

RateCount: Learning-Free Device Counting by Wi-Fi Probe Listening

TIANLANG HE, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

ZHANGYU CHANG, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China and College of Artificial Intelligence, Yango University, Fuzhou, China

ZHONGMING LIN, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

S.-H. GARY CHAN, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

Counting Wi-Fi devices within access point (AP) coverage by listening to their probe request frames (PRFs) is a well-established research problem, fundamental to many Internet of Things (IoT) applications such as crowd management and public transportation scheduling. While commendable counting accuracy has been reported, existing approaches fall short in deployment convenience due to their reliance on machine learning, which necessitates (1) extensive data collection and training efforts for system setup, and (2) specialized model fine-tuning for operational maintenance. We propose RateCount, an accurate, lightweight, and learning-free counting approach to lower deployment costs. RateCount employs a provably unbiased closed-form expression to estimate the device count based on the rate at which APs receive PRFs, along with an error model to compute the estimation variance. We also demonstrate its application in people counting by incorporating a device-to-person calibration scheme. Through extensive real-world experiments conducted at multiple sites spanning a wide range of counts, we show that RateCount, without any deployment costs for machine learning, achieves comparable counting accuracy to the state-of-the-art (SOTA) learning-based device counting and improves previous people counting schemes by a large margin.

CCS Concepts: • **Networks** → **Location based services**; • **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**;

Additional Key Words and Phrases: Wi-Fi probing, device counting, MAC address randomization, unbiased estimator, people counting

This work was supported, in part, by Research Grants Council Collaborative Research Fund (under grant number C1045-23G), and RGC-General Research Fund (under grant number 16201625).

Authors' Contact Information: Tianlang He (corresponding author), Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China; e-mail: theaf@cse.ust.hk; Zhangyu Chang, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China, and College of Artificial Intelligence, Yango University, Fuzhou, Fujian, China; e-mail: zchang@cse.ust.hk; Zhongming Lin, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China; e-mail: zmhkust@ust.hk; S.-H. Gary Chan, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China; e-mail: gchan@cse.ust.hk.



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

© 2027 Copyright held by the owner/author(s).

ACM 2577-6207/2027/02-ART3

<https://doi.org/10.1145/3833084>

ACM Reference Format:

Tianlang He, Zhangyu Chang, Zhongming Lin, and S.-H. Gary Chan. 2027. RateCount: Learning-Free Device Counting by Wi-Fi Probe Listening. *ACM Trans. Internet Things* 8, 1, Article 3 (February 2027), 22 pages. <https://doi.org/10.1145/3833084>

1 Introduction

A Wi-Fi-enabled device, or simply *Wi-Fi device* in this article, actively manages its wireless connection by sporadically broadcasting **probe request frames (PRFs)** to the **access points (APs)** in its proximity. The frames are broadcast no matter whether the device has been connected to an AP, with intervals generally ranging from seconds to minutes. Although each Wi-Fi device has a unique physical MAC address, to protect user privacy, unconnected devices often fabricate random (or virtual) MAC addresses in their PRFs, known as *MAC address randomization* [10, 37]. As a result, an AP may receive multiple distinct virtual MAC addresses from an unconnected device while the device remains in the coverage area.

As reported in much of the literature, the number of Wi-Fi devices in a *counting area* (defined as the joint coverage of one or more APs) may be estimated through listening to their PRFs broadcast within a window (of a few to tens of minutes) [6, 15, 29]. Such *device counting* is fundamental to many **Internet of Things (IoT)** applications, such as crowd estimation, Wi-Fi popularity assessment, and public transportation scheduling [5, 16, 22, 30, 39, 49]. The **state-of-the-art (SOTA)** approaches count devices by the number of MAC addresses. To prevent overcounting due to the MAC address randomization, they employ machine learning to associate PRFs with their broadcast devices [7, 9, 21, 27]. While accurate counting is achieved, their system deployment is often costly. For one thing, the model training for PRF association requires massive efforts for collecting and curating training data, and for tuning model hyperparameters [3, 50]. For another, the association model needs regular fine-tuning to accommodate new device brands and models [9, 28]. Hence, deploying these approaches is costly and time-consuming, requiring continuous human efforts, professional expertise, and computing power to support machine learning.

For fast deployment and easy maintenance, we study the number of PRFs over a window, i.e., *probing rate*. Intuitively, whether MAC addresses are randomized or not, APs typically receive more PRFs with more devices in a counting area. Thus, we may use probing rates to infer device counts, free from the need for PRF association or machine learning. While simple, this idea, to the best of our knowledge, has not been systematically studied in the literature, leaving three critical questions open: (1) What is the physical meaning of the device count estimated from the probing rate, and given that meaning, how can it be estimated unbiasedly? (2) How to interpret the counting error and determine the window size? (3) How to extend the device count to people counting?

To answer these questions, we propose RateCount, an accurate, lightweight, and learning-free Wi-Fi device counting approach under MAC address randomization. Figure 1 shows the counting process of RateCount. Given a sliding window and a time series of PRFs as listened by APs, RateCount uses two closed-form expressions—a rate model and an error model—to estimate device count and its variance, respectively, and it adapts to environmental changes by itself. RateCount makes the following contributions:

- *A closed-form rate model to count Wi-Fi devices and its unbiased estimator proof*: We propose a rate model—a lightweight, closed-form expression—that estimates device counts by the ratio of the probing rate of all devices within a window to the mean probing rate of a single device. We further prove that our rate model is an unbiased estimator of the device count time-averaged over the window.

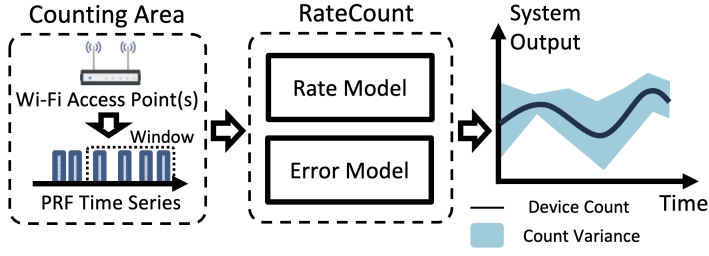


Fig. 1. The counting process of RateCount. Given a sliding window and a time series of PRFs as listened by APs, RateCount uses a rate model to estimate the device count over the window and an error model to compute the estimation variance.

- *A closed-form error model to estimate the count variance:* We derive a closed-form error model for the variance of the estimated counts, in terms of the window size, the number of PRFs, and the interval variance of device probing. We further provide a thorough analysis of the model to understand the counting error and instruct how to determine window size.
- *Extension to people counting:* As a showcase of one major application, we demonstrate how to apply RateCount to people counting, based on our deployment experiences from an existing counting system. Specifically, the system employs a multimodal calibration scheme to estimate the device-to-person ratio in the counting area.

We have implemented RateCount and conducted extensive experiments to count devices and people in various environments, including a lab room, a building entrance, a campus atrium, a study room, and a bus station. Our learning-free RateCount shows comparable accuracy with the SOTA learning-based device counting and significantly reduces the counting error of previous Wi-Fi-based people counting by 66%. Furthermore, our error model demonstrates a sound estimation of the count variance.

The remainder of the article is organized as follows. We first introduce the preliminaries in Section 2. Then, we discuss the design of RateCount in Section 3, followed by its extension to people counting in Section 4. We present illustrative experimental results in Section 5. We finally review related work in Section 6, and conclude in Section 7. We prove that RateCount gives an unbiased estimator in Appendix A.

2 Preliminaries

In this section, we introduce the background knowledge of Wi-Fi probing (Section 2.1) and device counting (Section 2.2). The major symbols are listed in Table 1. Unless otherwise specified, all concepts are relative to a specific window size within a specific counting area.

2.1 Wi-Fi Probing

Figure 2 illustrates three important concepts of Wi-Fi probing that relate to device counting: probing burst, MAC address randomization, and probing interval.

Probing Burst: Wi-Fi devices sporadically broadcast PRFs to probe the nearby APs, known as Wi-Fi probing. To enhance network discovery, Wi-Fi devices broadcast PRFs in bursts, i.e., device repeats PRF broadcast within a very short period (typically less than 4 seconds), with a much longer interval between two bursts (ranging from 30 seconds to a few minutes) [10]. In Figure 2, the duration of a *probing burst* is presented as a colored bar. We define the start time of a probing burst as the *probing instant*, shown as t_i ($i = 0, 1, \dots$), and the interval between two consecutive probing instants as the *probing interval*, denoted as τ_j ($j = 0, 1, \dots$).

Table 1. Summary of Notations (Unless Specified, All Concepts Are Relative to a Specific Window in a Specific Counting Area)

Notation	Definition
U	Set of devices $\{u\}$
V	Set of counting areas $\{v\}$
w	Window size
M	Number of people
$M(t)$	Number of people at time instant t
$\bar{M}$	Average number of people ($\bar{M} = (1/w) \int_0^w M(t)dt$)
N	Number of devices
$N(t)$	Number of devices at time instant t
$\bar{N}$	Average number of devices ($\bar{N} = (1/w) \int_0^w N(t)dt$)
$\hat{N}$	Estimator of $\bar{N}$
p_m	Dwell time of the m th person
d_n	Dwell time of the n th device
D	Total dwell time of all devices ($D = \sum_{n=1}^N d_n$)
b_n	Number of probing bursts from the n th device
B	Total number of probing bursts ($B = \sum_{n=1}^N b_n$)
$\bar{\tau}$	Mean of probing intervals ($\bar{\tau} = \int_0^\infty \tau Pr(\tau v) d\tau$)
σ_τ	Standard deviation of probing intervals
$\bar{r}$	Mean probing rate of all devices ($\bar{r} = 1/\bar{\tau}$)
R	Aggregated probing rate ($R = B/w$)
α	Device-to-person ratio ($\alpha = N/M$)

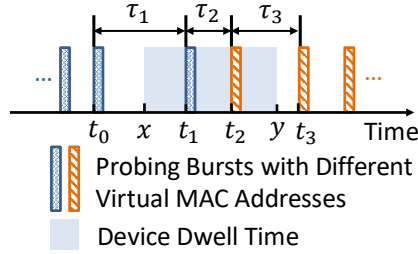


Fig. 2. Probing pattern of a single device and its dwell time in a counting area. The start time of a probing burst is defined as the probing instant, shown as t , and the interval between two consecutive probing instants is defined as the probing interval, denoted as τ . The device dwells in a counting area from time x to y .

MAC Address Randomization: At the data-link layer, PRF carries the MAC address of the device. Although each Wi-Fi device has a unique physical MAC address, to protect user privacy, most devices irregularly fabricate virtual MAC addresses in their PRFs (when unconnected to an AP), known as MAC address randomization [9]. As shown in Figure 2, the probing bursts at t_1 and t_2 carry different MAC addresses (differentiated by color and hatch) due to the MAC address randomization. Hence, an AP can recognize two bursts from the same device if they share the same MAC address, but two bursts from the same device may not carry the same MAC address.¹

¹Here, we ignore the minimal probability that multiple devices share the same virtual MAC address.

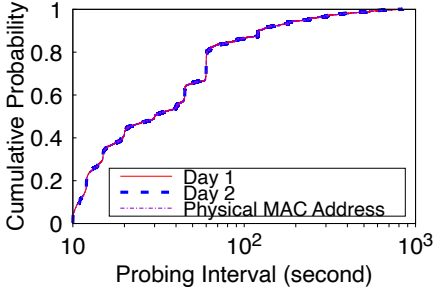


Fig. 3. Distributions of probing interval under different conditions in a counting area.

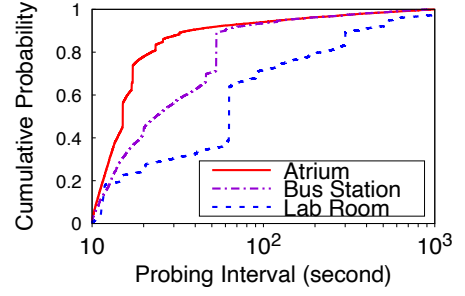


Fig. 4. Distributions of probing interval in different counting areas.

Distribution of Probing Interval: Probing interval is the idle time of a device before it initiates the next probing burst. Typically, probing interval depends on the device’s internal setting (e.g., hardware specifications) and external environment (e.g., signal reception quality); often, it also depends on randomness [10]. Thus, we could interpret the probing interval (τ) as an **independent and identically distributed (IID)** random variable sampled from the statistical distribution that depends on both the device and the external environment. Formally, for a single device $u \in U$ in a counting area $v \in V$, the probability that a probing interval is τ can be expressed as

$$Pr(\tau | u, v), \quad (1)$$

where U is the set of devices $\{u\}$, and V is the set of counting areas $\{v\}$.

We will later reveal that knowing the distribution of probing interval is important to achieve unbiased device counting. Unfortunately, knowing the distribution for each device in each counting area is next to impossible, given the unpredictably diverse brands and models of Wi-Fi devices.² Thus, we have to forgo the device information and collect the interval distribution in each counting area, i.e., the marginal distribution of Equation (1). Formally, the probability that a probing interval is τ in a counting area $v \in V$ is expressed as

$$Pr(\tau | v) = \sum_{u \in U} Pr(\tau | u, v) Pr(u | v). \quad (2)$$

Simply, we can estimate Equation (2) by collecting massive PRFs in a counting area over an extended period.

As a showcase of the data collection, we employ the existing APs from three counting areas (a university atrium, a bus station, and a lab room) for collecting PRFs for multiple days. On average, around 266,000 PRFs were received in each area per day. Figures 3 and 4 show the distributions of probing intervals based on a sliding window of a day, with the intervals calculated from those having the same MAC addresses (note that the entire collection process is automatic, without any human effort). From the two figures, we can conclude that

- *The data collection is sufficient to reflect the area-based distribution:* Figure 3 shows the distributions of one counting area in different days and randomization states (by checking the 7th bit of the information element in PRFs [19]), as the distributions show insignificant discrepancy.³ This suggests that the data collection is sufficient to reflect the interval distribution of this area.

²The distribution may differ from device to device, even among devices of the same brand and model.

³Quantitatively, we made a hypothesis that probing interval is IID given a counting area. Through *Ljung–Box* test and *Kolmogorov–Smirnov* test to validate the independence and identical distribution, respectively, both tests *failed to reject* the hypothesis with *p-values* > 0.9 .

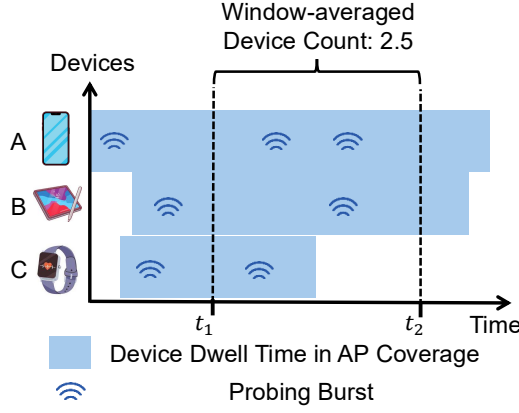


Fig. 5. Example of device counting (with PRFs already associated with their devices). The window-averaged device count is 2.5, with two devices (A and B) fully appearing in the window and one device (C) appearing for half of the window.

- *The data collection is necessary for each counting area:* Figure 4 shows the distribution of probing intervals in different counting areas. Due to the different distributions, the interval distribution collected in one counting area cannot be used for another. Hence, the on-site data collection is necessary.

2.2 Device Counting

Two common definitions of device counting are (1) the number of devices that appear in a window, denoted as $\mathbf{N}$, (2) the number of devices at a specific instant t , denoted as $N(t)$. In this article, we focus on $N(t)$ due to its widespread applications. For example, knowing the $N(t)$ of bus stations benefits the public transportation schedule. Yet, this knowledge is also useful for other crowd management scenarios in reflecting queuing situations (or crowdedness) at public venues, such as canteens, study rooms, banks, hospitals, and washrooms.

As mentioned, the SOTA approach counts devices by counting MAC addresses. While suitable for counting $\mathbf{N}$, it cannot be naively extended to estimate $N(t)$. This is because the probing bursts from different devices are transient and asynchronous. The number of MAC addresses, in nature, cannot indicate the *dwell time* of devices, and thus, we cannot know the device count at any instant. To tackle this, the existing expedient assumes that the device dwell time in a counting area equals the probing intervals. However, this could statistically underestimate the device counts, because the actual device dwell time is expected to be longer than the probing intervals.

To substantiate this, we give a counting example in Figure 5, showing the dwell time and probing bursts of three devices (for simplicity, the probing bursts have already been associated with their devices). In this example, due to the short probing bursts, we cannot know the device counts at any instant (say, t_1 and t_2). Even though we assume the probing intervals to be the devices' dwell time, the counts would be statistically underestimated. For instance, while the counting at t_1 is three, which is accurate (because the probing intervals of A, B, and C overlap at t_1), the two devices at t_2 (i.e., A and B) would be undercounted since the probing intervals are shorter than the actual dwell time.

To address this, we propose to approximate $N(t)$ by the number of devices time-averaged over a window of size w , termed the window-averaged device count, defined as

$$\bar{N} = \frac{1}{w} \int_0^w N(t) dt. \quad (3)$$

As illustrated in Figure 5, the window-averaged device count from t_1 to t_2 is 2.5, because two devices (A and B) appear fully in the window, and one device (C) appears for half of the window. One advantage is that this counting objective removes the need for assuming the device dwell time, thus avoiding the underestimation of devices. Furthermore, so long as we find a proper window size (w), the window-averaged device count can effectively reflect $N(t)$ with an acceptable delay.

In the following, we will discuss how to find a window, and compute the unbiased estimator of $\bar{N}$ and the count variance without using machine learning.

3 RateCount: Learning-Free Device Counting

In this section, we discuss the design of RateCount. We present the rate model for device counting (Section 3.1), explain its approximation of ground truth (Section 3.2), and give a proof of unbiasedness (Appendix A). Finally, we discuss the error model to estimate counting errors (Section 3.3).

3.1 Rate Model

Recall that a counting area is the joint coverage of one or more APs. To estimate the device count ($\bar{N}$) in a counting area given a sliding window (of size w), we propose a closed-form expression, named the rate model. Simply put, the rate model counts devices by calculating the ratio of the aggregated probing rate of all devices to the mean probing rate of a single device.

First, we define the *aggregated probing rate* as the rate at which APs receive probing bursts in a window, which is calculated as

$$R = \frac{B}{w}, \quad (4)$$

where B is the total number of probing bursts received in the window.

Second, we define the *mean probing rate* as the average rate that a single device probes in a counting area, which is calculated as

$$\bar{r} = \frac{1}{\bar{\tau}}, \quad (5)$$

where $\bar{\tau}$ is the *mean probing interval* in the counting area. Here, the mean probing interval can be derived from the area-based distribution in Equation (2), which is defined as

$$\bar{\tau} = \int_0^{\infty} \tau Pr(\tau | v) d\tau. \quad (6)$$

Third, the rate model counts devices by the *ratio* of the aggregated probing rate to the mean probing rate, which is given as

$$\hat{N} = \frac{R}{\bar{r}} = \frac{B}{w\bar{r}} = \frac{B\bar{\tau}}{w}. \quad (7)$$

While simple, the rate model ($\hat{N}$) is an unbiased estimator of the ground truth ($\bar{N}$), which will be explained next.

3.2 Approximation of Ground Truth

Explanation: The rate model ($\hat{N} = R/\bar{r}$) intuitively reflects the device counts in a counting area. By definition, the aggregated probing rate (R) describes how frequently all devices probe, and the mean probing rate ($\bar{r}$) describes how frequently a single device probes on average. If there is only one device, R is expected to equal $\bar{r}$, so we have $E(R/\bar{r}) = 1$. If there are N such devices, R is expected to be N times as frequent as it is in the single-device scenario, i.e., $E(R) = N\bar{r}$, such that we have $E(R/\bar{r}) = N$. Here, the explanation is based on intuition, without considering device dwell time (i.e., $N = \bar{N}$). In practice, R needs to be measured from a window (of size w). Below, we explain that the rate model ($\hat{N} = B\bar{\tau}/w$) approximates the window-averaged device count ($\bar{N}$).

As mentioned, N is the total number of devices that appear in a window. Let d_n denote the dwell time of the n th device, during which it broadcasts b_n bursts ($n = 0, 1, \dots, N$). The total dwell time of all devices in the window is

$$D = \sum_{n=1}^N d_n. \quad (8)$$

Recall that $\bar{\tau}$ is the mean probing interval, which estimates the expectation of the probing intervals in a counting area. One device is expected to probe once if it dwells in the area for a duration of $\bar{\tau}$. Conversely, we expect one device to dwell *longer* when it broadcasts *more* probing bursts.⁴ Therefore, the expectation of the dwell time of the n th device is $b_n \bar{\tau}$, which gives us this approximation:

$$d_n \approx b_n \bar{\tau}. \quad (9)$$

Extending to the case of N devices, the total dwell time can be approximated by

$$D = \sum_{n=1}^N d_n \approx \sum_{n=1}^N (b_n \bar{\tau}) = B \bar{\tau}, \quad (10)$$

note that the approximation error is negligible for sufficiently large N and w , by the *law of large numbers*. On the other hand, according to the definition of $\bar{N}$ in Equation (3), the total dwell time is

$$D = \int_0^w N(t) dt = \bar{N} w. \quad (11)$$

Combining Equations (10) and (11), we have

$$D = \bar{N} w \approx B \bar{\tau}. \quad (12)$$

Recall that $\hat{N}$ is the rate model, and $\bar{N}$ is the ground truth of the window-averaged device count.

With Equations (4) and (5), we rewrite Equation (12) as

$$\bar{N} \approx \frac{B/w}{1/\bar{\tau}} = \frac{R}{\bar{r}} = \hat{N}. \quad (13)$$

This explains that the rate model ($\hat{N}$) approximates the window-average device count ($\bar{N}$).

Example: We elaborate on the approximation using an example of a single-device scenario, as shown in Figure 2, which can be extended to the case of multiple devices. In the figure, the APs received $b_n = 2$ bursts separately at t_1 and t_2 , and the device enters the area at time x and leaves at y . Since the entering and leaving times are largely independent of the probing, x could be any value between t_0 and t_1 , and y could be any value between t_2 and t_3 .⁵ Therefore, the expected dwell time of this device is calculated as

$$\begin{aligned} E(d_n) &= E(y - x) \\ &= \frac{1}{t_3 - t_2} \int_{t_3}^{t_2} \left(\frac{1}{t_1 - t_0} \int_{t_1}^{t_0} (y - x) dx \right) dy \\ &= \frac{t_3 + t_2}{2} - \frac{t_1 + t_0}{2} = \frac{\tau_1 + 2\tau_2 + \tau_3}{2}. \end{aligned} \quad (14)$$

Since τ is an IID random variable sampled from the interval distribution in this counting area, $\bar{\tau}$ is the expectation of the variables τ_1, τ_2 and τ_3 . We have $E(d_n) = 2\bar{\tau} = b_n \bar{\tau}$, where $b_n = 2$ in this example.

⁴The device may broadcast zero or multiple bursts during $\bar{\tau}$, but the expected value is the mean across all possibilities.

⁵It should be noted that x cannot be too close to t_0 since a burst of PRFs could last for a few seconds, but this is negligible because $\bar{\tau}$ is much larger.

Proof: We further prove that the rate model is an unbiased estimator of the window-averaged device count, which is *independent* of the distribution of probing interval. The proof is based on the *Elementary Renewal Theorem* [12]. Interested readers could find it in Appendix A.

3.3 Error Model

We study the **mean squared error (MSE)** of the rate model. The error consists of counting variance, denoted as $\text{Var}(\hat{N})$, and counting bias, denoted as $\text{Bias}(\hat{N}, \bar{N})$, given as

$$\begin{aligned} \text{MSE}(\hat{N}) &= \mathbb{E}[(\hat{N} - \bar{N})^2] \\ &= \mathbb{E}[(\hat{N} - \mathbb{E}(\hat{N}))^2] + \mathbb{E}[(\mathbb{E}(\hat{N}) - \bar{N})^2] \\ &= \text{Var}(\hat{N}) + \text{Bias}^2(\hat{N}, \bar{N}). \end{aligned}$$

Since the rate model is an unbiased estimator, we neglect the bias term. The MSE can be estimated by counting variance, shown as

$$\text{MSE}(\hat{N}) \approx \text{Var}(\hat{N}). \quad (15)$$

Here, the number of bursts (B) and the window size (w) are parameters. The count variance depends on the variance of the mean probing interval, shown as

$$\text{Var}(\hat{N}) = \text{Var}\left(\frac{B\bar{\tau}}{w}\right) = \frac{B^2}{w^2} \text{Var}(\bar{\tau}). \quad (16)$$

Note that $\text{Var}(\bar{\tau})$ represents the variance of the mean value over a group of samples with a sample size of B (i.e., the number of bursts). Since τ is an IID random variable, the count variance depends on the variance of its sample mean, i.e.,

$$\text{Var}(\bar{\tau}) = \frac{\sigma_\tau^2}{B}, \quad (17)$$

where σ_τ is the standard deviation of τ that is derived from Equation (2). By combining Equations (16) and (17), the error model is given as

$$\text{MSE}(\hat{N}) \approx \text{Var}(\hat{N}) = \frac{B\sigma_\tau^2}{w^2}. \quad (18)$$

Overall, the error of the rate model depends on the window size (w), the number of bursts in the window (B), and the standard deviation of the probing intervals (σ_τ). The error model conveys the following messages:

- *How to determine the window size:* The window size needs to balance the real-time performance and counting accuracy. Intuitively, we prefer a shorter window to approximate $N(t)$ for lower counting delay. However, according to the error model, a too small window could lead to sparse samples of probing bursts, resulting in large counting errors. Therefore, *we should select the largest window as long as the delay is acceptable*. For example, a short window (say, 3 minutes) may be used at a bus stop to promptly reflect the crowd changes, while a longer window (say, 15 minutes) may be suitable for a study room to tune air conditioners.
- *Why does probing interval affect counting error:* From the AP perspective, probing bursts can be viewed as the samples of the device dwell time, based on which the rate model estimates device counts. Thus, a larger variance of probing interval leads to larger uncertainty of device counting in the counting area, resulting in errors.
- *Why does MSE increase with the number of probing bursts (B):* The increase of ground-truth device count concurrently leads to the increase of both B and MSE, not that B leads to MSE.

This is supported by the fact that the MSE relative to the ground truth (i.e., NRMSE) decreases with the number of probing bursts, which is shown as

$$\text{NRMSE}(\hat{N}) = \frac{\sqrt{\text{MSE}(\hat{N})}}{\bar{N}} \approx \frac{\sigma_\tau}{\sqrt{\bar{N}w\bar{\tau}}} = \frac{\sigma_\tau}{\bar{\tau}\sqrt{B}}, \quad (19)$$

note that $\bar{N} \approx B\bar{\tau}/w$ from Equation (7). The conclusion is that the absolute counting error *increases* with the number of ground-truth devices, while the relative error *decreases* with the ground truth. In other words, the counting of the rate model is more accurate in crowded scenarios in terms of relative errors.

4 Application in People Counting

In this section, we demonstrate how to extend RateCount to people counting, drawing on our experience with an existing system. We will discuss the system deployment (Section 4.1), explain the device-to-person ratio (Section 4.2), and present the error analysis (Section 4.3).

4.1 System Deployment

Literature has shown that the number of devices can effectively reflect crowd size [13, 15]. Yet, some scenarios prefer the more fine-grained people counts. To fulfill this need, we introduce a calibration scheme that extends RateCount to people counting.

The calibration scheme is intuitive. Device and person counts differ by how many devices a person carries. Thus, we estimate the ratio of the number of devices that a person carries in a counting area, i.e., the *device-to-person* ratio, defined as

$$\alpha = \frac{\mathbf{N}}{\mathbf{M}}, \quad (20)$$

where $\mathbf{M}$ is the number of people that appear in a window. With this ratio, the window-averaged people count in the counting area is estimated as

$$\begin{aligned} \bar{M} &= \frac{1}{w} \int_0^w M(t) dt \\ &\approx \frac{1}{w} \int_0^w \frac{N(t)}{\alpha} dt \approx \frac{\hat{N}}{\alpha}, \end{aligned} \quad (21)$$

Here, $M(t)$ is the number of people at time instant t , and recall that $\hat{N}$ is the rate model.

The device-to-person ratio is shown to be highly related to counting scenarios [14, 34, 43]. For example, in an elderly home, the ratio is usually less than 1 because many residents do not use Wi-Fi devices. In contrast, the ratio at a university is often greater than 1, since most students have more than one device (e.g., smartphones, wristbands, laptops). Therefore, to estimate the ratio in a counting area, we may calculate the ratio from the same scenario in the vicinity. For example, we may calculate the device-to-person ratio at a campus entrance and apply the ratio to a classroom, leveraging the people-sensing facilities (such as a surveillance camera) that campus entrance usually installs. There, we refer to the sensing area of people sensing in the vicinity as the *calibration region*.

Figure 6 illustrates our existing system deployment in a university campus. The calibration region is in an atrium, and we want to count people at a bus station, a study room, and a building entrance. Figure 7 shows the sensor deployment in the calibration region, where 6 APs and 1 camera are used to estimate the device-to-person ratio. Specifically, we use RateCount to compute $\hat{N}$ and the object detection [41] to estimate the window-averaged people count, denoted as $\hat{M}$ (an example of

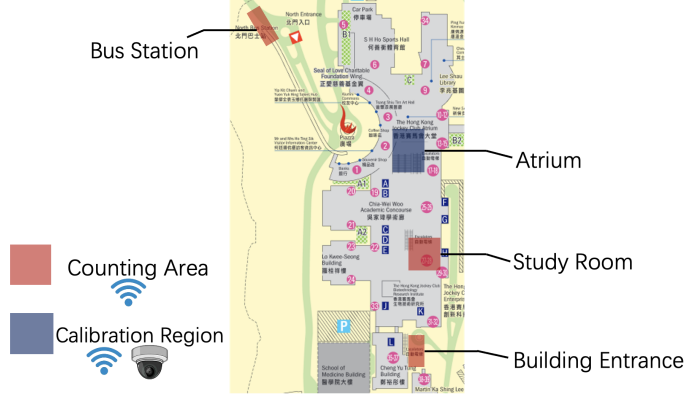


Fig. 6. System deployment of people counting in a university campus. The device-to-person ratio is estimated from a multimodal calibration region and is used for people counting in other areas.

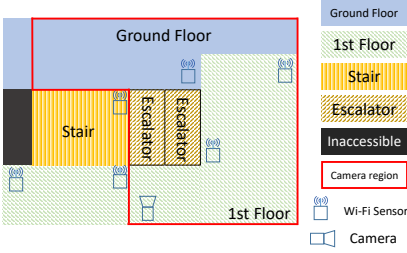


Fig. 7. Sensor deployment in calibration region. The infrastructural APs and cameras are used to estimate the device-to-person ratio.

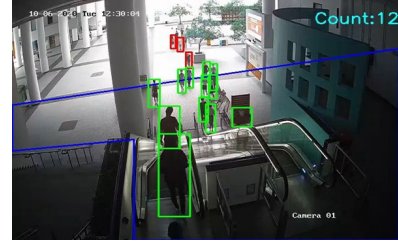


Fig. 8. Example of system operation in calibration region.

estimating $\hat{M}$ is shown in Figure 8). Then, we estimate the device-to-person ratio by

$$\alpha \approx \frac{\hat{N}}{\hat{M}}. \quad (22)$$

The approximation will be explained later.

For the multimodal calibration, the sensing area of an AP is often wider than that of a camera. To avoid the spill-over of Wi-Fi signals, we only count the probing bursts from the devices within the camera's counting area. Specifically, the probing bursts are counted by

$$B' = \sum_{i=1}^B Pr(u_i \in v), \quad (23)$$

where v is the counting area of camera, and u_i is the device that broadcasts the i th burst. In our deployment, we employ SVM to estimate $Pr(u_i \in v)$ using the RSSI of PRFs (more details of such *geofencing technology* can be found in [38]). This technology could also apply to the Wi-Fi counting areas that require a clear semantic boundary.

4.2 Device-to-Person Ratio

As mentioned earlier, the device-to-person ratio is defined as $\alpha = \mathbf{N}/\mathbf{M}$, while in system deployment, we estimate the ratio by $\alpha \approx \hat{N}/\hat{M}$. We have shown that $\hat{N} \approx \bar{N}$, and it is intuitive that $\hat{M} \approx \bar{M}$.

Thus, to justify the system deployment, we explain that

$$\alpha \approx \frac{\bar{N}}{\bar{M}}. \quad (24)$$

Recall that the total dwell time of all devices is the sum of the dwell time of each device. On the other hand, the dwell time of devices equals their users' dwell time.⁶ Thus, the dwell time of all devices is given as

$$D = \sum_{n=1}^N d_n = \sum_{m=1}^M d_m \alpha_m, \quad (25)$$

where d_m is the dwell time of the m th person who has α_m as the number of devices in total.⁷ Here, we consider a general case where a person's dwell time is independent of how many devices it carries. With a sufficiently large number of people, we have

$$\sum_{m=1}^M d_m \alpha_m \approx \frac{N}{M} \sum_{m=1}^M d_m, \quad (26)$$

recalling that N/M is the mean number of devices carried by a person. By combining Equations (25) and (26), we get

$$\frac{\bar{N}}{\bar{M}} = \frac{\frac{1}{N} \sum_{n=1}^N d_n}{\frac{1}{N} \sum_{m=1}^M p_m} \approx \frac{\frac{N}{M} \sum_{m=1}^M p_m}{\sum_{m=1}^M p_m} = \alpha. \quad (27)$$

This explains that α can be estimated by Equation ((24). From the analysis, the calibration region is preferably set up in an area with large crowds, such as a campus atrium or the entrance of a shopping mall. Also, a large window size is preferred in the calibration region.

4.3 Error Analysis

The error of people counting is similar to that of the device counting. The difference is that we need to consider the error incurred in the calibration scheme. Let $\hat{M}_c$ and $\hat{N}_c$ denote the people count and device count estimated in the calibration region, respectively, and recall that $\hat{N}$ denotes the device count estimated in the counting area. The error of people counting is estimated by the error of device counting in the counting area and the propagation of the estimation errors of the two modalities in the calibration region, which is given as

$$\text{NRMSE}(\hat{M}) = \sqrt{\text{NRMSE}^2(\hat{M}_c) + \text{NRMSE}^2(\hat{N}_c) + \text{NRMSE}^2(\hat{N})}.$$

Here, we estimate the NRMSE of device counting using the error model in Equation (19). The error of people counting may be subject to the sensing technologies. In our experiment, we use camera-based people counting, and its error ($\text{NRMSE}(\hat{M}_c)$) is empirically estimated.

5 Illustrative Experimental Results

In this section, we show implementations and extensive experiments to validate RateCount. We discuss the experimental setting (Section 5.1), and present experimental results of device counting (Section 5.2) and people counting (Section 5.3).

⁶Devices that are not associated with any users should be excluded using the method in [37].

⁷Rigorously, α_m represents the device-to-person ratio. Since it is specific to one person (the m th person), its value can be interpreted as the number of devices that the person carries.

5.1 Experimental Setup and Performance Metrics

Our experiments include device counting and people counting in different subareas of a university campus. To validate device counting, we implement RateCount in a laboratory room, where we manually track the entering and leaving times of each device to provide ground truth. To validate people counting, we set up a calibration region in an atrium and count people at a bus station, a building entrance, and a study room, as mentioned in Section 4.1. At each site, we employ the infrastructural Wi-Fi APs (GL-AR150 with OpenWrt 18.06) to capture PRFs based on Wi-Fi sniffers implemented with libcap. Once received, the PRFs are transmitted to a personal computer (with an Intel Core i7-4770 3.6 GHz CPU and 32 GB RAM) that runs RateCount. To obtain the near-ground truth of people counting, we temporarily install cameras in those sites. Note that the system hardware was chosen empirically, and our approach is independent of this choice.

We compare RateCount with the following device counting and people counting approaches:

- *MAC Counting* [29] is a classic device-counting method that counts devices by the number of unique MAC addresses. In the experiment, we implement the method by computing the counts averaged over a window. To account for the MAC address randomization, we apply a ratio to calibrate the counts, termed *MAC Counting (Cali)*.
- *Learning-based Counting* [15] is the SOTA device-counting approach that uses machine learning to associate the PRFs from the same devices. In the experiment, we implement the method based on the association results over a window.
- *CountMeIn* [34] is the SOTA probing-based people-counting approach that calibrates the number of MAC addresses in a window to people counts using a camera-based calibration scheme.

We evaluate our approach in terms of window-averaged device counting. Let $\hat{y}$ represent a counting result, $\bar{y}$ represent the ground truth, and J represent the number of observations in the experiment. We use the following metrics:

- *Root-Mean-Square Error (RMSE)*:

$$\text{RMSE}(\hat{y}) = \sqrt{\frac{1}{J} \sum_{j=1}^J (\hat{y}_j - \bar{y}_j)^2}. \quad (28)$$

- *Mean Absolute Percentage Error (MAPE)*:

$$\text{MAPE}(\hat{y}) = \frac{1}{J} \sum_{j=1}^J \frac{|\hat{y}_j - \bar{y}_j|}{\bar{y}_j}. \quad (29)$$

- *Normalized Root-Mean-Square Error (NRMSE)*:

$$\text{NRMSE}(\hat{y}) = \frac{\text{RMSE}(\hat{y})}{\frac{1}{J} \sum_{j=1}^J \bar{y}_j}. \quad (30)$$

Specifically, RMSE calculates the counting error in terms of the number of devices or people, and MAPE and NRMSE reflect the error relative to the ground truth. It is worth noting that RMSE and NRMSE calculate errors based on the second-order norm, which is more sensitive to large errors than MAPE.

Unless otherwise specified, the distributions of probing intervals were obtained using a 1-day sliding window (examples are shown in Figure 4). We use a 15-minute window for counting in the lab room (since users there often stay for extended periods) and a 3-minute window in other sites. The step size of the sliding window is 3 minutes.

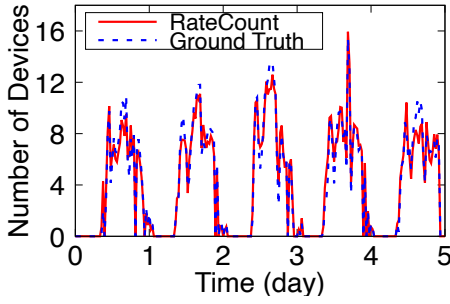


Fig. 9. Device counting in a lab room.

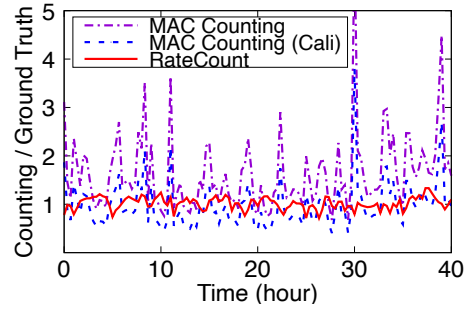


Fig. 10. Comparison between RateCount and the MAC-counting approach.

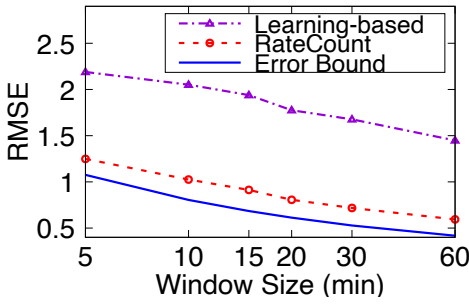


Fig. 11. RateCount achieves comparable accuracy with the SOTA learning-based approach.

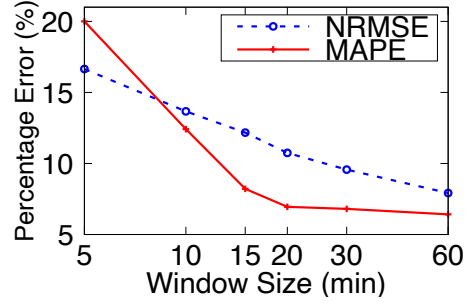


Fig. 12. Error analysis in device counting.

5.2 Device Counting

Figure 9 shows the results of device counting in a lab room over five days. Therein, RateCount accurately tracks the ground truth. In particular, the counting results effectively reflect the usage pattern of the lab room: most lab users arrive in the morning and leave in the evening, with the number of devices peaking at noon. This result, using only the infrastructural Wi-Fi APs without any sensor installation or learning from datasets, has been used to tune the air conditioning in the lab room, effectively.

In Figure 10, we compare RateCount with the classic MAC counting method in device counting, showing the counting results over the ground truth (the cases of divided-by-zero have been removed). While both approaches are free from machine learning, RateCount outperforms MAC counting by a large margin. This is because the probing rate is independent of MAC address randomization, making it more stable than the number of MAC addresses for device counts. Furthermore, simply calibrating the MAC counts cannot address the randomization issue (here, we empirically set the calibration ratio to 0.5, though other ratios lead to the same conclusion). This validates that achieving accurate learning-free device counting is nontrivial.

In Figure 11, we compare RateCount with the SOTA learning-based device counting, showing RMSE versus window size. Overall, the counting accuracy grows with window size, which agrees with our error model that a larger window leads to smaller counting errors (see (18)). In the figure, RateCount achieves comparable (or even slightly better) results than the learning-based approach, without any deployment costs for machine learning. For one thing, RateCount is free from the PRF association that learning-based approaches necessitate, which might lead to minor errors (subject to the effectiveness of the association model). For another, the rate model of RateCount

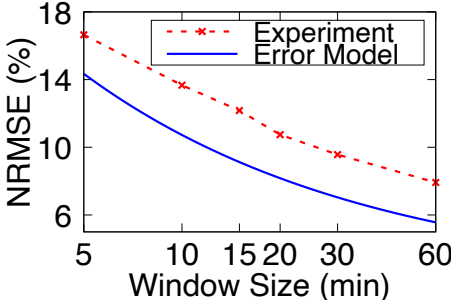


Fig. 13. Comparison between error model and experimental errors in device counting.

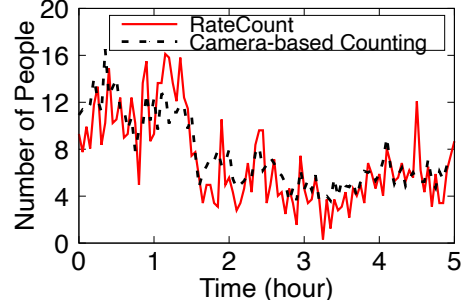


Fig. 14. People counting at a campus atrium.

is an unbiased estimator of the device counts, which is free from the underestimation issue, as discussed in Section 2.2.

In Figure 12, we present an error analysis of RateCount. As mentioned, the error reduces with an increasing window size, as a larger window leads to lower count variance. Furthermore, the error is more evenly distributed in NRMSE than in MAPE. This indicates that the counting is accurate most of the time, while in very rare cases, there are large errors due to drastic changes in the number of devices.

In Figure 13, we compare the error model with the experimental errors. In the figure, the error model not only outlines the error lower bound but also accurately estimates the counting error. Specifically, the experimental errors are larger than the error model by 2% in NRMSE, which is negligible (say, if there are 10 devices in total, the gap is 0.2 devices). This has validated the applicability of our error model in practice.

5.3 People Counting

In our experiment of people counting, the device-to-person ratios of the three sites (i.e., the atrium, bus station, and building entrance) are similar (around 1.14), because these sites share a similar scenario, as mentioned in Section 4.1. Figure 14 presents the people counting of RateCount in the atrium. There, our privacy-preserving Wi-Fi-based approach accurately tracks the outputs of the camera-based people counting, showing comparable counting accuracy with the camera-based approach. Note that the camera-based counting approach only provides the near-ground truths, which might not be entirely accurate. In our experiment, we observe that the error of the camera-based people counting is around 0.08 in terms of NRMSE.

In Figure 15, we compare RateCount with the SOTA probing-based people counting approach (i.e., CountMeIn). The difference of the two approaches is that CountMeIn calibrates MAC counting to the people counts, while ours is based on RateCount. In the figure, the counts of CountMeIn fluctuate significantly, since the fluctuating counts of the MAC-counting approach propagate to people counting (due to the MAC address randomization). In comparison, our people counting achieves much lower counting variance owing to the stable and accurate device counting of the rate model.

Figure 16 shows the performance of RateCount at a bus station during a peak hour. This site is more crowded compared with the atrium scenario. In the figure, RateCount shows consistent results with camera-based people counting, effectively capturing the usage pattern that students went to the bus station after the last class concluded at 5:30 PM. Such information is important for planning the bus schedule, acquired in a privacy-preserving way. Overall, this experiment validates RateCount's scalability to high crowd levels.

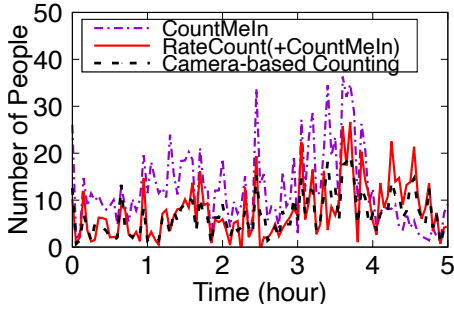


Fig. 15. Evidence that RateCount improves the SOTA Wi-Fi-based people counting scheme, demonstrated at a building entrance.

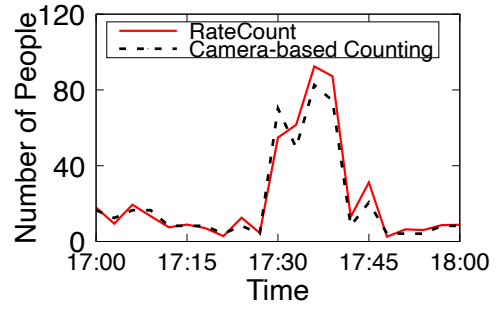


Fig. 16. People counting at a bus station during rush hour.

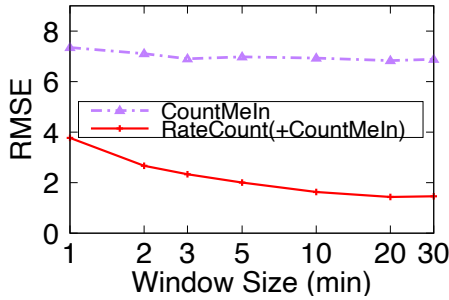


Fig. 17. Error of people counting at a building entrance.

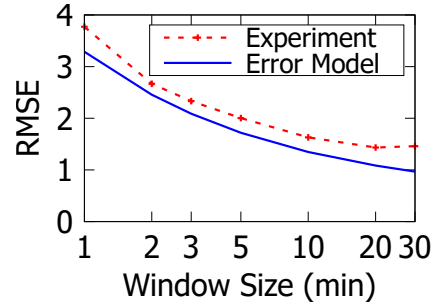


Fig. 18. Comparison between error model and experimental errors in people counting.

Figure 17 shows the RMSE of CountMeIn and RateCount at a building entrance, using camera-based people counting as the near-ground truth. In the figure, the RMSE of both approaches reduces with the window size, which is as expected. However, the RMSE of RateCount is not only much lower than that of CountMeIn, but it also reduces faster as the window size increases. This is because RateCount is based on probing rate, free from the perturbation of the MAC address randomization. Thus, RateCount shows more similar results to the camera-based people counting. Quantitatively, RateCount reduces around 4.57 RMSE (around 0.66 NRMSE) from the SOTA Wi-Fi-based people-counting approach.

Figure 18 shows an error analysis of people counting in the three sites (the atrium, building entrance, and bus stop). Compared with that of the device counting, the error model of people counting additionally includes the estimation of the people sensing in the calibration region. Thus, the result is consistent with the device counting, where the model satisfactorily reflects the actual errors. This validates that our error model can also effectively provide a sound estimate of the error in people-counting systems.

Finally, we demonstrate the **user interface (UI)** of our existing counting system based on RateCount in Figure 19. The demonstrated counting area is a public study room on a campus, whose 3D model is shown in Figure 19(a). In the area, the real-time people counts are estimated using the onsite APs, as shown in Figure 19(b), and the device-to-person ratio is derived from the atrium. Furthermore, the system visualizes the counting history, as shown in Figure 19(c). The UI has been shown on an LED display and a public website, for use by the university community.

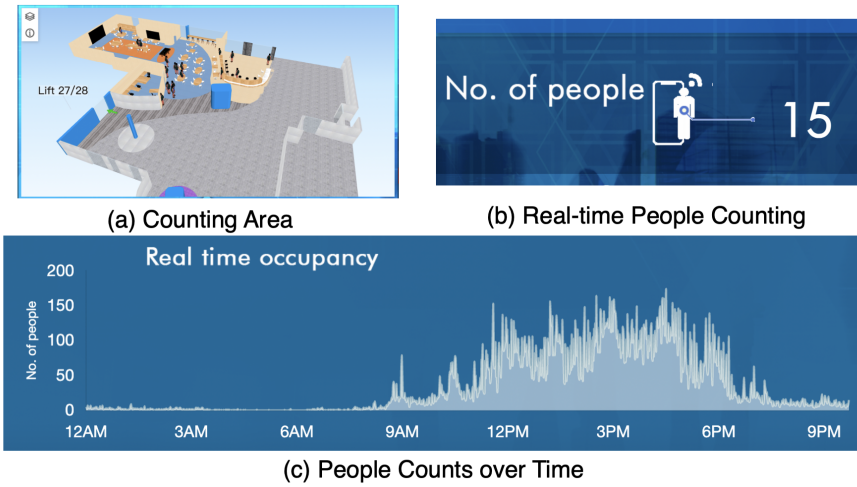


Fig. 19. User interface (UI) of the counting system (based on RateCount) in a study room.

6 Related Work

In this section, we review the related works in Wi-Fi-based device counting and the existing sensing technologies for people counting.

The Counting of Wi-Fi Devices: The simplest method to count Wi-Fi devices is to count the number of devices connected to APs [8, 45, 47]. While straightforward, the number of Wi-Fi connections does not involve the unconnected devices, leading to large counting errors [1, 23, 31, 48]. To account for the unconnected ones, subsequent works count devices using Wi-Fi probing, which counts the number of unique MAC addresses in PRFs within a window, termed the MAC-counting approach [2, 29, 44]. However, this approach is vulnerable to MAC address randomization, where one device may have multiple virtual MAC addresses. Since the randomization process is often difficult to reverse [11, 25, 36, 40], SOTA approaches associate PRFs with their broadcast device using PRF messages, which often rely on machine learning [6, 7, 15, 27]. For example, numerous machine learning models, such as Bayesian networks, decision trees, Gaussian mixture models, and deep learning models, have been studied to associate PRFs based on their messages, including information elements, sequence numbers, received signal strength, and arrival times [6, 15, 21, 26, 32, 37]. Although accurate association has been achieved, deploying these association models is inefficient and time-consuming, as they require dedicated efforts to support machine learning, and their association model needs regular professional expertise to accommodate new Wi-Fi devices and models for system maintenance [3, 50]. Consequently, SOTA approaches are unsuitable for scenarios that require fast deployment, easy maintenance, or those with restricted computing power. In addition, previous approaches often lack a theoretical understanding of the counting bias. To bridge this gap, we propose RateCount, the first learning-free approach to count Wi-Fi devices under MAC address randomization, with theoretically derived counting unbiasedness and error model.

People Counting Technologies: Several sensing technologies have been considered and used for people counting, such as camera, LiDAR, and mmWave radar [24, 30, 35]. However, these sensors often have limited sensing range, incur extra cost due to infrastructure setup, and might pose privacy concerns. To address these issues, Wi-Fi-based people counting has been explored, which can be categorized into active and passive approaches. The active approach counts people by analyzing the physical Wi-Fi signal propagating over the counting area (i.e., channel state

information), which often requires specialized sensing devices [20, 33, 42, 46]. On the other hand, the passive approach listens to the PRFs and studies how to calibrate the number of PRFs to estimate the people count, which, unfortunately, is often inaccurate [4, 13, 17, 18, 43]. In this article, we demonstrate that RateCount, incorporating the calibration scheme, can significantly improve the accuracy of Wi-Fi-based passive people counting.

7 Conclusion

Wi-Fi devices, for network discovery, sporadically broadcast PRFs with randomized MAC addresses to the nearby APs. We study counting the number of Wi-Fi devices within AP coverage by listening to their PRFs over a window. While such a well-established research problem has been achieved accurate, previous counting schemes address MAC address randomization by means of using machine learning to associate the PRFs with their broadcast devices. This, however, results in inefficient and time-consuming system setup (training data curation and cleaning, model training, hyperparameter tuning), sophisticated system maintenance (e.g., model adaptation to accommodate new Wi-Fi device brands and models), and high computing requirements for machine learning.

To lower deployment costs, we propose RateCount, an accurate, lightweight, and learning-free approach that counts devices based on the rate at which APs receive PRFs, i.e., probing rate. RateCount estimates the device count by a simple and effective closed-form rate model. We have proved that the rate model gives an unbiased estimator of the device count time-averaged over a window and deduced the count variance in terms of the window size, number of PRFs in the window, and the variance of the probing interval. Furthermore, we demonstrate how to extend RateCount to people counting via a multimodal calibration scheme based on our experience on an existing counting system. We have implemented RateCount and conducted extensive experiments on device counting and people counting in various diverse environments with various count levels, specifically, in a lab room, a building entrance, a campus atrium, a bus station, and a study room. The experimental results show that our learning-free RateCount can achieve comparable counting accuracy with SOTA learning-based device counting and significantly improve previous Wi-Fi-based people counting by cutting 66% errors. Additionally, our results show that the error model provides a sound estimation of the counting errors.

RateCount is learning-free, unbiased, and scales gracefully to crowded scenarios, which are key properties for deployments that demand fast setup and low maintenance. In future work, we plan to further improve its counting accuracy and environmental adaptability. This may be achieved by incorporating additional PRF features and designing adaptive window-size tuning.

References

- [1] Lannois Carroll-Woolery. 2023. *Localization and Counting of Indoor Populations on A University Campus Using Wi-Fi Connection Logs and Floor Plans*. Master's thesis. University of Waterloo.
- [2] Jean-François Determe, Sophia Azzagnuni, Utkarsh Singh, François Horlin, and Philippe De Doncker. 2022. Monitoring large crowds with WiFi: A privacy-preserving approach. *IEEE Systems Journal* 16, 2 (2022), 2148–2159.
- [3] Shi Dong, Ping Wang, and Khushnood Abbas. 2021. A survey on deep learning and its applications. *Computer Science Review* 40, C (2021), 100379.
- [4] Dorine C. Duives, Tim van Oijen, and Serge P. Hoogendoorn. 2020. Enhancing crowd monitoring system functionality through data fusion: Estimating flow rate from Wi-Fi traces and automated counting system data. *Sensors* 20, 21 (Jan. 2020), 6032. DOI: <https://doi.org/10.3390/s20216032>
- [5] Zefeng Chen, Wensheng Gan, Jiayang Wu, Hong Lin, and Chien-Ming Chen. 2024. Metaverse for smart cities: A surveys. *Internet of Things and Cyber-Physical Systems* 4 (2024), 203–216.
- [6] Abraham Pérez-Hernández, Maydelis N. Barreras-Martín, Juan A. Becerra, María José Madero-Ayora, and Pablo Aguilera. 2024. De-randomization of MAC addresses using fingerprints and RSSI with ML for Wi-Fi analytics. *IEEE Access* 12 (2024), 150857–150868.

- [7] Pegah Torkamandi, Ljubica Kärkkäinen, and Jörg Ott. 2021. An online method for estimating the wireless device count via privacy-preserving Wi-Fi fingerprinting. In *Proceedings of the 22nd International Conference on Passive and Active Measurement, PAM 2021*. Springer-Verlag, Berlin, 406–423. DOI: https://doi.org/10.1007/978-3-030-72582-2_24
- [8] Amee Trivedi, Camellia Zakaria, Rajesh Balan, Ann Becker, George Corey, and Prashant Shenoy. 2021. WiFiTrace: Network-based contact tracing for infectious diseases using passive WiFi sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1, Article 37 (Mar. 2021), 26 pages. DOI: <https://doi.org/10.1145/3448084>
- [9] Ellis Fenske, Dane Brown, Jeremy Martin, Travis Mayberry, Peter Ryan, and Erik Rye. 2021. Three years later: A study of MAC address randomization in mobile devices and when it succeeds. *Proceedings on Privacy Enhancing Technologies* 3 (2021), 164–181.
- [10] Julien Freudiger. 2015. How talkative is your mobile device? An experimental study of Wi-Fi probe requests. In *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. ACM, New York, USA, 1–6.
- [11] Yuki Furuya, Hiromu Asahina, Masashi Yoshida, and Iwao Sasase. 2021. Indoor crowd estimation scheme using the number of Wi-Fi probe requests under MAC address randomization. *IEICE Transactions on Information and Systems* 104, 9 (2021), 1420–1426.
- [12] Robert G. Gallager. 2013. *Stochastic Processes: Theory for Applications*. Cambridge University Press, New York, USA.
- [13] Lifei Hao, Baoqi Huang, Bing Jia, and Guoqiang Mao. 2023. On the fine-grained crowd analysis via passive WiFi sensing. *IEEE Transactions on Mobile Computing* 1, 1 (2023), 1–15. DOI: <https://doi.org/10.1109/TMC.2023.3324334>
- [14] Lifei Hao, Baoqi Huang, Bing Jia, and Guoqiang Mao. 2024. Heterogeneous dual-attentional network for WiFi and video-fused multi-modal crowd counting. *IEEE Transactions on Mobile Computing* 23, 12 (2024), 14233–14247.
- [15] Tianlang He, Jiajie Tan, and S.-H. Gary Chan. 2023. Self-supervised association of Wi-Fi probe requests under MAC address randomization. *IEEE Transactions on Mobile Computing* 22, 12 (2023), 7044–7056. DOI: <https://doi.org/10.1109/TMC.2022.3205924>
- [16] Tianlang He, Zhiqiu Xia, and S.-H. Gary Chan. 2025. Elevator, escalator, or neither? Classifying conveyor state using smartphone under arbitrary pedestrian behavior. *IEEE Transactions on Mobile Computing* 24, 11 (2025), 12626–12639.
- [17] Hande Hong, Girisha Durrel De Silva, and Mun Choon Chan. 2018. CrowdProbe: Non-invasive crowd monitoring with Wi-Fi probe. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3, Article 115 (Sep. 2018), 23 pages. DOI: <https://doi.org/10.1145/3264925>
- [18] Baoqi Huang, Guoqiang Mao, Yong Qin, and Yun Wei. 2021. Pedestrian flow estimation through passive WiFi sensing. *IEEE Transactions on Mobile Computing* 20, 4 (Apr. 2021), 1529–1542. DOI: <https://doi.org/10.1109/TMC.2019.2959610>
- [19] IEEE. 2017. *Guidelines for Use of Extended Unique Identifier (EUI), Organizationally Unique Identifier (OUI), and Company ID (CID)*. Technical Report. IEEE. 1–19.
- [20] Hongbo Jiang, Siyu Chen, Zhu Xiao, Jingyang Hu, Jiangchuan Liu, and Schahram Dustdar. 2023. Pa-count: Passenger counting in vehicles using Wi-Fi signals. *IEEE Transactions on Mobile Computing* 23, 4 (2023), 2684–2697.
- [21] Hongyu Jin and Panos Papadimitratos. 2024. Over-the-air runtime Wi-Fi MAC address re-randomization. In *Proceedings of the 17th ACM WiSec*. ACM, 8–13.
- [22] Danista Khan and Ivan Wang-Hei Ho. 2022. CrossCount: Efficient device-free crowd counting by leveraging transfer learning. *IEEE Internet of Things Journal* 10, 5 (2022), 4049–4058.
- [23] Kai Li, Chau Yuen, and Salil Kanhere. 2015. SenseFlow: An experimental study of people tracking. In *Proceedings of the RealWSN (RealWSN'15)*. ACM, New York, NY, USA, 31–34. DOI: <https://doi.org/10.1145/2820990.2820994>
- [24] Xiyang Liu, Jie Yang, Wenrui Ding, Tieqiang Wang, Zhijin Wang, and Junjun Xiong. 2020. Adaptive mixture regression network with local counting map for crowd counting. In *Proceedings of the ECCV 2020*. Springer, Online, 241–257.
- [25] Jeremy Martin, Travis Mayberry, Collin Donahue, Lucas Foppe, Lamont Brown, Chadwick Riggins, Erik C. Rye, and Dane Brown. 2017. A study of MAC address randomization in mobile devices and when it fails. In *Proceedings of the 2017 PoPETs*, Vol. 2017. De Gruyter Open, Minneapolis, USA, 365–383. DOI: <https://doi.org/10.1515/popets-2017-0054>
- [26] Célestin Matte, Mathieu Cunche, Franck Rousseau, and Mathy Vanhoef. 2016. Defeating MAC address randomization through timing attacks. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. ACM, 15–20.
- [27] Michele Nitti, Francesca Pinna, Lucia Pintor, Virginia Pilloni, and Benedetto Barabino. 2020. iABACUS: A Wi-Fi-based automatic bus passenger counting system. *Energies* 13, 6 (Jan. 2020), 1446. DOI: <https://doi.org/10.3390/en13061446>
- [28] Sheida Nozari, Chiara Garibotto, Andrea Sciarone, Igor Bisio, and Fabio Lavagetto. 2024. Analyzing broadcast patterns and randomization techniques in Wi-Fi probe request frames. *IEEE Network* 39, 4 (2024), 228–234.
- [29] Shokirkhon Oppokhonov, Jae-Hyun Lee, and Jun-young Moon. 2022. An analysis of Wi-Fi probe request for crowd counting through MAC-address classification. *Journal of the Korea Institute of Information and Communication Engineering* 26, 4 (2022), 612–623.
- [30] Zhuoxuan Peng and S.-H. Gary Chan. 2024. Single domain generalization for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Seattle WA, USA, 28025–28034.

- [31] Meng Qu. 2023. Exploring patron behavior in an academic library: A Wi-Fi-connection data analysis. *Education and Information Technologies* 1, 1 (Oct. 2023), 1–22. DOI : <https://doi.org/10.1007/s10639-023-12248-9>
- [32] Pieter Robyns, Bram Bonné, Peter Quax, and Wim Lamotte. 2017. Noncooperative 802.11 MAC layer fingerprinting and tracking of mobile devices. *Security and Communication Networks* 2017, 1 (2017), 6235484.
- [33] Iker Sobron, Javier Del Ser, Inaki Eizmendi, and Manuel Velez. 2018. Device-free people counting in IoT environments: New insights, results, and open challenges. *IEEE Internet of Things Journal* 5, 6 (2018), 4396–4408.
- [34] Gürkan Solmaz, Pankaj Baranwal, and Flavio Cirillo. 2022. CountMeIn: Adaptive crowd estimation with Wi-Fi in smart cities. In *Proceedings of the 2022 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 187–196.
- [35] Qingyu Song, Changan Wang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Jian Wu, and Jiayi Ma. 2021. To choose or to fuse? Scale selection for crowd counting. In *Proceedings of the AAAI*. AAAI, Online, 2576–2583.
- [36] Valeriu-Daniel Stanciu, Maarten van Steen, Ciprian Dobre, and Andreas Peter. 2023. Privacy-friendly statistical counting for pedestrian dynamics. *Computer Communications* 211, C (Nov. 2023), 178–192. DOI : <https://doi.org/10.1016/j.comcom.2023.09.009>
- [37] Jiajie Tan and S.-H. Gary Chan. 2021. Efficient association of Wi-Fi probe requests under MAC address randomization. In *Proceedings of the INFOCOM 2021*. IEEE, Virtual Conference, 1–10.
- [38] Jiajie Tan, Edmund Sumpena, Weipeng Zhuo, Ziqi Zhao, Mengyun Liu, and S.-H. Gary Chan. 2020. IoT geofencing for COVID-19 home quarantine enforcement. *IEEE Internet of Things Magazine* 3, 3 (2020), 24–29.
- [39] Denis Ushakov, Egor Dudukalov, Elena Kozlova, and Khodor Shatila. 2022. The internet of things impact on smart public transportation. *Transportation Research Procedia* 63 (2022), 2392–2400.
- [40] Mathy Vanhoef, Célestin Matte, Mathieu Cunche, Leonardo S. Cardoso, and Frank Piessens. 2016. Why MAC address randomization is not enough: An analysis of Wi-Fi network discovery mechanisms. In *Proceedings of the 11th Asia CCS (ASIA CCS'16)*. ACM, 413–424. DOI : <https://doi.org/10.1145/2897845.2897883>
- [41] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. 2023. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 7464–7475.
- [42] Fengyu Wang, Feng Zhang, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. 2020. Passive people counting using commodity WiFi. In *Proceedings of the 2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*. IEEE, Virtual Conference, 1–6.
- [43] Fang-Jing Wu and Gürkan Solmaz. 2018. CrowdEstimator: Approximating crowd sizes with multi-modal data for internet-of-things services. In *Proceedings of the 16th Mosisys (MobiSys'18)*. ACM, New York, NY, USA, 337–349. DOI : <https://doi.org/10.1145/3210240.3210320>
- [44] Ooi Boon Yaik, Kong Zan Wai, Ian K. T. Tan, and Ooi Boon Sheng. 2016. Measuring the accuracy of crowd counting using Wi-Fi probe-request-frame counting technique. *JTEC* 8, 2 (May 2016), 79–81.
- [45] Xi Yang, Suining He, Bing Wang, and Mahan Tabatabaie. 2022. Spatio-temporal graph attention embedding for joint crowd flow and transition predictions: A Wi-Fi-based mobility case study. *IMWUT* 5, 4, Article 187 (Dec. 2022), 24 pages. DOI : <https://doi.org/10.1145/3495003>
- [46] Yanni Yang, Jiannong Cao, Xuefeng Liu, and Xiulong Liu. 2018. Wi-count: Passing people counting with COTS WiFi devices. In *Proceedings of the 2018 27th ICCCN*. IEEE, 1–9.
- [47] Qi Zhang, Hengshu Zhu, Peng Wang, Enhong Chen, and Hui Xiong. 2023. Hierarchical Wi-Fi trajectory embedding for indoor user mobility pattern analysis. *IMWUT* 7, 2, Article 82 (Jun 2023), 21 pages. DOI : <https://doi.org/10.1145/3596237>
- [48] Shiyu Zhang, Bangchao Deng, and Dingqi Yang. 2023. CrowdTelescope: Wi-Fi-positioning-based multi-grained spatiotemporal crowd flow prediction for smart campus. *CCF Transactions on Pervasive Computing and Interaction* 5, 1 (Mar. 2023), 31–44. DOI : <https://doi.org/10.1007/s42486-022-00121-6>
- [49] Shuhan Zhong, Weipeng Zhuo, Sizhe Song, Guanyao Li, Zhongyi Yu, and S.-H. Gary Chan. 2025. MTM: A multi-scale token mixing transformer for irregular multivariate time series classification. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 4074–4085.
- [50] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. 2022. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 4 (2022), 4396–4415.

Appendix

A Proof of the Unbiasedness of Rate Model

We provide a proof of $E(\hat{N}) = \bar{N}$ below by showing that the whole system is equivalent to a *renewal process*. In a renewal process, we consider the summation of a sequence of positive IID random variables X_k given as

$$S_K = \sum_{k=1}^K X_k, \quad (31)$$

where $E(X_k) = \bar{X}$. We define a *counting random variable* $L(S)$ as the maximum number of IID random variables (X_k) such that their summation $S_{L(S)} \leq S$. Namely, we have $\sum_{k=1}^{L(S)} X_k \leq S$ and $\sum_{k=1}^{L(S)+1} X_k > S$.

An important property of the renewal process is the *Elementary Renewal Theorem*, which states that

$$\frac{E(L(S))}{S} = \frac{1}{\bar{X}} \quad (32)$$

when S is sufficiently large.

We elaborate that our system is equivalent to a *renewal process* using the single-device example shown in Figure 2. In this figure, the APs received $b_n = 2$ bursts separately at t_1 and t_2 , and the device enters the area at time x and leaves at y . The dwell time of this device is calculated as

$$d_n = y - x = \beta_1 \tau_1 + \tau_2 + \beta_3 \tau_3, \quad (33)$$

where β_1 and β_3 represent the fraction of τ_1 and τ_3 with $0 < \beta_1, \beta_3 < 1$. Since the device enters and leaves the area largely independent of the probing, with equal probabilities, x could be any value between t_0 and t_1 , and y could be any value between t_2 and t_3 . Therefore, β_1 and β_3 are IID *uniform random variables* with $E(\beta_1) = E(\beta_3) = 1/2$.

When computing the total dwell time (D), we use T_F to denote the set of *fractional* intervals $\{\beta_i \tau_i\}$ where only part of τ_i is summed up (e.g., $\beta_1 \tau_1$ and $\beta_3 \tau_3$), and T_I to denote the set of *integral* intervals $\{\tau_i\}$ where the whole of τ_i is summed up (e.g., τ_2). It is important to note that both T_F and T_I are sets of IID random variables where $E(\beta_i \tau_i) = \bar{\tau}/2$ and $E(\tau_i) = \bar{\tau}$.

Generally speaking, N devices with B probing bursts would have $2N$ fractional intervals and $B - N$ integral intervals. Therefore, the dwell time of all the devices can be written as

$$D = \sum_{\beta_i \tau_i \in T_F} \beta_i \tau_i + \sum_{\tau_i \in T_I} \tau_i. \quad (34)$$

We let $D_F = \sum \beta_i \tau_i$ for $\beta_i \tau_i \in T_F$ and $D_I = \sum \tau_i$ for $\tau_i \in T_I$. Both D_F and D_I are the sums of IID random variables similar to S_K in Equation (31). Clearly, we have $D = D_F + D_I$. By applying the Elementary Renewal Theorem given in (32), we have

$$\frac{E(B - N)}{D_I} = \frac{1}{\bar{\tau}} \quad (35)$$

and

$$\frac{E(2N)}{D_F} = \frac{2}{\bar{\tau}} \quad (36)$$

when D_F and D_I are sufficiently large.

Given Equation (7), we also have

$$\begin{aligned} \mathbb{E}(\hat{N}) &= \mathbb{E}\left(\frac{R}{\bar{r}}\right) = \mathbb{E}\left(\frac{B\bar{r}}{w}\right) = \frac{\bar{r}}{w}\mathbb{E}(B) \\ &= \frac{\bar{r}}{w}\left(\mathbb{E}(B - \mathbf{N}) + \frac{1}{2}\mathbb{E}(2\mathbf{N})\right). \end{aligned} \quad (37)$$

Based on Equations (3), (35), (36), and (37), when $\bar{N}$ and w are sufficiently large so that D is also sufficiently large, we have

$$\mathbb{E}(\hat{N}) = \frac{\bar{r}}{w}\left(\frac{D_I}{\bar{r}} + \frac{1}{2}\frac{2D_F}{\bar{r}}\right) = \frac{D}{w} = \bar{N}. \quad (38)$$

This shows that the rate model is an unbiased estimator of the window-averaged device count, which ends the proof.

Received 16 October 2025; revised 12 June 2026; accepted 10 July 2026