

# Proposta de um Ambiente Interacional baseado em um Espaço Inteligente Programável

Pedro Vinicius dos Santos Custodio<sup>1</sup>, Felippe M. de Queiroz<sup>1</sup>,  
Douglas Almonfrey<sup>2</sup>, Wagner Augusto Aranda Cotta<sup>1</sup>  
Alexandre Pereira do Carmo<sup>2</sup>, Raquel Frizera Vassallo<sup>1</sup>

<sup>1</sup>Universidade Federal do Espírito Santo  
Departamento de Engenharia Elétrica - Vitória, ES, Brasil

<sup>2</sup>Instituto Federal do Espírito Santo  
Vitória/Guarapari, ES, Brasil

{dalmonfrey,alexandre.carmo}@ifes.edu.br, raquel@ele.ufes.br  
{vdscpedro,mendonca.felippe,waacotta}@gmail.com

**Abstract.** Recently, the creation of smart spaces has been the focus of many works. One of the features, usually sought for these environments, is the interaction with users. Aiming at this direction, this work proposes the creation of an Interactional Environment based on a Programmable Intelligent Space, which guarantees flexibility and scalability to the system. Thus, in this paper some essential services, such as 2D detection and 3D skeleton localization, to support more complex ones, are developed and tested. To validate the proposed approach, an application that makes use of all the essential services presented is executed and evaluated.

**Resumo.** Nos últimos anos, a criação de espaços inteligentes tem sido o foco de muitos trabalhos. Uma das funcionalidades que se busca em alguns desses ambientes é a interação com usuários. Nesse sentido, esse trabalho propõe um Ambiente Interacional baseado em um Espaço Inteligente Programável, que garante flexibilidade e escalabilidade ao sistema. Assim, neste artigo alguns serviços essenciais, como detecção 2D e localização 3D de esqueletos, para suportar outros mais complexos, são desenvolvidos e testados. Para validação do ambiente proposto, uma aplicação que usa todos os serviços aqui apresentados é executada e avaliada.

## 1. Introdução

Na última década, o conceito de espaços inteligentes (*smart spaces*) se consolidou na literatura, apesar de existirem diferentes definições para esse conceito [Bhardwaj et al. 2018]. De maneira geral, um espaço inteligente pode ser considerado como um ambiente, equipado com uma rede de sensores como câmeras, microfones e outros sensores, capazes de adquirir informação do ambiente, e atuadores como robôs e outros dispositivos, aptos a atuar sobre o meio. Esses dispositivos físicos podem ser controlados por serviços computacionais implementados no espaço, e são responsáveis por coletar e analisar dados, tomar decisões e atuar no ambiente quando necessário. [Carmo et al. 2019].

Quando uma das principais funções, planejadas para um espaço inteligente, é interagir com seus usuários, seus sensores, atuadores e serviços devem ser escolhidos e projetados para incluírem essa capacidade. Nesse caso, um conjunto de recursos fundamentais deve ser considerado, tais como formas de interação e interfaces adequadas [Vega-Barbas et al. 2018], programação ubíqua [Möller 2016], formas de detecção de pessoas [Chen et al. 2019] e métodos de reconhecimento de gestos e/ou ações [Zhang et al. 2019, Mitra and Acharya 2007]. É por meio desses recursos e uma infraestrutura de hardware e software, que dê suporte a sua realização, que se pode agregar a devida interatividade ao espaço e estendê-la a uma cidade inteligente.

Nesse contexto, o trabalho aqui proposto busca criar um Ambiente Interacional baseado em um Espaço Inteligente Programável (PIS - *Programmable Intelligent Space*), com serviços voltados para o desenvolvimento de aplicações interativas. Esse PIS possui uma arquitetura baseada em Internet das Coisas (IoT - *Internet of Things*), de fácil utilização e que abstrai toda infraestrutura e equipamentos disponíveis no ambiente. A arquitetura de microsserviço (MSA) [Fowler and Lewis 2014], aplicada na infraestrutura de software, fornece a capacidade de programação e reutilização a nível de serviço. Assim, novos equipamentos e aplicações podem ser incorporados facilmente à plataforma. Os serviços são orquestrados para serem flexíveis e confiáveis, apresentando escalabilidade, dentro da realidade de suas operações.

O PIS utilizado é baseado em visão computacional, tendo câmeras como os principais sensores do ambiente. Além de fornecerem um grande volume de informação, câmeras são um dos sensores mais utilizados em interfaces de interação. Nos espaços onde as câmeras são comumente usadas para videomonitoramento, adicionar a funcionalidade de interação com seus usuários pode ser bastante interessante.

Nessa direção, alguns trabalhos que abordam interfaces modulares e programáveis para o PIS [Picoreti et al. 2018], assim como métodos de detecção e localização de seres humanos e robôs [Almonfrey et al. 2018, Queiroz et al. 2018], já foram desenvolvidos pelo nosso grupo de pesquisa visando a criação de um ambiente interacional. Assim, neste artigo, algumas dessas funcionalidades serão transformadas em serviços de suporte para interação, assim como outros serviços serão incluídos. É mandatório que tais serviços sejam desenvolvidos visando aplicações em tempo real, de forma que questões como latência e sincronismo serão sempre consideradas.

Entre os serviços implementados estão o de detecção de esqueletos e suas juntas em imagens 2D, o serviço de reconstrução 3D dos esqueletos, além de serviços essenciais como captura de imagens e transformação de referencial, todos desenvolvidos para dar suporte a um Ambiente Interacional. Alguns testes para avaliação desses serviços serão apresentados e discutidos. Ao final, uma aplicação que faça uso de todos os serviços conjuntamente é implementada para validar o ambiente. Vale ressaltar que os serviços aqui apresentados servem de base para futuros serviços de reconhecimento de gestos ou ações. Entretanto, esses últimos ainda não aparecem no escopo deste artigo, uma vez que representam um trabalho atualmente em andamento [dos-Santos et al. 2020] no grupo de pesquisa.

Assim, este artigo está organizado da seguinte forma: a Seção 2 traz alguns trabalhos relacionados; a Seção 3 descreve rapidamente o PIS e os serviços implemen-

tados para o Ambiente Interacional; os experimentos e os resultados são discutidos na Seção 4 e, por fim, a Seção 5 apresenta as conclusões e trabalhos futuros.

## 2. Trabalhos Relacionados

Segundo [Ryan 1999, Gherghescu 2018], um sistema interativo deve ser capaz de responder às ações dos usuários. Nesse sentido, a capacidade de ler e compreender a atividade do usuário é feita por meio de sensores e nós de computação que coletam e processam as informações, possibilitando, com isso, a tomada de decisões e execução de respostas. Esse é o contexto do Ambiente Interacional que se propõe na Seção 3, o qual é baseado em um espaço inteligente, concebido como plataforma base para permitir o desenvolvimento de serviços para a interação com usuários.

Em [Wang et al. 2015], os autores propõem um método baseado em gestos para uma interação com um espaço inteligente. A interação acontece através de um vocabulário pré-definido para representar tarefas desejadas. Apesar de interessante, a interação não é algo realmente natural e intuitiva, sendo necessário o aprendizado do vocabulário escolhido. Já em [Casillas-Perez et al. 2016], é proposto um sistema de reconhecimento de gestos para que usuários controlem o ambiente. Os resultados são promissores, mas a infraestrutura do ambiente não é o foco do trabalho e, portanto, não apresenta a flexibilidade e escalabilidade do ambiente aqui proposto.

Já o trabalho de [Vega-Barbas et al. 2018] faz uma discussão de aspectos cognitivos e interacionais que devem ser levados em conta no projeto de espaços inteligentes que se propõem a serem interativos. Segundo os autores, o uso de certos padrões de interação pode aumentar a confiança que os usuários desenvolvem quando usam dispositivos ou serviços de IoT.

Por fim, em [Glas. et al. 2015] desenvolve-se uma rede de robôs sociais para uso em centros comerciais. O sistema é baseado em sensores do tipo varredura, como lasers ou RGB-D, e desenvolvido como uma solução única. Diferentemente, o trabalho aqui proposto usa câmeras convencionais, conceitos de IoT e computação em nuvem, onde os robôs são apenas mais um serviço dentre outros que compõem o ambiente interativo e o tornam mais facilmente replicável.

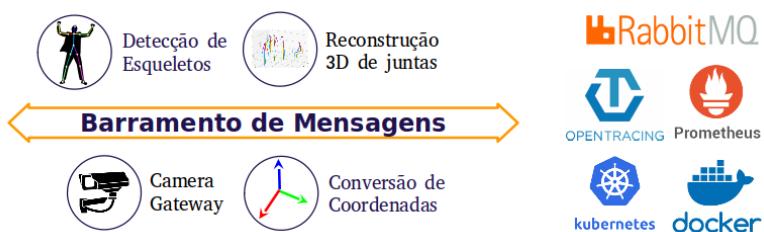
## 3. Ambiente Interacional baseado no PIS

O Ambiente Interacional aqui proposto é suportado por um PIS baseado em visão computacional, que possui uma rede de câmeras como seu principal sensor. Essas câmeras transmitem imagens para uma nuvem, responsável por processar os dados e determinar, entre outras informações, a localização de seres humanos no ambiente, assim como as juntas de seus esqueletos. Além de pessoas, o espaço também é capaz de localizar robôs ou outros objetos no seu interior.

A infraestrutura de hardware e software, que compõe o PIS, possui diferentes níveis de programabilidade. Parâmetros como frequência de amostragem das câmeras e velocidade dos robôs podem ser alterados a nível do dispositivo. *Switches SDN (Software-Defined Networking)* e um barramento virtual de mensagens garantem a programabilidade no nível da comunicação via rede. O nível de serviços contempla: calibração de câmeras, conversão de coordenadas, detecção e localização de esqueletos e barramento de mensagens. Os serviços são autônomos e com baixo nível

de acoplamento, facilitando a alocação de recursos e reusabilidade. Finalmente, no nível da aplicação, diferentes tecnologias para acessar a plataforma são fornecidas, como protobuf ou REST (*Representational State Transfer*). Toda essa programabilidade permite que requisitos do domínio de visão computacional e interação sejam atendidos durante o desenvolvimento da aplicação, com alto grau de abstração.

A Figura 1 exibe alguns dos serviços implementados para o Ambiente Interacional. A base da arquitetura do espaço inteligente é um barramento de mensagens, que emprega o protocolo AMQP (*Advanced Message Queuing Protocol*). Todos os serviços se comunicam por meio desse barramento, publicando e consumindo mensagens. Nesse barramento, o gerenciamento de filas e de conexões é feito usando o *broker* RabbitMQ (<http://www.rabbitmq.com/>). Os serviços são implementados usando contêineres Docker (<https://www.docker.com/>) como modelo de virtualização. Os contêineres são gerenciados por meio do Kubernetes (<https://kubernetes.io/>), que funciona como orquestrador da nuvem. Essa junção de tecnologias garante que os serviços sejam escaláveis tanto em termos de recursos quanto em número de instâncias. Além disso, a confiabilidade é garantida devido à possibilidade de monitoramento e reinicialização de serviços que se tornem indisponíveis devido a falhas de hardware ou software.



**Figura 1. Alguns dos serviços implementados para o Ambiente Interacional.**

Para a implementação do Ambiente Interacional foi desenvolvido um conjunto de serviços, dentre os quais destacam-se: câmera gateway; conversão de coordenadas; detecção de esqueletos; reconstrução 3D de juntas e barramento de mensagens. Tais serviços permitem localizar seres humanos e suas juntas, sendo um suporte para aplicações interativas mais complexas como reconhecimento de gestos e pessoas.

É válido comentar que, embora não esteja no escopo deste trabalho, o espaço inteligente possui ainda um serviço de localização de padrões do tipo ArUco [Garrido-Jurado et al. 2014] e um serviço de reconhecimento de objetos baseado no YOLOV3 [Redmon and Farhadi 2018]. Todos esses serviços, já implementados e em funcionamento, poderão ser usados em futuras aplicações de interação e cooperação, assim como localização e controle de robôs móveis, nesse Ambiente Interacional.

### 3.1. Serviços desenvolvidos até o momento para o Ambiente Interacional

De forma a viabilizar o desenvolvimento de um ambiente interacional, os seguintes serviços foram implementados no contexto do PIS.

- **Camera Gateway**

Cada instância do gateway é responsável por estabelecer a conexão com uma ou mais câmeras, utilizando o protocolo adequado para comunicação e confi-

guração. Após a conexão, o gateway recebe as imagens no formato *RAW*, as converte para *JPEG*, tornando-as disponíveis para os demais serviços.

- **Conversão de Coordenadas**

Em uma rede de câmeras, cada uma possui um referencial em relação ao referencial global do ambiente. Logo, um serviço para realizar a conversão de coordenadas entre referenciais se faz necessário para que as informações adquiridas, em cada uma das câmeras, possam ser representadas em um referencial global comum.

- **Barramento de Mensagens**

Responsável por criar um canal por onde os demais serviços irão se comunicar, o barramento de mensagens permite que serviços possam ser facilmente adicionados ou removidos do PIS. Além disso, esse serviço também gerencia as filas e conexões usando o protocolo AMQP e o *broker* RabbitMQ, tornando essa complexa gestão transparente aos desenvolvedores. Essa forma de comunicação é um mecanismo de troca de mensagens baseado em tópicos, em um esquema publicação/inscrição (*publish/subscribe*), agregam flexibilidade aos serviços desenvolvidos.

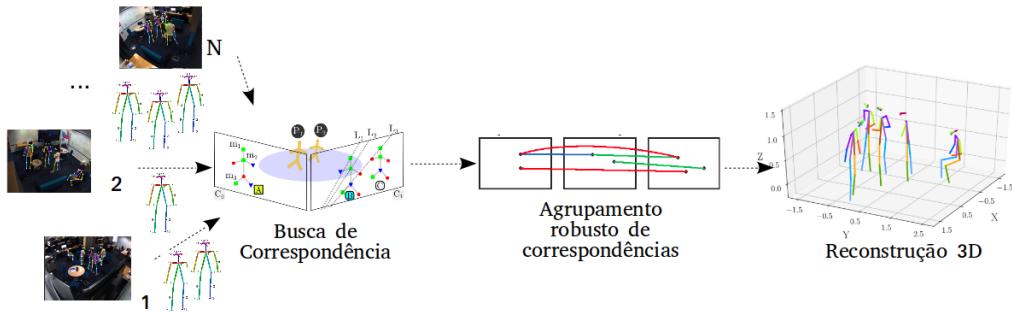
- **Serviço de Detecção de Esqueletos**

O serviço de detecção de esqueletos foi desenvolvido com base em [Queiroz et al. 2018], onde se realiza a detecção 2D das juntas de esqueletos presentes em uma imagem. Cada esqueleto é constituído por um conjunto de até 18 juntas representadas como coordenadas (u, v) na imagem da câmera na qual foram detectadas. Tal serviço é baseado no detector OpenPose [Wei et al. 2016], que se executado em GPU, é capaz de detectar múltiplos esqueletos na mesma imagem em algumas dezenas de milissegundos.

- **Serviço de Reconstrução 3D das Juntas dos Esqueletos**

O serviço de reconstrução 3D das juntas também é baseado em [Queiroz et al. 2018] e utiliza os esqueletos detectados pelo serviço descrito anteriormente. A cada instante de tempo, os esqueletos detectados numa mesma imagem recebem um identificador único, relacionado à câmera na qual foram identificados. Note que, para que a reconstrução 3D seja possível, as imagens capturadas devem pertencer a um sistema de câmeras calibrado. O serviço possui basicamente três etapas: (1) busca de correspondências, (2) agrupamento robusto de correspondências e (3) reconstrução 3D das juntas (vide Figura 2). Inicialmente, todas as imagens são analisadas duas a duas, usando-se geometria epipolar, para se encontrar as possíveis correspondências entre as juntas detectadas. Isso gera uma lista de correspondências, a qual é passada para a segunda etapa, onde o intuito é percorrer e verificar essa lista, eliminando as correspondências errôneas e agrupando todas aquelas associadas à mesma junta. Finalmente, a partir das correspondências consideradas corretas, realiza-se a reconstrução 3D através de um sistema de equações, que relaciona as coordenadas das juntas correspondentes.

Por fim, é importante ressaltar que a arquitetura do PIS provê escalabilidade horizontal, por meio do aumento do número de instâncias, e vertical, por meio do aumento do poder computacional alocado para cada instância, de um determinado serviço. Por exemplo, caso um nó de processamento tenha alcançado seu limite de



**Figura 2. Fluxo de reconstrução 3D do esqueleto.**

recursos computacionais, novas instâncias de serviços necessários poderão ser criadas em outro nó. Isso permite que o processamento e o tráfego de dados seja distribuído ao longo da infraestrutura, não gerando um gargalo no sistema.

#### 4. Experimentos e Resultados

A avaliação do Ambiente Interacional e seus serviços será feita por meio de uma aplicação que emprega, de forma integrada, os serviços descritos anteriormente. Serão analisados o erro de localização 3D das juntas, a taxa de detecção dos esqueletos, assim como os tempos de processamento dos serviços implementados. A aplicação corresponde à construção de um mapa de ocupação do ambiente em tempo real, o qual representa os lugares por onde pessoas passaram ao longo do tempo.

Conforme já mencionado, os serviços de detecção dos esqueletos e reconstrução 3D das juntas foram implementados a partir de [Queiroz et al. 2018]. Nesse trabalho específico, foram utilizadas as sequências “160224 haggling1”, “160226 haggling1” e “160422 haggling1” do *dataset* público Panoptic [Joo et al. 2015]<sup>1</sup>, o modelo de esqueleto com 18 juntas e 10 de 31 câmeras HD disponíveis no *dataset*, com o intuito de avaliar o erro da detecção de esqueletos, assim como o erro final de reconstrução 3D das juntas.

##### 4.1. Taxa de Perda dos Esqueletos

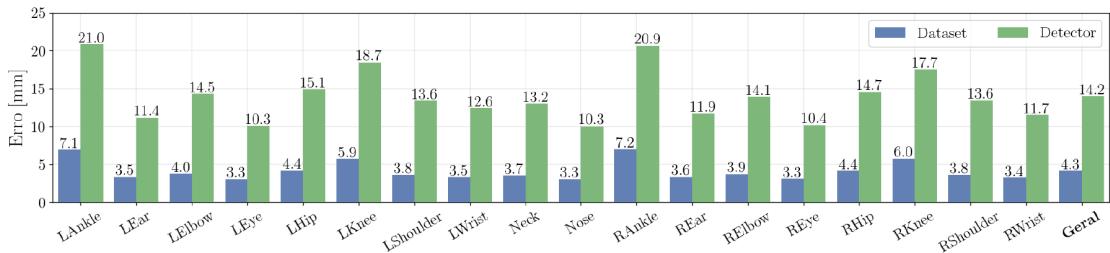
Para o serviço de detecção dos esqueletos, apenas cerca de 10% das anotações do conjunto de dados selecionados do *dataset* não foram recuperadas. Mesmo sendo considerada suficiente para o Ambiente Interacional proposto, acredita-se que o emprego futuro de alguma técnica de rastreamento das juntas pode melhorar e tornar essa taxa menor. A discussão detalhada pode ser vista em [Queiroz et al. 2018].

##### 4.2. Erro de localização 3D das juntas dos esqueletos

O erro final de localização 3D depende tanto do serviço de detecção dos esqueletos nas imagens, quanto do serviço de reconstrução. O primeiro insere erros na posição 2D das juntas, devido às imprecisões inerentes ao processamento das imagens, enquanto o segundo insere mais uma parcela de erro devido às incertezas do processo de correspondência e solução do sistema de equações da geometria epipolar.

<sup>1</sup><http://domedb.perception.cs.cmu.edu/dataset.html>

Inicialmente, para se analisar o erro devido apenas ao serviço de reconstrução, as anotações 3D das juntas das pessoas do *dataset* (*ground truth*), foram reprojetadas nas imagens, usando-se os parâmetros de calibração das câmeras. A seguir, as juntas reprojetadas foram passadas ao serviço de reconstrução, cujo resultado foi comparado ao *ground truth*, gerando os valores da Figura 3 com a legenda “Dataset”. Já a análise, representada pela legenda “Detector”, inclui tanto o serviço de detecção de esqueletos, quanto o serviço de reconstrução 3D de juntas. Nesse caso, o que se obteve foi o erro do processo completo de localização dos esqueletos 3D.



**Figura 3. Erro médio de reconstrução de cada junta e geral, considerando *ground truth* e as juntas obtidas pelo serviço de detecção [Queiroz et al. 2018]**

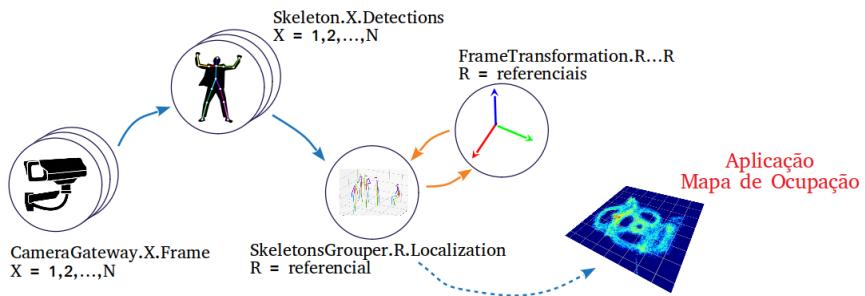
Note que, para o processo completo envolvendo os dois serviços avaliados, o erro médio de cada junta e erro médio geral são menores, respectivamente, que 22 mm e 15 mm (legenda “Detector”). É importante mencionar que o valor desse erro depende também da resolução das imagens e da calibração do sistema multicâmeras. Mesmo que esse erro possa sofrer variações, os valores obtidos foram considerados aceitáveis para o Ambiente Interacional proposto.

#### 4.3. Mapa de Ocupação do Ambiente

Para validar o funcionamento do Ambiente Interacional foi desenvolvida uma aplicação que cria um mapa de ocupação do ambiente. Essa aplicação usa todos os serviços descritos nesse artigo, mostrando como todo o sistema funciona em conjunto.

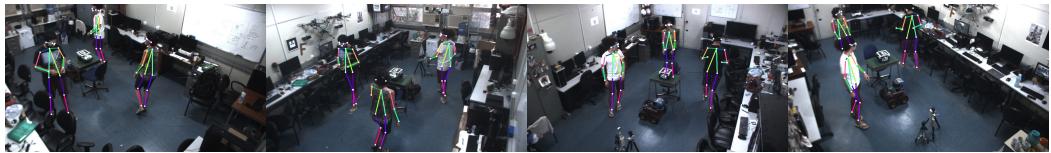
Conforme a Figura 4, as câmeras publicam suas imagens nos tópicos CameraGateway.X.Frame, onde X representa o ID de cada câmera. Em seguida, o serviço de detecção de esqueletos, inscrito nesse tópico, recebe a imagem, realiza o processamento e publica os *keypoints* das juntas no tópico Skeletons.X.Detections (vide Figura 5). Cabe, então, ao serviço de reconstrução 3D de juntas, inscrito em todas as instâncias do tópico Skeletons.X.Detections, receber todas as detecções e realizar o agrupamento e a reconstrução 3D dos esqueletos. Este serviço ainda faz uso do serviço de Conversão de Coordenadas, o qual realiza a transformação entre os referenciais existentes no ambiente e publica em FrameTransformation.R...R. Por fim, as juntas 3D são publicadas no tópico SkeletonsGrouper.R.Localization, onde R representa o referencial escolhido para a reconstrução 3D. Por meio desse tópico, a localização tridimensional das juntas dos esqueletos é então usada pela aplicação, a qual acumula tais localizações no tempo e constrói o mapa de ocupação do ambiente.

Durante o experimento, indivíduos se deslocaram pela área de trabalho (Figura 6a), e a análise temporal cumulativa das suas localizações resultou no mapa de

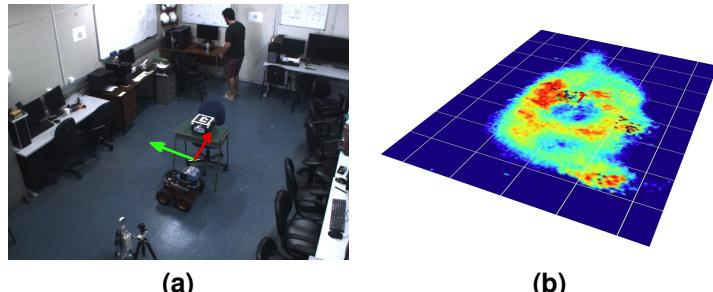


**Figura 4. Fluxo dos serviços usados na Aplicação do Mapa de Ocupação.**

ocupação da Figura 6b. Repare que as áreas ocupadas pela mobília do ambiente não apresentam ocupação relevante e os locais, representados com cores mais quentes, indicam onde houve mais ocupação por mais tempo.



**Figura 5. Reconstrução dos esqueletos 3D na perspectiva das câmeras.**



**Figura 6. (a) Sala usada para os experimentos. (b) Mapa de ocupação gerado.**

#### 4.4. Tempos de processamento dos serviços

Para execução em tempo real, o tempo de processamento dos serviços deve ser menor que o intervalo definido pela taxa de amostragem das câmeras, para que não haja perda de quadros. Assim, foi calculado o tempo médio de processamento dos serviços mais pesados envolvidos na aplicação, na qual usou-se uma infraestrutura de computação em nuvem com escalonamento automático de GPU e uma taxa de captura de imagens de 10 fps, o que define um intervalo de 100 ms. Os tempos de processamento de cada serviço estão mostrados na Tabela 1.

A soma dos tempos de processamento da cadeia de serviços é menor do que 100 ms, mostrando que eles podem ser usados com aplicações que incluem interação com o usuário em tempo real, sem causar desconfortos ou atrasos na sua resposta.

### 5. Conclusão

Neste trabalho, apresentou-se um Ambiente Interacional baseado em um Espaço Inteligente Programável. Os serviços implementados foram descritos e avaliados considerando-se sempre a sua utilização em aplicações de tempo real.

Serviço	Tempo Médio de Processamento
Câmera gateway (captura de imagem)	15.46 ms
Detecção de esqueletos	61.60 ms
Reconstrução 3D	5.29 ms

**Tabela 1. Tempo de processamento dos serviços.**

Uma aplicação para construção de um mapa de ocupação foi implementada, fazendo uso de todos os serviços apresentados. Tais serviços foram capazes de atender os requisitos de sincronização e tempo de resposta da aplicação, o que atesta a viabilidade do ambiente proposto para aplicações em tempo real.

Além disso, uma vez que serviços básicos são fornecidos, novas aplicações para o Ambiente Interacional podem ser implementadas de forma rápida e transparente, dada a modularização e programabilidade do PIS. Desta forma, como trabalhos futuros, está prevista a construção de novos serviços e aplicações de interação com usuários, que envolvam recursos de detecção e reconhecimento de gestos e ações.

## Agradecimentos

Este projeto conta com o suporte da FAPES - Fundação de Amparo à Pesquisa e Inovação do Espírito Santo, através do projeto Cidades Inteligentes e Segurança Pública no Espírito Santo, EDITAL FAPES/SEP No 20/2018.

## Referências

- Almonfrey, D., do Carmo, A. P., de Queiroz, F. M., Picoreti, R., Vassallo, R. F., and Salles, E. O. T. (2018). A flexible human detection service suitable for intelligent spaces based on a multi-camera network. *International Journal of Distributed Sensor Networks*, 14(3):1550147718763550.
- Bhardwaj, S., Ozcelebi, T., Lukkien, J. J., and Lee, K. M. (2018). Smart space concepts, properties and architectures. *IEEE Access*, 6:70088–70112.
- Carmo, A. P., Vassallo, R. F., Queiroz, F. M., Picoreti, R., Fernandes, M. R., Gomes, R. L., Martinello, M., Dominicini, C. K., Guimarães, R., Garcia, A. S., Ribeiro, M. R. N., and Simeonidou, D. (2019). Programmable intelligent spaces for Industry 4.0: Indoor visual localization driving attocell networks. *Transactions on Emerging Telecommunications Technologies*, page e3610.
- Casillas-Perez, D., Macias-Guarasa, J., Marron-Romera, M., Fuentes-Jimenez, D., and Fernandez-Rincon, A. (2016). Full body gesture recognition for human-machine interaction in intelligent spaces. In Ortúñoz, F. and Rojas, I., editors, *Bioinformatics and Biomedical Engineering. IWBBIO 2016*, volume 9656, Cham. Springer International Publishing.
- Chen, R., Wang, X., Liu, Y., Wang, S., and Huang, S. (2019). A survey of pedestrian detection based on deep learning. In *International Conference in Communications, Signal Processing, and Systems*, pages 1511–1516. Springer.
- dos-Santos, C. C., Samatelo, J. L. A., and Vassallo, R. F. (2020). Dynamic gesture recognition by using cnns and star rgb: A temporal information condensation. *Neurocomputing*, in Press.

- Fowler, M. and Lewis, J. (2014). Microservices a definition of this new architectural term. <http://martinfowler.com/articles/microservices.html>.
- Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F., and Marín-Jiménez, M. (2014). Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280 – 2292.
- Gherghescu, I. (2018). Interactive spaces a change in scenographic aesthetics. In *2018 IEEE 14th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 159–163.
- Glas., D. F., Kamei, K., Kanda, T., Miyashita, T., and Hagita, N. (2015). Human-robot interaction in public and smart spaces. *Intelligent Assistive Robots: Recent Advances in Assistive Robotics for Everyday Activities*, 106:235–273.
- Joo, H., Liu, H., Tan, L., Gui, L., Nabbe, B., Matthews, I., Kanade, T., Nobuhara, S., and Sheikh, Y. (2015). Panoptic studio: A massively multiview system for social motion capture. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3334–3342.
- Mitra, S. and Acharya, T. (2007). Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(3):311–324.
- Möller, D. P. F. (2016). *Ubiquitous Computing*, pages 185–234. Springer International Publishing, Cham.
- Picoreti, R., Pereira do Carmo, A., Mendonça de Queiroz, F., Salles Garcia, A., Frizera Vassallo, R., and Simeonidou, D. (2018). Multilevel observability in cloud orchestration. In *2018 IEEE 16th International Conference on DASC/PiCom/DataCom/CyberSciTech*, pages 776–784.
- Queiroz, F., Picoreti, R., Canuto, C., Fernandes, M., and Vassallo, R. (2018). Estimating tridimensional coordinates of skeleton joints in a multicamera system. In *Anais do XIV Workshop de Visão Computacional*.
- Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv*.
- Ryan, M.-L. (1999). Immersion vs. interactivity: Virtual reality and literary theory. *SubStance*, 28(2):110–137.
- Vega-Barbas, M., Pau, I., Augusto, J. C., and Seoane, F. (2018). Interaction patterns for smart spaces: A confident interaction design solution for pervasive sensitive iot services. *IEEE Access*, 6:1126–1136.
- Wang, X., Bernardos, A. M., Besada, J. A., Metola, E., and Casar, J. R. (2015). A gesture-based method for natural interaction in smart spaces. *Journal of Ambient Intelligence and Smart Environments*, 7:535–562.
- Wei, S.-E., Ramakrishna, V., Kanade, T., and Sheikh, Y. (2016). Convolutional pose machines. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732.
- Zhang, H.-B., Zhang, Y.-X., Zhong, B., Lei, Q., Yang, L., Du, J.-X., and Chen, D.-S. (2019). A comprehensive survey of vision-based human action recognition methods. *Sensors*, 19(5):1005.