CORTEN: A Real-Time Accurate Indoor White Space Prediction Mechanism

Hejun Xiao, Dongxin Liu, Fan Wu, Linghe Kong, Guihai Chen
Shanghai Key Laboratory of Scalable Computing and Systems
Shanghai Jiao Tong University, China
{hjxiao, liu-dx, linghe.kong}@sjtu.edu.cn, {fwu, gchen}@cs.sjtu.edu.cn

***Abstract*—Exploring and utilizing indoor white spaces (vacant VHF and UHF TV channels) have been recognized as an effective way to satisfy the rapid growth of the radio frequency (RF) demand. Although a few methods of exploring indoor white spaces have been proposed in recent years, they only focus on the exploration of the current indoor white spaces. However, due to the dynamic nature of the spectrum and the time delay in the process of exploration, users often cannot get accurate white space information in time, resulting in issues, such as spectrum utilization conflicts or inadequate white space utilization. To solve the problem, in this paper, we first perform an indoor TV spectrum measurement to study how the spectrum state changes over time and the spatio-temporal-spectral correlation of spectrum. Then, we propose a real-time aCcurate indoOR whiTe spacE predictioN mechanism, called CORTEN. CORTEN can predict the white space distribution for various time spans with high accuracy. Furthermore, we build a prototype of CORTEN and evaluate its performance based on the real-world measured data. The evaluation results show that CORTEN can predict accurately 38.7% more indoor white spaces with 51.3% less false alarms compared with the baseline approach when predicting the white spaces one hour ahead.**

I. INTRODUCTION

In 2008, Federal Communications Commission (FCC) issued a historic ruling that allowed unlicensed devices to use the TV spectrum that is not locally occupied by licensed devices (The unoccupied TV spectrum is often referred to as TV white space or simply white space). After that, the TV white spaces receive more and more attentions from Dynamic Spectrum Access (DSA) developers. Although white spaces are open for unlicensed users now, FCC also required that unlicensed white space devices should not interfere with the licensed devices (TV broadcasts). Therefore, it is essential for all user devices (especially the unlicensed ones) to find out whether a spectrum band is available before using it in communications. FCC proposed a geo-location database approach [1], [2] to protect the licensed TV transmissions, which required the unlicensed users to query a geo-location database in order to obtain the white space availability at different outdoor locations. However, studies have shown that in the most of time people stay indoors, which leads to the 70% demand for wireless spectrum being from indoor scenarios [3], [4]. However, due to the indoor complex environment such as walls and other obstacles, directly applying the outdoor method to the indoor environment would lead to a too conservative result.

In 2013, Ying et al. [5] studied the characteristics of indoor white spaces, and proposed the first indoor white space exploration system, named WISER. Then, different methods were proposed to boost the performance of indoor white space exploration [6], [7]. The existing indoor white space exploration systems aim at constructing an indoor white space availability map, which indicates the availability of the TV spectrums at different indoor locations. Users could submit their indoor locations, and then receive the corresponding list of white spaces according to the current indoor white space availability map. However, due to the dynamic nature of the spectrum and the time delay in the process of exploration, these methods can only return the current or even previous spectrum exploration results to the users. Availability of the spectrum may change, for example, some channels are vacant a little time before, but now occupied or vice versa. The above problem will make some white spaces undiscovered or the unlicensed users interfere with the licensed users, which all is not what we would like to see.

To solve the above problem, we propose a real-time aCcurate indoOR whiTe spacE predictioN mechanism, called CORTEN. CORTEN can predict the white space distribution in the future period of time accurately. In CORTEN, we can use only a limited number of RF sensors to get not only the current white space availability map but also the accurate spectrum distribution in the future period of time. To achieve accurate prediction of the white space, the main contributions of this paper are as follows:

- We perform indoor white space measurements in a building for a week. The statistical results further describe the correlation between the different locations and channels. In addition, we find that the correlation between the signal strength of different time is more significant and there is significant periodicity. This provides the basis for the application of the autoregressive integrated moving averages (ARIMA) [8] model to accurately predict the white spaces in the future period of time.

This work was supported in part by the State Key Development Program for Basic Research of China (973 project 2014CB340303), in part by China NSF grant 61672348, 61672353, and 61472252, and in part by Shanghai Science and Technology fund 15220721300. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

F. Wu is the corresponding author.

- We propose CORTEN, a real-time accurate indoor white space prediction mechanism. To the best of our knowledge, it is the first system to predict the indoor white space distribution. By taking spatio-temporal-spectral correlation and the periodicity into consideration, we propose a novel white space prediction algorithm based on the ARIMA model, k-medoids algorithm [9] and compressive sensing technique.
- We buid a prototype of CORTEN, and evaluate its performance with real data. When we define a time slot being an hour, and use CORTEN to predict the white space distribution of an hour ahead, in average, CORTEN can get the 0.58% of the FA Rate and 22.47% of the WS Loss Rate (The FA Rate and WS Loss Rate will be defined in Section V) by choosing different number of prediction points. In average, CORTEN can predict accurately 38.7% more indoor white spaces with 51.3% less false alarms compared with the baseline approach.

The remainder of the paper is organized as follows. We introduce our indoor white space measurements in Section II. Section III presents the system model of CORTEN. In Section IV, we describe the detailed algorithms of CORTEN, including ARIMA model and k-medoids. In Section V, we present evaluation results. Related work and conclusions are shown in Section VI and Section VII, respectively.

II. INDOOR WHITE SPACE AVAILABILITY MEASUREMENT

In this part, we perform our indoor white space measurements in the third floor of a building. The purpose of the indoor white space measurements is to study how the spectrum state changes over time and to verify the location-channel dependence, meantime to explore the time dependence of indoor white spaces. These features will be utilized in designing the indoor white space prediction mechanism later. In addition, the data obtained in this part will be used as the data for the evaluation of CORTEN in Section V.

A. Measurement Setup

The measurement devices we used consist of 22 sets of USRP N210 [10], omni-directional log periodic PCB antenna (400-1000 MHz), laptops, which is shown in Figure 1. The daughterboard of our USRP is SBX with 5-10 dBm noise figure. We calibrate the device using a RF signal generator to get the accurate signal strength. As suggested by FCC [11], the gain of the antenna is also taken into consideration during the calibration process.

A number of different methods were proposed for identifying the presence of signal transmissions, such as energy detection, waveform-based sensing, and matched-filtering [12]. Here, we choose energy detection, since it is the most common way with low computation and implementation complexities. In our measurements, we judge whether a channel is vacant or occupied by comparing the signal strength of the channel with a threshold, which depends on the noise floor [13]. If the signal strength of a TV channel is greater than the threshold, we label this channel as locally occupied, otherwise the channel is labeled as vacant. We measure the digital TV channels between 470 MHz - 566 MHZ and 606 MHz - 870 MHz with 8 MHz channel bandwidth, and use the same threshold -84.5 dBm/8 MHz as [5], [6] for digital TV signals. Due to the hardware limitations, the vacant channels determined by the above mentioned threshold may be not safe to use, but the observations drawn from the measurement are general, and our mechanism is not limited to any specific threshold. We believe that if the sensitivity of the measurement hardware can support a threshold of -114 dBm as suggested by FCC [11], our mechanism can be safely used in practice.

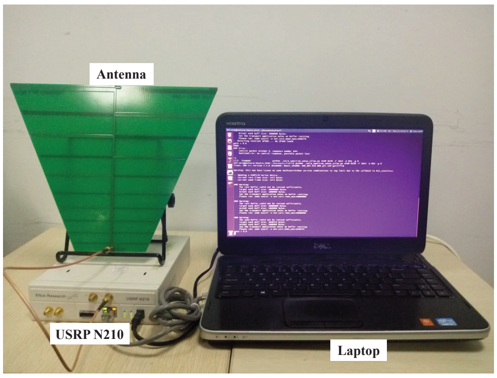


Figure 1. The measurement devices.

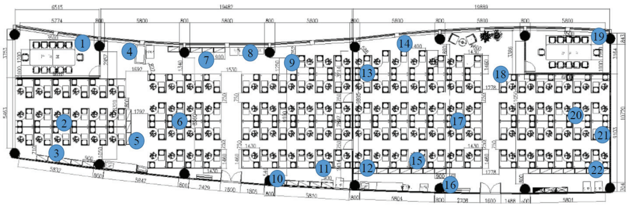


Figure 2. The experimental points.

Just like the prior works [5], [6], we implement the energy detector to detect the existence of signal transmissions. The energy detector is based on the fast Fourier transform (FFT) with bin size 1024 and sample rate 4 MHz. The signal strength of a channel is calculated by averaging the value in the corresponding bins.

We choose two largest consecutive labs (10m×45m) in our laboratory building as our experimental site. We measure the spectral signal strength of 22 measurement points, which is shown in Figure 2. In the course of the experiment, every measurement points in our laboratory are deployed with a USRP coupled with a laptop and an omni-directional log periodic PCB antenna to continuously measure the signal strength of the selected 45 digital TV channels for a week (Dec. 29, 2015-Jan. 4, 2016). The measurement devices are calibrated using a RF signal generator and synchronized using "crontab" of Ubuntu. We synchronously measure the signal strength of 45 TV channels every 5 minutes for a week. The observations are as follows.

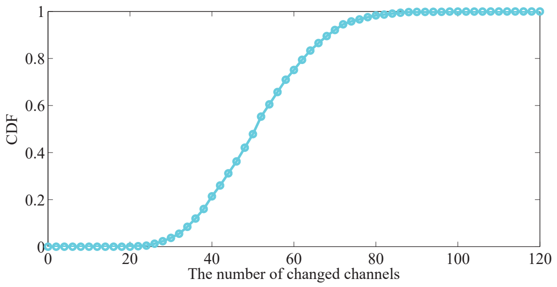


Figure 3. The CDF of the number of changed channels.

B. The Changes of The White Spaces

Five minutes as a time slot, there is a two-dimensional matrix data of 22 locations and 45 channels in each time slot, and we have a total of 2016 set of data in a week. We compare the signal strength of the data set in every two adjacent time slots and find that the signal strength of many channels fluctuates around the threshold, which means the availability of many channels will change over time. If the signal strength of a certain channel in a certain location is smaller than the threshold in the previous time slot, and it is greater than the threshold in the next time slot, we say that it changes from the vacant channel to the occupied channel. And if on the contrary, we say that it changes from the occupied channel to the vacant channel. We count these two kinds of changes in the 2015 groups adjacent data set and depict the CDF (Cumulative Distribution Function) curves of the sum of these two kinds of changes in Figure 3

As shown in the Figure 3, the sum of these two changes are floating between 20 and 100 channels, and the average number of changed channels is 52.6, which means more than 5% of the total channels (22×45) changes in each pair of adjacent time slots. If we do not predict the white spaces in the next time slot, but only use the white spaces according to the current exploration results, it will not only reduce the utilization rate of the white spaces and even cause the use conflict with licensed users. Therefore, in order to know which channel's availability will change, so as to make more reasonable use of white spaces, it is essential to accurately predict the white spaces of the next time slot.

C. The Spatio-Temporal-Spectral Correlation of Spectrum

In this part, we mainly study the spectrum correlation between locations, frequency and time slots. The previous work [5], [6] have already described the correlation between the spectrum in different locations and frequency. We will verify their conclusions in a more intuitive way, and present the time correlation of the spectrum.

In order to obtain the spectrum correlation between different locations based on the experimental data, we get a vector for each measured location that contains the signal strengths of all TV channels over a week interval. Then we calculate the Pearson product-moment correlation coefficients [14] of all pairs of locations, and draw Figure 4(a) based on the results. Through Figure 4(a), we can see that the correlation coefficient between each pair of locations is relatively large, which shows that the spectrum between the 22 sites exists a high degree correlation. In addition, we observe the spectrum correlation in different frequency channels. We use the same method with the location correlation, and get Figure 4(b). Through Figure 4(b), we can see that some channels have strong correlation, and there are no significant correlations between some other channels.

Then we focus on the time correlation of the spectrum. The previous work does not concern it very much, but it is the basis of our accurate prediction. At first, we use the same method to calculate the spectral correlation between every pair of time slots, resulting in Figure 4(c). Here, for the sake of clarity, we use one hour as a time slot. It shows the signal strength correlation in time is stronger than the location correlation and the channel correlation. Then we randomly select a channel at one location and then display the changes of its signal strength at a week interval, which is shown in Figure 4(d). Through Figure 4(d), it is particularly important that we observe the change in signal strength has a certain periodicity. It is easy to see that the change in signal strength occurs seven cycles roughly at 168 hours, about 24 hours per cycle. That means very similar change occurs in signal strength at the same time every day. The special characteristics of the white space in time provides the basis of accurate prediction.

D. Experimental Summary

Through the one-week continuous indoor white space measurements, we get the following information:

- According to the results of the measurements, the availability of many channels will change over time, which is the necessity of our prediction.
- The spatio-temporal-spectral correlation between the spectrum is very strong, and especially the periodicity in time is particularly significant, which is the basis of accurate white space prediction.

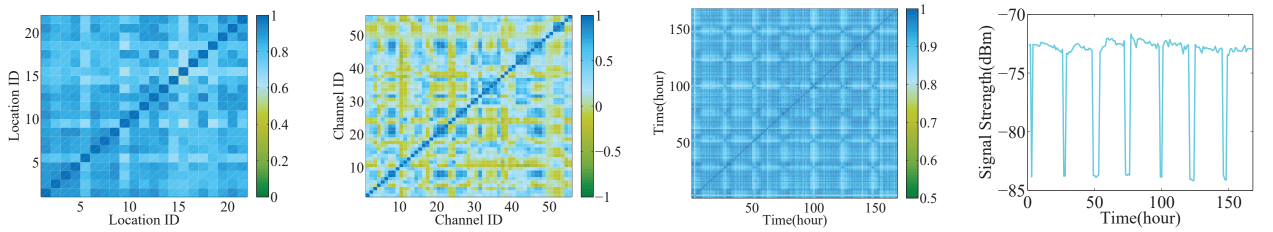
III. System Model

CORTEN aims to accurately predict white spaces in the future period of time, and the time period can be determined in advance based on the time granularity of our exploration data. As shown in Figure 5, CORTEN is mainly composed of three parts: real time sensing module, real time prediction module and central server. In this section, we will introduce these three parts using our white space measurements of 22 locations as an example.

A. Real Time Sensing Module

The job of the real time sensing module is selecting parts of these candidate locations, placing a sensor at each selected location, performing a "partial sensing" (since not all locations have sensors deployed), and sending the sensing data to the central server.

For ease of presentation, we define $\mathbf{X}_t$ to denote the location-channel matrix in time slot t, which is a 22×45



(a) Spatial correlation

(b) Spectral correlation

(c) Temporal correlation

(d) Time series of signal strength

Figure 4. Results of indoor white space measurements

matrix recording the signal strength, where $\mathbf{X}_t(i,\ j)$ is the signal strength of channel j at location i in time slot t. If we deploy a sensor at location i, we can get $\mathbf{X}_t(i,\ j)(1\leq j\leq 45)$ at time t. Then the real time sensing module will send the sensing data to the central server in real time.

B. Real Time Prediction Module

The real time prediction module can get data matrices of the signal strength in some continuous time slots from the central server as training data and predict the white spaces in the next time slot, then send back the prediction results to the central server. The work of real time prediction module consists of three steps:

- Slection of the prediction points: Because we can't deploy spectrum sensors in every point in practical application. In order to accurately predict white spaces, we should select parts of the points where are deployed spectrum sensors as the prediction points.
- Prediction of the selected points: After selecting the prediction points, we will predict every channel of the selected points and get the corresponding data of the next time slot based on the training data.
- Recovery of the prediction matrix: The prediction matrix is not complete after the prediction of the selected points, so we should use matrix recovery algorithm to recover the matrix based on the dependence of the matrix data. Then we will get the whole prediction matrix.

C. Central Server

First of all, the central server will process the data uploaded from the real time sensing module and use the matrix recovery algorithm to obtain the complete matrix data of the current time slot, and then send the training data to the real time prediction module. At last, the central server gets the prediction matrix of white spaces in the next time slot from the real time prediction module. If a user requests a white space, then the central server returns a white space list to the user based on the current and future data.

In FIWEX [6], the central server returns the white space list to users after questioned, but the process only consider the results of the current white space exploration. In CORTEN, we get the white space data in next time slot, so we can use a more efficient white space allocation algorithm to improve the utilization of white spaces and reduce the rate of collision.

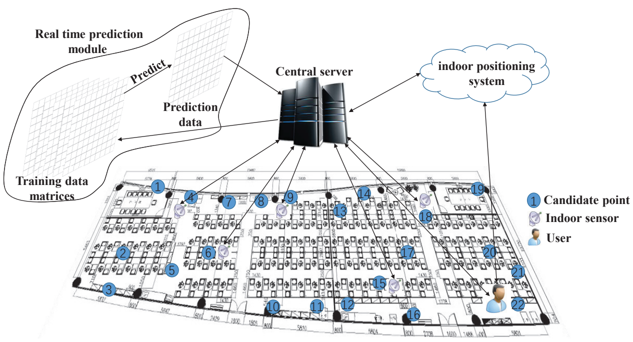


Figure 5. System architecture of CORTEN.

IV. PREDICTION MODEL AND ALGORITHM

In this section, we present our ARIMA model based indoor white space prediction algorithm. The ARIMA model is a prediction model mentioned by Box and Jenkins in the 1970s and it is mainly used in the field of economics, such as stock forecasting. Through continuous improvement, the ARIMA model has been widely applied to different realms [15], [16]. Because the time series of the spectral signal strength has a strong temporal correlation and a significant periodicity, we use ARIMA model to predict white spaces. In order to predict more accurately, we use the sliding window method to improve the ARIMA model. And then based on the location correlation of the spectrum, we propose a k-medoids method to select prediction points.

A. ARIMA Model

We begin with the ARMA($p,\ q$) model, which can be expressed as:

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \epsilon_t \tag{1}$$

where x_t is the value of a time series at time t, parameter ϵ_t represents the random noise at time t that satisfies N(0, σ^2), then α_i and β_i are the coefficients of the linear combination, besides p and q represent the order.

ARMA($p,\ q$) model can only deal with stationary time series. The so-called stationary is the mean of the time series of any section maintaining within a certain range. If the time series is not stationary, a very efficient way to deal with it is to do the time series difference until it is stationary, which can be represented as follows:

$$\nabla x_t = x_t - x_{t-1} \tag{2}$$

Similarly, the d order difference is as follows:

$$\nabla^d x_t = \nabla^{d-1} x_t - \nabla^{d-1} x_{t-1} \tag{3}$$

If the new time series $\nabla^d x_t$ generated after the difference satisfies ARMA(p, q), then we say that the original time series x_t satisfies the ARIMA(p, d, q):

$$\nabla^d x_t = \sum_{i=1}^{p} \alpha_i \nabla^d x_{t-i} + \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \epsilon_t \tag{4}$$

where d is the order of difference.

Before we use the ARIMA(p, d, q) model to predict, we first need to know p, d, q and the weight vector $\boldsymbol{\alpha} \in \mathbf{R}^p$ and $\boldsymbol{\beta} \in \mathbf{R}^q$. Parameter d is the difference order, and we can get its value by determining how many times the original time series do difference. After d-order differences, we get a stationary time series.

For a set of stationary time series $x_1, x_2, \ldots, x_t$, the k-order autocorrelation coefficient is

$$\rho_k = \frac{E[(x_i - \overline{x})(x_{i+k} - \overline{x})]}{\sigma^2} \tag{5}$$

where $\overline{x}$ and σ^2 are the mean and the variance of the time series. Then, we can calculate the autocorrelation coefficient of each order, and get q in ARIMA(p, d, q) according to the truncation of autocorrelation coefficient, which means when the q+1 order autocorrelation coefficient tends to 0, we can get the value of q [8].

We can solve the Yule-Walker equation [8] to get the k-order partial autocorrelation coefficients of the time series, and get p in ARIMA(p, d, q) according to the truncation of partial autocorrelation coefficient, which is similar to the method mentioned above. The Yule-Walker equation is

$$\begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} \Phi_{k1} \\ \Phi_{k2} \\ \vdots \\ \Phi_{kk} \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_k \end{bmatrix} \tag{6}$$

which can also be denoted by

$$\mathbf{P_k} \boldsymbol{\Phi}_\mathbf{k} = \boldsymbol{\rho}_\mathbf{k} \tag{7}$$

where Φ_{kj} represents the j-th coefficient of the k-th order regression expression, and we can judge the value of p by the truncation of Φ_{kk}.

In (4), vector $\boldsymbol{\alpha}$ is the set of autoregressive coefficients, we can also get its value through the Yule-Walker equation

$$\begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{p-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_p \end{bmatrix} \tag{8}$$

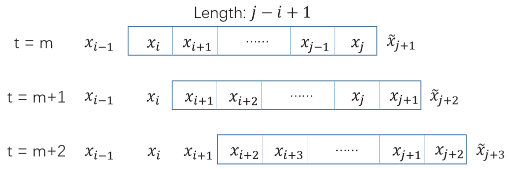


Figure 6. The sliding window.

Vector $\boldsymbol{\alpha}$ can be solved by the following formula

$$\boldsymbol{\alpha}_p = \mathbf{P}_p^{\mathbf{-1}} \boldsymbol{\rho}_p \tag{9}$$

which means when p equals to k, then we have

$$\boldsymbol{\alpha}_p = \boldsymbol{\Phi_k} \tag{10}$$

Since the relationship between vector $\boldsymbol{\beta}$ and autocorrelation coefficients is non-linear, vector $\boldsymbol{\beta}$ can not be solved by the Yule-Walker equation, but iteration method can be used to converge to maximum likelihood estimation, and get the approximate optimal value of vector $\boldsymbol{\beta}$. Due to the space limitation, we omit the calculation process. Please refer to [8] for more details.

Then we can use ARIMA(p, d, q) model to predict white spaces. In the article, we directly use the integrated ARIMA method in the R forecast package [17], we can improve the accuracy of which by using the sliding window.

B. The Sliding Window in ARIMA Model

The ARIMA model is only suitable for short-term prediction. For the ARIMA model, if the prediction step size is shorter, the accuracy of prediction is much higher, and when the prediction step size L = 1, the prediction result is the most accurate and effective. For the white space prediction problem, we just want to get the prediction results of the next time slot, so we set the prediction step size L = 1. In order to improve the accuracy of the prediction, we use a sliding window method to do real time updating of the training data.

The so-called sliding window, that is with the time going, the training data with fixed length in a window would continue to be updated. The sliding window is shown in Figure 6. In order to accurately and quickly model, we set a fixed length to the sliding window. And the length of the sliding window should be able to ensure the accuracy of prediction at the same time as short as possible, so that it can guarantee both rapid modeling and accurate prediction. And the sliding means we need to update the training data in the window in real time. In CORTEN, our sliding window will update the training data with the real time sensing data. The training data at the end of the sliding window is taken out of the window, and the latest sensing data of the current time slot is filled in the forefront of the sliding window. And then we need to do re-modeling and predict the white space distribution in next time slot based on new training data set, followed by the cycle to run CORTEN.

C. Selection of The Prediction Points

We need to select parts of all the measurement points as prediction points. In Section II, we have introduced the strong correlation between the spectrum at different locations. So when we select M locations in N locations to predict, the remaining N-M locations can be recovered by the matrix recovery algorithm. In order to make the recovery matrix data to be most accurate, we should try to select those locations where the dependency is as weak as possible so that we can guarantee the prediction data containing as much useful information as possible. In order to select the M locations to predict, our work mainly consists of two parts:

- According to the correlation between different locations, N locations are clustered into M groups.
- Select one of the most suitable location in each group as a prediction point.

According to the features of the prediction points, we use k-medoids algorithm to cluster and select the clustering centers as the prediction points. As shown in Algorithm 1, in the process of selecting prediction points, we select the data of *ts* continuous time slots as the training data to calculate the correlation of every pair of locations, and the *ts* here is equal to the length of the sliding window we mentioned before. The group number r can be determined in advance according to the accuracy requirement, which is shown in Section V. In line 2-5, we get the correlation coefficient between all pair locations through the training data, and the PCorre in line 5 is the method of calculating the Pearson product-moment correlation coefficient. We calculate the sum of the absolute values of the correlation coefficients of the training data set, which can get rid of accidental factors. Calculating the absolute value is because we only focus on their correlation, but do not care it is positive or negative correlation. In line 6, we take the negative of the correlation coefficient matrix, because it makes the larger the value is, the weaker the correlation is, which satisfies the k-medoids clustering criteria. In line 7-10, 100 iterations are performed to get the most optimal clustering results. Finally, the selected prediction points are stored in $\mathbf{SL}_{opt}$.

D. Prediction Matrix Recovery

After selecting prediction points and predicting the selected points, the prediction matrix only contains the data of the selected points and we need to use the matrix recovery algorithm to get a complete prediction matrix. Compressive sensing is a generic data reconstruction technique based on the structure and redundancy of real world signals or datasets [18], [19], and compressive sensing has been widely applied to different realms [20], [21]. Here, we combine compressive sensing with white space matrix recovery problem to perform matrix recovery just like [22].

Algorithm 1: SelectPoint($\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}\ldots\mathbf{X}^{(ts)}$, *ts*, *r*,$v_{opt}$)

```
Input  : 1) $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}\ldots\mathbf{X}^{(ts)}$: the signal strength
           data of ts continuous time slots; 2) ts: the
           length of training data; 3) r: the number of
           prediction points; 4) $v_{opt}$: initially a sufficient
           large number.
Output: $\mathbf{SL}_{opt}$: the set of selected prediction points.
1  $m \leftarrow$ size($\mathbf{X}^{(1)}$, 1), COR $\leftarrow \mathbf{0}^{(m\times m)}$;
2  for $i = 1$ to $m$ do
3      for $j = 1$ to $m$ do
4          for $k = 1$ to $ts$ do
5              COR$(i,j) \leftarrow$COR$(i, j)$+|PCorre($\mathbf{X}_i^{(k)}$,$\mathbf{X}_j^{(k)}$)|;
6  COR $\leftarrow$ -COR;
7  for $i = 1$ to $100$ do
8      [SL, $v$] $\leftarrow$ kmedoids(COR, $r$);
9      if v<v$_{opt}$ then
10         $\mathbf{SL}_{opt} \leftarrow$ SL, $v_{opt} \leftarrow v$;
11 return $\mathbf{SL}_{opt}$;
```

For the recovery of the white space matrix, due to the existence of some properties such as location-channel dependence in signal strength, we use compressive sensing technique to transform the white space matrix recovery problem into a optimization problem:

$$\begin{aligned} Minimize||B_s \circ (LR^T) - D_s||_F^2 + \lambda_1(||L||_F^2 + ||R||_F^2) \\ +\lambda_2||P(LR^T) - P_0||_F^2 + \lambda_3||(LR^T)C - C_0||_F^2 \end{aligned} \tag{11}$$

where $\circ$ refer to the Hadamard product ($X = Y \circ Z$ means $X(i,j) = Y(i,j) \times Z(i,j)$), and B_s is the prediction points and strong channel matrix, if it is a prediction point or a strong channel in matrix B_s, the value is 1, otherwise the value is 0. Matrix D_s contains the corresponding prediction value or the signal strength of strong channel. P and P_0 are the location dependence constraint matrices, C and C_0 represent the channel dependence constraint matrices. Lagrange multipliers λ_1 λ_2 and λ_3 are used to balance the weight of each part. L and R are the Singular Value Decomposition (SVD) of final prediction matrix $\widetilde{X}$

$$\widetilde{X} = LR^T \tag{12}$$

To solve (11) and get the optimal solution of L and R, we use the alternating steepest descent algorithm [23], which is commonly utilized to do the minimization in the low rank matrix completion. Then we can get the final prediction matrix $\widetilde{X}$. Due to the space limitation, we omit the derivation process. Please refer to [22] for more details.

E. The Overall Algorithm of The Real Time Prediction Module

In this part, we will introduce the overall algorithm of the real time prediction module in CORTEN. When the central server sends the real time training data to the real time prediction module, the new round of prediction will begin, as shown in Algorithm 2. When the real time prediction module gets training data, the work of selecting prediction points is done at first in line 3. After selecting points, it is necessary to model all the channels in all the selected points by ARIMA and to predict $\widetilde{\mathbf{X}}^{(ts+1)}$, which is shown in line 4-7. In line 6, we directly use the integrated ARIMA method in the R forecast package [17]. As shown in algorithm line 8, we will call the Alternating Steepest Descent (ASD) [22] method in FIWEX to recover the prediction matrix.

When the algorithm is finished, we get complete prediction matrix $\widetilde{\mathbf{X}}^{(ts+1)}$ at next time slot. The real time prediction module uploads the prediction results to the central server, so that the prediction module completes once work. When next time begins, the training data will be updated and the algorithm will be carried out again and do the cyclic process.

```
Algorithm 2: Predict($\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ ...$\mathbf{X}^{(ts)}$, $ts$, $r$, $v_{opt}$)

Input  : 1) $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ ...$\mathbf{X}^{(ts)}$: the signal strength
         data of $ts$ continuous time slots; 2) $ts$: the
         length of training data; 3) $r$: the number of
         prediction points; 4) $v_{opt}$: initially a
         sufficiently large number.
Output: $\widetilde{\mathbf{X}}^{(ts+1)}$: the prediction signal strength matrix.

1 $[m, n] \leftarrow$ size($\mathbf{X}^{(1)}$);
2 $\widetilde{\mathbf{X}}^{(ts+1)} \leftarrow \mathit{0}^{(m \times n)}$;
3 SL$_{opt}$ ← SelectPoint($\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ ...$\mathbf{X}^{(ts)}$, $ts$, $r$, $v_{opt}$);
4 foreach $P \in SL_{opt}$ do
5     foreach $C \in P$ do
6         $\widetilde{x}_{ts+1} \leftarrow$ ARIMA($x_1$, $x_2$ ...$x_{ts}$);
7         Put $\widetilde{x}_{ts+1}$ into the corresponding location of
            $\widetilde{\mathbf{X}}^{(ts+1)}$;
8 $\widetilde{\mathbf{X}}^{(ts+1)} \leftarrow$ ASD($\widetilde{\mathbf{X}}^{(ts+1)}$);
9 return $\widetilde{\mathbf{X}}^{(ts+1)}$;
```

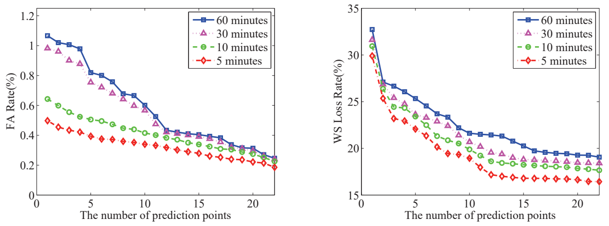
V. PERFORMANCE EVALUATION

In this section, we perform experiments to evaluate the performance of CORTEN. First, we evaluate the performance of CORTEN at different lengths of time slots and different number of prediction points. Then in order to evaluate CORTEN's performance, we define a baseline approach, and compare the prediction results with CORTEN.

A. Methodology

The system performance evaluation is based on the real data we measured in Section II. Although we only use the data in one building to evaluate the performance of CORTEN, but CORTEN is not for any particular indoor environment, so that using the data measured in other buildings will get the similar results. We choose FA Rate and WS Loss Rate as the performance metrics. Their definitions are as follows:

- **False Alarm Rate (FA Rate):** the ratio between the number of channels that a system mis-identifies as vacant and the total number of vacant channels identified by the system.
- **White Space Loss Rate (WS Loss Rate):** the ratio between the number of channels that a system mis-identifies as occupied and the total number of actually vacant channels.



(a) Results of FA Rate

(b) Results of WS Loss Rate

Figure 7. Prediction results at different lengths of time slot and different number of prediction points

We have a group of experimental data every five minutes in 7 consecutive days, which is a total of 2016 groups experimental data. We evaluate the prediction performance of CORTEN in different lengths of time slot, containing 5 minutes, 10 minutes, 30 minutes and an hour to illustrate CORTEN can achieve accurate and efficient prediction results in different lengths of time slot. Due to the exploration time and the delay of the sensors, we can only set the finest granular time slot to be 5 minutes. In practical applications, we can choose the appropriate time slot according to the specific situation. After experimenting with different length of the sliding window, we choose the optimal length 50, which can not only guarantee the prediction accuracy but also can quickly model. In the process of selecting prediction points, we directly use the data in the sliding window as training data to select the appropriate prediction points according to the location correlation. In the prediction matrix recovery, the method and parameter settings follow the method in FIWEX. In addition, in order to reduce the FA rate and avoid interference with licensed users, we set the protection range PR = -0.7 dBm for the threshold value. That means when we determine whether a channel is a white space or not after obtaining the prediction matrix, we should compare the prediction data with the sum of threshold and PR.

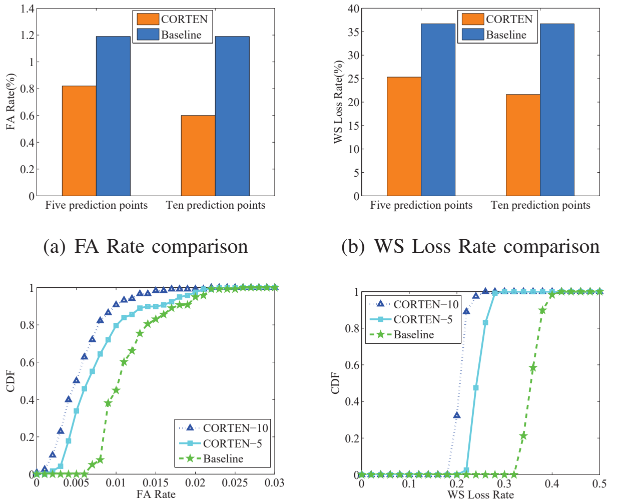
B. Performance at Different Lengths of Time Slot and Different Number of Prediction Points

In this part, we will illustrate the performance of CORTEN at different lengths of time slot and discuss how to choose the number of prediction points. Our experimental data includes 22 measurement points, and we run CORTEN using from 1 to 22 prediction points at four different lengths of time slot, containing 5 minutes, 10 minutes, 30 minutes and an hour. When the time slot is an hour, we totally have 168 groups data in a week, so we also selected continuous 168 groups data at other three lengths of time slot. Then we use the first 50 groups data as training data to run CORTEN ten times respectively at different lengths of time slot and different number of prediction points, and obtain the prediction results of the remaining 118 groups data. Then we calculate the average of the 118 sets of prediction results at every number of of prediction points and every length of time slot, which are plotted in Figure 7.

Through Figure 7, we can see that as for both the FA Rate and the WS Loss Rate, when the time slot is shorter, the prediction results are more accurate. The reason is when the time slot is shorter, the change of signal strength between adjacent time slots is smaller and the prediction is more accurate. The average FA Rate and WS Loss Rate at different length of time slot are shown in Table I. The results of Table I indicate CORTEN can obtain accurate prediction at different lengths of time slot. In addition, when the number of prediction points changes from 1 to 22, the results are to show a gradually reduced trend. And it is clear from the figure that the more prediction points are, the better the prediction results are. In practical applications, we can choose the appropriate length of time slot and the number of prediction points based on our requirements for the accuracy and how many spectrum sensors we have.

C. Performance Comparison with the Baseline Approach

To the best of our knowledge, CORTEN is the first mechanism to do indoor white space prediction, so we do not have a suitable contrast. In order to illustrate the prediction performance of CORTEN, we need to define a suitable baseline approach. We know FIWEX is the most accurate mechanism to explore the current indoor white space, and the authors used strong channels as the fixed channels to improve the accuracy of exploration. The strong channel means the signal strength of a channel is always much greater than the white space threshold for a continuous period of time. So in FIWEX, the only information we know about next time slot is the fixed signal strength of the strong channels, thus we can only use the strong channels to recover the prediction matrix. We then use compressive sensing technique to recover the prediction matrix based on the strong channels in our measurements as our baseline approach. In this part, due to the space limitation, we only illustrate the prediction results comparison between CORTEN and the baseline approach when the time slot is an hour and there are 5 and 10 prediction points. Similar results will be obtained when other time slots and prediction points are selected. In the 22 experimental points, we select 5 and 10 prediction points respectively and run CORTEN in the same way with the previous part. Similarly, we use the baseline approach to predict the same data groups to get the prediction results at the same time. Then we compare the prediction results of CORTEN with the baseline approach. In our data set, the location and the number of strong channels are fixed. Therefore, the prediction accuracy of the baseline approach is fixed, and it has nothing to do with the number of prediction points, so the values of FA Rate and WS Loss Rate are the same at 5 and 10 prediction points. The prediction results comparison is shown in Figure 8.



(a) FA Rate comparison

(b) WS Loss Rate comparison

(c) CDF comparison of FA Rate

(d) CDF comparison of WS Loss Rate

Figure 8. Results of comparison with the baseline approach

As shown in Figure 8(a) and Figure 8(b), when there are 5 prediction points, the average FA Rate and WS Loss Rate of CORTEN are 0.82% and 25.33% respectively, while the baseline approach performs an average FA Rate 1.19% and an average WS Loss Rate 36.68%. And the average FA Rate and WS Loss rate of CORTEN are 0.60% and 21.62% respectively at 10 prediction points. The results show that CORTEN has lower or better FA Rate and WS Loss Rate compared with the baseline approach when 5 and 10 prediction points are selected. We also depict the CDF (Cumulative Distribution Function) curves of the FA Rate and the WS Loss Rate after calculating the FA Rate and the WS Loss Rate of each time slot. Figure 8(c) shows the CDF curves of the FA Rate of the baseline approach and CORTEN at 5 and 10 prediction points. In the baseline approach, 61.86% time slots suffer a FA Rate that is higher than 1%, while the number is only 23.73% and 10.17% respectively at 5 and 10 prediction points in CORTEN. And Figure 8(d) shows the CDF curves of WS Loss Rate of the baseline approach and CORTRN, from which we can know that most time slots suffer a WS Loss Rate that is high than 30% in the baseline approach, while there are few time slots is up to 30% in CORTEN.

From Table I, we know the average 0.58% FA Rate and 22.47% WS Loss Rate at all different number of prediction points from 1 to 22 when the time slot is an hour. Compared with the FA Rate of 1.19% and the WS Loss Rate of 36.68% in the baseline approach, CORTEN has an average absolute FA Rate improvement of 0.61% and an average relative FA Rate improvement of 51.3%, an average absolute WS Loss Rate improvement of 14.21% and an average relative WS Loss Rate improvement of 38.7%.

VI. Related Work

Most of the existing works on indoor TV white spaces focused on how to accurately explore the current indoor TV white spaces. For example, in [5], the authors proposed the first indoor white space identification system, called WISER. WISER utilized the channel-location clustering based algorithm to explore indoor white spaces, and obtained great improvement compared to the baseline approaches. Then in [6], the authors designed a cost-efficient indoor white space exploration based on compressive sensing, named FIWEX, to improve the accuracy of exploration. In [7], the authors

Table I
The average FA Rate and WS Loss Rate at different length of time slots

	5 minutes	10 minutes	30 minutes	60 minutes
average FA Rate (%)	0.32	0.41	0.54	0.58
average WS Loss Rate (%)	19.32	20.59	21.36	22.47

presented the first exploration system that was not based on training data.

As the signal strength of spectrum changes over time, in order to more accurately and effectively use the TV white spaces, we need to accurately predict the spectrum. There were some works on outdoor white space prediction before, for example, in [24]–[26], 1st-order Markov Chain model was used to predict the spectrum, but the prediction accuracy was not satisfactory. In [27], the authors used the improvement frequent pattern mining in data mining [28] method to predict the outdoor white space and got a better prediction result. In this paper, we designed a real-time accurate indoor white space prediction mechanism named CORTEN based on the ARIMA model in time series analysis, and proved that CORTEN had a good prediction effect. To the best of our knowledge, this is the first system to predict the indoor white spaces.

VII. Conclusion

In this paper, we performed indoor white space measurements in a real building to study the characteristics of indoor white spaces. The measurement results confirm that the white space changes over time and has a significant periodicity, which is the basis of our prediction. Motivated by these observations, we proposed a real-time accurate indoor white space prediction mechanism, called CORTEN. At first, we selected the prediction points based on the correlation between the spectral signal strength at different locations. And then the spectral signal strength was predicted according to the correlation and periodicity of the spectrum in time. Finally, the prediction matrix was recovered based on the correlation of the spectral signal strength at different locations and different frequency. Through validation by the real experimental data, CORTEN achieved good prediction effects.

References

[1] Xiaojun Feng, Jin Zhang, and Qian Zhang. “Database-assisted multi-ap network on tv white spaces: Architecture, spectrum allocation and ap discovery”. In *DySPAN*, 2011.

[2] David Gurney, Greg Buchwald, Larry Ecklund, Steve Kuffner, and John Grosspietsch. “Geo-location database techniques for incumbent protection in the TV white space”. In *DySPAN*, 2008.

[3] Neil E. Klepeis, William C. Nelson, Wayne R. Ott, John P. Robinson, Andy M. Tsang, Paul Switzer, Joseph V. Behar, Stephen C. Hern, and William H. Engelmann. “The National Human Activity Pattern Survey (NHAPS): a resource for assessing exposure to environmental pollutants”. *Journal of Exposure Science and Environmental Epidemiology*, 11(3):231, 2001.

[4] Vikram Chandrasekhar, Jeffrey G. Andrews, and Alan Gatherer. “Femtocell Networks: A Survey”. *IEEE Communications magazine*, 46(9):59–67, 2008.

[5] Xuhang Ying, Jincheng Zhang, Lichao Yan, Guanglin Zhang, Minghua Chen, and Ranveer Chandra. “Exploring Indoor White Spaces in Metropolises”. In *MOBICOM*, 2013.

[6] Dongxin Liu, Zhihao Wu, Fan Wu, Yuan Zhang, and Guihai Chen. “FIWEX: Compressive Sensing Based Cost-Efficient Indoor White Space Exploration”. In *MOBIHOC*, 2015.

[7] Dongxin Liu, Fan Wu, Linghe Kong, Shaojie Tang, Yuan Luo, and Guihai Chen. “Training-Free Indoor White Space Exploration”. *IEEE Journal on Selected Areas in Communications*, 34(10):2589–2604, 2016.

[8] George E.P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[9] Hae-Sang Park and Chi-Hyuck Jun. “A simple and fast algorithm for K-medoids clustering”. *Expert systems with applications*, 36(2):3336–3341, 2009.

[10] Universal software radio peripheral. https://www.ettus.com.

[11] Second memorandum option and order. https://apps.fcc.gov/edocs%5fpublic/attachmatch/FCC-10-174A1.pdf.

[12] Tevfik Yucek and Huseyin Arslan. “A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications”. *IEEE communications surveys & tutorials*, 11(1):116–130, 2009.

[13] Harry Urkowitz. “Energy Detection of Unknown Deterministic Signals”. *Proceedings of the IEEE*, 55(4):523–531, 1967.

[14] Joseph Lee Rodgers and W.Alan Nicewander. “Thirteen Ways to Look at the Correlation Coefficient”. *The American Statistician*, 42(1):59–66, 1988.

[15] Rodrigo N. Calheiros, Enayat Masoumi, Rajiv Ranjan, and Rajkumar Buyya. “Workload Prediction Using ARIMA Model and Its Impact on Cloud Applications’ QoS”. *IEEE Transactions on Cloud Computing*, 3(4):449–458, 2015.

[16] Lisham L Singh, Al Muhsen Abbas, Flaih Ahmad, and Srinivasan Ramaswamy. “Predicting Software Bugs Using ARIMA Model”. In *Proceedings of the 48th Annual Southeast Regional Conference*, 2010.

[17] Rob J Hyndman and Yeasmin Khandakar. *Automatic time series for forecasting: the forecast package for R*. Number 6/07. Monash University, Department of Econometrics and Business Statistics, 2007.

[18] Emmanuel J Candès, Justin Romberg, and Terence Tao. “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information”. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[19] Yichao Chen, Lili Qiu, Yin Zhang, Guangtao Xue, and Zhenxian Hu. “Robust network compressive sensing”. In *MOBICOM*, 2014.

[20] Linghe Kong, Mingyuan Xia, Xiaoyang Liu, Minyou Wu, and Xue Liu. “Data loss and reconstruction in sensor networks”. In *INFOCOM*, 2013.

[21] Swati Rallapalli, Lili Qiu, Yin Zhang, and Yi-Chao Chen. “Exploiting temporal stability and low-rank structure for localization in mobile networks”. In *MOBICOM*, 2010.

[22] Fan Wu, Dongxin Liu, Zhihao Wu, Yuan Zhang, and Guihai Chen. “Cost-Efficient Indoor White Space Exploration Through Compressive Sensing”. *IEEE/ACM Transactions on Networking*, PP(99):1–17, 2017.

[23] Jared Tanner and Ke Wei. “Low rank matrix completion by alternating steepest descent methods”. *Applied and Computational Harmonic Analysis*, 40(2):417–429, 2016.

[24] Hang Su and Xi Zhang. “Cross-Layer Based Opportunistic MAC Protocols for QoS Provisionings Over Cognitive Radio Wireless Networks”. *IEEE Journal on selected areas in communications*, 26(1):118–129, 2008.

[25] Matthias Wellens, Alexandre de Baynast, and Petri Mahonen. “Exploiting Historical Spectrum Occupancy Information for Adaptive Spectrum Sensing”. In *WCNC*, 2008.

[26] Qing Zhao, Lang Tong, and Ananthram Swami. “Decentralized cognitive MAC for dynamic spectrum access”. In *DySPAN*, 2005.

[27] Dawei Chen, Sixing Yin, Qian Zhang, Mingyan Liu, and Shufang Li. “Mining Spectrum Usage Data: A Large-scale Spectrum Measurement Study”. In *MOBICOM*, 2009.

[28] Gao Cong, Kian-Lee Tan, Anthony K.H. Tung, and Feng Pan. “Mining Frequent Closed Patterns in Microarray Data”. In *ICDM*, 2004.