

USO DE UM ESPAÇO INTELIGENTE BASEADO EM VISÃO COMPUTACIONAL PARA O CONTROLE DE FORMAÇÃO DE ROBÔS MÓVEIS

Alexandre Pereira do Carmo², Felipe Mendonça de Queiroz¹,
Clebeson Canuto dos Santos¹, Leonardo de Assis Silva²,
Raquel Frizera Vassallo¹

¹Universidade Federal do Espírito Santo
Departamento de Engenharia Elétrica
Vitória, ES, Brasil

²Instituto Federal do Espírito Santo
Coordenadoria de Eletrotécnica
Guarapari, ES, Brasil

{mendonca.felippe,clebeson.canuto}@gmail.com

{alexandre.carmo,leonardo.assis}@ifes.edu.br, raquel@ele.ufes.br

Abstract. *It is well known that using only odometry for robot localization leads to an increasing error accumulation. This problem becomes greater when one seeks to coordinate the movements of more than one robot to act in formation. Therefore, this work proposes the use of an intelligent space based on computer vision as a service infrastructure for mobile robotics applications, such as formation control. The architecture based on micro-services and cloud computing offers the application flexibility, low latency and scalability. The results obtained demonstrate the feasibility of using intelligent spaces as a provider and manager of this type of application.*

Resumo. *É sabido que usar apenas odometria na localização de robôs móveis leva a um crescente acúmulo de erro. Esse problema torna-se maior quando se busca coordenar os movimentos de mais de um robô para atuarem em formação. Sendo assim, este trabalho propõe o uso de um espaço inteligente baseado em visão computacional como uma infraestrutura de serviços para aplicações de robótica móvel, como exemplo, o controle de formação de robôs. A arquitetura baseada em microserviços e computação em nuvem oferece à aplicação flexibilidade, baixa latência e escalabilidade. Os resultados obtidos demonstram a viabilidade de se utilizar espaços inteligentes como provedor e gerenciador desse tipo de aplicação.*

1. Introdução

Saber a postura precisa de um robô em seu ambiente de trabalho é um dos problemas centrais da robótica móvel, pois a tarefa a ser realizada pelo mesmo pode ser melhor executada caso tais informações sejam conhecidas.

A fim de solucionar esse problema, comumente utiliza-se a odometria, método que calcula a posição e orientação de um robô móvel por meio da integração no tempo

dos giros de suas rodas [O’Kane 2006]. Apesar de ser muito utilizada, a odometria acaba acumulando erros no decorrer do tempo. Isso ocorre devido à imprecisão dos sensores utilizados na medição e a situações que causam deslizamentos das rodas, como a falta de atrito com o solo ou superfícies irregulares, por exemplo, em que a localização é atualizada sem que o robô tenha se movido. Dessa maneira, quanto mais tempo o robô estiver movimentando-se, maior será o erro em sua postura.

Para diminuir esse erro e melhorar a estimativa da postura de robôs móveis, geralmente a informação de outros sensores é utilizada além dos odômetros, como leitores RFID e sensores RGB-D [Mi and Takahashi 2015, Ganganath and Leung 2012]. Contudo, acaba sendo necessário que sejam adicionados novos dispositivos ao ambiente e ao próprio robô. Isso pode tornar-se um problema quando se tem um grande número de robôs trabalhando no ambiente, ou ainda, quando a plataforma robótica não possibilita a inclusão de novos dispositivos.

Dessa forma, é necessário utilizar uma abordagem que permita a localização de robôs no ambiente, de maneira que diminua a dependência de configuração do espaço físico e do próprio dispositivo. Assim, por se basear em informações fornecidas por uma ou mais câmeras presentes no ambiente, a odometria visual pode ser uma candidata à resolução desse problema. Uma das vantagens dessa abordagem é que o erro na estimativa da postura é relativo apenas à qualidade da calibração de cada câmera e, por isso, não se acumula no tempo [Scaramuzza and Fraundorfer 2011], o que é importante para a realização de tarefas complexas. Por outro lado, se a calibração do sistema visual não for bem estimada, o erro pode ser considerável para a realização de atividades que exigem mais precisão de posicionamento.

Além disso, nos últimos anos o videomonitoramento vem sendo utilizado em diversas áreas, como segurança residencial, monitoramento de tráfego, dentre outros [Wang 2013]. Com isso, nesses ambientes que já possuem uma infraestrutura de câmeras instaladas, é possível utilizar tal infraestrutura para a execução de tarefas que requerem a localização de dispositivos sem muitas modificações. No entanto, para que a odometria visual possa ser melhor empregada, é necessário que as câmeras estejam bem calibradas e que o sistema como um todo realize o processamento das imagens e identificação das características relevantes. Ambientes desse tipo podem ser classificados como espaços inteligentes baseados em visão computacional.

Segundo [Lee et al. 1999], um espaço inteligente pode ser entendido como um ambiente capaz de tomar decisões por meio de vários sensores e computadores que se comunicam em rede. Nesse contexto, além de atuadores do ambiente [Rampinelli et al. 2014], robôs podem coletar dados que ajudam nas tomadas de decisão, funcionando como sensores do sistema. Isso permite ao espaço obter as informações que precisa enquanto gerencia os robôs na realização de suas tarefas.

O espaço inteligente utilizado neste trabalho está descrito em [Almonfrey et al. 2018]. Dentro desse espaço, o robô trabalha apenas em uma superfície plana, e a odometria visual pode ser estimada por uma única câmera. No entanto, quanto maior for o número de câmeras, melhor será a estimativa da postura do robô e maior será a área de cobertura onde o mesmo poderá atuar. Por esse motivo, o espaço inteligente é equipado com quatro câmeras monoculares

devidamente calibradas que fornecem dados visuais para vários serviços distribuídos em rede. Esses serviços são responsáveis por processar as imagens, calcular a localização visual do robô, dentre outras tarefas, possibilitando assim, a obtenção de informações precisas sobre a posição e orientação do robô no ambiente.

Com tudo o que foi posto, o objetivo principal deste trabalho é mostrar como a odometria visual fornecida pelo espaço inteligente supracitado pode ser utilizada em tarefas robóticas, de maneira distribuída e com o mínimo de erro na estimação da postura. Logo, para dar comprovações do que se propõe, foi desenvolvido um controlador de formação final entre dois robôs móveis, que é realimentado com informações fornecidas por vários serviços distribuídos em rede no espaço em questão.

Sendo assim, as seguintes seções abordarão, respectivamente: (2) um comparativo entre diferentes métodos de localização; (3) um resumo da arquitetura do espaço inteligente baseado em visão computacional; (4) a descrição do modelo de localização visual utilizado; (5) a descrição do controle de formação; (6) os experimentos e os resultados obtidos, e por fim; (7) as conclusões e trabalhos futuros.

2. Comparação entre Métodos de Localização

Um comparativo entre as principais técnicas de localização em ambientes internos é mostrado em [R.F. et al. 2017]. Nesse trabalho, mostra-se que muitas abordagens exploram tecnologias como ultrassom, WiFi, ZigBee, RFID, Bluetooth entre outras para localização. Também são avaliadas tecnologias passivas como aquelas baseadas em geomagnetismo, luz, som, incluindo a inercial.

Entre as abordagens consideradas, mostra-se que visão computacional pode fornecer uma precisão de localização na ordem de poucos centímetros, uma vez que imagens podem fornecer uma riqueza de informações que, processadas devidamente, são capazes de gerar estimativas de localização bastante precisas.

A Tabela 1, mostra esse comparativo, onde pode-se ver que técnicas de localização por visão computacional possuem uma melhor precisão quando comparadas com outras técnicas tradicionais em ambientes internos.

Tecnologia	Precisão Aproximada	Desvantagem
Tecnologias com codificação de sinal		
Infravermelho	57cm ~ 23m	Interferência do Sol
VLC	10cm	Alta custo
Ultrassom	1cm ~ 2m	Interferências
WiFi	1.5m	Vulnerável à mudanças no AP
Bluetooth	30cm ~ 10m	Necessita mapeamento de sinal
ZigBee	25cm	Necessita equipamento específico
RFID	1m ~ 5m	Precisão muito baixa
UWB	15cm	Alto custo
Tecnologias passivas sem codificação de sinal		
Geomagnético	2m	Necessita de mapeamento
Inercial	2m	Acumula erro
Som ambiente	-	Sem precisão
Luz ambiente	10cm ~ alguns metros	Sensibilidade a luminosidade
Visão Computacional	1cm ~ 1m	Sensibilidade a luminosidade

Tabela 1 - Comparativo entre métodos de localização indoor [R.F. et al. 2017].

3. Espaço Inteligente baseado em Visão Computacional

Espaços inteligentes podem ser descritos como ambientes interativos equipados com uma rede de sensores (e.g., câmeras, microfones, ultrassons) capazes de extrair informações do meio, e um conjunto de atuadores (e.g., robôs, dispositivos móveis, telas informativas), que podem ser controlados por diferentes serviços computacionais para agir e modificar o ambiente. Além de controlar os atuadores, os serviços computacionais podem analisar as informações adquiridas para ajudar na execução de tarefas e tomada de decisões.

Os sensores, atuadores e serviços computacionais são suportados por uma infraestrutura de software encarregada de fornecer a capacidade de comunicação e abstrações de acesso necessárias. Serviços e recursos de um dispositivo específico (sensor ou atuador) podem ser acessados e utilizados por diferentes entidades, como outros serviços, aplicações ou até mesmo outros dispositivos.

Como mencionado, nosso espaço inteligente é baseado em visão computacional [Almonfrey et al. 2018, Carmo et al. 2019]. Portanto, ele é instrumentado com uma rede de câmeras IP capazes de capturar imagens e vídeos digitais. O sistema também é capaz de controlar atuadores como um robô, e câmeras são usadas como os principais sensores do ambiente. Para alcançar uma compreensão de alto nível do ambiente, uma infraestrutura de software foi projetada para lidar com a tarefa de processar e analisar dados extraídos das câmeras distribuídas em tempo real.

A infraestrutura de software do espaço inteligente proposto é concebida como uma plataforma de desenvolvimento, i.e. como um PaaS (Platform as a Service). Portanto, os desenvolvedores de aplicações podem usar diferentes serviços computacionais, mesmo que alguns deles tenham sido projetados inicialmente para serem usados por outra aplicação específica. É por isso que esses serviços devem ser flexíveis o suficiente para atender, ao mesmo tempo, requisitos específicos das aplicações, mas também fornecer uma abstração de programação de alto nível para os desenvolvedores. O estilo da arquitetura de microsserviço (MSA) foi aplicado no projeto dessa infraestrutura de software para fornecer a capacidade de programação e reutilização necessárias a nível de serviço. Esses recursos facilitam a criação e a implantação de aplicações de tempo real para os desenvolvedores, além de permitir que elas integrem novos serviços à plataforma.

Além disso, nossa plataforma é implantada sobre uma infraestrutura de nuvem como IaaS (Infrastructure as a Service) para atender a requisitos específicos de aplicações de visão computacional, como baixa latência, grande largura de banda e alta capacidade de processamento. A capacidade de programação deste IaaS permite que a plataforma atenda aos requisitos rigorosos, citados anteriormente, além de fornecer a escalabilidade necessária para aplicações de visão computacional e robótica em tempo real.

4. Localização Visual

Neste trabalho, foram utilizados dois robôs móveis, um *Pioneer 3-AT* e um *P3-DX*, [MobileRobots 2017]. Sobre cada um deles, foram colocados dois círculos coloridos, nas cores vermelho e amarelo (*P3-AT*) e verde e amarelo (*P3-DX*), mostrados na Figura 1. O centroide de cada círculo é estimado após a segmentação por

cor das imagens capturadas no espaço inteligente. Tal segmentação foi realizada nas imagens convertidas para espaço de cor LUV, o qual é menos ruidoso à variação de luz e oferece uma distinção clara das três cores usadas na identificação dos robôs.

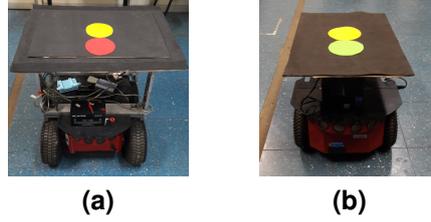


Figura 1. Robôs utilizados e seus respectivos padrões de identificação: (a) círculos amarelo e vermelho, e (b) círculos amarelo e verde.

A seguir, a localização dos robôs no espaço inteligente é obtida convertendo-se as coordenadas dos centroides dos círculos que compõe o padrão em um ponto tridimensional. Para isso, as câmeras do espaço inteligente estão calibradas, i.e., seus parâmetros intrínsecos e extrínsecos são conhecidos.

O modelo de câmera utilizado foi o modelo *pinhole*, apresentado na Equação 1 em coordenadas homogêneas. Nessa equação, λ_i é um fator de escala, $\tilde{\mathbf{m}}_i = [u_i \ v_i \ 1]^T$ é um ponto na imagem da câmera i , \mathbf{K}_i é a matriz de parâmetros intrínsecos, $\mathbf{\Pi}$ a matriz de projeção, $[\mathbf{R}_i, \mathbf{T}_i]$ a matriz de parâmetros extrínsecos, composta por uma rotação e uma translação e, por fim, $\tilde{\mathbf{M}} = [x \ y \ z \ 1]^T$ corresponde ao ponto tridimensional, no referencial global no qual as câmeras foram calibradas, que gera as projeções $\tilde{\mathbf{m}}_i$ em cada imagem. O subíndice i tem objetivo de diferenciar as câmeras, e as variáveis indicadas com $\tilde{}$ estão representadas em coordenadas homogêneas.

$$\lambda_i \tilde{\mathbf{m}}_i = \mathbf{K}_i \mathbf{\Pi} [\mathbf{R}_i, \mathbf{T}_i] \tilde{\mathbf{M}} \quad (1)$$

Uma vez que o robô se desloca no plano $z = 0$ e a altura do padrão é constante, pode-se assumir que a coordenada z dos pontos que se deseja reconstruir é sempre conhecida. Desse modo, na Equação 1, o termo $\mathbf{K}_i \mathbf{\Pi} [\mathbf{R}_i, \mathbf{T}_i]$ pode ser representado por uma matriz \mathbf{A}_i de dimensão 3×4 , cujas colunas serão indicadas por \mathbf{a}_i^c .

$$\lambda_i \tilde{\mathbf{m}}_i = \mathbf{A}_i [x \ y \ z \ 1]^T \quad (2)$$

Então, a partir da Equação 2 obtém-se o sistema da Equação 3, que possui solução única para as incógnitas x , y e λ_i , pois z é constante, o ponto $\tilde{\mathbf{m}}_i$ é obtido na imagem e a matriz \mathbf{A}_i é conhecida porque o sistema é calibrado.

$$\begin{bmatrix} \mathbf{a}_i^1 & \mathbf{a}_i^2 & -\tilde{\mathbf{m}}_i \end{bmatrix} \begin{bmatrix} x \\ y \\ \lambda_i \end{bmatrix} = \begin{bmatrix} -z\mathbf{a}_i^3 - \mathbf{a}_i^4 \end{bmatrix} \quad (3)$$

Entretanto, se forem utilizadas duas ou mais câmeras para a localização, não é necessária uma informação extra. Pode-se montar um sistema de equações com os pontos $\tilde{\mathbf{m}}_i$ nas n imagens que observam o ponto $\tilde{\mathbf{M}}$, $\forall n \geq 2$. Nesse caso, reescreve-se a Equação 1 na forma da Equação 4, na qual as incógnitas são λ_i e $\tilde{\mathbf{M}}$, sem a sua

representação em coordenadas homogêneas.

$$\lambda_i \tilde{\mathbf{m}}_i = \mathbf{K}_i (\mathbf{R}_i \mathbf{M} + \mathbf{T}_i) \quad (4)$$

Manipulando-se a Equação 4 e escrevendo na forma matricial, tem-se

$$\begin{bmatrix} -\mathbf{I} & (\mathbf{K}_i \mathbf{R}_i)^{-1} \tilde{\mathbf{m}}_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \lambda_i \end{bmatrix} = \mathbf{R}_i^{-1} \mathbf{T}_i \quad (5)$$

onde \mathbf{I} é uma matriz identidade 3×3 .

A Equação 5 pode ser empilhada formando um sistema de equações, onde para cada câmera adicionada, inclui-se mais um fator de escala λ , sendo o vetor \mathbf{M} o mesmo para todas as câmeras. Logo, tem-se a Equação 6, na qual $(\mathbf{K}_i \mathbf{R}_i)^{-1} \tilde{\mathbf{m}}_i$ é representado por \mathbf{P}_i , e $\mathbf{0}_{m \times n}$ é uma matriz de zeros de dimensão $m \times n$.

$$\begin{bmatrix} -\mathbf{I} & \mathbf{P}_1 & \mathbf{0}_{3 \times n-1} & \\ \vdots & \vdots & \vdots & \vdots \\ -\mathbf{I} & \mathbf{0}_{3 \times i-1} & \mathbf{P}_i & \mathbf{0}_{3 \times n-i} \\ \vdots & \vdots & \vdots & \vdots \\ -\mathbf{I} & & \mathbf{0}_{3 \times n-1} & \mathbf{P}_n \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \lambda_1 \\ \vdots \\ \lambda_i \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^{-1} \mathbf{T}_1 \\ \vdots \\ \mathbf{R}_i^{-1} \mathbf{T}_i \\ \vdots \\ \mathbf{R}_n^{-1} \mathbf{T}_n \end{bmatrix} \quad (6)$$

Portanto, identificando o padrão sobre o robô em uma ou mais câmeras no mesmo instante, é possível determinar as coordenadas tridimensionais dos centroides dos círculos em relação ao referencial global utilizando a Equação 3 ou 6. A partir dos centroides, a posição e a orientação de cada um dos robôs é determinada.

5. Controle de Formação

O estudo de caso desse trabalho é realizar o controle de posição final de uma formação composta por dois robôs móveis através de um espaço inteligente. O modelo de controle de formação foi baseado em [Brandao et al. 2009] e está mostrado na Figura 2a. As variáveis de formação são representadas no vetor $\mathbf{q} = [x_f \ y_f \ \rho_f \ \alpha_f]^T$, no qual x_f e y_f representam o ponto médio entre os robôs, ρ_f é a distância entre eles, e α_f é a orientação da reta que passa pelo centro dos mesmos.

O controlador proposto possui uma estrutura multicamadas, que subdivide cada etapa do controle de formação em um módulo independente, tornando a estrutura mais flexível. A Figura 2b mostra a estrutura e as variáveis de cada etapa.

Os vetores \mathbf{h}_1 e \mathbf{h}_2 correspondem às posições (x_1, y_1) e (x_2, y_2) de cada robô e formam o vetor $\mathbf{x} = [\mathbf{h}_1^T \ \mathbf{h}_2^T]^T$. Já a função $f(\mathbf{x})$ transforma as posições dos robôs nas variáveis de formação x_f, y_f, ρ_f e α_f , sendo definida por

$$f(\mathbf{x}) = \begin{bmatrix} x_f \\ y_f \\ \rho_f \\ \alpha_f \end{bmatrix} = \begin{bmatrix} (x_1 + x_2)/2 \\ (y_1 + y_2)/2 \\ \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \end{bmatrix}. \quad (7)$$

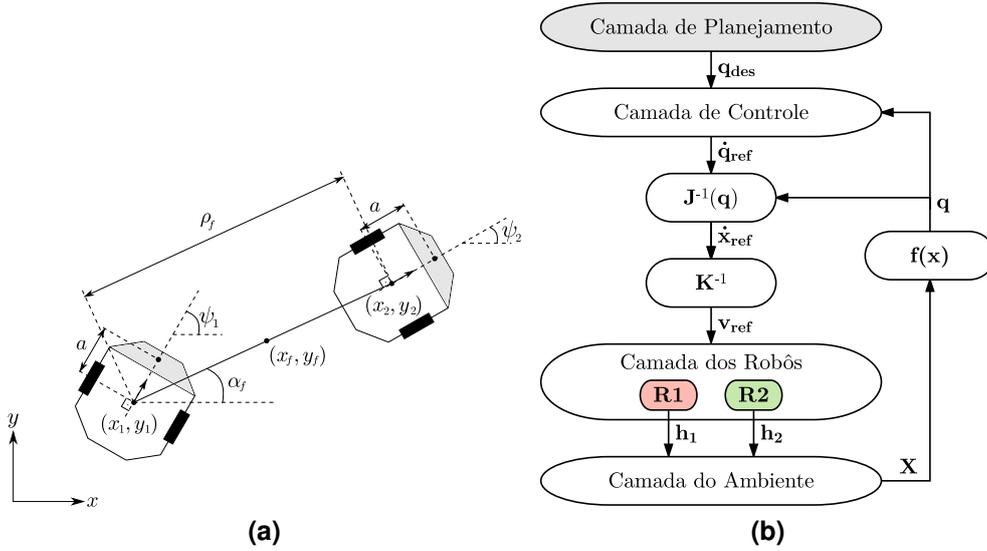


Figura 2. (a) Representação do modelo de formação. (b) Estrutura multicamadas utilizada para a modelagem do controlador.

A matriz $\mathbf{J}^{-1}(\mathbf{q})$ é a função jacobiano que mapeia as velocidades das variáveis de formação para as velocidades dos robôs, calculada a partir das derivadas parciais de $\mathbf{J}^{-1}(\mathbf{q}) = \partial \mathbf{x}_{m \times 1} / \partial \mathbf{q}_{n \times 1}$.

$$\mathbf{J}^{-1}(\mathbf{q}) = \begin{bmatrix} \frac{\partial x_1}{\partial x_f} & \frac{\partial x_1}{\partial y_f} & \frac{\partial x_1}{\partial \rho_f} & \frac{\partial x_1}{\partial \alpha_f} \\ \frac{\partial y_1}{\partial x_f} & \frac{\partial y_1}{\partial y_f} & \frac{\partial y_1}{\partial \rho_f} & \frac{\partial y_1}{\partial \alpha_f} \\ \frac{\partial x_2}{\partial x_f} & \frac{\partial x_2}{\partial y_f} & \frac{\partial x_2}{\partial \rho_f} & \frac{\partial x_2}{\partial \alpha_f} \\ \frac{\partial y_2}{\partial x_f} & \frac{\partial y_2}{\partial y_f} & \frac{\partial y_2}{\partial \rho_f} & \frac{\partial y_2}{\partial \alpha_f} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\frac{\cos(\alpha_f)}{2} & \rho_f \frac{\sin(\alpha_f)}{2} \\ 0 & 1 & -\frac{\sin(\alpha_f)}{2} & -\rho_f \frac{\cos(\alpha_f)}{2} \\ 1 & 0 & \frac{\cos(\alpha_f)}{2} & -\rho_f \frac{\sin(\alpha_f)}{2} \\ 0 & 1 & \frac{\sin(\alpha_f)}{2} & \rho_f \frac{\cos(\alpha_f)}{2} \end{bmatrix} \quad (8)$$

Por sua vez a matriz \mathbf{K}^{-1} é composta pelas matrizes de cinemática inversa de cada robô.

$$\mathbf{K}^{-1} = \begin{bmatrix} \mathbf{K}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2^{-1} \end{bmatrix} = \begin{bmatrix} \cos(\alpha_1) & \sin(\alpha_1) & 0 & 0 \\ -\frac{\sin(\alpha_1)}{a_1} & \frac{\cos(\alpha_1)}{a_1} & 0 & 0 \\ 0 & 0 & \cos(\alpha_2) & \sin(\alpha_2) \\ 0 & 0 & -\frac{\sin(\alpha_2)}{a_2} & \frac{\cos(\alpha_2)}{a_2} \end{bmatrix} \quad (9)$$

Os vetores \mathbf{q}_{des} e $\dot{\mathbf{q}}_{ref}$ correspondem ao estado desejado da formação e às velocidades da formação, as quais são calculadas pela Equação 10, na qual \mathbf{L} é a matriz diagonal com os fatores de escala para cada sinal de controle, \mathbf{k}_p é também diagonal e contém os ganhos proporcionais de cada variável controlada, e $\hat{\mathbf{q}}$ é o erro dado por $\mathbf{q}_{des} - \mathbf{q}$.

$$\dot{\mathbf{q}}_{ref} = \mathbf{L} \tanh(\mathbf{L}^{-1} \mathbf{k}_p \hat{\mathbf{q}}), \quad (10)$$

Por fim, o vetor \mathbf{v}_{ref} é o vetor com as velocidades linear e angular que devem ser aplicadas a cada robô da formação. Vale ressaltar que o controlador também é um serviço do espaço inteligente, que envia as informações de velocidade angular e linear para cada robô. Desse modo, cabe ao robô apenas a função de receber e executar os comandos enviados pelo sistema do espaço inteligente.

6. Experimentos

Para realização dos experimentos foi desenvolvida uma aplicação de visualização e envio de comandos aos robôs. Essa aplicação é composta de diferentes serviços reutilizáveis, da mesma forma como visto em [Carmo et al. 2019]. Neste artigo, será dado foco a dois deles: o serviço de localização visual e o serviço de controle de formação de robôs. Para isso, o passo inicial é selecionar a posição final desejada de cada robô móvel na tela de visualização das imagens das câmeras. Tais posições além de indicar a posição final de cada robô, definem o aspecto desejado para a formação, i.e., a distância entre robôs e a orientação da reta que os une.

A Figura 3 mostra a tela do usuário após a marcação das posições finais desejadas – marcadores vermelho e verde. Os pontos devem ser selecionados na imagem de apenas uma das câmeras e as coordenadas na imagem são convertidas para o referencial do mundo para que seja definido o vetor \mathbf{q}_{des} . Vale ressaltar que é possível realizar a reconstrução do ponto selecionado na imagem de uma câmera, resultando em um ponto no referencial do mundo porque é definido *a priori* que o ponto selecionado está localizado sempre no piso do laboratório, ou seja, $z = 0$.

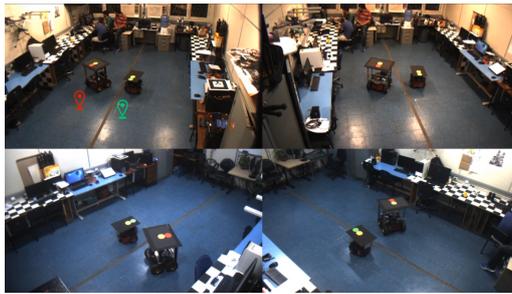


Figura 3. Aplicação desenvolvida para visualização e envio de comandos.

Para o experimento em questão, o controlador recebe o comando de posição final da formação $\mathbf{q}_{des} = [-1516.51, -337.862, 1454.79, 74.7578]$, no qual as coordenadas do ponto central da formação, x_f e y_f , e a distância entre os centros dos robôs ρ_f estão em milímetros; e a orientação da formação (α_f) está em graus. Nesse experimento, a orientação final do robô estava a 180° da orientação inicial, de modo que toda a formação era obrigada a rotacionar, o que poderia atrapalhar o processo de controle e de identificação dos mesmos pelo sistema visual.

A Figura 4a mostra as posições dos robôs durante a execução da tarefa de controle de formação, além da indicação de sete instantes da execução da tarefa, a fim de mostrar a evolução da formação. Além disso, são apresentados os erros de posição, orientação e distância da formação na Figura 4b, na qual verifica-se que todos os erros das variáveis de formação convergiram assintoticamente para zero.

Além disso, na Figura 5, está apresentado o número de detecções simultâneas do padrão para cada robô. Nota-se que existem instantes nos quais o robô 1 é detectado apenas por uma câmera. Mesmo assim, foi possível determinar sua postura. A não detecção feita pelas outras câmeras pode ter vários motivos, como oclusão ou condições de iluminação que impossibilitem a segmentação. Portanto, isso mostra a vantagem de se utilizar mais de uma câmera com sobreposição de vistas para localização dos robôs no espaço inteligente.

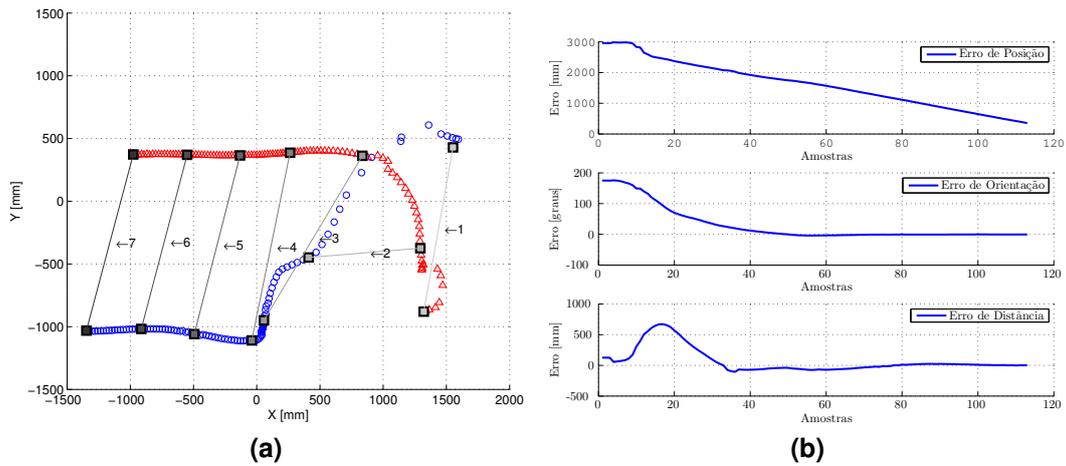


Figura 4. (a) Evolução das posições dos robôs e (b) erros de posição, orientação e distância da formação.

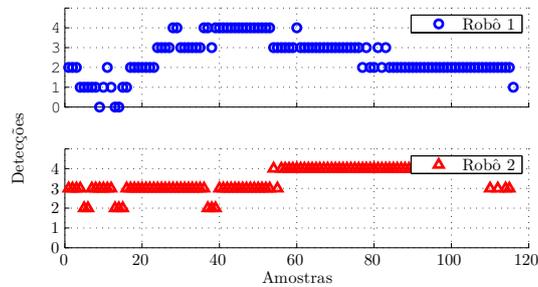


Figura 5. Número de detecções simultâneas do padrão para cada robô.

7. Conclusão

Este trabalho teve como principal objetivo apresentar uma das utilizações de um espaço inteligente baseado em visão computacional para robótica móvel. Para isso, foi desenvolvida uma aplicação de controle de formação, na qual as posturas dos robôs são determinadas apenas por informação visual.

A partir do experimento descrito, observa-se que o desempenho do controlador utilizado foi satisfatório. Desse modo, é possível concluir que a utilização do espaço inteligente como provedor e gerenciador de aplicações que necessitam de odometria visual para realizar tarefas traz bons resultados.

Várias vantagens podem ser citadas no uso de câmeras para localização de robôs móveis, como o erro fixo, redundância de sensores, evitando assim problemas de oclusão, e diminuição do sensoriamento embarcado nos dispositivos robóticos. Além disso, câmeras permitem diversas aplicações dentro da área de espaços e cidades inteligentes, como detecção de pessoas, gestos, segurança pública, entre outros.

Uma vez que a infraestrutura de espaço inteligente utilizada é baseada em serviços, novas aplicações poderão ser implementadas a partir da composição destes e de outros serviços. Essa característica de reutilização de serviços transforma o espaço inteligente baseado em visão computacional em uma promissora plataforma que servirá de base para novos trabalhos na área de robótica.

Agradecimentos

Este projeto conta com o suporte da RNP - Rede Nacional de Ensino e Pesquisa através do GT-MobiLysa aprovado no Edital de 2019. Agradecemos também à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES pelo suporte financeiro dado através da Bolsa de Doutorado concedida ao terceiro autor.

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):155014771876355.
- Brandao, A. S., Martins, F. N., Rampinelli, V. T. L., Sarcinelli-Filho, M., Bastos-Filho, T. F., and Carelli, R. (2009). A multi-layer control scheme for multi-robot formations with adaptive dynamic compensation. In *2009 IEEE International Conference on Mechatronics*, pages 1–6.
- 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.
- Ganganath, N. and Leung, H. (2012). Mobile robot localization using odometry and kinect sensor. In *2012 IEEE International Conference on Emerging Signal Processing Applications*, pages 91–94. IEEE.
- Lee, J.-H., Ando, N., and Hashimoto, H. (1999). Intelligent space for human and mobile robot. In *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*, pages 784–784. IEEE.
- Mi, J. and Takahashi, Y. (2015). Performance analysis of mobile robot self-localization based on different configurations of RFID system. In *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1591–1596. IEEE.
- MobileRobots, O. A. (2017). Adept mobile robots.
- O’Kane, J. M. (2006). Global localization using odometry. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 37–42. IEEE.
- Rampinelli, M., Covre, V. B., de Queiroz, F. M., Vassallo, R. F., Bastos-Filho, T. F., and Mazo, M. (2014). An intelligent space for mobile robot localization using a multi-camera system. *Sensors*, 14(8):15039–15064.
- R.F., B., J.P., G.-V., C.E., G.-T., D., M.-R., C., V.-R., and J., F. (2017). *Evolution of indoor positioning technologies: a survey*. *Sensors*.
- Scaramuzza, D. and Fraundorfer, F. (2011). Visual odometry [tutorial]. *IEEE robotics & automation magazine*, 18(4):80–92.
- Wang, X. (2013). Intelligent multi-camera video surveillance: A review.