

Recebido em 7 de junho de 2021, aceito em 4 de julho de 2021, data de publicação em 8 de julho de 2021, data da versão atual em 16 de julho de 2021.

Identificador de Objeto Digital 10.1109/ACCESS.2021.3095546

# Localização interna usando aumento de dados via Redes Adversariais Gerativas Seletivas

Wafa Njima <sup>1</sup>, (Membro, IEEE), Marwa Chafii <sup>1</sup>, (Membro, IEEE), H. Vincent Poor <sup>1</sup>, Sênior, IEEE, Raed M. Shubair Arsenia Chorti E <sup>2</sup>, (Membro Sênior, IEEE), 3, (Membro Vitalício, IEEE)

<sup>1</sup>ETIS UMR 8051, CY Paris Université, ENSEA, CNRS, 95000 Cergy, França.

<sup>2</sup>Departamento de Engenharia Elétrica e de Computação, Universidade de Nova York (NYU) Abu Dhabi, Abu Dhabi, Emirados Árabes Unidos <sup>3</sup>Departamento de Engenharia Elétrica e de Computação, Universidade de Princeton, Princeton,

NJ 08544, EUA Autor correspondente: Wafa Njima (wafa.njima@ensea.fr)

Este trabalho foi apoiado em parte pela Iniciativa de Excelência do CY (bolsa "Investissements d'Avenir" ANR-16-IDEX-0008) e pelo projeto DELICATE financiado pelo CNRS/INS2I.

**RESUMO** Vários serviços baseados em localização exigem informações de localização precisas em ambientes internos.

Recentemente, foi demonstrado que as impressões digitais do indicador de intensidade do sinal recebido (RSSI) baseadas em redes neurais profundas (DNN) alcançam alto desempenho de localização com baixa complexidade online. No entanto, tais métodos exigem uma quantidade muito grande de dados de treinamento para projetar e otimizar adequadamente o modelo DNN, o que torna a coleta de dados muito custosa. Neste artigo, propomos redes adversariais generativas para aumento de dados RSSI que geram dados RSSI falsos com base em um pequeno conjunto de dados rotulados reais coletados. O modelo desenvolvido utiliza aprendizado semissupervisionado para prever os pseudo-rótulos dos RSSIs gerados. Uma seleção adequada dos dados gerados é proposta para cobrir todo o ambiente interno considerado e reduzir o erro de geração de dados selecionando apenas os RSSIs falsos mais realistas.

Experimentos numéricos extensivos mostram que o esquema proposto de aumento e seleção de dados leva a uma melhoria na precisão de localização de 21,69% para dados simulados e 15,36% para dados experimentais.

**TERMOS DE ÍNDICE** Localização interna, indicador de intensidade do sinal recebido (RSSI), rede neural profunda (DNN), rede adversária generativa (GAN), aprendizado semissupervisionado.

## 1. INTRODUÇÃO

A proliferação de dispositivos sem fio inteligentes que aproveitam a conectividade onipresente, particularmente a internet das coisas (IoT), gerou um interesse crescente em uma ampla gama de serviços, incluindo localização interna [1], [2]. Em particular, a disponibilidade de informações de localização se tornou um fator-chave em serviços baseados em localização, como saúde, assistência domiciliar, monitoramento, rastreamento, etc. [3], [4]. Para fornecer informações de localização do usuário, diferentes tecnologias sem fio e emergentes (por exemplo, Bluetooth [5], WiFi [6], banda ultralarga (UWB) [7], luz visível [8], campo magnético [9]) e parâmetros de sinal (por exemplo, tempo de chegada (ToA) [10], ângulo de chegada (AoA) [11], informações de estado do canal (CSI) [12] e indicador de intensidade do sinal recebido (RSSI) [13]) foram investigados.

Os sistemas tradicionais de localização interna são baseados principalmente em métodos geométricos [14] (por exemplo, trilateração e triangulação) e baseados em impressão digital [15]. O desempenho de

O mapeamento geométrico é fortemente afetado por efeitos de propagação multipercursos, que obscurecem cada vez mais as relações entre medições físicas e distâncias. Consequentemente, a modelagem de propagação torna-se complicada e a precisão da localização é degenerada. Como alternativa aos métodos geométricos, métodos baseados em impressão digital têm sido propostos, adotando um processo de correspondência de padrões. Tais métodos começam com uma tarefa de levantamento de campo, coletando características de sinal em posições de treinamento na área de interesse para construir um banco de dados de impressão digital, também chamado de mapa de rádio. A localização é então realizada online, calculando a similaridade entre a impressão digital em um local desconhecido e o banco de dados de impressão digital. A localização estimada corresponde à impressão digital mais bem ajustada, aplicando diferentes algoritmos de estimativa. Algoritmos clássicos de avaliação e estimativa de similaridade são muito exigentes em termos de energia e tempo computacional, uma vez que todo o processo é online e requer a navegação em enormes bancos de dados de impressão digital coletados. Para superar esses problemas, sistemas de localização interna baseados em aprendizado de máquina (ML) foram propostos [16], em parti-

O editor associado que coordena a revisão deste manuscrito e aprovando-o para publicação foi Kegen Yu <sup>1</sup>.

Utilizando tais métodos, um modelo de DL salvo com base em um banco de dados de impressões digitais coletado, treinado offline, é usado online para previsão precisa da localização de informações, o que minimiza a complexidade online. Uma variedade de sistemas de localização baseados em métodos de DL (por exemplo, para complementação de dados coletados [20], minimização de ruído [21] e estimativa de localização [22]) foram propostos na literatura e fornecem bom desempenho de localização. No entanto, para construir um modelo de DL eficiente, esses métodos são intensivos em dados, exigindo grandes quantidades de dados de treinamento, cuja aquisição é uma tarefa que consome tempo e energia. Para abordar essa questão, algumas abordagens recentes que alavancam o aprendizado semissupervisionado foram introduzidas, combinando um pequeno conjunto de dados rotulados coletados e um grande conjunto de dados não rotulados coletados [23]–[25]. No entanto, a coleta de dados não rotulados ainda pode ser cara e, portanto, neste artigo, investigamos o uso de modelos generativos [26]–[28] para aumento de dados sem qualquer coleta de dados extra.

Nos últimos anos, os modelos generativos têm atraído um interesse significativo na comunidade científica devido aos seus benefícios promissores em diferentes áreas. Eles têm demonstrado uma boa capacidade de produzir suplementos realistas para diversos tipos de dados, como imagens, textos e sons.

Em particular, redes adversárias generativas (GANs) foram empregadas [29]–[31] para gerar medições adicionais, o que reduz o tempo de coleta e economiza esforço humano. Essas redes geram amostras com diversidade aprimorada e expandem o banco de dados de treinamento para garantir um projeto adequado de redes neurais profundas (DNNs) em diferentes campos, incluindo localização. Em [32], GANs são usadas com aprendizado semissupervisionado, onde GANs usam dados rotulados e não rotulados para compartilhar pesos com um classificador de localização a fim de se beneficiar de dados contidos em informações não rotuladas quando dados rotulados não são suficientes. Os autores em [33] visam construir um mapa de rádio eficiente cobrindo espaço livre (por exemplo, espaços abertos e corredores) e espaços restritos. Neste experimento, os dados correspondentes ao ambiente de espaço livre são medidos, enquanto os dados para o espaço restrito são gerados artificialmente. Isso é justificado pelas dificuldades em realizar medições sob restrições de espaço. Nos trabalhos descritos em [34] e [35], o banco de dados de impressões digitais de treinamento inicial foi expandido, aumentando a quantidade de dados de treinamento coletados em cada ponto de referência com base em GANs, a fim de aumentar sua riqueza. Para isso, uma coleta de medições de sinal foi realizada em diferentes pontos de referência e GANs foram utilizadas para gerar medições adicionais em cada ponto de referência, a fim de aumentar a diversidade dos dados coletados em cada posição de referência.

Neste artigo, ao contrário de [32], não assumimos a disponibilidade de dados não rotulados para melhorar o treinamento do modelo generativo e não assumimos ter dados coletados suficientes para regiões específicas, conforme assumido em [33] para o espaço livre. Consideramos um caso extremo e realista, em que apenas uma pequena quantidade de dados rotulados está disponível. Também não utilizamos GANs para aumentar a diversidade dos dados RSSI para dados específicos conhecidos.

posições de referência, conforme proposto em [34], mas geramos dados RSSI para novas posições desconhecidas, a fim de cobrir todo o ambiente. Além disso, propomos neste trabalho aplicar um processo de seleção dos dados falsos gerados que considere restrições tanto na cobertura quanto na credibilidade dos dados gerados.

O algoritmo proposto é validado e testado em ambientes simulados utilizando parâmetros realistas do modelo de propagação medidos em experimentos. Para fundamentar os resultados da simulação, aplicamos o sistema proposto em medições reais do banco de dados público UJIndoorLoc [36].

O restante deste artigo está organizado da seguinte forma: a Seção II formula o problema de localização e descreve o modelo do sistema. As GANs semissupervisionadas seletivas propostas para ampliação de dados são apresentadas na Seção III. O cenário ambiental e os resultados da simulação obtidos são apresentados e discutidos na Seção IV. Resultados experimentais baseados em medições reais do banco de dados UJIndoorLoc são apresentados na Seção V. Por fim, a Seção VI conclui nosso trabalho.

## II. CENÁRIOS DE LOCALIZAÇÃO INTERNA E TÉCNICA DE

### IMPRESSÃO DIGITAL

Nesta seção, descrevemos brevemente as limitações dos sistemas de localização interna existentes com base em impressões digitais RSSI.

E detalhamos diferentes etapas do modelo de sistema proposto desenvolvido para abordar o problema de interesse.

#### A. DESCRIÇÃO DO MODELO DE SISTEMA

Consideramos um ambiente interno com área de  $(C \times L)$  m<sup>2</sup>, onde a tecnologia WiFi é implantada utilizando  $M$  pontos de acesso (APs). Conforme ilustrado na Figura 1, o sistema consiste em duas partes principais: a unidade central (UC), que realiza todo o processamento e localização dos dados, e os nós sensores móveis. Existem dois tipos de nós sensores móveis: aqueles usados durante a fase de treinamento para coletar medições de RSSI para a construção do banco de dados de treinamento e aqueles que solicitam a localização online. Para aumentar a riqueza do banco de dados RSSI de impressões digitais, GANs são usadas para gerar impressões digitais RSSI falsas.

Seja  $p_{ij}$  o RSSI medido na  $i$ -ésima posição do sinal transmitido pelo  $j$ -ésimo AP.

#### B. LOCALIZAÇÃO BASEADA EM IMPRESSÕES DIGITAIS

Nesta seção, apresentamos a técnica padrão de impressão digital usada com modelos DNN. Em seguida, detalhamos brevemente as etapas do modelo do nosso sistema, incluindo coleta e treinamento de dados.

##### 1) IMPRESSÃO DIGITAL BASEADA EM

DNN Os sistemas de posicionamento baseados em RSSI podem ser divididos em duas categorias: métodos geométricos e métodos baseados em impressões digitais. Dentre elas, a impressão digital é amplamente estudada e adotada devido à sua precisão de localização e simplicidade de implementação. Consiste em duas fases principais: uma fase offline, também chamada de fase de treinamento, e uma fase online, chamada de fase de teste. Durante a fase offline, as medições RSSI são coletadas em cada local de medição (posição de treinamento).

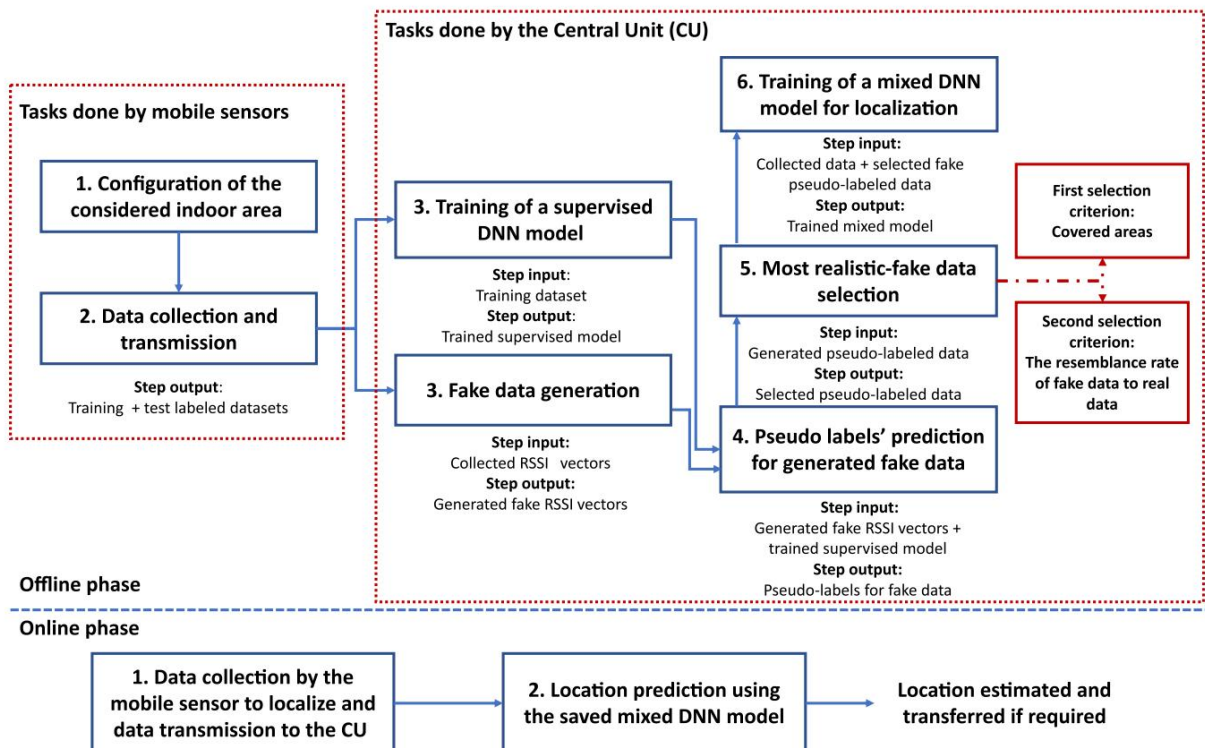


FIGURA 1. O pipeline do método de localização baseado em GANs seletivos proposto durante a fase de treinamento.

de diferentes transmissores de sinais de rádio, essencialmente APs quando se considera sinais WiFi. Esses RSSIs coletados com suas coordenadas de localização são transferidos para uma CU e armazenados em um banco de dados de treinamento. Na fase online, os RSSIs de diferentes APs são medidos e comparados com os coletados e armazenados no banco de dados de treinamento para prever a localização do alvo. Essa previsão pode ser realizada por um modelo DNN que pode ser treinado offline e aplicado diretamente online. O uso de modelos DNN para previsão de localização é recomendado, uma vez que a maior parte da complexidade online é transferida para a fase offline. O DNN é recomendado para localização interna precisa devido ao fato de ser capaz de aprender flutuações de sinal ao longo do tempo e dinamicidade ambiental devido às suas funções mais profundas que mapeiam a entrada para a saída [37], [38].

Para obter boa precisão de localização usando um modelo DNN treinado de forma otimizada, uma grande quantidade de dados coletados de forma dispendiosa é necessária durante a fase offline, o que torna o método de fingerprinting trabalhoso e demorado. Para solucionar esse problema, uma estrutura de localização interna GAN seletiva combinada com aprendizado semissupervisionado é proposta e detalhada neste artigo, a qual realiza a previsão de localização com base em dados rotulados coletados e dados pseudo-rotulados gerados falsamente. Este sistema aproveita a geração de dados falsos e a mistura deles com dados rotulados no processo de treinamento para reduzir a dependência de dados coletados de forma dispendiosa. No entanto, os dados gerados podem ser irrealistas e podem criar classes desbalanceadas quando os dados gerados são mais densos para regiões específicas. Portanto,

um protocolo de seleção de dados adequado deve ser desenvolvido para superar esses problemas.

## 2) COLETA DE DADOS E TREINAMENTO DE DADOS

Nesta parte, apresentamos em detalhes a coleta de dados rotulados, bem como todo o processo de pré-processamento e treinamento dos dados coletados.

### a. Coleta de banco de dados de impressões digitais: Neste artigo,

consideramos um ambiente interno ruidoso como a área de interesse para coletar impressões digitais RSSI durante a fase offline, incluindo dados de treinamento e teste. b. **Aumento de**

**dados:** O aumento de dados é usado para complementar dados quando eles são muito caros e raros. Assim, visa aumentar o tamanho e a diversidade de um conjunto de dados gerando novas amostras falsas com base em amostras reais. • **Geração de dados:** Com base nos vetores RSSI coletados, dados falsos extras são gerados usando uma GAN para expandir o conjunto de dados e garantir sua diversidade. • **Pseudo-rotulagem de dados:** Uma vez que os vetores RSSI falsos são gerados, um pseudo-rótulo artificial é associado a cada vetor usando aprendizado semissupervisionado.

c. **Seleção de dados:** Após a geração e a pseudo-rotulagem dos dados, aplicamos um método de seleção de dados. Primeiro, geramos um grande número de amostras falsas para cobrir todo o ambiente e, em seguida, eliminamos dados RSSI desnecessários e imprecisos para melhorar a precisão da localização.

**d. Localização baseada em DNNs:** Para localizar um alvo, um modelo DNN treinado é aplicado para estimar suas coordenadas com base no vetor RSSI coletado correspondente.

Durante o processo de treinamento, todos os vetores RSSI (por exemplo, vetores reais coletados e vetores falsos gerados selecionados) são ajustados ao modelo como entradas e as coordenadas correspondentes são consideradas como saídas.

### III. GAN SELETIVO SEMI-SUPERVISIONADO PROPOSTO PARA AUMENTO

#### DE DADOS DE LOCALIZAÇÃO

Nesta seção, descrevemos em detalhes diferentes etapas do sistema de localização proposto.

#### A. GERAÇÃO DE DADOS

##### 1) INTUIÇÃO POR TRÁS DAS

GANs Modelos generativos visam aprender a verdadeira distribuição de dados de um conjunto de treinamento para gerar novas amostras realistas com algumas variações. Assim, tentamos produzir amostras com uma distribuição que seja o mais semelhante possível à verdadeira distribuição de dados. GANs são modelos generativos que incluem dois componentes [39]: o gerador e o discriminador, como mostrado na Fig. 2. O modelo gerador recebe ruído aleatório como entrada e aprende como produzir uma representação de saída realista semelhante aos dados reais, enquanto o discriminador aprende como distinguir entre amostras falsas e reais.

Esses dois modelos são treinados em conjunto até que o gerador seja capaz de gerar exemplos realistas a partir do ruído de entrada. GANs alcançaram um desempenho impressionante em uma infinidade de tarefas (por exemplo, geração de faces, geração de objetos 3D e tradução de imagens de um domínio para outro) e muitas empresas as utilizam, incluindo o Google para geração de texto, a IBM para aumento de dados, a Adobe para Photoshop de última geração e o Snapchat-TikTok para filtros de imagem.

Em nosso caso, GANs são usadas para aumentar os dados, o que significa que os dados gerados podem ser usados para complementar dados reais, aumentando o tamanho e a diversidade do conjunto de dados por meio do uso de amostras geradas. O objetivo é aumentar o conjunto de dados quando os dados reais são muito caros para coletar.

##### 2) TREINAMENTO DE

GANs: GANs consistem em duas redes neurais diferentes: a geradora e a discriminadora. Para treinar uma GAN, alternamos o treinamento do modelo gerador de DNN  $G$  e o treinamento da DNN

modelo discriminador  $D$ . Para cada modelo DNN, introduzimos um vetor de entrada  $i(0) \in \mathbb{R}^{N_0 \times 1}$  e seu vetor de saída associado  $\hat{y} \in \mathbb{R}^{N_H+1 \times 1}$

Seja  $N_h$  o número de neurônios para a camada  $h$ , tal que  $0 \leq h \leq H+1$ , onde  $H$  é o número de camadas ocultas.  $b_h \in \mathbb{R}^{N_h \times 1}$  e  $W_h \in \mathbb{R}^{N_h \times N_{h-1}}$  denotam as matrizes de vieses e pesos, respectivamente. O vetor de saída  $th$  da camada  $h$

camada pode ser expressa como

$$o(h) = g(b(h) + W(h) i(h)), \quad i(h) = o(h), \quad (1)$$

onde o vetor de entrada  $i(h)$  sofre uma transformação linear representada por  $W(h)$ , um vetor de polarização  $b(h)$  e, em seguida, uma função de ativação não linear  $g(h)$  é aplicada. Durante o treinamento DNN, a função de perda  $L(\tilde{y}) = L(W, b)$  é calculada para atualizar iterativamente os parâmetros DNN  $\tilde{y} = (W, b)$ .

• **Treinamento do gerador:** Seja  $Z \in \mathbb{R}^{M \times m_f}$  o conjunto de dados de entrada  $(i)$  do gerador, de modo que cada coluna  $z \in \mathbb{R}^{M \times 1}$   $i = 1, \dots, m_f$  corresponda a um vetor de ruído aleatório cujas amostras são distribuídas uniformemente em  $[y_1, 1]$ . Na saída do gerador DNN, cada entrada  $(i)$  produz um vetor RSSI falso  $G(z(i))$  e o vetor de saída falso é gerado seguindo (1),

$$\text{vetor de ruído } z \in \mathbb{R}^{M \times 1} \quad \text{onde } i(0) = z \in \mathbb{R}^{N_H+1 \times 1} = (i), \quad N_0 = M, \quad o(H+1) = G(z(i)) \text{ e}$$

$M$ . Então,  $G(z(i))$  é passado para o discriminador DNN  $D(\cdot)$ , que faz uma previsão de quão realista  $G(z(i))$  é. A perda do gerador é então calculada minimizando a função de perda:

$$L_G = \sum_{i=1}^{m_f} -\log D(G(z(i))) \quad (2)$$

Uma vez que a função de perda é minimizada, e os parâmetros  $\tilde{y} = (W, b)$  não mudam significativamente após iterações adicionais, podemos salvar os parâmetros DNN do gerador.

• **Treinamento do discriminador:** O discriminador aprende a distinguir o real do falso. No início do treinamento, ele não sabe quais vetores são reais e quais são falsos. No entanto, ele tem acesso a dados reais para compará-los com vetores de entrada e classificar os vetores de entrada como reais ou falsos. Assim, o discriminador recebe exemplos cada vez mais realistas a cada rodada do gerador até que os exemplos sejam bons o suficiente para preencher o discriminador. O discriminador atua como um classificador binário e calcula a probabilidade de um exemplo ser falso  $D(G(z(i)))$ , seguindo (1), resultando em

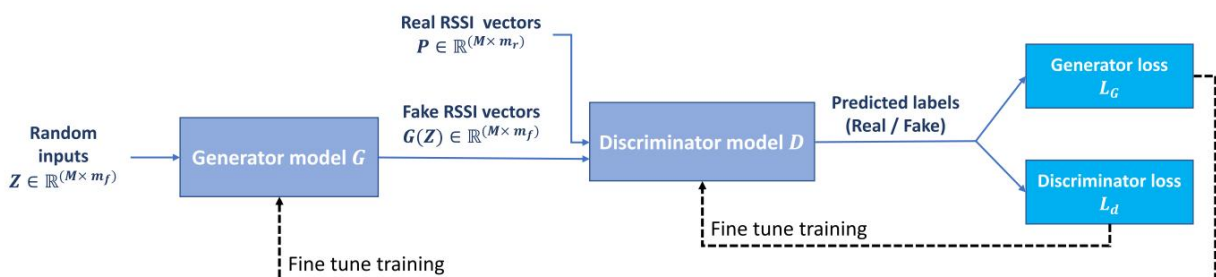


FIGURA 2. Rede GAN para geração de dados RSSI falsos durante o treinamento.

ambas as amostras falsas geradas  $G(Z)$ , e o conjunto de dados real  $P$   $\tilde{y}$   $R$   $M \times m_r$  de vetores RSSIs  $m_r$  coletados denotados como  $(i)$   $i = 1, \dots, \text{senhor}$ . Esta probabilidade será dada ao gerador  $p$  para melhorar seu desempenho conforme expresso em (2). O treinamento do discriminador é realizado minimizando a função de perda derivada da entropia cruzada binária (BCE) entre os dados reais e gerados [40]:

$$L_d = \tilde{y}^T d \tilde{y} \quad m_r \quad \frac{1}{\text{senhor}} \sum_{i=1}^{\text{senhor}} \log D(p \quad (i))$$

$$- \frac{1}{\text{senhor}} \sum_{i=1}^{\text{senhor}} \log 1 \tilde{y} D G(z \quad (i)) \quad (3)$$

onde  $\tilde{y}d = (Wd, bd)$  são os parâmetros do modelo DNN discriminador  $D(\cdot)$ .  $\log(D(p \quad (i)))$  refere-se à probabilidade logarítmica de que o discriminador esteja classificando corretamente os exemplos reais

enquanto maximiza  $\log 1 \tilde{y} D G(z \quad (i))$

ajudaria o discriminador a rotular corretamente a amostra falsa que vem do gerador.

### 3) GERAÇÃO DE DADOS FALSOS COM BASE EM GANs Dado

um pequeno número de amostras de dados rotulados de treinamento, vetores RSSI falsos extras são gerados com base no GAN. O gerador toma como entradas vetores de ruído que serão ajustados ao discriminador com os vetores RSSI reais. Com base nas saídas do discriminador, os modelos do gerador e do discriminador são atualizados para melhorar seu desempenho de geração e classificação. Geramos vetores RSSI falsos  $m_f$   $\tilde{y} R (M \times 1)$  com base no conjunto de dados de vetores RSSI coletados,  $m_f$  é fixado experimentalmente para escolher o melhor número de vetores gerados que fornece a melhor melhoria de localização.

### B. SEMI-SUPERVISADO PARA PSEUDO-ROTULAÇÃO DE DADOS GERADOS

A pseudo-rotulagem visa estimar os rótulos de um conjunto de dados não rotulado, dado um modelo DNN treinado em um modelo rotulado. No contexto de localização, os rótulos correspondem às informações de localização dos usuários (ID da sala, ID do andar, identificador de zona, coordenadas 2D/3D, etc.). As coordenadas 2D são os rótulos utilizados em nosso trabalho. As etapas do processo de pseudo-rotulagem são resumidas da seguinte forma:

- Etapa 1: O modelo DNN é treinado em RSSI rotulado apenas impressões digitais, de forma supervisionada.
- Etapa 2: Com base no modelo DNN treinado, 'pseudo-rótulos' são previstos para vetores RSSI não rotulados gerados.
- Etapa 3: Um modelo DNN misto é treinado combinando dados rotulados e pseudo-rotulados selecionados. Para treinar tal modelo, os vetores RSSI coletados e selecionados são usados como entradas, e os rótulos associados correspondentes (rótulos reais e pseudo-rótulos artificiais) são usados como saídas.

Durante a etapa de geração de dados falsos, produzimos vetores RSSI não rotulados. Para serem integrados à nossa localização

No processo, esses vetores falsos serão pseudorotulados e selecionados. Para isso, os vetores RSSI coletados são usados como entradas para um modelo DNN supervisionado que recebe como saídas os rótulos correspondentes, por exemplo, coordenadas durante o treinamento. Após o treinamento, vetores RSSI gerados artificialmente são fornecidos ao modelo para prever os pseudorótulos artificiais associados. Em seguida, um modelo geral usado para localização é treinado com dados reais e selecionados. O procedimento de seleção é descrito em detalhes na subseção III-C.

### C. CRITÉRIOS DE SELEÇÃO DE DADOS FALSOS GERADOS

Inicialmente, a ideia deste trabalho era gerar vetores RSSI, para gerar mais dados e cobrir todo o ambiente, e então estimar seus pseudo-rótulos para construir um conjunto de dados misto e rico. Geramos uma grande quantidade de dados falsos e testamos o sistema resultante. No entanto, ao plotar os pseudo-rótulos dos RSSIs gerados, ou seja, posições, notamos que (i) as posições geradas não cobrem toda a área do ambiente considerado, (ii) essas posições às vezes são condensadas em uma área específica, e (iii) gerar um número maior de dados falsos não leva necessariamente a uma melhor precisão de localização. Portanto, introduzimos aqui critérios de seleção para escolher apenas um subconjunto útil dos dados falsos gerados:

- **Critério 1: Cobertura do ambiente:** Aqui, consideramos que nosso ambiente  $E$  é uniformemente dividido em zonas  $e_j$ , cada uma cobrindo  $(l_j \times w_j) m_2$ , de modo que  $\sum_j l_j \times w_j = L \times W m_2$  uma grande. Após gerar aleatoriamente quantidade de dados (vetores RSSI e as pseudoposições), construímos um novo conjunto de dados selecionando para cada zona  $e_j$  um número  $N_j$  de amostras de dados proporcional à sua área de superfície  $(l_j \times w_j) m_2$  ou seja, as pseudoposições caem na zona desejada  $e_j$ . Assim, dados redundantes são eliminados, e todo o ambiente  $E$  é coberto e cada zona  $e_j$  é igualmente representada no novo conjunto de dados que contém  $j N_j$  amostras selecionadas. Primeiro, tentamos selecionar aleatoriamente  $N_j$   $\text{senhora} =$  amostras por zona  $e_j$ , que essa seleção aleatória pode ser melhorada selecionando apenas os dados  $m_s$  mais realistas, o que nos leva ao Critério 2. • **Critério 2: Dados falsos mais realistas:** A seleção das  $N_j$  amostras de dados falsos mais realistas por zona  $e_j$  é realizada comparando a pontuação de cada RSSI gerado  $G(z \quad (i))$  na saída do discriminador  $D(G(z \quad (i)))$ .

Assim, em cada zona  $e_j$ , selecionamos as posições  $N_j$  que têm maior probabilidade de serem reais, ou seja, associadas aos menores valores da função de perdas  $N_j$ :

$$L_d^{(eu)} = \tilde{y}^T d \tilde{y} \quad \frac{1}{\text{senhor}} \sum_{\text{senhor}=1}^{\text{senhor}} \log D(p \quad (i))$$

$$- \tilde{y} \log 1 \tilde{y} D G(z \quad (i)) \quad (4)$$

onde  $p, j = 1, \dots, m_r$  é o  $j$  vetor RSSI real coletado.



#### IV. RESULTADOS BASEADOS EM DADOS SIMULADOS Nesta

seção, apresentamos a configuração do ambiente e as diferentes arquiteturas e hiperparâmetros de DNN utilizados. Também apresentamos o desempenho de localização do nosso sistema proposto com base em GANs seletivas e o comparamos com outros métodos para demonstrar sua utilidade em termos de precisão de localização e custo de coleta de dados em um ambiente simulado.

##### A. AMBIENTE DE PROPAGAÇÃO E CONFIGURAÇÃO AMBIENTAL

Para modelar o ambiente de propagação, consideramos que os sinais podem ser degradados e bloqueados pelos obstáculos:

- **Bloqueio:** 40% dos dados medidos pelo RSSI são desconhecidos.

Atribuímos o valor de -110 dBm aos APs não detectados para obter um sinal fraco que não afete o processo de cálculo. A escolha desse valor baseia-se em diversos experimentos em nossos ambientes internos simulados.

- **A degradação** é considerada uma combinação de perda de trajetória e efeitos de sombreamento.

Seja  $p_{ij}$  o RSSI medido na  $i$ -ésima posição do sinal transmitido pelo  $j$ -ésimo AP. Pode ser expresso como

$$p_{ij} = p_t - p_{Lij} + B\tilde{y} \text{ [dBm]}, \text{ onde } p_t \text{ é} \quad (5)$$

uma potência transmitida constante,  $B\tilde{y}$  é uma média zero Variável aleatória distribuída gaussiana com desvio padrão

$\tilde{y}$  representando os efeitos de sombreamento e  $p_{Lij}$  é a perda de caminho calculado da seguinte forma:

$$p_{Lij} = p_{L0} + 20 \log_{10}(f) + 10\mu \log_{10} \frac{d_{ij}}{d_0}, \quad (6)$$

onde  $p_{L0}$  denota o valor da perda de caminho a uma distância de referência  $d_0$ ,  $f$  a frequência da portadora,  $\mu$  o expoente da perda de caminho cujo valor caracteriza um ambiente específico e pode ser calculado empiricamente com base em medições coletadas e  $d_{ij}$  é a distância entre o  $j$ -ésimo AP e a  $i$ -ésima posição.

Nesta parte, consideramos uma rede de sensores composta por  $M = 10$  APs dispostos aleatoriamente em um ambiente interno com área de  $C \times L = 400 \text{ m}^2$ . Para avaliar a precisão da localização, utilizamos um banco de dados de treinamento composto por  $m_r = 1000$  vetores RSSI coletados em 100 posições rotuladas por suas coordenadas. Em cada posição rotulada, coletamos 10 medições de RSSI, o que ajuda a minimizar as flutuações temporais de RSSI causadas por efeitos de sombreamento e desvanecimento.

Essas posições são distribuídas de maneira uniforme, conforme ilustrado na Fig. 3. Dividimos o ambiente interno cobrindo  $L \times W \text{ m}^2$  em zonas/ classes  $e_j$  de tamanho  $l_j \times w_j$  de modo que  $\tilde{y}_j$ ,  $l_j = w_j = l$  e colocamos uma posição rotulada no centro de cada zona para cobrir uniformemente toda a área.

Os vetores RSSI são construídos conforme descrito acima. Detalhamos na Tabela 1 os parâmetros utilizados para o modelo de propagação e os parâmetros de simulação. O banco de dados de teste é construído utilizando o mesmo modelo de propagação utilizado para a construção dos dados de treinamento, mas com trajetórias diferentes. Consideramos  $m_t = 8000$  impressões digitais RSSI rotuladas para teste, coletadas em 800 posições de teste. Para obter um conjunto de dados de teste representativo de

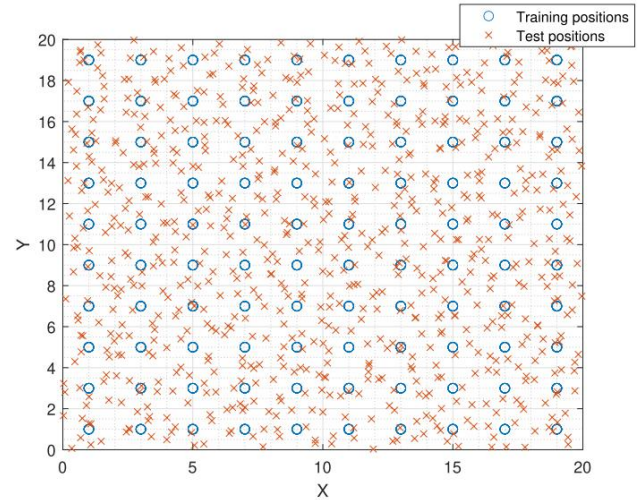


FIGURA 3. Localização das posições de treinamento e teste no ambiente interno simulado cobrindo 400 m<sup>2</sup>.

TABELA 1. Diferentes parâmetros usados durante o processo de simulação.

| Parameter         | Definition                              | Value                    |
|-------------------|---|--------------------------|
| $p_t$             | Transmitted power                       | 20 dBm                   |
| $f$               | Frequency                               | 2.4 GHz                  |
| $\mu$             | Pathloss exponent                       | 3.23                     |
| $d_0$             | Reference distance                      | 1 m                      |
| $B_{\sigma_{ij}}$ | Shadowing value                         | $\sim \mathcal{N}(0, 4)$ |
| $L$               | Length of the indoor environment        | 20 m                     |
| $W$               | Width of the indoor environment         | 20 m                     |
| $l$               | Length of each zone                     | 2 m                      |
| $M$               | Number of APs                           | 10                       |
| $m_r$             | Number of labeled training measurements | 1000                     |
| $m_f$             | Number of generated fake positions      | 40000                    |
| $m_t$             | Number of test measurements             | 8000                     |

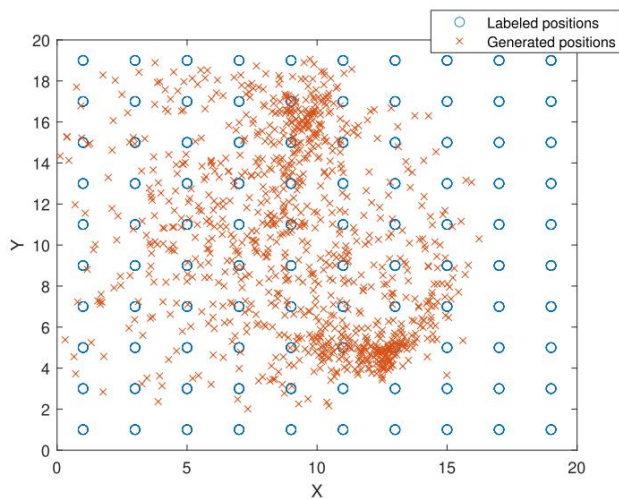
em todo o ambiente, colocamos aleatoriamente duas posições de teste em cada zona de tamanho 1 m<sup>2</sup>, conforme apresentado com cruzes vermelhas na Fig. 3.

##### B. MODELOS DE GERAÇÃO DE DADOS

Nesta seção, detalhamos as arquiteturas e os parâmetros de treinamento usados por diferentes modelos DNN durante o processo de geração de dados.

###### 1) ARQUITETURAS E PARÂMETROS GAN. Gerar

dados falsos úteis não é simples. Por exemplo, ao gerar vetores RSSI falsos  $m_f = 1000$  com base em vetores RSSI reais  $m_r = 1000$  e prever suas pseudoposições, percebemos que os dados gerados não cobrem todo o ambiente, como mostrado na Fig. 4. Assim, geramos um número muito grande de vetores RSSI para obter pseudoposições que cubram toda a área. Com base em



**FIGURA 4.** 1000 posições geradas com base em 100 posições rotuladas de treinamento.

simulações extensivas, gerando  $mf = 40000$  amostras falsas parecem cobrir nosso ambiente de 400 m<sup>2</sup>. Por exemplo, ao gerar  $mf = 10000$  amostras de dados falsos, cobrimos apenas 362 zonas de tamanho 1 m<sup>2</sup> cada, enquanto com  $mf = 40000$ , cobrimos 396 zonas de tamanho 1 m<sup>2</sup>, como pode ser visto na Fig. 5a.

Para melhorar a diversidade da GAN, não geramos 40.000 vetores RSSI de uma só vez, mas geramos 10 vezes 4.000 amostras falsas com base nas mesmas amostras reais. Neste artigo, introduzimos GANs baseadas em uma DNN otimizada com estimativa de momento adaptativo (ADAM), usando 0,01 como taxa de aprendizado durante 200 épocas. A função de ativação usada por  $G$  é a função de unidade linear retificada (ReLU), usada em uma camada oculta com 10 neurônios. A função de ativação de  $D$  é a função ReLU, enquanto a última camada usa a função sigmoide. Um discriminador de camada oculta com 10 neurônios é usado. Essas escolhas são baseadas em vários experimentos e

testes.

2) ARQUITETURAS E PARÂMETROS DE PSEUDO-ROTULAÇÃO: Uma vez gerados todos os dados, uma DNN é treinada com base nos dados rotulados. Este modelo é então usado para prever pseudo-rótulos.

para vetores RSSI gerados. Usamos ADAM como algoritmo de otimização [41] para todos os modelos DNN usados neste trabalho, seja para pseudo-rotulagem ou localização. Os experimentos intensivos levaram a uma arquitetura DNN de 2 camadas ocultas com 30 neurônios na primeira camada e 20 neurônios na segunda usando 200/250 épocas e um tamanho de minilote igual a 50/100. Uma vez gerado e pseudo-rotulado  $mf = 40000$ , 1000 são selecionados, selecionando amostras por zona coberta de 2 tamanhos 4m. Para cada zona coberta, as amostras selecionadas são associadas aos RSSIs falsos mais realistas. Observe que se pularmos o Critério 1 e selecionarmos diretamente as 1000 amostras de dados falsos mais realistas, acabaremos com amostras não uniformemente distribuídas por todo o ambiente, conforme apresentado na Fig. 5b.

### C. MODELOS DNN USADOS PARA LOCALIZAÇÃO:

#### ARQUITETURAS E PARÂMETROS DE TREINAMENTO

Para fins de avaliação, comparamos a precisão da localização combinando dados falsos gerados, rotulados e selecionados.

Chamamos cada algoritmo dependendo da natureza e do número de amostras usadas da seguinte forma: •

- Supervisionado( $mr, mp$ ): ao usar um método supervisionado baseado em dados rotulados  $mr$  coletados em posições  $mp$  diferentes.
- Selective-SS-GAN( $mr, mp, ms$ ) é o método de localização em que combinamos amostras marcadas com  $mr$  coletadas em posições diferentes de  $mp$  e amostras falsas geradas selecionadas pseudomarcadas com  $ms$  para localização.

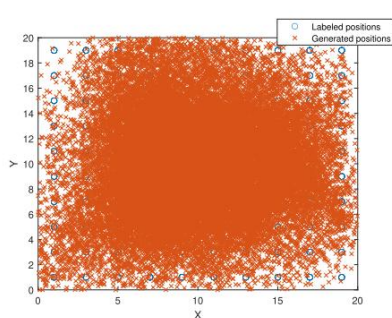
Para aplicar a localização, precisamos treinar um modelo DNN com dados coletados rotulados e dados pseudorotulados. Este modelo utiliza os vetores RSSI como entradas e as saídas são os rótulos correspondentes, que são as coordenadas das posições.

Uma taxa de aprendizagem igual a 0,01 foi selecionada usando 250 épocas e um tamanho de mini lote igual a 100. As arquiteturas DNN usadas estão resumidas na Tabela 2, onde  $FCi(\cdot)$  se refere ao número de neurônios no  $i$

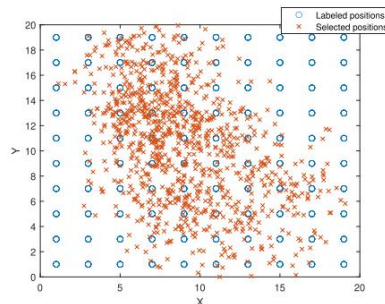
camada totalmente conectada.

### D. PRECISÃO DE LOCALIZAÇÃO

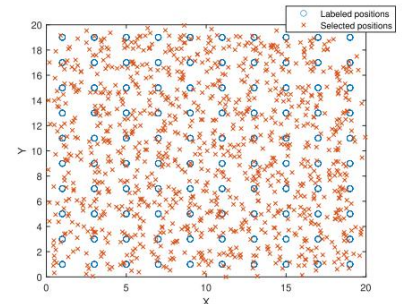
Nesta seção, apresentamos a precisão de localização do nosso sistema proposto com base no conjunto de dados aumentado e



(a) 40000 generated fake positions



(b) 1000 selected most realistic-fake positions



(c) 1000 selected most realistic-fake positions per zone

**FIGURA 5.** Posições geradas e posições selecionadas com base em critérios de seleção definidos.

TABELA 2. Desempenho de localização obtido considerando 1000 amostras de dados rotuladas.

|                                  | Localization accuracy | Accuracy increase vs supervised(1000,100) | Architecture of the DNN used for localization                       |
|----------------------------------|-----------------------|---|---|
| Supervised(1000,100)             | 1.06 m                | —   | FC <sub>1</sub> (30) and FC <sub>2</sub> (20)                       |
| Selective-SS-GAN (1000,100,100)  | 0.87 m                | 19 cm   17.92%                            | FC <sub>1</sub> (100),FC <sub>2</sub> (80) and FC <sub>3</sub> (20) |
| Selective-SS-GAN (1000,100,500)  | 0.84 m                | 22 cm   20.75%                            | FC <sub>1</sub> (100),FC <sub>2</sub> (50) and FC <sub>3</sub> (10) |
| Selective-SS-GAN (1000,100,1000) | <b>0.83 m</b>         | 23 cm   21.96%                            | FC <sub>1</sub> (80),FC <sub>2</sub> (60) and FC <sub>3</sub> (20)  |
| Selective-SS-GAN (1000,100,2000) | 0.85 m                | 21 cm   19.81%                            | FC <sub>1</sub> (100),FC <sub>2</sub> (80) and FC <sub>3</sub> (20) |
| Selective-SS-GAN (1000,100,3000) | 0.86 m                | 20 cm   18.86%                            | FC <sub>1</sub> (80),FC <sub>2</sub> (60) and FC <sub>3</sub> (20)  |
| Selective-SS-GAN (1000,100,4000) | 0.86 m                | 20 cm   18.86%                            | FC <sub>1</sub> (80),FC <sub>2</sub> (60) and FC <sub>3</sub> (20)  |
| Supervised(2000,1100)            | 0.81 m                | 25 cm   23.58%                            | FC <sub>1</sub> (30) and FC <sub>2</sub> (20)                       |

Compare-o com o esquema em que apenas amostras de dados coletados e rotulados são usadas para localização. Os resultados correspondentes a esses dados são mostrados na Tabela 2.

Na Tabela 2, apresentamos a precisão de localização (ou seja, erro médio de localização) para o Selective-SS-GAN( $mr$ ,  $mp$ ,  $ms$ ) treinado em  $mr = 1000$  amostras de dados rotuladas e diferentes números de amostras de dados falsos selecionadas vs. modelo de aprendizado supervisionado treinado apenas em  $mr = 1000$  amostras rotuladas. Selecionamos diferentes números de amostras de dados falsos [100 ÷ 4000] do conjunto de  $mf = 40000$  amostras geradas enquanto satisfazemos o Critério 1 e o Critério 2. Notamos que para todos os conjuntos de dados aumentados, a precisão de localização é melhorada em comparação a um conjunto de dados limitado apenas a dados reais. A melhoria de localização varia entre 17,92% e 21,96%. A melhor precisão de localização é obtida com  $ms = 1000$  amostras de dados falsos selecionadas, onde alcançamos um aumento de 21,96% na precisão de localização vs. o algoritmo supervisionado convencional sem qualquer custo adicional na coleta de dados adicionais. Essa melhoria é explicada pela consideração de que a DNN foi treinada em um conjunto de dados maior, que abrange novas posições não observadas no conjunto de dados limitado com base nos dados coletados. Além disso, podemos observar que gerar mais dados falsos não necessariamente melhora a precisão da localização, o que pode ser explicado pelo erro de geração da GAN devido à geração de vetores RSSI menos realistas. Observe que, a partir de 2.000 amostras geradas, o desempenho satura e nenhuma melhoria pode ser alcançada pela geração de dados falsos adicionais. Isso pode ser explicado pelo fato de que, com base em 1.000 vetores rotulados coletados em 100 posições de treinamento, não podemos fornecer uma maior diversidade de medição para a GAN.

A Fig. 6 ilustra as funções de distribuição cumulativas (CDFs) de três algoritmos que combinam subconjuntos de dados de diferentes tipos (rotulados e gerados). Podemos facilmente notar que o sistema de localização interna supervisionado, baseado em 1.000 amostras rotuladas coletadas em 100 posições conhecidas, corresponde à pior precisão de localização. Para uma comparação justa, usamos o mesmo conjunto de dados de posições rotuladas, ao qual adicionamos (i) 1.000 medições rotuladas coletadas em 1.000 posições rotuladas diferentes, colocadas aleatoriamente na área considerada, ou seja, Supervisionada (2.000, 1.100), e (ii) com base nesses dados, geramos e selecionamos 1.000 posições falsas, ou seja, GAN-SS-Seletivo (1.000, 1.000, 1.000). Observamos que a localização

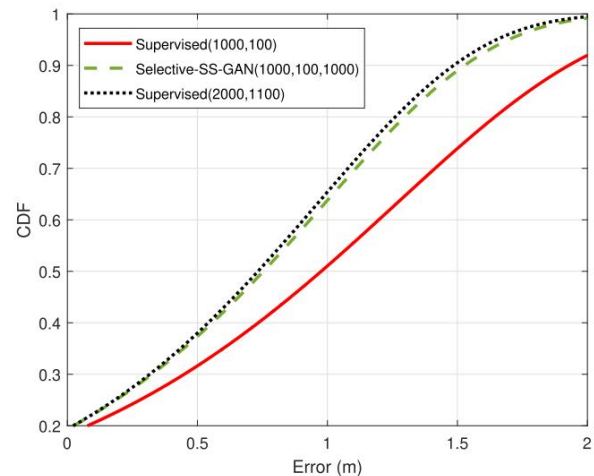


FIGURA 6. Comparação de desempenho de localização.

O desempenho é próximo, com apenas 2 cm de diferença, o que significa que podemos alcançar quase o mesmo desempenho coletando apenas metade dos dados rotulados e gerando artificialmente a outra metade. Assim, o processo de geração e seleção de dados proposto proporciona uma melhoria na precisão da localização sem custo adicional de coleta.

## V. RESULTADOS BASEADOS EM DADOS EXPERIMENTAIS

Para fundamentar os resultados da simulação, validamos e testamos nosso sistema em medições reais de RSSI do banco de dados público UJIndoorLoc. Este banco de dados abrange a área de

Quase 110.000 m<sup>2</sup>, incluindo três prédios de quatro ou cinco andares. Para simplificar a implementação e como trabalhamos em uma área de um andar, consideramos os dados coletados correspondentes ao segundo andar apenas do prédio 1. Há medições de treinamento e de validação coletadas quatro meses depois das de treinamento. Para o andar considerado, temos 1.395 impressões digitais de treinamento coletadas em 80 posições de treinamento e 40 posições de validação coletadas em diferentes posições de validação. Para uma comparação justa e como consideramos apenas 1.000 impressões digitais de treinamento como dados de entrada do sistema durante as simulações, escolhemos aleatoriamente 1.000 impressões digitais do conjunto de treinamento e adicionamos as 395 posições de treinamento restantes.



TABELA 3. Desempenho de localização obtido considerando 1000 impressões digitais reais correspondentes ao Edifício 1-Piso 2 do banco de dados UJIndoorLoc.

|                              | Localization mean error | Localization min-max errors | Localization mean error decrease vs supervised(1000) | Architecture of the DNN used for localization                         |
|------------------------------|-------------------------|-----------------------------|--|---|
| Supervised(1000)             | 4.1 m                   | 3.92 m - 4.53 m             | –  | FC <sub>1</sub> (200), FC <sub>2</sub> (100) and FC <sub>3</sub> (50) |
| Selective-SS-GAN (1000,700)  | 3.67 m                  | 3.58 m – 3.72 m             | 43 cm   10.48 %                                      | FC <sub>1</sub> (300), FC <sub>2</sub> (150) and FC <sub>3</sub> (30) |
| Selective-SS-GAN (1000,1000) | <b>3.47 m</b>           | 3.38 m – 3.57 m             | <b>63 cm   15.36 %</b>                               | FC <sub>1</sub> (300), FC <sub>2</sub> (150) and FC <sub>3</sub> (50) |
| Selective-SS-GAN (1000,1500) | 3.77 m                  | 3.74 m – 3.79 m             | 33 cm   8.04 %                                       | FC <sub>1</sub> (300), FC <sub>2</sub> (150) and FC <sub>3</sub> (30) |
| Selective-SS-GAN (1000,2000) | 3.83 m                  | 3.76 m – 3.94 m             | 27 cm   6.58 %                                       | FC <sub>1</sub> (300), FC <sub>2</sub> (150) and FC <sub>3</sub> (30) |

para as 40 posições de validação a serem usadas para teste. Durante a construção do UJIndoorLoc, 520 APs são usados. No entanto, apenas 18 APs são detectados em cada posição e 91% dos RSSIs coletados são desconhecidos. Consequentemente, para simplificar a implementação, consideramos apenas APs detectados pelo menos uma vez durante a coleta de dados que sejam iguais a 190.

Após a reorganização e redução dos dados, podemos aplicar nosso algoritmo. Primeiramente, geramos  $mf = 28.000$  vetores RSSI falsos com GANs baseados em DNN, otimizados usando 0,01 como taxa de aprendizado durante 200 épocas. Um discriminador de camada oculta com 200 neurônios e um gerador de camada oculta com 200 neurônios são considerados. Para o algoritmo de otimização e as funções de ativação, usamos aqueles usados durante as simulações. Em segundo lugar, uma DNN é treinada em medições rotuladas durante 200 épocas com 50 como tamanho de lote para estimativa de pseudo-rótulos. A melhor DNN alcançável é composta por três camadas ocultas contendo 200 neurônios, 100 neurônios e 50 neurônios, respectivamente. Para a seleção de dados, consideramos que dividimos a zona de interesse em 625 classes com  $l = 6,4959$  m e  $w = 5,9760$  m. O processo de seleção de dados é seguido pela estimativa de localização, aplicando uma DNN treinada offline. Uma taxa de aprendizado igual a 0,01 foi selecionada usando 150/250 épocas e um tamanho de minilote igual a 50/100.

As arquiteturas DNN usadas para localização são apresentadas na Tabela 3.

Avaliamos a precisão da localização de diferentes algoritmos usando dados reais e falsos selecionados, ou seja GAN-SS Seletivo ( $mr$ ,  $ms$ ). A Tabela 3 apresenta o desempenho da localização (por exemplo, erro médio de localização, erro mínimo de localização e erro máximo de localização) e a melhoria da localização em comparação com o sistema de localização utilizando apenas 1.000 dados rotulados, ou seja, Supervisionado (1.000). Selecionamos diferentes números de posições falsas [700 - 2.000] do conjunto de  $mf = 28.000$  posições geradas. Observamos que a melhoria da localização variou entre 6,58% e 15,36%. A melhor precisão de localização é obtida com  $ms = 1.000$  posições geradas selecionadas, onde alcançamos um aumento de 15,36% na precisão de localização em relação ao algoritmo supervisionado convencional. A partir de 1.500 posições geradas selecionadas, notamos que o desempenho está saturado e não podemos fornecer melhor precisão com base nas posições de treinamento. Assim, os resultados experimentais reforçam os resultados baseados em dados simulados. Mesmo trabalhando em um ambiente realista com dispositivos altamente dinâmicos e heterogêneos, nosso sistema proposto alcança boa precisão de localização.

## VI. CONCLUSÃO

Neste artigo, vimos que GANs podem ser usadas para produzir dados sintéticos, excepcionalmente realistas, para complementar um conjunto de dados real, a fim de aprimorar o treinamento de uma DNN usada para localização. Portanto, essa técnica é muito útil em situações em que a coleta de dados é cara e demorada, como na localização em ambientes internos. Em particular, apresentamos um sistema GAN semissupervisionado seletivo para localização em ambientes internos, onde geramos dados falsos com base em dados reais coletados e rotulados, a fim de aumentar o desempenho da localização. Nossas soluções propostas foram validadas e testadas com base em diversas simulações que mostram que a combinação de dados coletados e dados gerados selecionados é benéfica em termos de desempenho de localização e custo de coleta de dados. O processo de geração de seleção melhora a precisão da localização em 21,69% em comparação com o método supervisionado padrão baseado no mesmo subconjunto de dados rotulados ao considerar 1.000 amostras rotuladas coletadas em 100 posições diferentes. Nosso método também funciona com dados reais, do banco de dados público UJIndoorLoc, resultando em uma melhoria de 15,36% na precisão da localização. Os resultados promissores deste artigo motivam a expansão do modelo desenvolvido, explorando novos métodos de ampliação de dados.

## REFERÊNCIAS

- [1] F. Zafari, A. Gkelias e KK Leung, "Uma pesquisa sobre sistemas e tecnologias de localização interna", *IEEE Commun. Surveys Tuts.*, vol. 21, n.º 3, pp. 2568–2599, 3.º trimestre, 2019.
- [2] A. Yassin, Y. Nasser, M. Awad, A. Al-Dubai, R. Liu, C. Yuen, R. Raulefs e E. Aboutanios, "Avanços recentes na localização interna: uma pesquisa sobre abordagens teóricas e aplicações", *IEEE Commun. Pesquisas Tuts.*, vol. 19, não. 2, pp. 1327–1346, 2º quart., 2016.
- [3] JA Del Peral-Rosado et al., "Whitepaper sobre novos métodos de localização para sistemas sem fio 5G e a Internet das Coisas", 2018.
- [4] A. Bourdoux, AN Barreto, B. van Liempd, C. de Lima, D. Dardari, D. Belot, E.-S. Lohan, G. Seco-Granados, H. Sarieddeen, H. Wymeersch, J. Suutala, J. Saloranta, M. Guillaud, M. Isomursu, M. Valkama, MRK Aziz, R. Berkvens, T. Sanguanpuak, T. Svensson e Y. Miao, "whitepaper 6G sobre localização e detecção", 2020, *arXiv:2006.01779*.
- [Online]. Disponível em: <http://arxiv.org/abs/2006.01779>
- [5] S. Liu, Y. Jiang e A. Striegel, "Estimativa de proximidade face a face usando Bluetooth em smartphones", *IEEE Trans. Mobile Comput.*, vol. 13, no. 4, pp. 811–823, abril de 2013.
- [6] S. He e S.-HG Chan, "Posicionamento interno baseado em impressão digital Wi-Fi: avanços e comparações recentes", *IEEE Commun. Surveys Tuts.*, vol. 18, n.º 1, pp. 466–490, janeiro de 2015.
- [7] B. Kempke, P. Pannuto e P. Dutta, "SurePoint: Explorando inundação de banda ultralarga e diversidade para fornecer localização interna robusta, escalável e de alta fidelidade: Resumo de demonstração", em *Proc. 14ª Conferência ACM. Rede embarcada. Sensor Syst. CD-ROM*, novembro de 2016, pp. 137–149.

- [8] ABMM Rahman, T. Li e Y. Wang, "Avanços recentes na localização interna por meio de luzes visíveis: uma pesquisa", *Sensors*, vol. 20, no. 5, p. 1382, março de 2020.
- [9] W. Shao, H. Luo, F. Zhao, Y. Ma, Z. Zhao e A. Crivello, "Posicionamento interno baseado em imagem de impressão digital e aprendizado profundo", *IEEE Access*, vol. 6, pp. 74699–74712, 2018.
- [10] N. Tadayon, MT Rahman, S. Han, S. Valaee e W. Yu, "Variação de decímetro com informações de estado do canal", *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3453–3468, julho de 2019.
- [11] L. Chen, I. Ahriz e D. Le Ruyet, "Impressão digital probabilística de localização interna com reconhecimento de AoA usando informações de estado do canal", *IEEE Internet Things J.*, vol. 7, nº 11, pp. 10868–10883, novembro de 2020.
- [12] Z. Yang, Z. Zhou e Y. Liu, "De RSSI a CSI: localização interna via resposta de canal", *ACM Comput. Surv.*, vol. 46, no. 2, pp. 1–32, 2013.
- [13] P. Kumar, L. Reddy e S. Varma, "Esquema de medição de distância e estimativa de erro para localização baseada em RSSI em redes de sensores sem fio", em *Proc. 5ª Conferência Internacional de Redes de Sensores de Comunicação Sem Fio (WCSN)*, dezembro de 2009, pp. 1–4.
- [14] Z. Yang e Y. Liu, "Qualidade da trilateração: localização iterativa baseada em confiança", *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 5, pp. 631–640, maio de 2010.
- [15] S. Yiu, M. Dashti, H. Claussen e F. Perez-Cruz, "Localização de impressão digital RSSI sem fio", *Signal Process.*, vol. 131, pp. 235–244, fev. 2017.
- [16] MI AlHajri, NT Ali e RM Shubair, "Localização interna para IoT usando seleção adaptativa de recursos: uma abordagem de aprendizagem de máquina em cascata", *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 11, pp. 2306–2310, nov. 2019.
- [17] W. Njima, I. Ahriz, R. Zayani, M. Terre e R. Bouallegue, "Deep CNN para localização interna em sistemas de sensores IoT", *Sensors*, vol. 19, não. 14, pág. 3127, julho de 2019.
- [18] W. Njima, M. Chafii, A. Nimr e G. Fettweis, "Recuperação de dados baseada em aprendizagem profunda para localização", *IEEE Access*, vol. 8, pp. 175741–175752, 2020.
- [19] Z. Chen, MI AlHajri, M. Wu, NT Ali e RM Shubair, "Uma nova abordagem de aprendizado profundo em tempo real para localização interna com base na identificação do ambiente RF", *IEEE Sensors Lett.*, vol. 4, no. 6, pp. 1–4, junho de 2020.
- [20] R. Wang, Z. Li, H. Luo, F. Zhao, W. Shao e Q. Wang, "Um algoritmo robusto de posicionamento de impressão digital Wi-Fi usando autocodificador de redução de ruído empilhado e perceptron multicamadas", *Remote Sens.*, vol. 11, no. 11, p. 1293, maio de 2019.
- [21] J. Fan e T. Chow, "Completamento de matriz baseado em aprendizagem profunda", *Neurocomputing*, vol. 266, pp. 540–549, novembro de 2017.
- [22] R. Ayyalasomayajula, A. Arun, C. Wu, S. Sharma, AR Sethi, D. Vasist e D. Bharadia, "Localização sem fio baseada em aprendizagem profunda para navegação interna", em *Proc. 26º ano. Internacional Conf. Computação Móvel. Netw.*, abril de 2020, pp.
- [23] W. Li, C. Zhang e Y. Tanaka, "Localização interna descentralizada baseada em aprendizagem federada e pseudo-orientada por rótulos via crowdsourcing móvel", *IEEE Sensors J.*, vol. 20, nº 19, pp. 11556–11565, outubro de 2020.
- [24] Y. Xia, L. Ma, Z. Zhang e Y. Wang, "Algoritmo de posicionamento semi-supervisionado em ambiente WLAN interno", em *Proc. Conf. (VTC Primavera)*, maio de 2015, pp. 1–5.
- [25] M. Zhou, Y. Tang, W. Nie, L. Xie e X. Yang, "GrassMA: Alinhamento de coletor semi-supervisionado baseado em gráfico para localização de WLAN interna", *IEEE Sensors J.*, vol. 17, nº 21, pp. 7086–7095, novembro de 2017.
- [26] A. Antoniou, A. Storkey e H. Edwards, "Redes adversárias generativas de aumento de dados", 2017, *arXiv:1711.04340*. [Online]. Disponível em: <http://arxiv.org/abs/1711.04340>
- [27] C. Bowles, R. Gunn, A. Hammers e D. Rueckert, "Aprendizagem GANsfer: Combinando dados rotulados e não rotulados para aumento de dados baseado em GAN," 2018, *arXiv:1811.10669*. [Online]. Disponível em: <http://arxiv.org/abs/1811.10669> [28] A. Brock, J. Donahue e K. Simonyan, "Treinamento GAN em larga escala para síntese de imagem natural de alta fidelidade," 2018, *arXiv:1809.11096*. [Online]. Disponível em: <http://arxiv.org/abs/1809.11096>
- [29] A. Odena, "Aprendizagem semi-supervisionada com redes adversárias generativas," 2016, *arXiv:1606.01583*. [Online]. Disponível em: <http://arxiv.org/abs/1606.01583> [30] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, DA Dickie, MV Hernández, J. Wardlaw e D. Rueckert, "Aumento de GAN: aumentando dados de treinamento usando redes adversárias generativas," 2018, *arXiv:1810.10863*. [Online]. Disponível em: <http://arxiv.org/abs/1810.10863>
- [31] A. Belmonte-Hernández, G. Hernandez-Penalzo, DM Gutiérrez e F. Alvarez, "Modelo recorrente para recuperação de rastreamento e posicionamento interno sem fio usando redes generativas", *IEEE Sensors J.*, vol. 20, no. 6, pp. 3356–3365, março de 2020.
- [32] KM Chen e RY Chang, "Aprendizagem semi-supervisionada com GANs para localização interna de impressão digital sem dispositivo", em *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, dezembro de 2020, pp. 1–6.
- [33] H. Zou, C.-L. Chen, M. Li, J. Yang, Y. Zhou, L. Xie e CJ Spanos, "Construção e adaptação de mapa de rádio interno WiFi automático habilitado para aprendizagem adversarial com robô móvel", *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6946–6954, agosto de 2020.
- [34] Q. Li, H. Qu, Z. Liu, N. Zhou, W. Sun, S. Sigg e J. Li, "AF-DCGAN: Amplitude apresenta GAN convolucional profundo para construção de impressão digital em sistemas de localização interna", *IEEE Trans. Tópicos Comput. Intell.*, vol. 5, no. 3, pp. 468–480, junho de 2021.
- [35] Y. Lei, D. Li, H. Zhang e X. Li, "Localização de impressão digital externa com recurso Wavelet baseada em ResNet e GAN de convolução profunda", *Symmetry*, vol. 12, nº 9, p. 1565, setembro de 2020.
- [36] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, JP Avariento, TJ Arnau, M. Benedito-Bordonau e J. Huerta, "UJIIndoorLoc: Um novo banco de dados de vários edifícios e vários andares para problemas de localização interna baseados em impressões digitais WLAN", em *Proc. Internacional Conf. Posicionamento interno Navega interno. (IPIN)*, outubro de 2014, pp.
- [37] KS Kim, R. Wang, Z. Zhong, Z. Tan, H. Song, J. Cha e S. Lee, "Serviços de localização em larga escala em acesso: Classificação hierárquica de edifícios/andares e estimativa de localização usando impressão digital Wi-Fi baseada em redes neurais profundas", *Fiber Integr. Opt.*, vol. 37, no. 5, pp. 277–289, set. 2018.
- [38] KS Kim, S. Lee e K. Huang, "Uma arquitetura de rede neural profunda escalável para localização interna de vários edifícios e andares com base em impressão digital Wi-Fi", *Big Data Anal.*, vol. 3, no. 1, pp. 1–17, dezembro de 2018.
- [39] IJ Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville e Y. Bengio, "Redes adversárias generativas", 2014, *arXiv:1406.2661*. [Online]. Disponível em: <http://arxiv.org/abs/1406.2661> [40] L. Mescheder, A. Geiger e S. Nowozin, "Quais métodos de treinamento para GANs realmente convergem?" em *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3481–3490.
- [41] DP Kingma e J. Ba, "Adam: Um método para otimização estocástica," 2014, *arXiv:1412.6980*. [Online]. Disponível em: <http://arxiv.org/abs/1412.6980>



**Wafa Njima** (Membro, IEEE) nasceu na Tunísia, em 1991. Formou-se em engenharia pelo Institut National des Sciences Appliquées et de Technologies, Túnis, em 2015, e obteve o doutorado. licenciatura em radiocomunicações pelo Conservatoire National des Arts et Métiers, Paris, em 2019, em colaboração com a Ecole Supérieure des Communications de Tunis, sob a supervisão do Prof.

Ridha Bouallegue. Atualmente, é pesquisadora de pós-doutorado no CNRS, ETIS UMR 8051, ENSEA, Universidade CY Cergy Paris, França. Suas publicações abrangem diversas áreas de pesquisa, incluindo processamento de sinais, comunicações sem fio, dados esparsos, localização interna, IoT e aprendizado de máquina para comunicações.



**MARWA CHAFII** (Membro, IEEE) recebeu o título de mestre em sistemas avançados de comunicação sem fio (SAR) e o título de doutor em engenharia elétrica pela CentraleSupélec, França, em 2013 e 2016, respectivamente. De 2014 a 2016, ela foi pesquisadora visitante na Universidade de Tecnologia de Poznan, Polônia; na Universidade de York, Reino Unido; na Universidade Nacional de Yokohama, Japão; e na Universidade de Oxford, Reino Unido. Ela se juntou à Cátedra Vodafone de Sistemas de Comunicação Móvel, Universidade Técnica de Dresden, Alemanha, em fevereiro de 2018, como líder de grupo

de pesquisa. Desde setembro de 2018, ela é professora associada na ENSEA, França, onde ocupa a Cátedra de Excelência em inteligência artificial da CY Initiative. Seus interesses de pesquisa incluem design avançado de forma de onda, aprendizado de máquina para comunicações sem fio e localização interna. Ela recebeu o prêmio de Melhor Doutorado, na França nas áreas de sinal, imagem e visão, e foi indicada entre as dez Estrelas em Ascensão em Redes de Computadores e Comunicações pela N2Women em 2020. Atualmente, é vice-presidente do IEEE ComSoc ETI sobre Aprendizado de Máquina para Comunicações, lidera o Grupo de Trabalho em Educação, ETI sobre Sensoriamento Integrado e Comunicações, é líder de pesquisa com mulheres em IA e gerencia o Comitê de Gênero da Comunidade AI4EU. Desde 2019, atua como editora associada da IEEE COMMUNICATIONS LETTERS, onde recebeu o prêmio de Melhor Editora em 2020.



**RAED M. SHUBAIR** (Membro Sênior, IEEE) recebeu o título de Ph.D. em engenharia elétrica pela Universidade de Waterloo, Canadá, em 1993. De 1993 a 2017, foi Professor Titular de Engenharia Elétrica na Universidade Khalifa, Emirados Árabes Unidos. Atualmente, é Professor Titular de Engenharia Elétrica afiliado à Universidade de Nova York (NYU) em Abu Dhabi. Seus cargos acadêmicos e de pesquisa atuais e anteriores também incluem o Instituto de Tecnologia de Massachusetts (MIT), a Universidade Harvard e a Universidade de Waterloo. Suas publicações abrangem

diversas áreas de pesquisa, incluindo comunicações 6G e terahertz, antenas modernas e eletromagnetismo aplicado, processamento de sinais e matrizes, aprendizado de máquina, IoT e localização de sensores, sensoriamento médico e nanobiomedicina. É membro da Academia de Eletromagnetismo do MIT e membro fundador do MIT Scholars of the Emirates e de cinco capítulos da sociedade IEEE, Emirados Árabes Unidos. Recebeu o Prêmio de Dissertação de Doutorado Distinto da Universidade de Waterloo por seu título de Ph.D. Recebeu diversos prêmios internacionais, incluindo o Distinguished Service Award da ACES Society, EUA, e da MIT Electromagnetics Academy, EUA. É também membro titular dos conselhos editoriais de diversos periódicos internacionais e atua regularmente nos comitês de direção, organização e técnica das principais conferências do IEEE em antenas, comunicações e processamento de sinais, incluindo IEEE AP-S/URSI, EuCAP, IEEE GlobalSIP, IEEE WCNC e IEEE ICASSP. É também membro do Conselho da European School of Antennas e Diretor Regional da IEEE Signal Processing Society na Região 8 do IEEE no Oriente Médio, e atuou como Presidente Fundador do Programa de Iniciativas Educacionais da IEEE Antennas and Propagation Society. É também fundador e presidente do IEEE na NYUAD.

Ele também é Diretor da Iniciativa Técnica Emergente da IEEE ComSoc sobre Aprendizado de Máquina para Comunicações. É também o Diretor Fundador do Programa IEEE UAE Distinguished Seminar Series, pelo qual foi selecionado para receber, juntamente com Mohamed Al-Hajri, do MIT, o Prêmio IEEE UAE do Ano de 2020. É também editor do IEEE JOURNAL OF ELECTROMAGNETICS, RF AND MICROWAVES IN MEDICINE AND BIOLOGY e do IEEE OPEN JOURNAL OF ANTENNAS AND PROPAGATION.



**ARSENIA (ERSI) CHORTI** (Membro Sênior, IEEE) é atualmente professora da École Nationale Supérieure de l'Electronique et des ses Applications (ENSEA); chefe do Grupo de Informação, Comunicações e Imagem (ICI), Laboratório ETIS UMR 8051; e pesquisadora visitante da Universidade de Princeton e da Universidade de Essex. Seus interesses de pesquisa incluem comunicações sem fio e segurança de sistemas sem fio para 5G e 6G, com foco particular em segurança da camada física, segurança com reconhecimento de contexto, protocolos de autenticação multifator, 5G/6G e IoT, detecção de anomalias, aprendizado de máquina para comunicações, acesso múltiplo não ortogonal (NOMA) e sinalização mais rápida que Nyquist. Ela é membro do IEEE INGR em Segurança e do Grupo de Trabalho de Padronização IEEE P1951.1 (Cidades Inteligentes). Ela está em licença semi-sabática (delegação do CNRS) desde março de 2021.



**H. VINCENT POOR** (Life Fellow, IEEE) recebeu o título de Ph.D. em EECS pela Universidade de Princeton, em 1977. De 1977 a 1990, ele foi membro do corpo docente da Universidade de Illinois em Urbana-Champaign. Desde 1990, ele faz parte do corpo docente da Universidade de Princeton, onde atualmente é Professor Universitário Michael Henry Strater. De 2006 a 2016, ele atuou como Reitor da Escola de Engenharia e Ciências Aplicadas de Princeton. Ele também ocupou cargos de visitante em várias outras universidades, incluindo, mais recentemente, em Berkeley e Cambridge. Seus interesses de pesquisa incluem teoria da informação, aprendizado de máquina e ciência de redes, e suas aplicações em redes sem fio, sistemas de energia e campos relacionados. Entre suas publicações nessas áreas está o próximo livro sobre aprendizado de máquina e comunicações sem fio (Cambridge University Press, 2021). Ele é membro da Academia Nacional de Engenharia e da Academia Nacional de Ciências, e membro estrangeiro da Academia Chinesa de Ciências, da Royal Society e de outras academias nacionais e internacionais. Seu trabalho recebeu reconhecimentos recentes, incluindo a Medalha Alexander Graham Bell do IEEE de 2017 e o título de Doutor em Engenharia.

Honoris Causa da Universidade de Waterloo concedido, em 2019.

• • •