ProxyFAUG: Proximity-based Fingerprint Augmentation

Grigorios G. Anagnostopoulos, Alexandros Kalousis
Geneva School of Business Administration, HES-SO Geneva, Switzerland
Email: $\{grigorios.anagnostopoulos, alexandros.kalousis\}$@hesge.ch

Abstract—**The proliferation of data-demanding machine learning methods has brought to light the necessity for methodologies which can enlarge the size of training datasets, with simple, rule-based methods. In-line with this concept, the fingerprint augmentation scheme proposed in this work aims to augment fingerprint datasets which are used to train positioning models. The proposed method utilizes fingerprints which are recorded in spacial proximity, in order to perform fingerprint augmentation, creating new fingerprints which combine the features of the original ones. The proposed method of composing the new, augmented fingerprints is inspired by the crossover and mutation operators of genetic algorithms. The ProxyFAUG method aims to improve the achievable positioning accuracy of fingerprint datasets, by introducing a rule-based, stochastic, proximity-based method of fingerprint augmentation. The performance of ProxyFAUG is evaluated in an outdoor Sigfox setting using a public dataset. The best performing published positioning method on this dataset is improved by 40% in terms of median error and 6% in terms of mean error, with the use of the augmented dataset. The analysis of the results indicate a systematic and significant performance improvement at the lower error quartiles, as indicated by the impressive improvement of the median error.**

Keywords—**Data Augmentation, Genetic Operators, IoT, Fingerprinting, Sigfox, Localization, Positioning, Reproducibility, Machine Learning, knn**

I. Introduction

The fingerprint data collection is generally regarded as a tedious and costly process. For this reason, the data collection requirement is considered as a major drawback of fingerprinting methods [1], [2], [3], [4], [5]. The amount of data collected, its geographic distribution and density are factors that play a crucial role in the determination of the achievable performance of positioning systems that build their models by training on these data. Therefore, the ability to meaningfully enrich the volume of the training datasets is a very relevant task.

The recent explosive advancement of the field of Machine Learning (ML), and particularly Deep Learning (DL), is closely related to and facilitated by the vast availability of huge volumes of data. The widely popularised methods of Deep Learning require big data volumes. At the same time, the availability of big data volumes has been one of the determinant enablers of the impressive breakthroughs of DL. Nevertheless, it is often the case that the available datasets over which deep learning models are meant to be trained upon do not offer the variety or the volume that the models would need to perform efficiently. In this context, data augmentation techniques have been widely adopted, in order to provide the data volumes that the models need to reduce their overfitting and to achieve increased accuracy and stability.

Data augmentation techniques aim at increasing the amount of available training data, either by introducing simple rule-based methods that slightly alter existing data, or by synthesizing new data with more elaborate techniques, such as generative modelling. Typical usage of rule-based data augmentation methods are those commonly performed on image datasets [6], where new samples are created by preforming simple actions on existing images, such as: cropping, flipping (horizontally or vertically), rotating, injecting noise, transforming the color distribution, or even mixing images.

Recent works have proposed fingerprinting augmentation methods [7], [8], [9], evaluating the methods' potential to provide performance improvements, and presenting promising results. Those works rely on the assumption that, at each location appearing on the training dataset, multiple receptions are available. While this assumption is met in some fingerprint datasets, it is not always the case. Such a requirement can be very restrictive, setting a large group of datasets unable to profit from the advantages of fingerprint augmentation.

In this work, the requirement of having multiple receptions recorded at the exact same ground truth location is removed, since ProxyFAUG operates with the notion of spacial proximity, utilizing fingerprints in the same vicinity. The intuition behind the proposed method is rooted on the fact that signal receptions (fingerprints) in locations of close proximity, as well as at the exact same location, might contain unmatching sets of receiving basestations, or have fluctuations on the values of the receiving basestations. Such discrepancies can be expected to appear in various combinations. Exploring a combinatorial approach of altering the existing proximal fingerprints is the path that this work has followed in order to meaningfully augment training datasets.

The rest of this paper is organized as follows. In Section II, the related work is discussed. Section III presents ProxyFAUG, the proposed fingerprint augmentation scheme, in its detail. The concise description of the experimental setup, as well as the extensive presentation and discussion of the results is

Code: https://doi.org/10.5281/zenodo.4457353
Data: https://doi.org/10.5281/zenodo.4457391

Author's accepted manuscript version. Accepted for publication in the 2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN).
The current author's accepted manuscript version is released for the purpose of meeting public availability requirements.
Please refer to the published version, for citing this work or for other possible usage.

the content of Section IV. Finally, Section V summarizes this work, presenting conclusions drawn and future directions.

II. Related Work

There has been an increasing interest on the subject of fingerprint augmentation over the last years . The existing works on fingerprint augmentation can be categorized into two main approaches. The first one consists of *rule-based fingerprint augmentation* methods, while the second one concerns the generation of synthetic fingerprints through *generative modelling* methods. In this section, we present an overview of the related work of both these approaches.

The methods of augmenting fingerprints through *rule-based* approaches, as the one proposed in the current work, utilize the concepts of adding noise or permuting received values of training fingerprints. Xiao et al. [7] propose a fingerprint augmentation method, exemplifying it using a public Wi-Fi dataset, which has multiple receptions at each training location. Assuming that at each training location a data matrix of size m by n is recorded, corresponding to m receptions for each of the n access points, the authors of [7] propose the generation of a new data matrix of equal size (m by n), which is produced upon permuting the elements of each column of the original data matrix. The authors report an improvement, in terms of mean localization error, of approximately 10%.

In a recent series of two consecutive works, Sinha et al. [8], [9], have proposed three fingerprint augmentation schemes. In the first work [8], the authors utilize an indoor dataset in a single floor setting, where a very big volume of data, of over 10000 data points, is collected. Assuming the same m by n data matrix available at each fingerprint, as in [7], they propose the generation of multiple augmented m by n data matrices, each of which differs from the original matrix simply by the subtraction of a constant number from a random reception reported on the matrix. Alternatively, the authors propose another strategy where a uniformly random value (selected inside the range of values of the certain access point) is used instead of the subtraction of a constant. In their follow-up work, Sinha et al. [9] propose a method that offers greater variety of generated fingerprints. In that work, each row of the generated matrices results by randomly selecting one of the m original receptions for each of the n access points, from the real training set. In both these works [8], [9], each m by n data matrix creates a grey scale image which feeds a CNN model. Each such image is treated as one fingerprint. The most recent work [9] yields the best results, reporting significant improvements in comparison to the case of only using the original training set.

It is noteworthy that the above presented works assume the availability of a multitude of measurements on each of the training locations. Given the fact that the collection of training data is considered a costly and timely process, which is often mentioned as the main barrier of the utilization of fingerprinting techniques, removing such a requirement can be a significant advantage. To the best of our knowledge, ProxyFAUG is the first relevant work that does not require multiple measurements per location, and which instead operates with the notion of proximity. Not requiring multiple measurements at the same location in order to perform fingerprint augmentation, allows the application of fingerprint augmentation in a wider range of fingerprint datasets, collected by different surveying techniques. The Sigfox dataset used in this work, along with other Low Power Wide Area Network (LPWAN) datasets which were published together [10], were collected with a particular surveying method. More specifically, data were collected by postal service cars which circulated around the city center of Antwerp, while carrying the relevant hardware. Such datasets could profit from proximity based augmentation, as they cannot always meet the strict requirement of multiple measurements per location.

Lastly, the utilization of *generative modelling* in the generation of synthetic fingerprints has recently started getting traction. Zou et al. [3] propose the combination of Conditional Generative Adversarial Networks (GAN) conditioned on the outcome of a Gaussian process regression, to generate fingerprints in indoor locations that were not reachable by a robot which collected the original set of fingerprints. An improvement of 33% per cent in terms of mean error is reported [3]. Similarly, GAN architectures were also utilized by two other relevant, recent works [4], [5]. Nabati et al. [4], exemplify GAN-based fingerprint generation, in a four-room (four-class) classification setting, with 250 training samples for each class. Results indicated that using only 10% of the training data to synthesize a volume of fingerprints equal to the 90% of the original dataset, can result in a synthetic dataset that achieves a performance similar to the original dataset. Li et al. [5], explore GAN-based fingerprint generation, in a limited, 7-by-7 meter setting, exemplifying the feasibility of collection effort reduction without an expense in terms of accuracy.

III. The Proposed ProxyFAUG Method

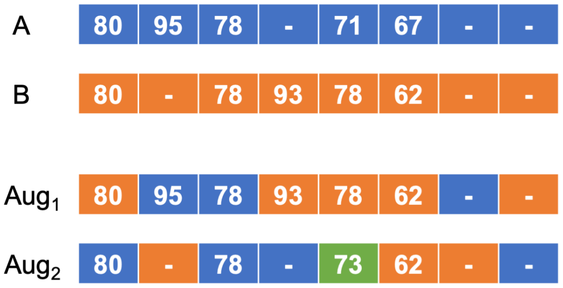
A. The ProxyFAUG Concept

The main conceptual idea behind the proposed augmentation scheme is presented in this section. It is often the case that fingerprints taken either at exactly the same location or in close proximity present noticeable differences. These differences among two fingerprints may concern either distinct sets of receiving basestations between the two fingerprints, or different measurement values from the same basestation (for instance, in the signal strength, for the RSSI case) among the two fingerprints. Such differences among fingerprints are exemplified in Figure 1. More particularly, in Figure 1, Fingerprints A and B have received the same values from the 1_{st} and 3_{rd} basestation. Basestation 2 was heard only from Fingerprint A while basestation 4 was heard only from Fingerprint B . Basestations 5 and 6 were heard in both fingerprints but with different received values. Lastly, the 7_{th} and 8_{th} basestation were not heard in either of the fingerprints.

Similar fluctuations among proximal fingerprints are a common issue in fingerprint datasets. The idea that ProxyFAUG explores is the creation of new fingerprints, as a combination of signal receptions among proximal fingerprints. The assumption is that a valid fingerprint reception in the area between proximal Fingerprints A and B could contain a combination of values of the two fingerprints. For instance, regarding the example of Figure 1, a fingerprint could reasonably have values from both Basestations 2 and 4 or from none of them. Such two cases are exemplified by the augmented Fingerprints Aug_1 and Aug_2 of Figure 1, which are produced with the ProxyFAUG operators presented bellow.

B. The ProxyFAUG Operators

ProxyFAUG proposes an adjustment of the *crossover* and *mutation* operators of genetic algorithms, in order to implement the rules that will create the augmented fingerprints, in accordance with the logic presented above. In the *crossover* operation, two real fingerprints from the training set, *the parents*, are combined to produce a new, augmented fingerprint. ProxyFAUG uses a *uniform crossover*, under which each feature (meaning the reception value of a specific basestation) is chosen from either of the two parents with an equal probability. Traditionally, the crossover operator of genetic algorithms (often referenced as *recombination* operator as well) implies that each pair of parents generates two complementary offsprings. ProxyFAUG doesn't impose this restriction and generates one augmented fingerprint per pair of parents, in order to better control the volume of the augmented dataset. Nevertheless, generating two complementary fingerprints for each pair of parents would be a valid option as well.



A	80	95	78	-	71	67	-	-
B	80	-	78	93	78	62	-	-
Aug$_1$	80	95	78	93	78	62	-	-
Aug$_2$	80	-	78	-	73	62	-	-

Fig. 1. Example of parent fingerprints A and B producing two augmented fingerprints, by the ProxyFAUG crossover and mutate operators. The augmented fingerprints indicate with the color of each cell, the origin of its value from the respective parent. A cell that underwent a mutation is indicated with green color.

The *mutation* operator, in the traditional genetic algorithm setting, concerns the arbitrary flip of a bit in a bit sequence, based on a designer-defined *mutation probability*. In ProxyFAUG, we use the *mutation* operator, in conjunction with the creation of a new, augmented fingerprint by the crossover operator. More specifically, in the augmented fingerprint resulting from the crossover operation, each feature has a probability equal to the *mutation probability* to be altered. If a feature is mutated, it is replaced by a random value uniformly selected in the range between the respective values of this feature for the two parents. For instance, in Figure 1, the value of the fifth feature of Aug_2 has been mutated. The resulting value 73 is obtained by a random sampling in the range $[71, 78]$, which is defined by the respective values of the fifth feature of the *parent* Fingerprints A and B. The reason for limiting the mutation outcome in this range, is for getting random mutations which range in plausible and meaningful values.

Since the adjusted versions of the two operators are used in conjunction and sequentially in ProxyFAUG, we will refer to their combined usage as *crossover-and-mutate*. Figure 1 exemplifies the two operators, with the creation of two, independent between them, augmented fingerprints.

The above defined operators describe the part of the augmented fingerprint that concerns the features, meaning the signals that characterize a certain location. Since in ProxyFAUG the fingerprints that are recombined in order to produce new augmented ones have not been necessarily collected at the same location, but in proximity, the location with which the augmented fingerprint is associated with remains to be defined. ProxyFAUG uses the midpoint location of the two parrent fingerprints, to characterize the resulting augmented fingerprint.

C. The ProxyFAUG Proccess

In this section, we will present the process that ProxyFAUG follows in order to augment a training set of fingerprints. The process is presented in pseudocode, in Algorithm 1. The process contains a number of parameters that determine it, and which the designer is called to tune. The parameters, which are summarized in Table I, are subsequently explained in detail.

```
Input: $training\_set, r, N, S_{max}, p_m$
Output: $augmented\_training\_set$
for $training\_point \in training\_set$ do
    Define $cluster$ of training points in range $r$ from
      $training\_point$;
    if $cluster$ $size$ $> S_{max}$ then
        randomly keep $S_{max}$ points in the $cluster$;
    end
    Define $cluster\_pairs$ containing all pairs of the
      $cluster$ elements;
    for $pair \in cluster\_pairs$ do
        Create $N$ augmented fingerprints by the
          crossover-and-mutate operator with $p_m$ on the
          $pair$;
        Add the $N$ augmented fingerprints in the
          $augmented\_training\_set$;
    end
end
Copy $training\_set$ in $augmented\_training\_set$;
Return $augmented\_training\_set$;
```

Algorithm 1: Pseudocode of the ProxyFAUG proccess.

Initially, the notion of proximity should be concretely defined by the range parameter r, which represents the maximum geographic distance between two fingerprints for them to be considered as proximal. Each fingerprint in the training set

TABLE I
THE PARAMETERS OF PROXYFAUG

Parameter	Explanation
r	**Range**: The range (maximum geometric distance) within which fingerprints will be considered to be proximal
S_{max}	**Maximum cluster size**: The maximum number of training fingerprints that will form a cluster, including the fingerprint functioning as a reference from which the $range$ parameter is measured.
N	**Crossovers per pair**: The amount of augmented fingerprints that are produced, by each pair of fingerprints selected from a cluster.
p_m	**Mutation probability**: The probability with which the value of a cell of the augmented fingerprint gets replaced by a random value. The value is sampled by a uniform distribution in the range between the respective cell values of the two parents of the augmented fingerprint.

will form, together with a set of randomly selected proximal fingerprints, a cluster. Since there might be a big amount of fingerprints in the proximity of a certain fingerprint, we define S_{max} as the maximum cluster size. With this limitation, each training fingerprint will form a cluster together with at most $S_{max} - 1$ other fingerprints that exist in a range of r meters from it. If there are up to $S_{max} - 1$ fingerprints in the range of a given fingerprint, they will compose, together with the said fingerprint, a cluster of size that is at most equal to S_{max}. Alternatively, if the are S_{max} or more fingerprints in range, a random selection takes place for forming the cluster of size S_{max}.

Once the cluster has been defined, all possible pairs of fingerprints are identified. Each pair will function as the *parents* over which the combined *crossover-and-mutation* operator, presented in III-B, will be performed. Since a pair might have multiple recombinations that can be useful, a parameter N is available to be tuned by the designer, which will determine the number of augmented fingerprints produced by a pair of *parents*, through the *crossover-and-mutate* operator. As mentioned in Section III-B, the midpoint location of the two *parent* fingerprints, is assigned as the ground truth location of each augmented fingerprint.

The process can be summarized as follows. Each fingerprint of the training set composes a cluster of a maximum size S_{max}, together with other randomly selected fingerprints of the training set, which should all lay in its vicinity, in a geometric distance that is at most equal to the range parameter r. Each possible pair of fingerprints of a cluster produces an amount of augmented fingerprints equal to the parameter N, through the *crossover-and-mutate* operator.

The volume of the resulting augmented dataset, as well as its quality, will depend on the selected parameter values. We will now calculate the maximum size that the augmented training set may have, to obtain an intuition about the impact of the selected parameter values on the resulting dataset size. Let M be the size of the training set. Assume that the range parameter r and the dataset's spatial distribution is such that there exist S_{max} proximal fingerprints for each of the training fingerprints. Then, M clusters of size S_{max} will be formed. Each cluster will create an amount of pairs of *parents* equal to the number of combinations of S_{max} objects taken 2 at a time. Moreover, each of these pairs will create N augmented fingerprints. Overall, the size of the augmented training set M_{Aug}, is upper bounded by the term described in Inequality 1.

$$M_{Aug} \leq M + M\binom{S_{max}}{2}N = M + M\frac{S_{max}!}{2!(S_{max}-2)!}N \quad (1)$$

From the above formula, it can be concluded that parameter N shows a linear relation with the resulting volume of the augmented training set. On the contrary, the maximum cluster size parameter S_{max} may affect the size of the augmented training set with a factorial term, and therefore, should be used with caution.

IV. EXPERIMENTATION AND RESULTS

A. Setting Selection and Motivation

In this section, we will investigate the effect of the ProxyFAUG data augmentation scheme on the performance of a positioning system, utilizing a public dataset of a real-world Sigfox deployment [10]. The Sigfox dataset used in this study has been collected in the urban area of Antwerp, in Belgium. Datasets of LPWANs, such as this Sigfox dataset, constitute appropriate test-beds for the ProxyFAUG augmentation scheme for following reasons.

Firstly, each recorded fingerprint contains the RSSI values of the transmitted signal as received by each of the 84 basestations that are present in the dataset, plus the spatial ground truth of the signal's transmission location, as estimated by a GPS device. The fact that an estimate, which is subject to error, is used as ground truth introduces a certain bias. Since the typical localization error of GPS is of a lower order than that of the Sigfox localization, its selection as a ground truth is acceptable [10]. Having clarified this fact, we proceed with our assumption. The uncertainty of the actual location of the recorded fingerprints facilitates us to make the assumption that, even if fingerprints are not assigned to the exact same locations but in close proximity (based on their GPS-defined ground truth) they could be characterized by similar signal receptions.

Secondly, such big urban datasets are often characterized by an uneven density of recorded fingerprints. Spatial zones with low density of fingerprints might not have enough variance in the training set so as to correctly match a new signal reception with one of the fingerprints of the zone. Moreover, since a big amount of basestations is present, it is often the case that distinct sets of basestations are present in fingerprints collected at the same (or in proximal) locations. One could assume that the numerous permutations of the sets of basesations that are present in these fingerprints could constitute valid potential fingerprints, as described in SectionIII-A and exemplified in Figure 1. The possibility to augment fingerprints in the way proposed by ProxyFAUG can increase the variance of the dataset and enhance the accuracy of a positioning system. Conceptually, ProxyFAUG aims to fill in the gaps of the data collection, an action which seems appropriate in this kind of datasets.

Lastly, the big volume of data of such datasets, comes hand in glove with the intention of ProxyFAUG to enrich the training set with multiple valid permutations of recorded fingerprints, so as to create a dense mesh of points in the signal space, that would allow new receptions to be correctly affiliated with fingerprints that are close to the actual location under question. Having presented the intuition behind the setting selection of the mentioned Sigfox dataset for exemplifying and testing ProxyFAUD, we proceed to the experimentation part.

B. Tuning

In order to evaluate the influence of the ProxyFAUG augmentation scheme on a positioning system, we will compare the performance of the same fingerprinting method in two cases: in the case of only using the original training set, against the case where the augmented training set is used. As the positioning method, we will use the one that, to the best of our knowledge, is the best performing published method dealing with the given dataset, presented by Anagnostopoulos and Kalousis [11]. As suggested in [11], the dataset is initially preprocessed by the $powed$ transformation, which has been initially introduced by Torres-Sospedra et al. [12], with the parameter β of the $powed$ transform set to 2.6. Moreover, the arbitrary value -200 which was set in the original dataset in replacement of the out-of-range missing values is replaced by the experimental minimum received RSSI value of the training set, that is -157. Lastly, the Bray-Curtis dissimilarity is used as a distance metric in a kNN setting, were $k = 6$ is set. It is worth noting that the same train / validation / test set split used in [11], which is publicly available [13], has been reused in the current work.

TABLE II
THE PROXYFAUG PARAMETERS SELECTED FOR THIS SETTING

Parameter	Explanation
r	20 meters
S_{max}	2
N	8
p_m	0.3

Upon experimentation with the ProxyFAUG parameters, the values reported in Table II were selected for the given setting. The selected parameter values suggest that each training fingerprint is coupled with another training fingerprint that is in a range of 20 meters (if such a fingerprint exists). Each such couple of fingerprints, is considered a cluster of $S_{max} = 2$. The two fingerprints of the cluster, are the only pair of *parents* of the cluster. Each pair creates $N = 8$ augmented fingerprints, by the *crossover-and-mutate* operator, with a mutation probability of $p_m = 0.3$. After the augmentation process, the augmented training set is composed of 68127 training points, as opposed to the 10063 points of the original training set.

Since the landscape of the training set has undegone a significant change with the augmentation process, re-evaluating the hyperparameter selection of the positioning system would be appropriate. Figure 2 presents the mean and median error on the validation set (with the blue and red continuous line respectively) for various values of k, for the case of training on the augmented training set. The dashed lines indicate the level of the performance of the optimal setting of the model trained on the original training set, as proposed in [11].

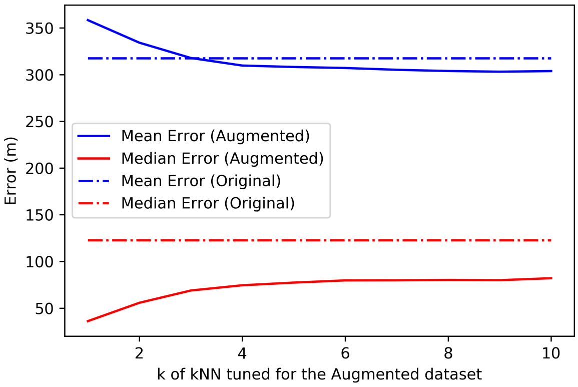


Fig. 2. Mean and median localization error on the validation set (in blue and red solid lines respectively) for various value of the k parameter, for the case of using the augmented training set. The constant value of the best performance on the validation set of the model trained with the original training set is indicated by the dotted lines.

There appears to be a trade-off between the mean and the median error, along the axis representing the amount of k Nearest Neighbours used for the position estimation. The higher values of mean error for low k values can be explained by the fact that, in zones of sparse data presence and for estimates of high localization error, averaging the location of many neighbours may reduce the error. Conversely, the fingerprints of the low quartiles are greatly benefited by the profusion of adequate neighbours, produced by the augmentation process. This fact is highlighted in Figure 2, which depicts the trend of the median error decreasing as lower values of k are evaluated. The median error goes as down as 36 meters (71% improvement) for the case of $k = 1$, though this setting increases the mean error to 358 meters (13% deterioration). The value of $k = 6$ used for the optimal setting of the original dataset appears to handle well the trade-off, improving both the mean and the median error, and will be used for training on the augmented dataset as well. Therefore, having solidified the hyperparameter setting by evaluating the performance on the validation set, we can proceed to report an unbiased evaluation of the performance on the testing set.

C. *Performance*

The performance of the positioning system for the two cases of using the original and the augmented dataset is presented in Table III and Figure 3. Table III reports the mean, median and 75th percentile of error, on the validation and the test set, and for two different training sets: the original and the augmented one. While for the mean and the 75th percentile of error the relative improvement is small, ranging around 6%, the improvement in terms of median error is impressive. More specifically, the use of the augmented dataset improves the median of the test set error by 40%.

TABLE III
LOCALIZATION ERROR COMPARISON BETWEEN THE ORIGINAL AND THE AUGMENTED TRAINING SET IN TERMS OF VALIDATION AND TEST ERROR

Training set	Validation Set Error			Test Set Error		
	mean	**median**	**75th**	**mean**	**median**	**75th**
Original	318	123	336	298	108	319
Augmented	307	80	324	280	65	300
Improvement	**3%**	**35%**	**4%**	**6%**	**40%**	**6%**

The discrepancy of the relative improvement between the mean and median error, is due to the fact that the ProxyFAUG augmentation appears to significantly improve the low quartiles of error in the studied setting. Moreover, the median error, as a metric, is resilient to the presence of outliers of high error that exist when using either of the two training sets. On the contrary, the mean error is mildly affected by the significant improvement of the low ranging errors, as the fewer cases of very high values of error affect it disproportionally. The significant performance improvement of ProxyFAUG in the lower quartiles of error is highlighted in the Cumulative Distribution Function (CDF) of the test error, of Figure 3.

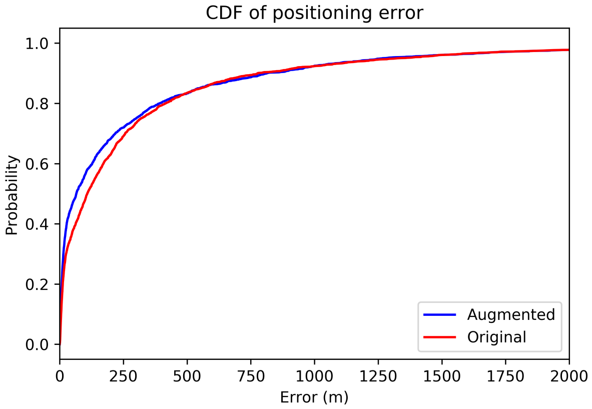


Fig. 3. The CDF of the test error, for the cases of using the original and the augmented training sets respectively.

V. CONCLUSIONS AND FUTURE WORK

In this work, we presented ProxyFAUG, a novel, concise methodology of proximity-based fingerprint augmentation. Unlike other common approaches of fingeprint augmentation, ProxyFAUG does not require multiple signal recordings at each training location, a fact that broadens the set of fingerprint datasets in which the augmentation method is applicable. ProxyFAUG proposes a set of parameters, which can allow the system designer to adapt the augmentation method to the particularities of the dataset and to the system requirements in terms of acceptable training data volume, complexity and achievable localization accuracy.

The results of the tests of this work, performed in a public, real-world dataset, highlight the significant performance improvements that are achievable with ProxyFAUG. More particularly, ProxyFAUG offers systematic performance improvements in the low quartiles of error, indicated by the 40% reduction of the median error in the independent test set, in comparison to the best performing published method on the same dataset. The CDF of error, depicted in Figure 3, highlights a clear performance discrepancy between the augmented and the original dataset.

Moreover, a significant difference in the relative improvement between the median and mean error has been observed. A relevant point is the fact that the outliers of high error, that exist when using either the original or the augmented dataset, greatly affect the mean error, while they don't have the same impact on the median metric. In addition, it was observed that ProxyFAUG provides significant and systematic performance improvements in the lower error quartiles, which may be witnessed in the CDF and the median error improvement.

Overall, ProxyFAUG presents a generic parametric method, which allows the designer to appropriately tune its functionality. As a future work, we plan to explore the possibility of alternative, dynamic versions of ProxyFAUG, in which the parameter values used will automatically adjust to the spatial characteristics of the training sets. For instance, the different spatial density of the training sets in different zones, could be used to dynamically adjust the ProxyFAUG parameters values, of parameters such as the range r or the amount of augmentations N. The assumption behind this future work direction is that a more targeted and customized arrangement of the augmentation process could potentially offer greater performance improvements. Lastly, we intent to explore the performance improvements of ProxyFAUG with other positioning methods and on other public fingerprint datasets.

It is noteworthy that in the spirit of repeatability, reproducibility, verifiability and comparability of results, the full code implementation of the current work as well as the utilized and resulting datasets are openly available at the Zenodo repository.

ACKNOWLEDGEMENTS

This work was funded by the Swiss National Science Foundation, under the Spark Funding scheme, in the context of the project Eratosthenes (project number 195964).

References

[1] P. Davidson and R. Piché, “A survey of selected indoor positioning methods for smartphones,” *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 1347–1370, 2017.

[2] R. W. Ouyang, A. K. Wong, C. Lea, and M. Chiang, “Indoor location estimation with reduced calibration exploiting unlabeled data via hybrid generative/discriminative learning,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 11, pp. 1613–1626, 2012.

[3] H. Zou, C. L. Chen, M. Li, J. Yang, Y. Zhou, L. Xie, and C. J. Spanos, “Adversarial learning-enabled automatic wifi indoor radio map construction and adaptation with mobile robot,” *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6946–6954, 2020.

[4] M. Nabati, H. Navidan, R. Shahbazian, S. A. Ghorashi, and D. Windridge, “Using synthetic data to enhance the accuracy of fingerprint-based localization: A deep learning approach,” *IEEE Sensors Letters*, vol. 4, no. 4, pp. 1–4, 2020.

[5] Q. Li, H. Qu, Z. Liu, N. Zhou, W. Sun, S. Sigg, and J. Li, “Af-dcgan: Amplitude feature deep convolutional gan for fingerprint construction in indoor localization systems,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, pp. 1–13, 2019.

[6] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0

[7] L. Xiao, A. Behboodi, and R. Mathar, “A deep learning approach to fingerprinting indoor localization solutions,” in *2017 27th International Telecommunication Networks and Applications Conference (ITNAC)*, 2017, pp. 1–7.

[8] R. S. Sinha, S.-M. Lee, M. Rim, and S.-H. Hwang, “Data augmentation schemes for deep learning in an indoor positioning application,” *Electronics*, vol. 8, no. 5, p. 554, May 2019. [Online]. Available: http://dx.doi.org/10.3390/electronics8050554

[9] R. S. Sinha and S.-H. Hwang, “Improved rssi-based data augmentation technique for fingerprint indoor localisation,” *Electronics*, vol. 9, no. 5, p. 851, May 2020. [Online]. Available: http://dx.doi.org/10.3390/electronics9050851

[10] M. Aernouts, R. Berkvens, K. Van Vlaenderen, and M. Weyn, “Sigfox and lorawan datasets for fingerprint localization in large urban and rural areas,” *Data*, vol. 3, no. 2, 2018. [Online]. Available: http://www.mdpi.com/2306-5729/3/2/13

[11] G. G. Anagnostopoulos and A. Kalousis, “A reproducible analysis of rssi fingerprinting for outdoor localization using sigfox: Preprocessing and hyperparameter tuning,” in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019, pp. 1–8.

[12] J. Torres-Sospedra, R. Montoliu, S. Trilles, Óscar Belmonte, and J. Huerta, “Comprehensive analysis of distance and similarity measures for wi-fi fingerprinting indoor positioning systems,” *Expert Systems with Applications*, vol. 42, no. 23, pp. 9263 – 9278, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417415005527

[13] A. Grigorios and K. Alexandros, “A Reproducible Analysis of RSSI Fingerprinting for Outdoors Localization Using Sigfox: Preprocessing and Hyperparameter Tuning (datasets),” May 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3228744