CSI-MIMO: K-nearest neighbor applied to indoor localization

Abdallah Sobehy, Eric Renault, Paul Mühlethaler

CSI-MIMO: K-nearest Neighbor applied to Indoor Localization

Abdallah Sobehy*, Éric Renault* and Paul Mühlethaler†
*Samovar, CNRS, Télécom SudParis, Institut Polytechnique de Paris, 9 Rue Charles Fourier, 91000 Évry, France
†Inria, 2 rue Simone Iff, 75589 Paris, France

***Abstract*—Indoor Localization has attracted interest in both academia and industry for its wide range of applications. In this paper, we propose an indoor localization solution based on Channel State Information (CSI). CSI is a fine-grain measure of the effect of the channel on the transmitted signal. It is computed for each subcarrier and each antenna in the Multiple-Input-Multiple-Output (MIMO) antenna case. It is also becoming a trend for indoor position fingerprinting. By using a K-nearest neighbor learning method a highly accurate indoor positioning is achieved. The input feature is the magnitude component of CSI which is preprocessed to reduce noise and allow for a quicker search. The euclidean distance between CSI is the criteria chosen for measuring the closeness between samples. The method is applied to a CSI dataset estimated at an 8 × 2 MIMO antenna that is published by the organizers of the Communication Theory Workshop Indoor Positioning Competition. The proposed method is compared with three other methods all based on deep learning approaches and tested with the same dataset. The K-nearest neighbor method presented in this paper achieves a Mean Square Error (MSE) of 2.4 cm which outperforms its counterparts.**

I. Introduction

Location services have attracted much research interest for the numerous applications that depend on them e.g. routing, Internet-Of-Things (IOT), military applications, etc. Despite the importance of localization and the numerous studies that address it, this problem is not fully resolved due to the challenges each context present; especially indoors [2]. In an outdoor environment, the Global Positioning System (GPS) provides sufficient localization accuracy for many applications [1]. However, in an indoor environment, GPS cannot be used due to the building structure that obstruct the signal. Received Signal Strength Index (RSSI) has been the dominant measure used to localize nodes indoors [3]. RSSI measures the strength of the signal as it is received by the receiver. The relative distance between the transmitter and the receiver is estimated from the RSSI since the strength of the signal decreases with distance. The estimated distance to several nodes can then be used to estimate the position [4], [5]. However, RSSI exhibits high sensitivity to environmental changes and multipath fading, leading to erroneous position estimation [9]. Therefore, the community is moving towards a more robust measure to compute accurate positions indoors.

The advent of 5G and its high data rate requirements led to the use of Multiple-Input-Multiple-Output (MIMO) antennas to increase transmission bandwidth. Moreover, by using Orthogonal Frequency Division Multiplexing (OFDM), multiple transmissions are sent simultaneously on orthogonal subcarriers. Channel State Information (CSI) is a fine-grain information calculated at the physical layer representing the effect of the channel on the transmitted signal. A transmission to multiple antennas over multiple subcarriers allows the channel response per subcarrier and per antenna to be computed [6].

The CSI indicates the change that the signal experiences while traversing the channel. This change is dependant on the position from which the signal is transmitted. This makes CSI a suitable measure for position estimation. Equation (1) illustrates the effect of CSI on the transmitted signal, where $T_{i,j}$ is the signal transmitted from antenna i on subcarrier j. $R_{i,j}$ is the received signal following the changes caused by channel $CSI_{i,j}$ and noise N.

$$R_{i,j} = T_{i,j} \cdot CSI_{i,j} + N \tag{1}$$

CSI is a complex number and thus can be represented in different forms such as Cartesian and Polar forms. The Cartesian form is made up of real and imaginary Components, while the Polar form is represented by Magnitude and Phase. Equations (2) and (3) show the two representations, while Equation (4) shows the conversion from one form to another.

$$CSI_{i,j} = |Mag| \angle \phi \tag{2}$$

$$CSI_{i,j} = Re + iIm \tag{3}$$

$$\begin{aligned} Mag &= \sqrt{Re^2 + Im^2} \\ \phi &= \arctan(Re, Im) \end{aligned} \tag{4}$$

The contribution of this paper is summarized as follows:

1) A statistical analysis of a publicly available CSI dataset that shows the temporal stability of the magnitude component compared to the other three components.
2) A method to reduce the CSI magnitude values to allow for a faster learning process for position estimation.
3) Position estimation using k-nearest neighbor method.

The rest of the paper is organized as follows. Sec. II includes some state-of-the-art solutions to the indoor positioning problem. It also includes a brief description of the experiment devised by the authors of [7] to create the dataset. In Sec. III, we present the analysis that led to the choice of the magnitude as the input feature, the noise reduction of magnitude values using polynomial regression, and the k-nearest neighbor step. In Sec. IV a comparison with the results of other state-of-the-art methods applied to the same dataset is presented. Finally, the conclusion and future work are discussed in Sec. V.

II. Related Work

A. CSI-based Solutions

FIFS [10] and FILA [6] are examples of the early attempts to use CSI-based indoor localization. The former utilizes MIMO antennas from several access points to build an offline radio map of CSI fingerprints to user position. This is followed by an online prediction phase where the input CSI readings are compared to the map using a probabilistic method [8] which was originally designed for RSSI fingerprinting. In FILA [6], the authors process the CSI readings over the subcarrier spectrum and reduce it to an effective value $CSI_{effective}$. The effective CSI is then used to estimate the distance from the antenna to the transmitter using a parametric equation whose parameters are estimated using supervised learning. Finally, using triangulation [11], the position of the transmitter is calculated from distances to multiple antennas. In [14], a k-nearest neighbor method is used on a fingerprinting database based on the magnitude of CSI. Their estimation results outperforms FILA [6] and FIFS [10].

Various studies have attempted to exploit different components of CSI. In [7], the authors propose a channel sounder and utilize both real and imaginary components of CSI with a Convolutional Neural Network (CNN) to achieve position fingerprinting. Their main contribution is the flexible channel sounder architecture that allows for CSI estimation at various frequency bands and environments. More importantly, the dataset collected from their experimentation is publicly available to the scientific community. This makes it possible to make fair comparison between different methods using the same testbed. The use of a CNN with real and imaginary components as input features yields an estimation error of 32 cm in a Line-Of-Sight (LOS) scenario. This accuracy can be greatly improved using other CSI components. One of the very first attempts to use the phase component is [12]. The authors use linear transformation to calibrate the phase component estimated at thirty subcarriers and three antennas of Intel's WiFi Link 5300 NIC. The calibrated phase is then used as input to a three-layer Neural Network to achieve position fingerprinting. The authors that localization using CSI with commodity hardware is more accurate than using RSSI. The mean error in the Line-Of-Sight experiment is $\approx$ 1 m. It is difficult to make a direct comparison with our results given the differences in the experiment area, the number of subcarriers, and the antennas. However, an important point of comparison with our method is the choice of the phase component. Their reasoning in choosing the phase over the magnitude is that it is less sensitive to obstacles and that it is more stable in general. Nevertheless, we show that the magnitude is more stable through a statistical analysis of the dataset. The authors in [13] also concluded on the stability of the magnitude component and chose it as their input feature over the phase.

NDR [15] is a deep learning solution that is tested on the same data set. NDR stands for Noise and Dimensionality Reduction of magnitude values which are then used as input to a Multi Layer Perceptron Neural Network (MLP). A polynomial regression method is used to describe the magnitudes over the subcarrier spectrum and only a subset of the magnitude along the lines is used as input. The method estimates an accurate polynomial line across the magnitude value but this is computationally expensive. We propose a computationally lighter method to reduce noise and dimensionality with a negligible loss in polynomial estimation accuracy. Another approach tested on the same dataset uses the difference between adjacent magnitude values [16] as the input to a Neural Network Ensemble. In addition, a data augmentation step is used to improve estimation accuracy.

B. Experimental Setup

As previously mentioned, the dataset used to test our method was published during the Indoor Positioning Competition organized during the IEEE CTW (Communication Theory Workshop). The transmission occurs between a transmitter and an 8 $\times$ 2 MIMO antenna. The channel sounder described in [7] is used to estimate the CSI values. They are then matched with the ground truth position from which the transmission occurred. The ground truth position is computed using a tachymeter with a 1 cm error. The transmitter is mounted on a robot that traverses a 4 $\times$ 2 meter table while transmitting to the MIMO antenna. The published dataset includes around 17k CSI samples together with their ground truth positions. The frequency of transmission is 1.25 GHz with a 20 MHz bandwidth over which 1024 subcarriers are used. Of the 1024 subcarriers, 10 % are used as guard bands. The remaining 924 subcarrier readings are available for processing along with their corresponding positions. Figure 1 shows the setup with a sketch of the MIMO antenna illustrating the position of its center in the coordinate system.

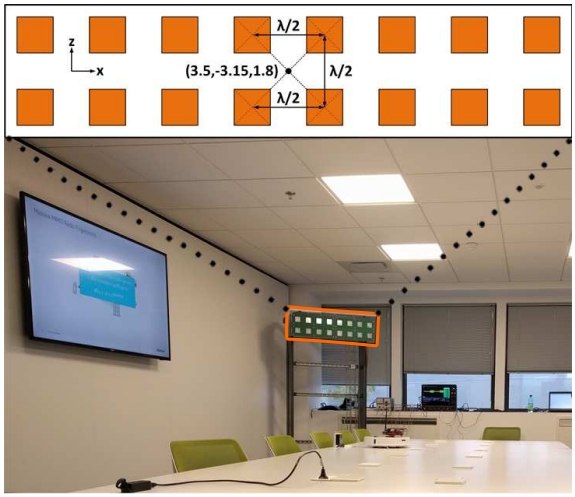

Fig. 1: Environmental setup [7].

III. Methodology

A. Feature selection

The first step consists of selecting the input feature for the k-nearest neighbor learning model. To this end, a statistical

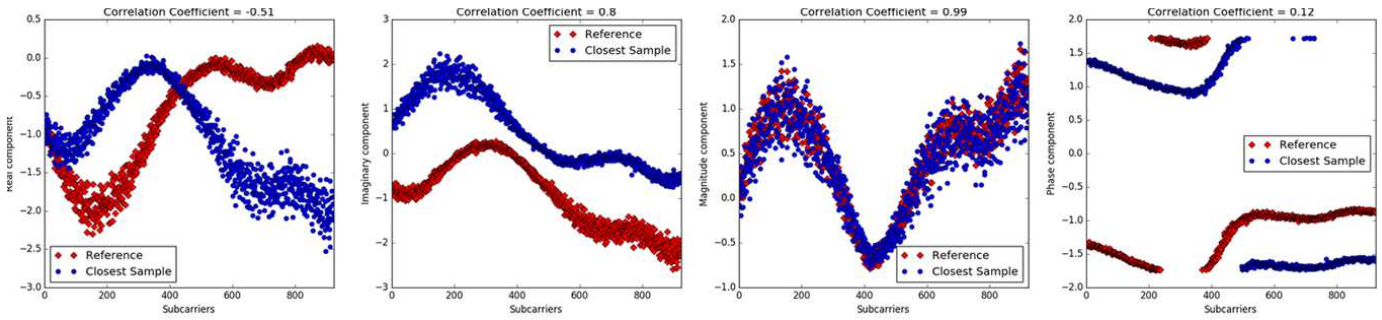

Fig. 2: CSI readings of reference sample and the sample of the closest position for Real, Imaginary, Magnitude and phase components.

analysis was performed to determine the most stable component of the CSI. The stability of a component is defined as the correlation between the values of CSI readings estimated from approximately the same or a very close position. In other words, the higher the stability of the component, the lower the change in its values when estimated from the same or a very close position. This reflects temporal robustness. Assume two transmissions occurred from positions p_1 and p_2 where the distance between p_1 and p_2 is a small value dp. The correlation between the CSI components from both positions at a given antenna for each of the 924 subcarriers is given by:

$$Corr_{p_1,p_2} = \frac{Cov\left(CSI_{p1}, CSI_{p2}\right)}{\alpha_{p_1} \times \alpha_{p_2}} \tag{5}$$

CSI_{p1} and CSI_{p2} are two vectors of one of the four CSI component values estimated at p_1 and p_2 respectively. The stability analysis is implemented by picking a position randomly from the dataset along with the corresponding CSI component values. Let's call this a reference sample. Next, the dataset is searched for the closest position to the reference position. The correlation between the reference and the closest samples is computed as depicted in Equation (5). This process is repeated for approximately 1000 reference samples. The average correlation coefficient over all the sample pairs is computed for the real, imaginary, magnitude, and phase components. Figure 2 shows an example of real, imaginary, magnitude, and phase components for a reference sample and its closest sample. The correlation coefficient value is written at the top of each sub figure. It can be noted that the magnitude conserves its trend better than the other components. This conclusion is further supported by the statistical results of the correlation mean over the 1000 pairs shown in Fig. 3.

The correlation for the magnitude component estimated at close positions is the highest among all the components. Also, the 95% confidence interval is the smallest, meaning that most of the correlation values are around 0.92. As a result of this analysis, the magnitude component is chosen to be the input feature for our learning model.

B. Magnitude Reduction using Least-Squares Polynomial Regression

Since the k-nearest neighbor technique requires a sweep of the training set for each prediction, we propose a method to

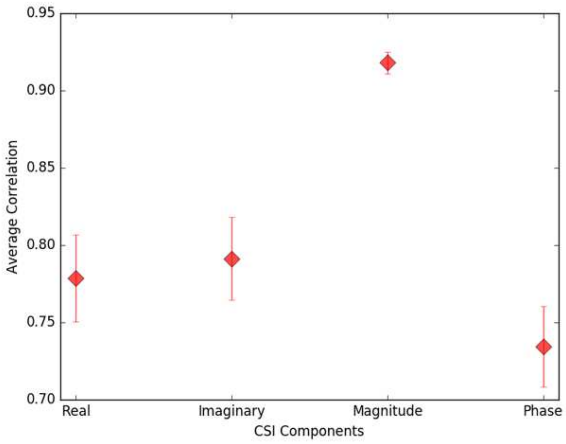

Fig. 3: Average Correlation for Real, Imaginary, Magnitude and Phase components.

reduce the 924 magnitudes to 33 values for a quicker and more accurate search. In [15], the subcarrier spectrum is divided into four overlapping batches. A polynomial regression with various degrees is performed on the points within each batch. The polynomial with the least error to points is chosen. Then, the polynomial lines of each batch are merged using a linear weighted averaging method. This results in an accurate representation of the component values. In our case, a polynomial degree is fixed and a least squares regression [17] is performed over the full subcarrier spectrum. Consequently, a gain in the computation time for a negligible loss in regression accuracy is achieved. The proposed k-nearest neighbor method is tested with both the regression method in [15] and our method. The experiment shows that the loss in regression accuracy does not affect the k-nearest neighbor estimation.

The first step consists of fixing a degree for the regression process. The degree should not be too low to be able to represent the variations in magnitude values. It also should not be too high to avoid computational complexity. Figure 4 shows the average fitting error and standard deviation (std) over all dataset samples along with the 95% confidence interval for degrees from 3 to 8. The confidence interval in the figure is very small which is a good indication of the stability of regression accuracy. It can be viewed that the error stabilizes from degree 6 onward. Hence, degree 6 is chosen for the

polynomial regression step.

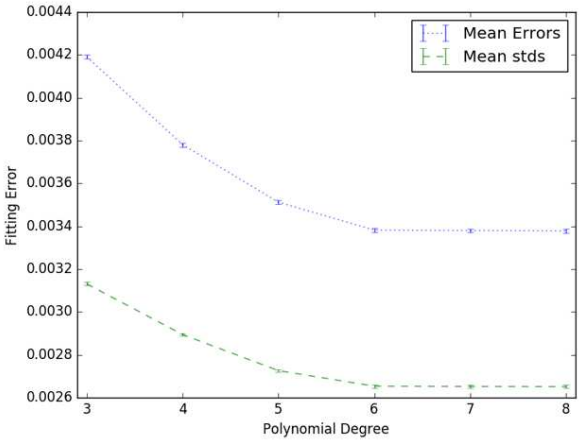

Fig. 4: Average fitting error for various polynomial degrees.

Using the estimated polynomial line, the magnitude values are reduced from 924 points to 33 equidistant points on the line. Value 33 was chosen empirically to hit a sweet-spot between computational cost and accuracy. When attempting to make predictions with number of points less than 33, the accuracy deteriorates. Figure 5 shows an example of the result of the polynomial regression with degree 6. The line in red is the regression outcome along which the 33 equidistant points are selected to represent the magnitude component.

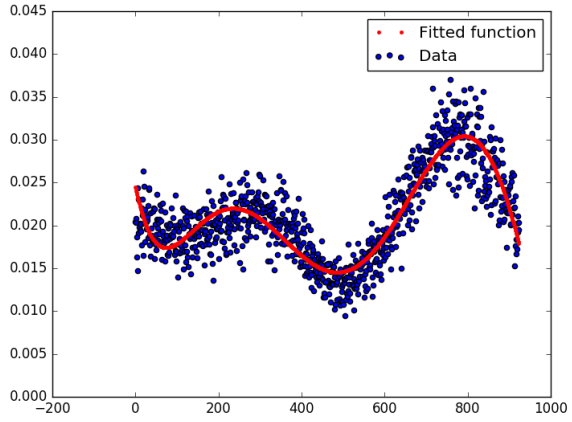

Fig. 5: Polynomial fitting example with degree 6

C. K-nearest neighbor

With the input feature chosen and pre-processed, the final step aims to build the k-nearest neighbor model. Two main parameters have to be chosen: the criteria to define closeness and the number of neighbors (k). For one position, there are 16×33 magnitude values which correspond to the number of reduced magnitudes at each antenna. A reasonable comparison criterion should be computationally light in order to avoid a long processing time. More importantly, it should be able to capture a meaningful difference between samples. Let M^1, M^2 be two sets of 33×16 magnitude values for all 16 antennas. We propose three criteria:

1) The *Absolute Difference* between corresponding magnitude values, which is averaged over all magnitude values for all antennas.

$$\left|diff_{M^1,M^2}\right| = \frac{1}{16}\sum_{a=1}^{16}\frac{1}{33}\sum_{n=1}^{33}\left|M^1_{a,n} - M^2_{a,n}\right| \quad (6)$$

2) The *Euclidean Distance* between two sets of 33 magnitude values averaged for all antennas.

$$dist_{M^1,M^2} = \frac{1}{16}\sum_{a=1}^{16}\sqrt{\sum_{n=1}^{33}\left(M^1_{a,n} - M^2_{a,n}\right)^2} \quad (7)$$

3) The *Correlation Coefficient* between two sets of 33 magnitude values averaged over all antennas.

$$Corr_{M^1,M^2} = \frac{1}{16}\sum_{a=1,n=1}^{16,33}\frac{Cov\left(M^1_{a,n}, M^2_{a,n}\right)}{\alpha_{M^1_{a,n}} \times \alpha_{M^2_{a,n}}} \quad (8)$$

Figure 6 shows the localization accuracy using these three closeness criteria. The x-axis represents the number of antennas used. The best performance for all criteria is when all the 16 antennas are used. The absolute difference and Euclidean distance yield higher accuracy than the correlation coefficient. The Euclidean distance has a slightly lower error than the absolute difference. Using all 16 antennas, the Mean Square Error (MSE) is 2.5 cm with the absolute difference and 2.4 cm with the Euclidean distance. Consequently, we choose the Euclidean distance as the closeness criteria since it gives the highest accuracy.

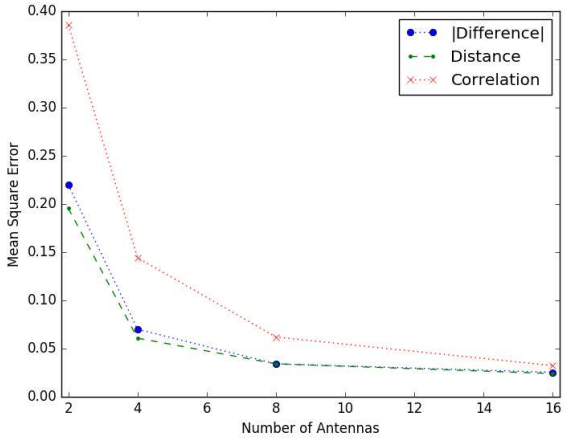

Fig. 6: Mean Square Error using different closeness criteria.

After choosing the comparison criterion, the next step consists of choosing the number of closest neighbors (the k value) from which the prediction is to be computed. Various k

values are tested where the position prediction is the average of the closest k neighbors' positions. Figure 7 shows the effect of the k value on the estimation accuracy. It appears that the larger the k value, the higher the error. This is due to the nature of CSI where the magnitude readings tend to experience abrupt changes from one position to another position that is not very close [13]. Thus, by adding more neighbors, the distances to some of the added neighbors are larger and it becomes difficult to relate the training CSI to the test CSI sample.

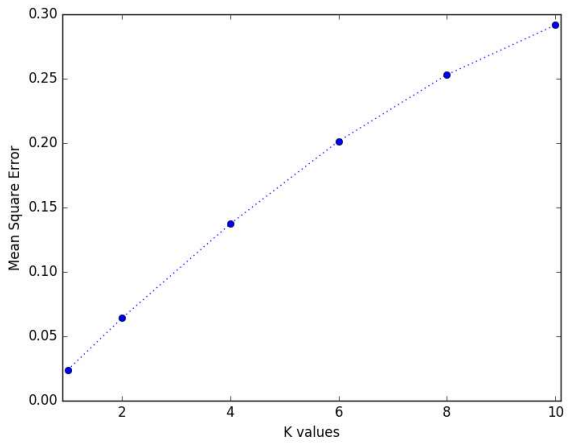

Fig. 7: Mean Square Error using different k values.

IV. Experimental Evaluation

Based on the analysis presented in the previous section, the k-nearest neighbor model is used with the value of k to equal one. For a fair comparison, estimation results are compared to solutions tested using the same dataset. The 17k samples are split into a 90% training set and a 10% test set. For each test set sample, the whole training set is traversed and the position corresponding to the test sample with the smallest distance is chosen to be the predicted position. The distance is computed using Equation (7).

The experiments were carried out on a PC with Ubuntu 14.04 operating system and a 3.8-GHz Intel(R) Xeon(R) quad core CPU E3-1270 v6 with 32 GB of RAM. The GPU is a 2-GB RAM NVIDIA Quadro K420. The time consumed to do the data preprocessing step with the least square optimization is 2.6 ms per antenna per position. The inference time to predict one position is $\approx$ 1.1 s. The estimation accuracy is compared with three solutions:

1) CNN [7]: The real and imaginary components are the input features of a CNN.
2) NDR [15]: The magnitude component is reduced using a polynomial regression and used as an input to an MLP.
3) Ensemble [16]: The differences between magnitude values are fed into an MLP Neural Network ensemble with data augmentation.

Figure 8 presents the position estimation accuracy of the proposed k-nearest neighbors solution and the previously mentioned comparable methods. The k-nearest neighbor method outperforms its counterparts when four or more antennas are used. This shows that the k-nearest neighbor technique is more sensitive to the number of training samples. However, when the number of training samples is sufficient, it is able to localize with lower error. When all the 16 antennas are used, the k-nearest neighbor solution achieves a 2.4 cm MSE compared to 3.1 cm for the Ensemble NN technique [16]. This represents a 16% improvement over the closest error.

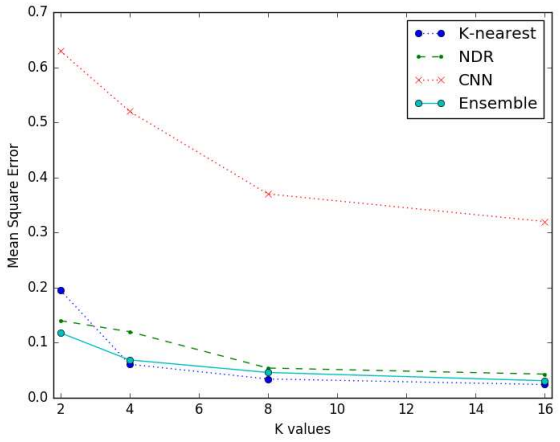

Fig. 8: Comparison between K-nearest neighbors and state-of-the-art methods.

Figure 9 shows the error distribution of the 1.7k estimated positions using the k-nearest neighbor method. The frequency is log-scaled to allow the scarce large errors to be visible since most errors are very close to zero. Outliers with large errors are scattered over the table length. These outliers are possibly due to some Non-Line-Of-Sight (NLOS) transmissions that were not possible to relate to the training set.

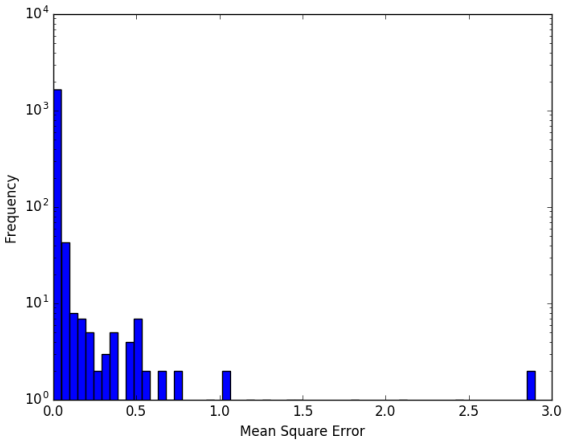

Fig. 9: Error distribution for 1.7k test samples.

Another point that is worth studying is the relation between the smallest euclidean distance to the selected training sample and the resulting prediction error. Figure 10 shows the relation

between the euclidean distance from the test magnitude values to the selected closest sample in the training set and the resulting prediction error. The x-axis shows the euclidean distance between the test and the closest training sample magnitude values. The y-axis represents the resulting prediction errors. Since most prediction errors are concentrated very close to 0, two sub-figures to the right and top are added to give a clearer insight by showing the distribution of points on both axes. The right sub-figure is actually a rotated version of figure 9 which is also log-scaled. The top sub-figure illustrates the distribution of the magnitude distances to the closest training sample which appears like a biased normal distribution. Most of the distances are very small and fewer cases have large distances to the closest training sample. When the distances are greater than 0.03, the outlier predictions begin to appear with prediction error higher than 20 cm, which is far from the average error. Hence, it can be concluded that a smaller euclidean distance leads to a better prediction in general. This is an interesting insight that can be used to detect outliers before making the prediction. In other words, if the euclidean distance to the closest training sample is larger than a certain threshold, there is a high probability that the estimation is far from the true position. It is difficult to reach a precise probabilistic analysis for the outliers since the number of outliers is small. However, it might be possible to reach a concrete probabilistic description with more data in the NLOS scenario.

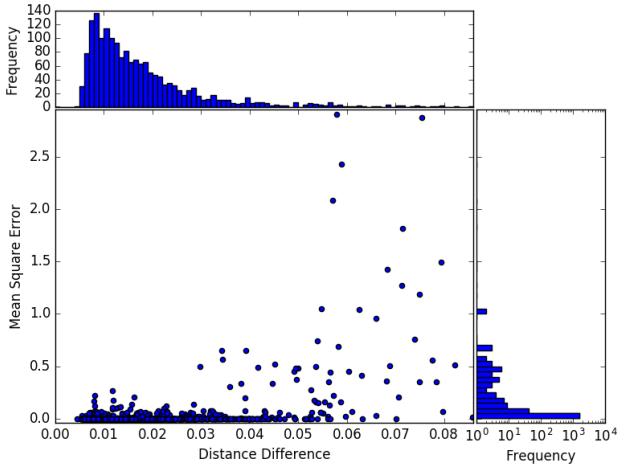

Fig. 10: Relating the Euclidean distance between the test and closest training sample to the prediction error.

V. Conclusion

In this paper, we presented a k-nearest neighbor method to estimate positions from CSI in indoor environments. The first step consisted of choosing the input feature from the CSI components: real, imaginary, magnitude, or phase. We used a statistical analysis to show that the magnitude is the most stable component, therefore, we selected it as the input feature. Magnitude values are then reduced using a polynomial line of degree 6 and least-squares optimization. Out of the 924 magnitude points, 33 equidistant points were chosen to represent the magnitude component. The Euclidean distance has then been used to represent the closeness between magnitude samples. With a k value equal to one, a k-nearest neighbor search was conducted over the training set for each test sample. With an MSE of 2.4 cm, the presented method outperforms three state-of-the-art methods, all based on deep learning techniques. Extension of this study includes an analysis to estimate a localization accuracy upper bound for the dataset. Also, detection of outliers might be possible with more data in NLOS scenarios.

References

[1] É. Renault, E. Amar, H. Costantini and S. Boumerdassi. "Semi-flooding location service." In 2010 IEEE 72nd Vehicular Technology Conference-Fall, pp. 1-5. IEEE, 2010.

[2] Z. Wu, X. Qiang, L. Jianan, F. Chenbo, X. Qi and X. Yun "Passive indoor localization based on csi and naive bayes classification." IEEE Transactions on Systems, Man, and Cybernetics: Systems 48, no. 9 (2017): 1566-1577.

[3] A. Rai, K. C. Krishna, N. P. Venkata, and S. Rijurekha "Zee: Zero-effort crowdsourcing for indoor localization." In Proceedings of the 18th annual international conference on Mobile computing and networking, pp. 293-304. ACM, 2012.

[4] F. Mourad, H. Snoussi, F. Abdallah, and C. Richard, "Guaranteed boxed localization in manets by interval analysis and constraints propagation techniques." In IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference, pp. 1-5. IEEE, 2008.

[5] A. Sobehy, É. Renault, and P. Mühlethaler, "Position Certainty Propagation: A Localization Service for Ad-Hoc Networks." Computers 8, no. 1, 2019. 6.

[6] K. Wu, J. Xiao, Y. Yi, M. Gao and L. M. Ni, "Fila: Fine-grained indoor localization." In 2012 Proceedings IEEE INFOCOM, pp. 2210-2218. IEEE, 2012.

[7] M. Arnold, J. Hoydis and S. T. Brink, "Novel Massive MIMO Channel Sounding Data applied to Deep Learning-based Indoor Positioning." In SCC 2019; 12th International ITG Conference on Systems, Communications and Coding, pp. 1-6. VDE, 2019.

[8] S. Fang, T. Lin, K. Lee, "A Novel Algorithm for Multipath Fingerprinting in Indoor WLAN Environments," in IEEE Transactions on Wireless Communnication, 2008.

[9] K. Wu, X. Jiang, Y. Youwen , C. Dihu, L. Xiaonan, and M. Ni. Lionel "CSI-based indoor localization." IEEE Transactions on Parallel and Distributed Systems 24, no. 7 (2012): 1300-1309.

[10] J. Xiao, W. Kaishun, Y. Youwen and M. Ni. Lionel "FIFS: Fine-grained indoor fingerprinting system." In 2012 21st international conference on computer communications and networks (ICCCN), pp. 1-7. IEEE, 2012.

[11] Y. Liu, Z. Yang, X. Wang and L. Jian, "Location, localization, and localizability." Journal of Computer Science and Technology 25, no. 2, 2010. 274-297.

[12] X. Wang, L. Gao, and M. Shiwen "PhaseFi: Phase fingerprinting for indoor localization with a deep learning approach." 2015 IEEE Global Communications Conference (GLOBECOM), pp. 1-6. IEEE, 2015.

[13] X. Wang, L. Gao, S. Mao and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information." 2015 IEEE wireless communications and networking conference (WCNC), pp. 1666-1671. IEEE, 2015.

[14] Q. Song, G. Songtao, L. Xing and Yuanyuan Yang "CSI amplitude fingerprinting-based NB-IoT indoor localization." IEEE Internet of Things Journal 5, no. 3 (2017): 1494-1504.

[15] A. Sobehy, É. Renault, P. Mühlethaler "NDR: Noise and Dimensionality Reduction of CSI for Indoor Positioning using Deep Learning." GlobeCom, Dec 2019, Hawaii, United States. <hal-023149>

[16] A. Sobehy, É. Renault and P. Mühlethaler "CSI based Indoor localization using Ensemble Neural Networks." Machine Learning for Networking, Dec 2019, Paris, France. <hal-02334588>

[17] E. Jones, T. Oliphant, P. Peterson, and others. "SciPy: Open source scientific tools for Python." Retrieved from "http://www.scipy.org/", 2001.