e.* 70.1% of the initial dataset) and 1,712 tracks (73.3%). Moreover, to avoid the high variation when it comes to short measurements, a more 'homogeneous' approach was considered. This approach was to remove (smartphone/user) measures that stay less than 30 s (65.3% of the initial dataset in terms of measurement points), 20 s (71.6% of the initial dataset) or 10 s (91.8% of the initial dataset), in each hexagon. These sub-datasets will be referred in the next paragraph to as 'filtered data'.

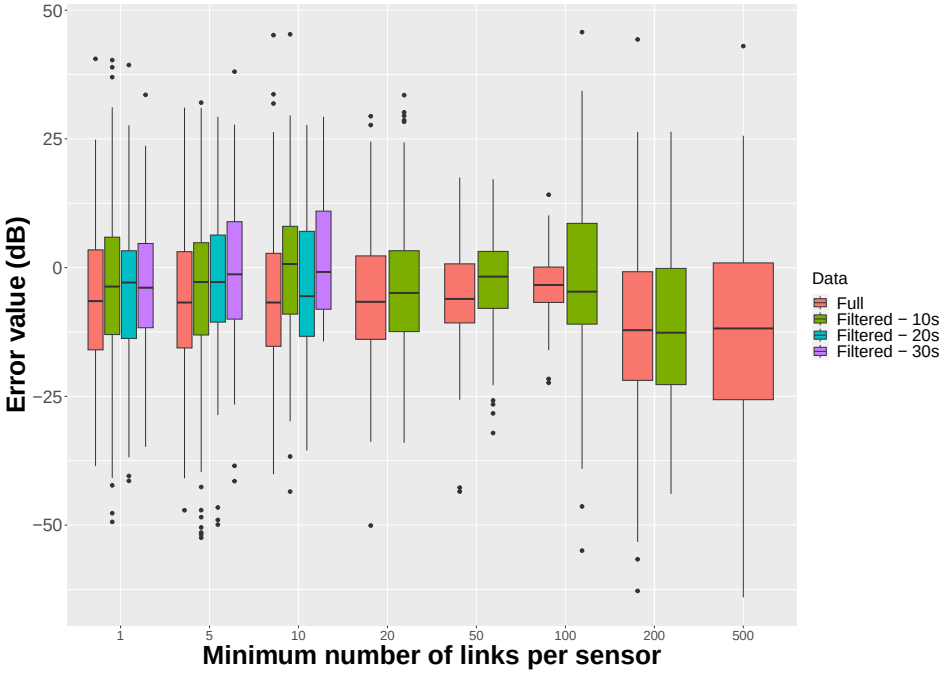


Figure 11. Application of the hybrid NGM-SMM methodology on the City of Rezé. Mean error (in dB) between the estimated drift and the initial calibration of smartphones in function of the number of links between couples (smartphone,user) from 1 (*i.e.*, the NGM reference) to 500, for each filter duration. The hybrid method is applied to both the 'full' dataset and the sub-dataset ('filtered') that correspond to a presence time of at least 30 s, 20 s and 10 s in a hexagon area.

Table 4. Errors between the gain calibration value of smartphone and the obtained drift value, in function of the number of links, in terms of mean error, median error and interquartile range (IQR), for the full dataset and the filtered data.

Minimum number of link per sensor	**1**	**5**	**10**	**20**	**50**	**100**	**200**	**500**
Full dataset								
IQR	19.4	18.6	18	16.2	11.4	6.8	21.1	26.6
Mean	-6	-6	-6	-5.4	-2.7	-3.4	-11.8	-12
Median	-6.5	-6.5	-6.5	-6	-2.5	-3.4	-12.2	-11.8
Number of (smartphone,user) pairs	201	169	155	145	94	72	37	30
Filtered dataset - 10 s								
IQR	18.9	17.9	17.1	15.7	11.1	19.6	22.6	
Mean	-4.2	-5.1	-2.1	-4	-2.9	-3.6	-9.9	
Median	-3.7	2.8	-1.3	-4.9	-1.8	-4.7	-12.6	
Number of (smartphone,user) pairs	163	131	108	85	57	26	19	
Filtered dataset - 20 s								
IQR	17	16.9	20.4					
Mean	-3.8	-3.6	-4.8					
Median	-2.9	-2.8	-5.5					
Number of (smartphone,user) pairs	101	45	18					
Filtered dataset - 30 s								
IQR	16.3	18.9	19					
Mean	-3.1	-0.7	-3.6					
Median	-3.9	-1.5	-0.8					
Number of (smartphone,user) pairs	63	20	12					

3.5.2. Time slot variability for a *rendez-vous*

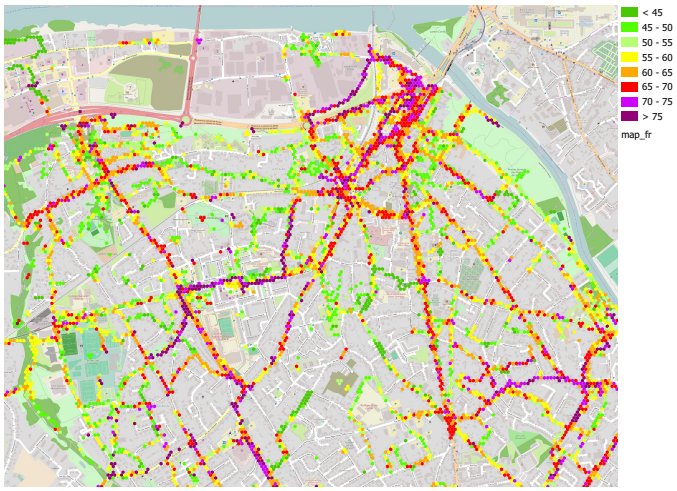
Similarly to Figure 8, Figure 11 illustrates the mean error and uncertainty of the hybrid method, applied to data collected for the city of Rezé, as a function of the minimum number of links between (smartphone,user) pairs. The approach is also applied on the sub-dataset with a minimum presence time of 30 s, 20 s and 10 s in a hexagon. Here again, the hybrid approach seems to give better results as the number of minimum links increases (the mean error decreases, as does the uncertainties). For the full dataset, the limit of improvement is reached *a priori* when the number of remaining (smartphone,user) pairs becomes insufficient. In the present case, this limit seems to appear for a number of links between 50 (94 remaining pairs) and 200 (37 remaining pairs), and is visible for a number of links equal to 100. In this case, the average error is -3.4 dB between the smartphone calibration values and the drift values obtained using the hybrid method. The uncertainty is also much lower in this situation.

When considering a minimum time of presence in an hexagon area, we observe that the mean error decreases in comparison with the full data (results for a minimum number of links of 5 and 10), while the uncertainty is quite similar and constant. For larger number of links, there are no more enough remaining (smartphone,user) pairs and the hybrid method can not give result. When comparing the results for the full dataset with the results for a time of presence of 10 s, we observe that the optimum minimum number of links is reached earlier for the filtered data. It is difficult to conclude, since there is not enough data for 20 and 30 s, but one could expect that increasing the temporal filter duration will increase the quality of the results of the hybrid method.

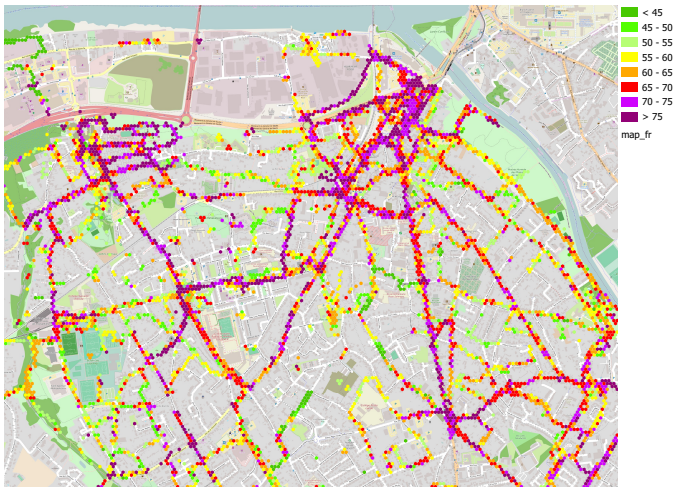
3.5.3. Qualitative Results

In addition, we now consider the application of the hybrid method on the City of Rezé, with a minimum number of links of 100, which corresponds to the best configuration for the full dataset. As an illustration, Figure 12 shows the comparison between calibrated noise maps, either by considering the individual smartphone calibration values (as measured on the smartphone), or by considering the calibration values obtained using the hybrid blind calibration method, for a small part of the City of Rezé:

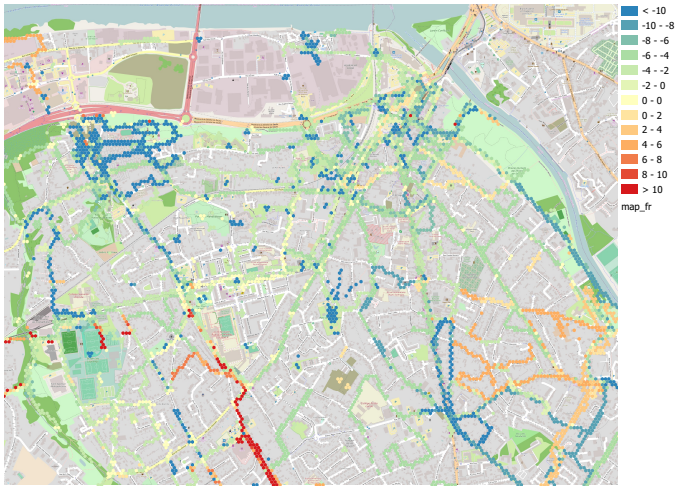
- The noise map (in dBA) produced with the initial calibration values (*Initial* noise map, figure 12a). It considers only data for smartphones with an initial calibration (134 pairs).
- The noise map (in dBA) obtained by applying the blind calibration, using the hybrid method with a minimum threshold of 100 links per smartphone, but only for the smartphones that were initially calibrated, (*Blind calibrated* noise map, figure 12b). In this case, 52.7% of smartphones were calibrated (54 using the NGM method and 53 using the SMM method), enabling 71.9% of measurement points to be corrected.
- The difference map (in dBA) between the Initial and the Blind calibrated noise maps (figure 12c); this difference map is calculated on the basis of the differences in the sound level in each hexagon. This map is completed in figure 13 by a representation of the distribution of sound level differences, as a percentage of the total number of corresponding hexagons in the whole City of Rezé (8,464 hexagons contain data on all 10,365 hexagons).



(a) Initial calibrated noise map (calibrated smartphones only)



(b) Noise map after blind calibration (calibrated smartphones only)



(c) (Initial-Blind calibration) noise map (c-a)

Figure 12. Noise maps of a part of the City of Rezé: (a) data with initial calibration (134 calibrated (smartphone,user) pairs); (b) data after applying the blind calibration on the initially calibrated smartphones only; (c) difference noise map (a-b), see also details of the differences at figure 13.

A qualitative comparison of the map produced using the calibration values initially entered by users, and the map produced after blind calibration provides some first insights into the method. The initial map (Figure 12a) cannot completely serve as a reference, because there may be errors in the calibration values entered by users. On the other hand, the blind calibration method also allows a calibration value to be estimated for smartphones that have not been calibrated, an asset that is not evaluated here.

The findings are as follows. The blind calibration method tends to result in a noise map with higher noise values in this case study. This is probably due to a bias linked to the assumption that the calibration values are centred on zero, which is not necessarily the case for a small number of smartphones. In fact, of the 134 smartphones, 10 correspond to almost half of the measurements, and in this particular case the average gain calibration given for these smartphones is negative and slightly overestimated by the method. This is visually accentuated in the neighbourhoods where few measurements contribute to the estimated value for each hexagon. This is the case for instance on the North-West, where the density of measurements is small (see Figure 6 of Reference [12]).

That said, Figure 13 shows that the dispersion of the differences between the two maps is fairly small, with a large part of the points concentrated between -8 dB and $+2$ dB, which confirms the validity of the method (this distribution would be probably centered for an input dataset whose calibration values are centered on zero). It will be interesting, in a further study, to test the behaviour of the method as a function of the input data sets, in order to adapt it to the study areas; this point is discussed in the following section.

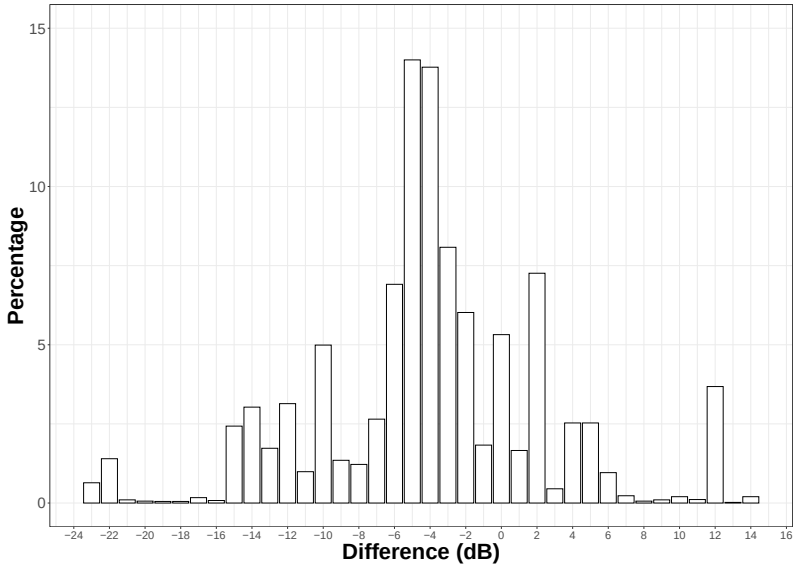


Figure 13. Distribution of the differences (in dB) of the noise level measured in each hexagonal zone, between the initial calibrated noise map and the noise map after blind calibration with the hybrid model, for the whole City of Rezé (for smartphones that were initially calibrated only). Differences are calculated for each hexagonal area (15 m) that composed the City of Rezé. *Y*-axis are given in terms of a percentage (%) of hexagonal area characterized by a given difference in dB.

4. Conclusion

Mobile noise measurements offer an alternative way of producing noise maps and collecting data on the noise environment through a participatory approach in which every citizen can become a data producer. Over and above the interest in contributing to the evaluation and development of public policies, this project raises real research questions, particularly in relation to the quality of the data produced and its use in an operational or regulatory context. Recent work on NoiseCapture data has shown a certain heterogeneity in the data collected, for example, in the absence of acoustic calibration of smartphones, a lack of expertise in the field of environmental acoustics by the contributors, or difficulties

in implementing a measurement protocol that could be shared by all contributors. Data cleaning and quality control are therefore essential stages in the relevant use of the information collected. The work presented in this article is part of this approach and was aimed more specifically at implementing a generic calibration method for all data simultaneously. It is now accepted that it will never be possible to calibrate each smartphone individually and that a mass calibration should therefore be considered instead.

Among the solutions envisaged, those based on blind calibration approaches, already tested on other studies such as for air quality measurements, are an interesting perspective. In the present article, we have exploited a method that takes advantage of the multiple *rendez-vous* of several smartphones "at the same place" and "at the same time", measuring the same acoustic observable. Written as a network graph model, the resolution of the associated matrix system can then be used to determine a mean drift for each smartphone, which is similar to a calibration correction in acoustics. The method relies on certain constraints, which are discussed in the paper, such as the temporal distribution of the data at our disposal to verify the "at the same time" condition, or more accurately at similar periods in the day, as well as the size of the spatial area to verify the "at the same place" condition. In addition, the number of *rendez-vous* a smartphone can have with others is an important factor for the quality of results. In particular, we proposed a hybrid approach to address this critical point, enabling us firstly to improve the quality of the calibration on a limited number of smartphones by using the Network Graph Model, then, secondly, using these calibrated smartphones to calibrate the other ones using a simpler approach. With regard to the first limitation, the progressive accumulation of new data over time should make it possible to obtain a more relevant temporal distribution of data. We have also observed that considering only smartphones with a minimum time of presence in each spatial area could be a way to enhance the behavior of the hybrid method. Regarding the second limitation related to the size of the spatial area, the results show that a 15 m radius spatial area was sufficient to verify a relatively homogeneous noise environment in the context of the hybrid method.

The obtained results seem particularly interesting and demonstrate the feasibility of such a blind calibration approach for mobile noise data. The method can also be improved by taking advantage of the simultaneous presence of *reference* sensors in a given area, such as noise observatories or calibrated smartphones, as suggested in [34].

The behavior of the method could also be studied on the basis of a perfectly controlled virtual mobile noise measurement dataset, as it was done in Section 2.3.3. For example, it would be possible to study in more detail the effects of time of presence in hexagons, temporal and spatial variability, minimum number of links, or presence of reference sensors. It could be useful to identify with more confidence the best conditions for applying the hybrid blind calibration method, and to adapt its parameter values to the characteristics of the dataset. A virtual mobile noise measurement dataset will also enable testing other spatial and temporal grids, replacing for instance hexagons by streets with similar traffic behavior, or refining the "at the same time" condition relying on temporal periods with similar sound levels. It will be of interest finally to test the sensibility of the method to datasets with different levels of heterogeneity in the participatory contributions, as this first analyse suggests that some main contributors might have an influence on the method if they collect a large proportion of the data and have calibration values not centered on zero.

More generally, to improve the method, it might also be useful to improve the quality of the data collected. This could be envisaged at source, by improving the mobile application to ensure better control of the measurement procedure. It can also be achieved *a posteriori*, by searching for and then removing any data collected that could be assimilated to anomalies. This can be considered for example by considering methods such as the Local Outlier Factor (LOF) [35] or the Isolation Forest [36] methods.

Author Contributions: Conceptualization, Ayoub Boumchich and Judicaël Picaut; Formal analysis, Ayoub Boumchich, Judicaël Picaut, Pierre Aumond and Arnaud Can; Funding acquisition, Judicaël Picaut; Investigation, Ayoub Boumchich, Judicaël Picaut, Pierre Aumond, Arnaud Can and Erwan

Bocher; Methodology, Ayoub Boumchich and Judicaël Picaut; Project administration, Judicaël Picaut and Erwan Bocher; Software, Ayoub Boumchich; Supervision, Judicaël Picaut and Erwan Bocher; Validation, Ayoub Boumchich, Judicaël Picaut, Pierre Aumond and Arnaud Can; Visualization, Ayoub Boumchich; Writing – original draft, Ayoub Boumchich and Judicaël Picaut; Writing – review & editing, Ayoub Boumchich, Judicaël Picaut, Pierre Aumond, Arnaud Can and Erwan Bocher.

Funding: This research was initially funded in the framework of the ENERGIC-OD Project (European Network for Redistributing Geospatial Information to user Communities - Open Data), under the ICT Policy Support Programme (ICT PSP) (CIP-ICT-PSP-2013-7) as part of the Competitiveness and Innovation Framework Programme by the European Community. A part of this research is funding by the Région Pays de La Loire grand number 2020_10361.

Data Availability Statement: The data presented in this study for the 2017-2020 period is openly available from Université Gustave Eiffel Dataverse Repository at https://doi.org/10.25578/J5DG3W.

Acknowledgments: The authors would like to thank Gwendall Petit and Nicolas Fortin, from the Environmental Acoustics Research Laboratory (UMRAE), for their help in manipulating and representing the NoiseCapture data. Map data are from OpenStreetMap licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF) (see https://www.openstreetmap.org/copyright/).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Can, A.; Dekoninck, L.; Botteldooren, D. Measurement network for urban noise assessment: Comparison of mobile measurements and spatial interpolation approaches. *Applied Acoustics* **2014**, *83*, 32–39. https://doi.org/https://doi.org/10.1016/j.apacoust.2014.03.012.
2. Maisonneuve, N.; Stevens, M.; Ochab, B. Participatory Noise Pollution Monitoring Using Mobile Phones. *Info. Pol.* **2010**, *15*, 51–71.
3. Kanjo, E. NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping. *Mobile Networks and Applications* **2010**, *15*, 562–574. https://doi.org/10.1007/s11036-009-0217-y.
4. D'Hondt, E.; Stevens, M.; Jacobs, A. Participatory noise mapping works! An evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring. *Pervasive and Mobile Computing* **2013**, *9*, 681–694. https://doi.org/10.1016/j.pmcj.2012.09.002.
5. Guillaume, G.; Can, A.; Petit, G.; Fortin, N.; Palominos, S.; Gauvreau, B.; Bocher, E.; Picaut, J. Noise mapping based on participative measurements. *Noise Mapping* **2016**, *3*, 140–156. https://doi.org/10.1515/noise-2016-0011.
6. Brambilla, G.; Pedrielli, F. Smartphone-Based Participatory Soundscape Mapping for a More Sustainable Acoustic Environment. *Sustainability* **2020**, *12*, 7899. https://doi.org/10.3390/su12197899.
7. Picaut, J.; Fortin, N.; Bocher, E.; Petit, G.; Aumond, P.; Guillaume, G. An open-science crowdsourcing approach for producing community noise maps using smartphones. *Building and Environment* **2019**, *148*, 20–33. https://doi.org/10.1016/j.buildenv.2018.10.049.
8. Picaut, J.; Boumchich, A.; Bocher, E.; Fortin, N.; Petit, G.; Aumond, P. A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *International Journal of Environmental Research and Public Health* **2021**, *18*, 7777. https://doi.org/10.3390/ijerph18157777.
9. Zipf, L.; Primack, R.B.; Rothendler, M. Citizen scientists and university students monitor noise pollution in cities and protected areas with smartphones. *PLOS ONE* **2020**, *15*, e0236785. https://doi.org/10.1371/journal.pone.0236785.
10. Guillaume, G.; Aumond, P.; Bocher, E.; Can, A.; Ecotière, D.; Fortin, N.; Foy, C.; Gauvreau, B.; Petit, G.; Picaut, J. NoiseCapture smartphone application as pedagogical support for education and public awareness. *The Journal of the Acoustical Society of America* **2022**, *151*, 3255–3265. https://doi.org/10.1121/10.0010531.
11. Lefevre, B.; Agarwal, R.; Issarny, V.; Mallet, V. Mobile crowd-sensing as a resource for contextualized urban public policies: a study using three use cases on noise and soundscape monitoring. *Cities & Health* **2021**, *5*, 179–197. https://doi.org/10.1080/23748834.2019.1617656.
12. Can, A.; Audubert, P.; Aumond, P.; Geisler, E.; Guiu, C.; Lorino, T.; Rossa, E. Framework for urban sound assessment at the city scale based on citizen action, with the smartphone application NoiseCapture as a lever for participation. *Noise Mapping* **2023**, *10*, 20220166. https://doi.org/doi:10.1515/noise-2022-0166.
13. Kardous, C.A.; Shaw, P.B. Evaluation of smartphone sound measurement applications. *The Journal of the Acoustical Society of America* **2014**, *135*, EL186–EL192. https://doi.org/10.1121/1.4865269.
14. Zhu, Y.; Li, J.; Liu, L.; Tham, C.K. iCal: Intervention-free Calibration for Measuring Noise with Smartphones. In Proceedings of the 2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS), 2015, pp. 85–91. https://doi.org/10.1109/ICPADS.2015.19.

15. Murphy, E.; King, E.A. Testing the accuracy of smartphones and sound level meter applications for measuring environmental noise. *Applied Acoustics* **2016**, *106*, 16–22. https://doi.org/10.1016/j.apacoust.2015.12.012.
16. Ventura, R.; Mallet, V.; Issarny, V.; Raverdy, P.G.; Rebhi, F. Evaluation and calibration of mobile phones for noise monitoring application. *The Journal of the Acoustical Society of America* **2017**, *142*, 3084–3093. https://doi.org/10.1121/1.5009448.
17. Nast, D.R.; Speer, W.S.; Prell, C.G.L. Sound level measurements using smartphone "apps": Useful or inaccurate? *Noise and Health* **2014**, *16*, 251. https://doi.org/10.4103/1463-1741.140495.
18. Aumond, P.; Lavandier, C.; Ribeiro, C.; Boix, E.G.; Kambona, K.; D'Hondt, E.; Delaitre, P. A study of the accuracy of mobile technology for measuring urban noise pollution in large scale participatory sensing campaigns. *Applied Acoustics* **2017**, *117*, 219–226. https://doi.org/10.1016/j.apacoust.2016.07.011.
19. Garg, S.; Lim, K.M.; Lee, H.P. An averaging method for accurately calibrating smartphone microphones for environmental noise measurement. *Applied Acoustics* **2019**, *143*, 222–228. https://doi.org/10.1016/j.apacoust.2018.08.013.
20. Aumond, P.; Can, A.; Rey Gozalo, G.; Fortin, N.; Suárez, E. Method for in situ acoustic calibration of smartphone-based sound measurement applications. *Applied Acoustics* **2020**, *166*, 107337. https://doi.org/10.1016/j.apacoust.2020.107337.
21. Kardous, C.A.; Shaw, P.B. Evaluation of smartphone sound measurement applications (apps) using external microphones—A follow-up study. *The Journal of the Acoustical Society of America* **2016**, *140*, EL327–EL333. https://doi.org/10.1121/1.4964639.
22. Roberts, B.; Kardous, C.; Neitzel, R. Improving the accuracy of smart devices to measure noise exposure. *Journal of Occupational and Environmental Hygiene* **2016**, *13*, 840–846. https://doi.org/10.1080/15459624.2016.1183014.
23. Celestina, M.; Hrovat, J.; Kardous, C.A. Smartphone-based sound level measurement apps: Evaluation of compliance with international sound level meter standards. *Applied Acoustics* **2018**, *139*, 119–128. https://doi.org/10.1016/j.apacoust.2018.04.011.
24. Celestina, M.; Kardous, C.A.; Trost, A. Smartphone-based sound level measurement apps: Evaluation of directional response. *Applied Acoustics* **2021**, *171*, 107673. https://doi.org/10.1016/j.apacoust.2020.107673.
25. Can, A.; Guillaume, G.; Picaut, J. Cross-calibration of participatory sensor networks for environmental noise mapping. *Applied Acoustics* **2016**, *110*, 99–109. https://doi.org/10.1016/j.apacoust.2016.03.013.
26. Pődör, A.; Szabó, S. Geo-tagged environmental noise measurement with smartphones: Accuracy and perspectives of crowd-sourced mapping. *Environment and Planning B: Urban Analytics and City Science* **2021**. https://doi.org/10.1177/2399808320987567.
27. Miluzzo, E.; Lane, N.D.; Campbell, A.T.; Olfati-Saber, R. CaliBree: A self-calibration system for mobile sensor networks. In Proceedings of the Distributed Computing in Sensor Systems; Nikoletseas, S.E.; Chlebus, B.S.; Johnson, D.B.; Krishnamachari, B., Eds.; Springer-Verlag Berlin: Berlin, 2008; Vol. 5067, pp. 314–331.
28. Wang, C.; Ramanathan, P.; Saluja, K.K. Moments based blind calibration in mobile sensor networks. In Proceedings of the 2008 Ieee International Conference on Communications, Proceedings, Vols 1-13; Ieee: New York, 2008; pp. 896–900. https://doi.org/10.1109/ICC.2008.176.
29. Wang, C.; Ramanathan, P.; Saluja, K.K. Blindly Calibrating Mobile Sensors Using Piecewise Linear Functions. In Proceedings of the 2009 6th Annual Ieee Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (secon 2009); Ieee: New York, 2009; pp. 216–224.
30. Lee, B.T.; Son, S.C.; Kang, K. A Blind Calibration Scheme Exploiting Mutual Calibration Relationships for a Dense Mobile Sensor Network. *Ieee Sensors Journal* **2014**, *14*, 1518–1526. https://doi.org/10.1109/JSEN.2013.2297714.
31. Whitehouse, K.; Culler, D. Calibration as Parameter Estimation in Sensor Networks. In Proceedings of the Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications; Association for Computing Machinery: New York, NY, USA, 2002; WSNA '02, p. 59–67. https://doi.org/10.1145/570738.570747.
32. Brocolini, L.; Lavandier, C.; Quoy, M.; Ribeiro, C.F. Measurements of acoustic environments for urban soundscapes: choice of homogeneous periods, optimization of durations, and selection of indicators. *The Journal of the Acoustical Society of America* **2013**, *134 1*, 813–21.
33. Bocher, E.; Petit, G.; Picaut, J.; Fortin, N.; Guillaume, G. Collaborative noise data collected from smartphones. *Data in Brief* **2017**, *14*, 498–503. https://doi.org/10.1016/j.dib.2017.07.039.
34. Dorffer, C.; Puigt, M.; Delmaire, G.; Roussel, G. Informed Nonnegative Matrix Factorization Methods for Mobile Sensor Network Calibration. *Ieee Transactions on Signal and Information Processing Over Networks* **2018**, *4*, 667–682. https://doi.org/10.1109/TSIPN.2018.2811962.
35. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: identifying density-based local outliers. *ACM SIGMOD Record* **2000**, *29*, 93–104. https://doi.org/10.1145/335191.335388.
36. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation Forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 2008, pp. 413–422. https://doi.org/10.1109/ICDM.2008.17.

Conclusion and perspectives

Conclusion and short-term perspectives

The use of a crowdsourcing approach in analyzing sound environments provides intriguing prospects due to the extensive spatial coverage and temporal dynamics offered by the collected data. The involvement of citizens in collaborative research brings an additional dimension to scientific investigations in this field. Initial concerns about the relevance of using such data for environmental purposes and evaluating noise reduction policies, health effects of noise, and perception of noise environments have been alleviated. Studies have demonstrated the efficacy of this approach, while highlighting important factors such as user engagement, critical mass of contributors, enhanced measurement accuracy, and the need for collective noise sensing sessions.

The development of the NoiseCapture application aligns perfectly with this alternative approach. Notably, the NoiseCapture approach distinguishes itself by providing a completely open-source platform, ensuring complete transparency in data collection and processing methods, and granting everyone the freedom to utilize the data. Moreover, efforts have been made to ensure the long-term sustainability of the project. These unique attributes contribute significantly to the success of the approach across various communities. Since its launch on August 29, 2017, the volume of collected data has been substantial.

While the amount of collected data is significant, exploiting the database for sound environment-related applications necessitates a thorough understanding of the data to mitigate analysis biases. The objective of the first chapter was to comprehensively review all the collected data, including its nature, content, and limitations, while identifying specific user behavior associated with the application. This analysis provided a precise framework for further data exploitation. Given the extensive data volume, it is evident that depending on the intended analysis, a considerable portion of the data may not be utilized due to its lack of relevance, completeness, or accuracy. Nevertheless, Several potential solutions to address this bias can be implemented within the NoiseCapture application, such as by using user notification to prompt them to activate their GPS and wait for improved geolocalization before starting measurements, by enhancing the user profile during application updates, by making certain supplementary information such as 'Pleasantness' and 'Tags,' mandatory, or by automatically identifying the sound sources instead of using tags.

Enhancing and controlling the data quality and measurement conditions are major prospects for improving the database. One way to do it is the use of machine learning methods, particularly supervised or semi-supervised approaches. Having say that, the utilization of such methods requires labeled data for training models. One approach to obtaining labeled data is through the organization of specific events called NC Parties. These events involve supervising contributors to ensure adherence to measurement protocols and smartphone calibration, making the collected data valuable reference data for model training. Nevertheless, the existing reference data from NC Parties alone are insufficient in number to ensure optimal learning quality, necessitating additional data in the reference database. As the NC application has expanded globally, other participants have organized NC Party-like events, which, once identified within the database, can enhance the reference database. Generally, these events generate higher spatial and temporal measurement point densities, making clustering methods well-suited for their detection.

In the second chapter, DBSCAN was applied to detect known NC Parties, successfully identifying them by carefully selecting the DBSCAN parameters, namely the measurement point search radius and the minimum number of points. In the next step, the method was applied to select countries to analyze the typology of the detected clusters. While several events similar to NC Parties were identified, varying the processing parameters also led to the detection of additional clusters without clear associations with specific events. Finally, the method was applied to the entire NC database using parameters aimed at detecting the most significant clusters, resulting in over 2000 clusters worldwide, some of which could be linked to research-published events.

It is evident that the method's effectiveness relies heavily on the DBSCAN `Eps` and `MinPts`parameters, requiring expert assistance based on expected cluster typologies and numbers. Regardless, the primary objective of building a reference database, post-processing techniques can be employed after clustering, such as merging detected clusters. Some measurements belonging to the same event may be grouped into separate clusters due to different locations or time periods. This issue can be resolved by examining the contributors of each cluster to determine if they are the same or by detecting overlapping areas between clusters. Furthermore, the reference database can be expanded by selecting the most important or relevant clusters and associating independently produced data from participants with these events. Ultimately, this approach would facilitate the creation of a larger reference database suitable for employing supervised machine learning methods to develop quality control protocols for NoiseCapture data. It may be possible in the future to enhance to performance for this approach by using a sliding temporal window instead of a fixed one. Another suggestion is to use the Density-Adaptive DBSCAN or the DA-DBSCAN methods instead [32]. As the algorithm progresses and forms a cluster, it calculates the density within the cluster. If the density is below a certain threshold, it adjusts the `Eps` value to a smaller distance and re-evaluates the points inside the cluster using this updated `Eps` value. The purpose of this adjustment is to capture clusters of varying densities accurately. By adaptively adjusting the `Eps` value based on the density within the cluster, DA-DBSCAN can handle datasets with clusters of different densities effectively. It allows for the discovery of clusters with varying local densities and provides more flexibility in identifying clusters in datasets with non-uniform density distributions. Additionally, other clustering methods, such as OPTICS [33] or trajectory-based methods like TRACLUS [34], could be explored and compared for performance enhancement.

The third chapter focuses on implementing a generic calibration method for smartphone data. Because an individual calibration for each smartphone is not feasible, we propose a mass calibration approach instead. Blind calibration methods have been explored and an approach based on smartphones *rendez-vous* at the same place and time, measuring the same observable was proposed. By formulating this as a network graph model, the matrix system can be solved to determine the average drift for each smartphone. However, the method has limitations, including temporal distribution, spatial area size, and limited availability of smartphones with multiple appointments. A hybrid approach was suggested to address these limitations and improve the calibration. The obtained results demonstrate the feasibility of blind calibration for mobile noise data, and further improvements can be made by incorporating reference sensors and removing anomalies in the data. Moreover, determining the best homogeneous period can also improve the results. Nevertheless, the usefulness of calibration may be questioned when dealing with significant amounts of data, as calibration can have minimal impact on noise maps in areas with extensive data. Integrating calibrated and uncalibrated smartphone data leads to noise maps further from reality. So, another idea could be correcting only uncalibrated smartphone, while trusting the calibration introduced by the users.

Other perspectives

Acoustic anomalies detection

Anomaly detection is a crucial aspect of utilizing the NoiseCapture database. By identifying anomalies, unwanted data can be filtered out, thereby enhancing the overall data quality. In the framework of our approach, acoustic anomalies are not considered as anomalous noise events or acoustic events such as sirens/horns or any specific sound sources, for example, that are needed to discriminate a given sound source (*i.e.* road traffic) from other sources [35, 36]. In our case, an anomaly is seen rather as a measurement that seems inconsistent within a set of measurements, and which could, for example, be associated with poor implementation of the measurement protocol or a very abnormal acoustic event in the vicinity of the user. Nevertheless, it is a crucial aspect that needs to be addressed. Among the various machine learning techniques available, DBSCAN, LOF [37], Isolation Tree [38] and Tukey's fence [39], are considered highly promising methods. However, the base approaches (DBSCAN, LOF, Isolation Tree) require careful calibration of their parameters to achieve improved results. Despite using hyperparameters, data visualization techniques, and expert knowledge, there is still a possibility of generating false positive outcomes, as shown in figure 3.1.

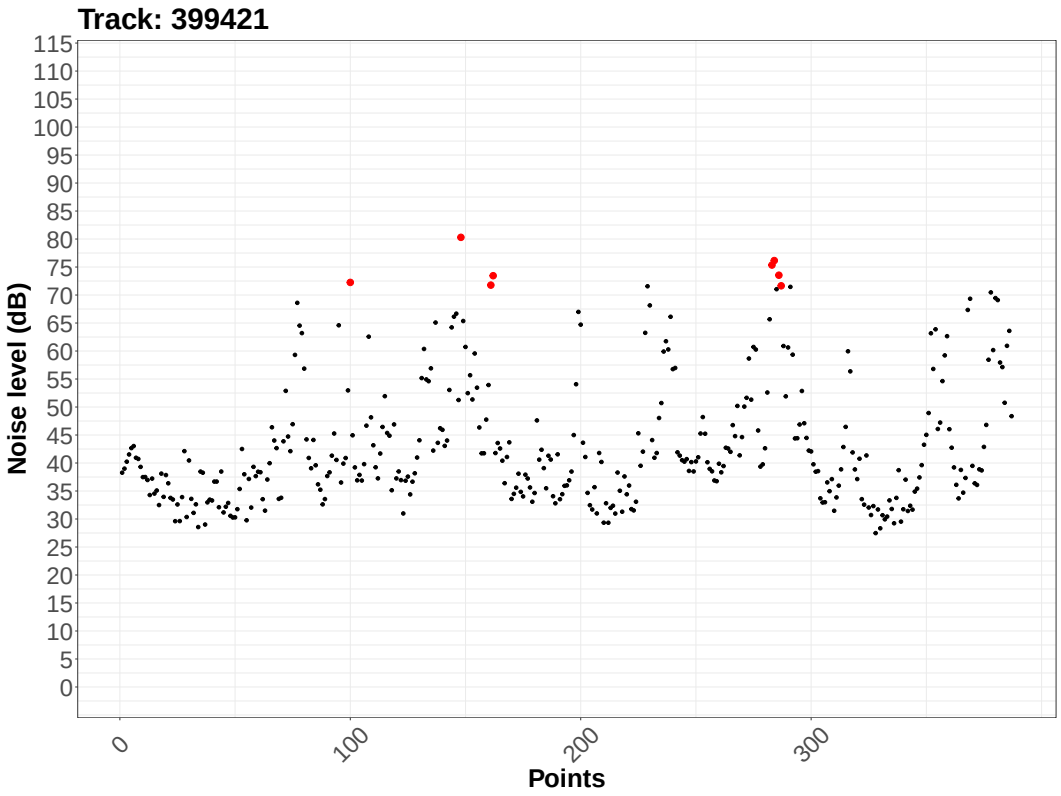


Figure 3.1: Graphical representation of the anomaly detection results utilizing the DBSCAN algorithm with parameters `Eps`=20 and `MinPts`=90%, on a NoiseCapture track (Track Id N°399421). Measurement points in red are detected as anomalies.

In other words, the approaches might incorrectly identify certain measurements as anomalies when they are actually not. To overcome this challenge, the employment of *Ensemble Learning* to compare and combine multiple approaches is important. These approaches can be thoroughly examined, compared and integrated to form an Ensemble Learning model. The model assesses the decisions made by each approach on the dataset and combines them to arrive at a final determination, as shown in table 3.1 and figure 3.2. However, the lack of information regarding the source of noise leads to additional challenges and limits the accuracy of the Ensemble Learning model's analysis. One way to around this is to introduce the temporal information (such as time of measurement, period in case of periodic distribution) as a factor while detecting the anomaly. Moreover, the evolution of acoustic indicators can be looked as audio signal, which makes the use of method such as constrained capsule network [40] also possible.

Table 3.1: Number of measurement points and corresponding percentage of anomalous points per the number of votes: 20,278 (5.86%) measurement points are found as anomalies across 1223 tracks. '1' stands for measurement points that were as detected as anomaly by one method and '4' stands for measurement points that were detected as anomaly by all the 4 methods.

	Points	
Vote	Nb	%
1	13,761	67.86
2	5,741	28.31
3	634	3.13
4	142	0.7

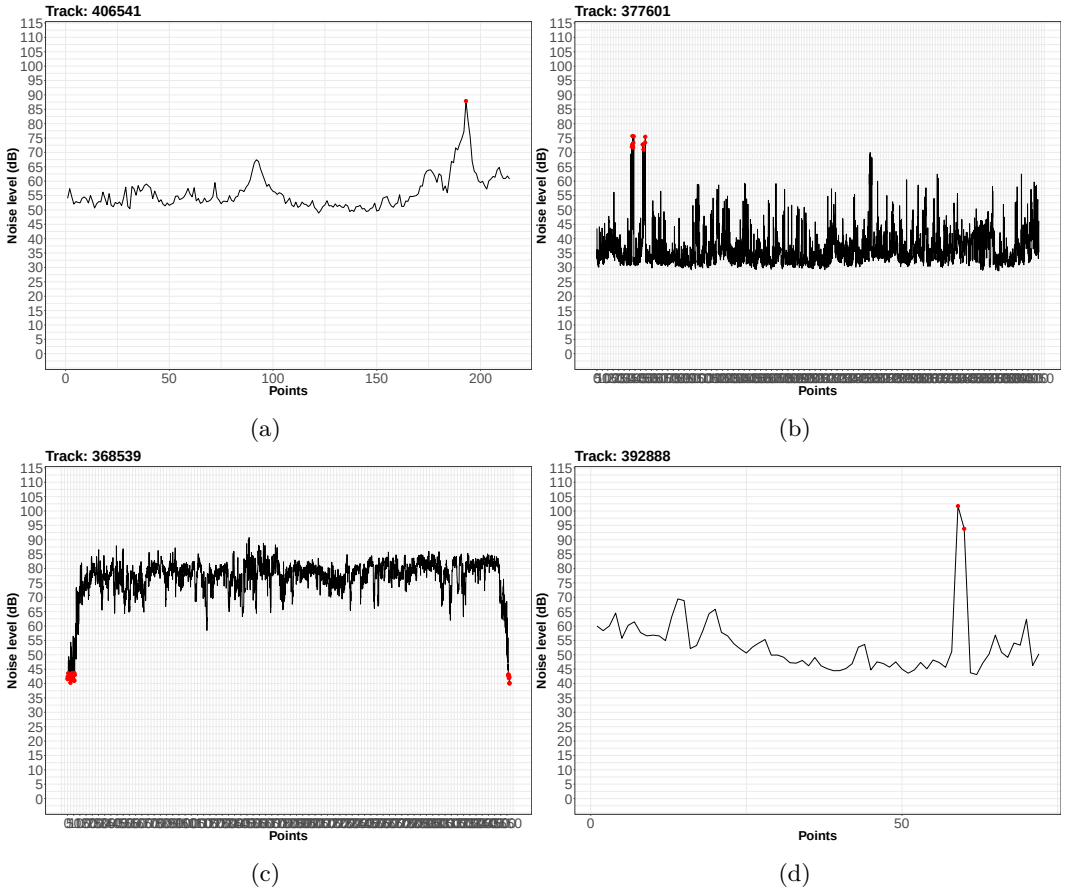


Figure 3.2: Graphical representation of 4 tracks with points (in red) that were detected as anomalies by the 4 methods.

Automatic sound sources identification

Implementing a model capable of accurately detecting the source of noise is a interesting feature to enhance the overall quality of the NoiseCapture dataset, instead of requesting users to tag the tracks. Identifying the source of the noise serves two primary purposes: firstly, it provides supplementary and valuable information to the users of the dataset, and secondly, it offers additional insights that can greatly benefit future models, particularly those related to anomaly detection.

There are several established methodologies documented in the literature that aims at identifying the origin of noise. One such method involves the utilization of frequency spectra [41, 42]. Applying this approach to the NoiseCapture dataset enables an in-depth analysis of frequencies, thereby facilitating the identification of noise sources present in each track. Another effective approach involves the implementation of Artificial Intelligence (AI) techniques for audio source identification. For instance, 'Yamnet' [43, 44] represents a sound-trained Convolutional Neural Network (CNN) capable of reporting the top-5 highest-scoring classes (predictions), averaged over all the frames of the input audio recording. Integrating this method into the NoiseCapture application enables seamless automated detection of noise sources, without necessitating manual user input.

Another viable method involves constructing a comprehensive geo-database that encompasses the spatial distribution of noise sources. This entails gathering information about the precise locations where noise is generated and recording it in a structured manner [45]. By incorporating this geo-spatial data into the NoiseCapture dataset, it becomes feasible to determine not only the spatio-temporal information of the noise but also the spatio-temporal information of its source. This integration enables the generation of more relevant and informative noise maps that provide a comprehensive understanding of the noise and its origins.

Taking this a step further, acquiring detailed information about the source of the noise can also help in the development of a predictive model for estimating noise levels in areas with missing or limited information. By leveraging the knowledge of noise sources and their characteristics, it becomes possible to extrapolate and predict the noise levels in locations where data might be incomplete. This predictive

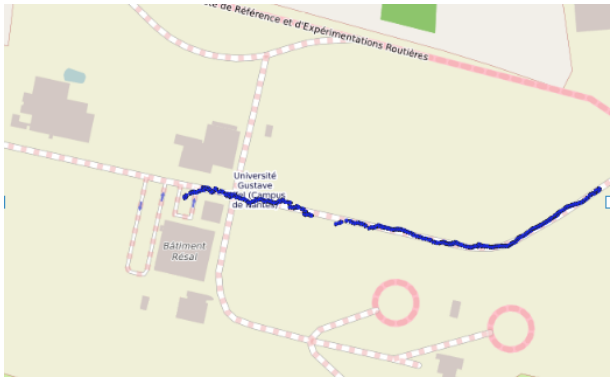
capability can greatly enhance the usability and effectiveness of the model, providing valuable insights for noise control and mitigation strategies in areas where information gaps exist.

Re-localization of measurement points

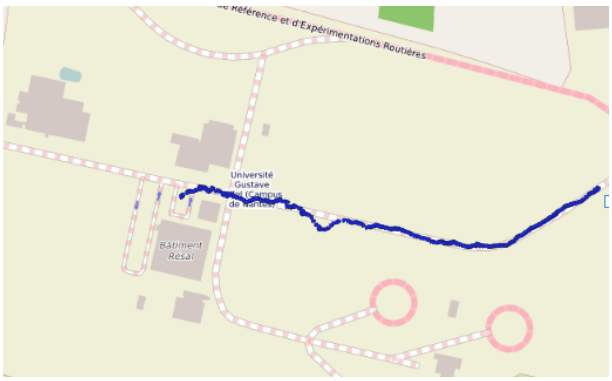
An alternative way to increase the quality of the NoiseCapture dataset involves mitigating data loss during the cleansing procedure, for example,by focusing on the data that is filtered out due to insufficient geolocalization or significant GPS inaccuracies. By performing a refinement of the dataset, eliminating entries lacking geolocalization information and exceeding a 15 m threshold, we observe a reduction of approximately 42% in measurement points and 45% in tracks data, removing to nearly half of the dataset being affected. Approximately 54% of these tracks are devoid of any geolocation information, rendering them unalterable. Conversely, the remaining 46% of tracks, though partially affected, present an opportunity for correction, thereby allowing us to preserve a greater portion of the data. 95.5% of these tracks were collected in stationary state, which makes predicting or correcting the geolocalization fairly easy. As for the remaining 4.5%, 'Dead reckoning' (DR) [46] is a navigation technique used to estimate the current position of a moving object or vehicle based on its previously known position, along with the direction and speed of its movement. This approach is particularly useful when there is no access to external positioning systems, such as GPS, or when GPS signals are unreliable or unavailable, such as in certain indoor environments or remote locations. Given that NoiseCapture measurements are acquired during walking activities, the application of Pedestrian Dead Reckoning (PDR) [47, 48] appears to be a more suitable approach. Approximately 40% of the tracks within the dataset exhibit gaps in geolocation data, either at the beginning or the end. As a consequence, predicting the starting or ending points accurately using the Pedestrian Dead Reckoning (PDR) approach becomes challenging and susceptible to bias. Consequently, our attention will be directed towards tracks with missing geolocation data in the middle section, where we can potentially address and refine the data with greater precision.

A preliminary application was conducted to assess tracks with either missing geolocalization or significant GPS accuracy values, utilizing the Pedestrian Dead Reckoning (PDR) method while calculating the bearing (direction) between each point. In Figure 3.3, two examples are presented: one illustrating the case of missing geolocalization (figures 3.3a and 3.3b), and the other demonstrating large GPS accuracy (figures 3.3c and 3.3d). Although both corrections show a linear alignment with the last point, this alignment does not consistently hold in real-world data, resulting in potential quality issues. Moreover, this approach encounters challenges when dealing with stationary data, making it difficult to accurately predict the correct localization.

To address these last limitations, alternative approaches can be employed. Time series analysis [49] can be beneficial if the data exhibits temporal patterns [50]. Spatial interpolation [51] can also offer a solution when dealing with correlated missing data points, which is often the case since these points belong to the same track and were collected by the same user during a specific period. Additionally, the application of deep learning models, such as Recurrent Neural Networks (RNNs) [52] or Convolutional Neural Networks (CNNs) [53], could be explored due to the ample amount of available data, potentially improving prediction accuracy and addressing the challenges posed by the existing method.

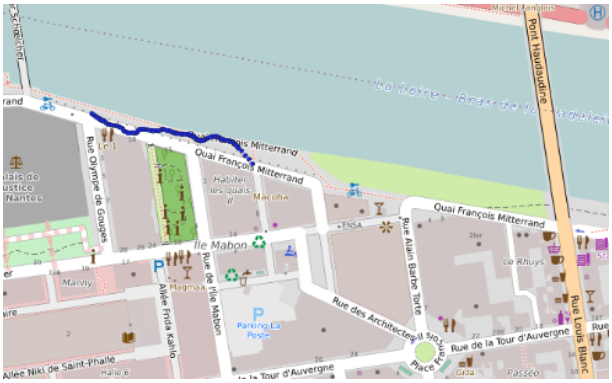


(a) Track ID N°266965 at University Gustave Eiffel, Nantes Campus (Original)



(b) Track ID N°266965 at University Gustave Eiffel, Nantes Campus (Corrected)

(c) Track ID N°53288 at Ile de Nantes (Original)



(d) Track ID N°53288 at Ile de Nantes (Corrected)

Figure 3.3: Examples of tracks that suffered form lack of geolocalization for some points (Track Id 266965) or high value for GPS accuracy (Track Id 453288). Application of Pedestrian Dead Reckoning (PDR) method to correct the wrong geolocalization of measurement points.

Appendix I

Statistical analysis of the correlation between NoiseCapture data

The following appendix presents additional details and materials related to past works realized during our internship at Gustave Eiffel University, in 2020, prior to the thesis, but on the same subject. The internship objective comprised two main components. Firstly, an in-depth exploration of the state-of-the-art definitions pertaining to noise annoyance, as well as an examination of the effects of noise annoyance and environmental noise mapping. Additionally, the study investigated how experts have utilized smartphones as sensors for collecting noise measurements. The second part of the internship objective centered around technical applications. This involved implementing a data cleaning process guided by expert knowledge, analyzing variable correlations to extract valuable insights, selecting an anomaly detection method, and utilizing the cleaned data for model training while employing separate data for testing purposes. This appendix will focus on the 'analyzing variable correlations to extract valuable insights' part.

To ascertain the impact of numerical variables on the *noise level*, a linear regression model [54,55] was fitted. The regression analysis revealed that among the variables considered, only the gain calibration variable exhibited a significant effect. The adjusted R-squared value of this model stood at 71.3%, indicating that the included variables accounted for approximately 71.3% of the variability in the noise level. Complementary techniques such as Feature Selection (*i.e.* LASSO [56]) and data visualization were employed, yielding consistent outcomes. The details of the regression results are provided in Table I.1.

Table I.1: Outputs of a fitted Regression model applied to noise levels collected with NoiseCapture.

Variables	**Estimate**	**P-value**
(Intercept)	2.96	<2e-16
Speed	1.21	0.32
Accuracy	3.4	0.18
Gain calibration	-3.05	<2e-16
Orientation	0.01	0.85

Regarding the categorical variables, a multi-way Analysis of Variance (ANOVA) [55,57] was conducted to assess their impact on the noise level. The ANOVA analysis identified that only the *time* and the *geo-spatial localization* of measurement variables exhibited a significant effect. Furthermore, the interaction between these two variables also demonstrated a significant effect. Conversely, the ANOVA analysis indicated that the device model had an effect, albeit statistically insignificant (*i.e.* a mere 0.05 dB in terms of acoustics). The adjusted R-squared value for this model was determined to be 69.9%, indicating that the included variables accounted for approximately 69.9% of the variability observed in the noise level. Complementary techniques, such as data visualization, were employed, resulting in consistent outcomes. For further details regarding the ANOVA results, please refer to Table I.2.

Table I.2: Outputs of Multi-way ANOVA applied to noise levels collected with NoiseCapture.

Variables	**Estimate**	**P-value**
(Intercept)	1.22	<0.08
Profile	0.95	0.25
Time	5.21	<2e-16
Space	3.88	<2e-16
calibration method	-1.21	<0.29
device model	0.05	0.04
Time:Space	4.62	<2e-16

Another ANOVA was conducted to investigate whether tags (*i.e.* sources of noise) have an impact on the noise level. For this analysis, only tracks with a single tag were considered, excluding the tags 'Indoor' and 'Test'. This subset accounted for 4,748,068 data points (7.96% of the total measurements) and 24,657 tracks (9.47% of the total tracks). The ANOVA analysis yielded results consistent with the previous ANOVA, indicating that time, localization, and their interaction significantly influenced the noise level. Additionally, the analysis revealed that tags (source) and the interaction between the tags and time also exhibited a significant effect. The adjusted R-squared value for this model was determined to be only 51.4%, indicating that the included variables accounted for approximately 51.4% of the variability observed in the noise level. For more detailed information regarding the ANOVA results, please refer to Table I.3.

Table I.3: Outputs of Multi-way ANOVA applied to noise levels collected with NoiseCapture.

Variables	**Estimate**	**P-value**
(Intercept)	1.22	<0.08
Tag	7.33	<2e-16
Time	4.85	<2e-16
Space	2.18	<2e-16
Tag:temp	21.97	2e-16
Tag:Space	2.55	0.07
Time:Space	3.96	<2e-16

During the analysis of the NoiseCapture dataset, several other noteworthy observations emerged. Firstly, it was observed that the device model has a discernible impact on the gain calibration value. Secondly, it was revealed that the variables of profile and calibration method are interdependent.

Based on our analysis, we can ascertain that the following variables play crucial roles in controlling the quality of the noise level: (1) temporal information, (2) spatial information, (3) gain calibration, and (4) tag (*i.e.* sound source). However, due to the insufficient availability of data with tag information, the decision was made to solely utilize the first three variables for further analysis.

Bibliography

[1] World Health Organization. Regional Office for Europe. *Burden of disease from environmental noise: quantification of healthy life years lost in Europe*. World Health Organization. Regional Office for Europe, 2011.

[2] Elise van Kempen, Maribel Casas, Göran Pershagen, Maria Foraster, Elise van Kempen, Maribel Casas, Göran Pershagen, and Maria Foraster. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Cardiovascular and Metabolic Effects: A Summary. *International Journal of Environmental Research and Public Health*, 15(2):379, February 2018.

[3] Charlotte Clark, Katarina Paunovic, Charlotte Clark, and Katarina Paunovic. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Cognition. *International Journal of Environmental Research and Public Health*, 15(2):285, February 2018.

[4] Mathias Basner, Sarah McGuire, Mathias Basner, and Sarah McGuire. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Effects on Sleep. *International Journal of Environmental Research and Public Health*, 15(3):519, March 2018.

[5] Mariola Śliwińska Kowalska, Kamil Zaborowski, Mariola Śliwińska Kowalska, and Kamil Zaborowski. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Permanent Hearing Loss and Tinnitus. *International Journal of Environmental Research and Public Health*, 14(10):1139, September 2017.

[6] Mark Nieuwenhuijsen, Gordana Ristovska, Payam Dadvand, Mark J. Nieuwenhuijsen, Gordana Ristovska, and Payam Dadvand. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Adverse Birth Outcomes. *International Journal of Environmental Research and Public Health*, 14(10):1252, October 2017.

[7] Rainer Guski, Dirk Schreckenberg, Rudolf Schuemer, Rainer Guski, Dirk Schreckenberg, and Rudolf Schuemer. WHO Environmental Noise Guidelines for the European Region: A Systematic Review on Environmental Noise and Annoyance. *International Journal of Environmental Research and Public Health*, 14(12):1539, December 2017.

[8] Alan Brown, Irene van Kamp, Alan Lex Brown, and Irene van Kamp. WHO Environmental Noise Guidelines for the European Region: A Systematic Review of Transport Noise Interventions and Their Impacts on Health. *International Journal of Environmental Research and Public Health*, 14(8):873, August 2017.

[9] Directive 2002/49/EC of the European Parliament and of the Council of 25 June 2002 relating to the assessment and management of environmental noise - Declaration by the Commission in the Conciliation Committee on the Directive relating to the assessment and management of environmental noise. http://data.europa.eu/eli/dir/2002/49/oj/eng, July 2002. (Accessed on 2021-04-26).

[10] CadnaA software. https://www.datakustik.com/products/cadnaa/cadnaa. [Online; accessed 26-June-2023].

[11] MithraSIG software - logiciel de cartographie acoustique. https://boutique.cstb.fr/sante-confort/550-mithrasig.html. [Online; accessed 25-July-2023].

[12] SoundPLAN software. https://www.soundplan.eu/en/software/. [Online; accessed 26-June-2023].

[13] Erwan Bocher, Gwenaël Guillaume, Judicaël Picaut, Gwendall Petit, and Nicolas Fortin. NoiseModelling: An Open Source GIS Based Tool to Produce Environmental Noise Maps. *ISPRS International Journal of Geo-Information*, 8(3):130, March 2019.

[14] Bruitparif. https://www.bruitparif.fr/. [Online; accessed 26-June-2023].

[15] Acoucité. https://www.acoucite.org/. [Online; accessed 26-June-2023].

[16] Francesco Asdrubali and Francesco D'Alessandro. Innovative Approaches for Noise Management in Smart Cities: a Review. *Current Pollution Reports*, 4(2):143–153, June 2018.

[17] Francesc Alias and Rosa Ma Alsina-Pages. Review of Wireless Acoustic Sensor Networks for Environmental Noise Monitoring in Smart Cities. *Journal of Sensors*, 2019:7634860, 2019.

[18] Judicael Picaut, Arnaud Can, Nicolas Fortin, Jeremy Ardouin, and Mathieu Lagrange. Low-Cost Sensors for Urban Noise Monitoring Networks-A Literature Review. *SENSORS*, 20(8):2256, April 2020.

[19] Ye Liu, Xiaoyuan Ma, Lei Shu, Qing Yang, Yu Zhang, Zhiqiang Huo, and Zhangbing Zhou. Internet of Things for Noise Mapping in Smart Cities: State of the Art and Future Directions. *IEEE Network*, 34(4):112–118, August 2020.

[20] A. L. Padilla-Ortiz, F. A. Machuca-Tzili, and D. Ibarra-Zarate. Smartphones, a tool for noise monitoring and noise mapping: an overview. *International Journal of Environmental Science and Technology*, 20(3):3521–3536, March 2023.

[21] Gwenael Guillaume, Arnaud Can, Gwendall Petit, Nicolas Fortin, Sylvain Palominos, Benoit Gauvreau, Erwan Bocher, and Judicael Picaut. Noise mapping based on participative measurements. *Noise Mapping*, 3(1):140–156, January 2016.

[22] Willian Zamora, Carlos T. Calafate, Juan-Carlos Cano, and Pietro Manzoni. A Survey on Smartphone-Based Crowdsensing Solutions. *Mobile Information Systems*, 2016, 2016.

[23] S. Santini, B. Ostermaier, and R. Adelmann. On the use of sensor nodes and mobile phones for the assessment of noise pollution levels in urban environments. In *2009 Sixth International Conference on Networked Sensing Systems (INSS)*, pages 1–8, June 2009.

[24] Rajib Kumar Rana, Chun Tung Chou, Salil S. Kanhere, Nirupama Bulusu, and Wen Hu. Ear-phone: An End-to-end Participatory Urban Noise Mapping System. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, IPSN '10, pages 105–116, New York, NY, USA, 2010. ACM.

[25] Eiman Kanjo. NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping. *Mobile Networks and Applications*, 15(4):562–574, August 2010.

[26] Nicolas Maisonneuve, Matthias Stevens, and Bartek Ochab. Participatory Noise Pollution Monitoring Using Mobile Phones. *Info. Pol.*, 15(1,2):51–71, April 2010.

[27] Noise-Planet website. Noise-Planet - Data. https://data.noise-planet.org/index.html, 2021. (Accessed on 2021-01-13).

[28] Luis Gasco, Chloé Clavel, Cesar Asensio, and Guillermo de Arcas. Beyond sound level monitoring: Exploitation of social media to gather citizens subjective response to noise. *Science of The Total Environment*, 658:69–79, March 2019.

[29] Luca Maria Aiello, Rossano Schifanella, Daniele Quercia, and Francesco Aletta. Chatty maps: constructing sound maps of urban areas from social media data. *Royal Society Open Science*, 3(3):150690.

[30] Judicaël Picaut, Ayoub Boumchich, Erwan Bocher, Nicolas Fortin, Gwendall Petit, and Pierre Aumond. A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *International Journal of Environmental Research and Public Health*, 18(15):7777, January 2021.

[31] Ayoub Boumchich, Judicaël Picaut, and Erwan Bocher. Using a Clustering Method to Detect Spatial Events in a Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment. *Sensors*, 22(22):8832, January 2022.

[32] Wei Zhou, Limin Wang, Xuming Han, Yizhang Wang, Yufei Zhang, and Zhiyao Jia. Adaptive density spatial clustering method fusing chameleon swarm algorithm. *Entropy*, 25(5), 2023.

[33] Mihael Ankerst, Markus M. Breunig, Hans-peter Kriegel, and Jörg Sander. OPTICS: Ordering Points To Identify the Clustering Structure. pages 49–60. ACM Press, 1999.

[34] Jae-Gil Lee, Jiawei Han, and Kyu-Young Whang. Trajectory clustering: A partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, SIGMOD '07, page 593–604, New York, NY, USA, 2007. Association for Computing Machinery.

[35] Rosa Ma Alsina-Pages, Francesc Alias, Joan Claudi Socoro, and Ferran Orga. Detection of Anomalous Noise Events on Low-Capacity Acoustic Nodes for Dynamic Road Traffic Noise Mapping within an Hybrid WASN. *Sensors*, 18(4), April 2018.

[36] Rosa Ma Alsina-Pagès, Roberto Benocci, Giovanni Brambilla, and Giovanni Zambon. Methods for Noise Event Detection and Assessment of the Sonic Environment by the Harmonica Index. *Applied Sciences*, 11(17):8031, January 2021.

[37] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: identifying density-based local outliers. *ACM SIGMOD Record*, 29(2):93–104, May 2000.

[38] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, December 2008. ISSN: 2374-8486.

[39] John Tukey. *Exploratory Data Analysis*. Pearson, Reading, Mass, 1st edition edition, January 1977.

[40] Nacwoo Kim, Hyunyong Lee, Jungi Lee, and Byungtak Lee. Sound-based anomaly detection using a locally constrained capsule network. In *12th International Conference on Ict Convergence (ictc 2021): Beyond the Pandemic Era with Ict Convergence Innovation*, pages 73–75, New York, 2021. Ieee.

[41] José A. Ballesteros, Ennes Sarradj, Marcos D. Fernández, Thomas Geyer, and Mª Jesús Ballesteros. Noise source identification with Beamforming in the pass-by of a car. *Applied Acoustics*, 93:106–119, June 2015.

[42] M. E. Wang and Malcolm J. Crocker. On the application of coherence techniques for source identification in a multiple noise source environment. *The Journal of the Acoustical Society of America*, 74(3):861–872, September 1983.

[43] Github - YAMNet. https://github.com/tensorflow/models/tree/master/research/audioset/yamnet. [Online; accessed 19-July-2023].

[44] Tensorflow - classification du son avec yamnet. https://www.tensorflow.org/hub/tutorials/yamnet?hl=fr. [Online; accessed 19-July-2023].

[45] Nicolas Roelandt, Pierre Aumond, and Ludovic Moisan. Crowdsourced acoustic open data analysis with foss4g tools. pages 387–393, August 2022.

[46] Ac Phillips. Low-Cost Dead Reckoning Sensors. In *Proceedings Of The National Technical Meeting Of The Institute Of Navigation: Evolution Through Integration Of Current And Emerging Systems*, pages 145–149, Washington, 1993. Inst Navigation.

[47] Yuan Wu, Hai-Bing Zhu, Qing-Xiu Du, and Shu-Ming Tang. A Survey of the Research Status of Pedestrian Dead Reckoning Systems Based on Inertial Sensors. *International Journal of Automation and Computing*, 16(1):65–83, February 2019.

[48] Xinyu Hou and Jeroen Bergmann. Pedestrian Dead Reckoning With Wearable Sensors: A Systematic Review. *IEEE Sensors Journal*, 21(1):143–152, January 2021.

[49] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time Series Analysis: Forecasting and Control*. Wiley, Hoboken, New Jersey, 5th edition edition, June 2015.

[50] Won-Chan Lee, You-Boo Jeon, Seong-Soo Han, and Chang-Sung Jeong. Position Prediction in Space System for Vehicles Using Artificial Intelligence. *Symmetry*, 14(6):1151, June 2022.

[51] Di Guo, Xiaobo Qu, Lianfen Huang, and Yan Yao. Sparsity-Based Spatial Interpolation in Wireless Sensor Networks. *Sensors*, 11(3):2385–2407, March 2011.

[52] Ryo Fujii, Jayakorn Vongkulbhisal, Ryo Hachiuma, and Hideo Saito. A Two-Block RNN-Based Trajectory Prediction From Incomplete Trajectory. *IEEE Access*, 9:56140–56151, 2021.

[53] De Zhao, Yan Zhang, Wei Wang, Xuedong Hua, and Min Yang. Car-following trajectory data imputation with adversarial convolutional neural network. *IET Intelligent Transport Systems*, 17(5):960–972, 2023.

[54] Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining. *Introduction to Linear Regression Analysis*. John Wiley & Sons, March 2021.

[55] Ronald Christensen. *Analysis of Variance, Design, and Regression: Linear Modeling for Unbalanced Data, Second Edition*. Chapman and Hall/CRC, Boca Raton, 2nd edition edition, December 2015.

[56] Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[57] Richard Arnold Johnson and Dean W. Wichern. *Applied Multivariate Statistical Analysis*. Pearson Prentice Hall, 2007.

Titre : Une base de données citoyenne pour l'évaluation du bruit dans l'environnement : du contrôle qualité des données à la production de cartes de bruit pertinentes

Mots clés : bruit environnemental, collecte citoyenne, application mobile, analyse de données, apprentissage automatique

Résumé : Ce travail de thèse s'inscrit dans le cadre de la maitrise des environnements sonores dans l'environnement. Face à cet enjeu majeur, il est usuel de recourir à des cartes de bruit, réalisées la plupart du temps par la modélisation numérique, mais au détriment d'un manque de réalisme. Une alternative récente, basée sur l'utilisation d'une application pour smartphone (NoiseCapture), plus réaliste, propose d'utiliser une approche participative citoyenne pour collecter des données, mais pose la question de la qualité des données ainsi produite et de leur utilisation à des fins opérationnelles.

A cet effet, le présent travail de thèse propose l'application de méthodes issues des Sciences des Données pour répondre à cette question.

Le travail a d'abord porté sur une analyse détaillée de la base de données NoiseCapture pour en déterminer les limites et incertitudes. Dans un second temps, et dans la perspective d'utilisation des techniques (semi-)supervisées issues des méthodes d'apprentissage, une méthode de *clustering* spatial (DB-SCAN) a été mise en œuvre afin d'identifier des données de référence dans la base de données NoiseCapture. Enfin, afin de pallier aux limites de l'étalonnage individuel des smartphones, une méthode d'étalonnage de masse, à l'aveugle, a été proposée puis appliquée sur la base de données NoiseCapture. Dans l'ensemble, la thèse a permis d'améliorer la qualité de la base de données NoiseCapture, ce qui offre des perspectives intéressantes pour l'utilisation de ces données à des fins opérationnelles.

Title : A Smartphone-Based Crowd-Sourced Database for Environmental Noise Assessment: from data quality assessment to the production of relevant noise maps

Keywords : environmental noise, crowd-sourced data, smartphone application, data analysis, machine learning

Abstract: This thesis is concerned with the control of environmental noise. Faced with this major challenge, it is common practice to use noise maps, usually produced by numerical modeling, but at the expense of a lack of realism. A recent alternative, based on the use of a smartphone application (NoiseCapture), is more realistic and proposes to use a participatory citizen approach to collect data, but raises the question of the quality of the data thus produced and its use for operational purposes.

To this end, the present thesis proposes the application of methods from Data Science to answer this question.

The work began with a detailed analysis of the NoiseCapture database to determine its limitations and uncertainties. Secondly, and with a perspective on the use of (semi-)supervised techniques derived from learning methods, a spatial clustering method (DB-SCAN) was implemented to identify reference data in the NoiseCapture database. Finally, to overcome the problem of individual smartphone miscalibration, a blind mass calibration method was proposed and applied to the NoiseCapture database. Overall, the thesis has improved the quality of the NoiseCapture database, offering interesting prospects for the use of this data for operational purposes.