VISCO (VIEW, SCAN, AND CONTROL IT): uso de visão computacional para descoberta de serviços em ambientes residenciais inteligentes

Paulo Filipe Dantas
Universidade Federal do Ceará
Fortaleza, Ceará, Brazil
paulodantas@great.ufc.br

José Gilvan Rodrigues Maia
Universidade Federal do Ceará
Fortaleza, Ceará, Brazil
gilvan@virtual.ufc.br

Windson Viana
Universidade Federal do Ceará
Fortaleza, Ceará, Brazil
windson@great.ufc.br

Abstract

The widespread of smart objects in our daily lives request the creation and analysis of new service discovery mechanisms and interaction techniques. In this work, we designed and evaluated a pointing-based interaction mechanism based on a Convolutional Neural Network classification method. We called it ViSCo (View, Scan, and Control it), which extends the openHAB service discovery mechanism of smart objets. ViSCo aggregates the users' field of view, captured by the camera of their smartphones, to reduce the service discovery results. 17 users evaluated the final solution remotely, in an environment with virtual devices. Participants used the ViSCo approach to find and control virtual devices by pointing to real objects in their homes (e.g., their TVs). System Usability Scale (SUS) survey about ViSCo results showed a good level of acceptance, with an average score of 83.97.

Keywords: IoT, *Smart Home*, CNN, computer vision, object classification

1 Introdução

A proliferação de dispositivos conectados à Internet das Coisas (do Inglês, *Internet of Things* - IoT) já apresenta impacto em várias facetas do cotidiano. Alguns desses dispositivos são chamados de Objetos Inteligentes (do Inglês, *Smart Object* - SO). Em [7] um SO é definido como um dispositivo físico (ou um conjunto de dispositivos) conectado à Internet e que interage tanto com usuários quanto com outros SOs . Um SO pode incluir serviços não físicos que visam ajudar usuários a realizar suas tarefas, tais como um assistente pessoal virtual (e.g., *Alexa*).

Diversas formas de interação de um usuário com um SO podem ser encontradas na literatura [4, 8, 9, 13]. Em [8], são apresentadas e avaliadas quatro formas de interação física com um SO que ainda permanecem atuais. **Tocar** (do Inglês, *touching*): a interação demanda que o usuário ponha o seu *smartphone* em contato com o SO com o qual deseja interagir. **Apontar** (do Inglês, *pointing*): onde o usuário interage com o SO apontando para ele com um smarphone e um dispositivo *laser pointer*. Essa interação pode ocorrer por *QR Codes*, sensores de luzes anexados ao SO [9] ou por visão computacional. **Escanear** (do Inglês, *scanning*): o usuário visualiza uma lista de SOs que estão próximos a ele com base em informações de um serviço de descoberta da rede de acessso (e.g., via *Bluetooth* ou por meio de Multicast DNS). E a **Mediada pelo usuário** (do Inglês, *user-mediated*): o usuário deve informar algum dado do SO com o qual deseja estabelecer a comunicação, como, por exemplo, um código de verificação ou um identificador, ou mais recentemente, comandos de voz. Um ponto em comum em todas essas formas de interação é que todos os objetos devem estar passíveis de conexão em alguma rede de comunicação. Caso o contrário, o usuário não poderá interagir com as diversas funções que este objeto apresenta por meio do seu *smartphone*, por exemplo.

Este trabalho apresenta uma abordagem do tipo **Apontar** que visa também aperfeiçoar o mecanismo de descobertas de SOs. Denominada ViSCo (*View, Scan, and Control it*), ela une um serviço de descoberta já existente com a capacidade dos dispositivos móveis atuais de executar algoritmos de reconhecimento de objetos por meio de visão computacional. Assim, o usuário aponta a câmera do *smartphone* para o objeto que deseja controlar e, via algoritmos de visão computacional, a categoria do SO é identificada e usada como um filtro contextual do serviço de descoberta.

2 View, Scan, and Control it - ViSCo

A Figura 1 detalha as etapas envolvidas no processo de identificação, busca e controle dos SOs (um vídeo ilustrando a abordagem também pode ser acessado no link http://bit.ly/visco-2021). A primeira etapa é a ação do usuário de informar qual SO ele tem o interesse de controlar. Para isso, basta que ele aponte a câmera do seu *smartphone* para o SO desejado. A partir deste momento, o algoritmo de classificação de objetos é acionado e passa a classificar as imagens obtidas pela câmera (etapa 2). As imagens são classificadas usando uma CNN (do Inglês, *Convolutional Neural Network*) construída nesta pesquisa. Esta CNN classifica as imagem da câmera em categorias preestabelecidas que representam os tipos de SOs

In: IV Concurso de Teses e Dissertações (CTD 2022), Curitiba, Brasil. Anais Estendidos do Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia). Porto Alegre: Sociedade Brasileira de Computação, 2022.

ISSN 2596-1683

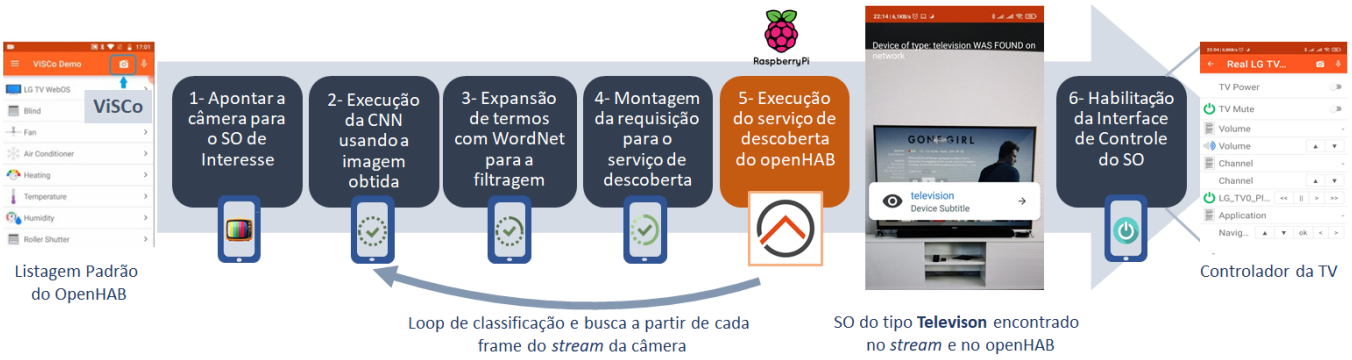

Figure 1. Etapas da busca de SOs com a abordagem ViSCo

(e.g., ar-condicionado, TV). Com a categoria do SO definida, a aplicação prepara um filtro/consulta a ser enviado ao serviço de descoberta. Este filtro usa um módulo de expansão de termos (etapa 3) criado para construir o padrão de busca. Na etapa 4, a aplicação móvel envia uma requisição para o serviço de descoberta do openHAB. Este por sua vez retorna todos os SOs compatíveis com a busca (etapa 5). A aplicação exibe, então, uma lista contendo apenas estes SOs. De posse dos dispositivos de interesse, o usuário pode então ativar a interface de controle do SO sugerido. As etapas 2 a 5 são executadas em *loop*, enquanto o *smartphone* estiver com a câmera ativa. As imagens são enviadas em sequência para o algoritmo.

2.1 View - classificação de SOs via CNN

Uma CNN foi criada para categorizar automaticamente o SO que está sendo apontado pelo usuário. Esta CNN foi baseada em um retreino (*fine-tuning*) da CNN MobileNet v1 [3] usando a técnica de *transfer learning*. O retreino foi realizado usando uma base própria de cerca 11.000 imagens separadas em 32 categorias (e.g., TVs, lâmpadas, ventiladores). Foi construído um algoritmo para realizar o treinamento da *CNN*, no qual as imagens foram separadas em conjuntos de treino, teste e validação. Teste e validação correspondem a 10% das amostras cada, restando 80% para treino. A CNN resultante apresentou 83% de acurácia no conjunto de validação.

Ao utilizar a *CNN* criada, o resultado será o nome de uma categoria de SO. No entanto, realizar a busca apenas informando a categoria identificada pode ser infrutífera, pois os fabricantes não possuem um padrão para nomenclatura de seus SOs. Vários nomes podem estar em suas descrições no serviço de descoberta e divergir do nome da categoria proposta pela CNN. Devido a esta falta de padrão, foi necessário construir e adicionar a aplicação móvel um módulo responsável pela expansão de termos por meio de um dicionário léxico. Este módulo recebe como entrada a classe identificada (e.g., TV) e devolve para a aplicação uma *string* que representa uma *regex* relacionada à categoria do SO. Uma biblioteca para acesso à base de dados WordNet ® foi usada [1]. É com a expansão dos termos que o padrão de busca dos SOs de interesse do usuário é construído. Assim, se o usuário aponta para uma TV, a categoria "TV" irá emergir da CNN e, em seguida, uma *string* contendo "*television*", "*telecasting*" e "*television receiver*" e alguns outros sinônimos será usada para montar a consulta a ser enviada ao openHAB.

Este processo *View* do ViSCO que engloba desde a classificação dos SOs até a geração da consulta expandida é **independente** do openHAB. Entretanto, para conseguirmos apresentar uma aplicação funcional estes componentes de software foram adicionados à aplicação móvel da plataforma openHAB. Para tal, foi criada uma nova tela que consiste em uma visualização da câmera. A imagem exibida nesta tela é classificada de forma constante e a *regex* é gerada para uso no serviço de busca. A classificação usa a biblioteca *TensorFlow Lite*.

2.2 Scan and Control - via openHAB

Para realizar o **Scan**, o ViSCo demandou duas novas funcionalidades do **serviço de descoberta** do openHAB. A primeira verifica se existem SOs do tipo identificado na análise de imagem do ViSCo, tendo como base uma categoria de objetos. A segunda funcionalidade é a melhoria da resposta que o serviço REST de busca deve retornar. Quando o usuário realizar a busca de um SO, esta resposta deve conter apenas itens identificados na análise de imagens, ou seja, o serviço REST que retorna as configurações dos menus e SO, deve receber um novo parâmetro que represente os objetos que devem ser retornados para o usuário. Para atender a primeira funcionalidade, um novo *endpoint* foi criado no componente REST do serviço de descoberta do openHAB. Este serviço recebe como parâmetro uma *string* contendo a categoria desejada e seus respectivos sinônimos. Ao receber esta *regex*, o serviço vai realizar uma busca em profundidade na estrutura dos menus de SOs buscando itens que possuam uma descrição que combine com a *regex* informada.

[1]Base léxica contendo sinônimos de uma palavra em inglês [12]

Para realizar o **Control**, a aplicação openHAB *mobile* usa comunicação via REST com o openHAB *core*. Um dos resultados do *Scan* é informar para o usuário se existe ou não dispositivos da categoria identificada na rede. Caso exista, é apresentado um botão para que seja confirmado a ação de filtrar usando o serviço de descoberta descrito a cima. Os dados retornados da descoberta são usados para montar uma estrutura de menus e atuadores, em tempo de execução, para que o usuário possa ver a lista dos SOs daquele tipo (e.g., TVs da rede).

3 Avaliação com usuários finais

Foi realizado um teste em ambiente simulado para avaliar a usabilidade do ViSCo, apresentando duas formas de interação com SOs. A primeira usava a interação do tipo **Escanear** e listava os SOs presentes na rede do usuário. O participante deveria identificar objetos de interesse dentro da lista, seja pelo nome ou pelo ícone do dispositivo. Já a segunda forma usava a interação **Apontar** fornecida pelo ViSCo. A avaliação tinha os seguintes objetivos: (1) mensurar o nível de aceitação da abordagem ViSCo em diferentes grupos de usuários; (2) identificar a preferência dos participantes entre a abordagem ViSCo e a abordagem de listagem simples; (3) verificar o nível de acerto da CNN construída quando usada de forma a classificar um *stream* de imagens ao invés de imagens estáticas como realizado antes; e (4) coletar opiniões dos participantes sobre a solução e identificar possíveis melhorias e problemas de usabilidade.

Alguns *Surveys* foram aplicados durante a pesquisa. O primeiro *Survey* foi desenhado para coletar a opinião pré-teste sobre a percepção do participante em relação à interação do tipo **Apontar**, antes do mesmo ter a experiência de usá-la. O segundo *Survey*, ao final da avaliação, teve como objetivo avaliar de forma quantitativa e qualitativa a abordagem ViSCo. Usou-se o SUS (do inglês, *System Usability Scale*) [5] Além das assertivas do SUS, algumas das questões objetivas realizadas no *survey* de opinião pré-teste são reavaliadas pelos participantes.

3.1 Perfil dos Participantes

A avaliação contou com **17** participantes. Cinco mulheres e doze homens com idade entre 21 e 49 anos. 11 dos 17 participantes pertenciam ao **Perfil 1**, i.e., trabalhava e/ou estudava TI e teve contato com *Smart Homes*. Os outros 6 se dividiam igualmente nos outros três perfis, a saber: **Perfil 2**: Não trabalhava nem estudava TI e teve contato com *Smart Home*; **Perfil 3**: Trabalhava e/ou estudava TI e não teve contato com *Smart Home*; e **Perfil 4**: Não trabalhava nem estudava TI e não teve contato com *Smart Home*. Ressalta-se que com cinco usuários já se pode identificar 85% dos problemas de usabilidade de um sistema. A maior parte dos problemas críticos tendem a ser descobertos com 15 participantes [6], portanto o quantitativo no presente estudo foi considerado adequado para seus objetivos. Os dados foram coletados em um survey que foi divulgado ao público geral com o intuito de identificar o perfil dos participantes, convida-los ao testes e realizar o agendamento do teste simulado.

3.2 Procedimento

De inicio os participantes foram contextualizados sobre o cenário do teste e seus objetivos. Lhes era informado que seriam usadas duas formas de localização de SOs, a do ViSCo e uma de listagem simples. Após a explicação, os participantes foram solicitados a responder ao primeiro *survey*. Em seguida, a instalação do aplicativo. O primeiro cenário tinha uma lista pequena de SOs, apenas 10 objetos inteligentes. Era apresentado para o participante a lista com todos os dispositivos, bem como um agrupamento semântico, simulando uma configuração onde os objetos estariam agrupados por cômodos da *Smart Home*

O participante escolhia uma ou mais categorias de SOs e tentava localizar os objetos na listagem para realizar uma interação com eles (e.g., ligar). Após realizar a localização e interagir com o SO, o participante devia tentar localizar novamente os SOs usando a abordagem ViSCo. Para isso, bastava que o participante clicasse no ícone da câmera na tela inicial e apontasse para o objeto de sua casa (semelhante a algum objeto da lista simulada) com a câmera do *smartphone*. Em seguida, uma pequena configuração no aplicativo era solicitada para se ter acesso ao segundo cenário do teste. Com isso, o participante tinha acesso a um segundo cenário simulado, onde a lista de SOs era bem mais extensa (i.e., 41 SOs simulados).

3.3 Resultados

A média do SUS foi 83,97 e o desvio padrão em 8,79, variando de 70 até 97,5. Nenhum dos participantes apresentou um score do SUS abaixo de 70. Isto aponta um bom nível de aceitação da abordagem ViSCo. De acordo com o estudo apresentado por [1], um software pode ser considerado como "bom" a partir do score 71,4. Apenas um participante não considerou o seu uso como "bom". Suas respostas indicaram um score muito próximo deste valor (i.e., 70).

As duas primeiras perguntas elaboradas para identificar a mudança de opinião sobre a interação **Apontar** foram relacionadas à utilidade do uso de smartphone e do uso da câmera (ViSCo) para interagir com os dispositivos. Comparando as respostas pré e pós teste pode-se notou-se uma tendência maior ao uso de *smartphone* e também ao uso do ViSCo. Em outra pergunta com o intuito de comparar as abordagens apresentadas (encontrar um SO em uma listagem simples e encontrar um SO utilizando o ViSCo) também apresentou uma tendência de maior aceitação.

Para a intenção de uso com maior frequência do ViSCo, em comparação com a abordagem de listagem simples, também apresentou uma aceitação maior no pós-teste do que no pré-teste. Uma das hipóteses aqui é que mesmo o usuário

gostando e/ou achando que fosse mais simples de utilizá-la, ele poderia escolher não usar a abordagem ViSCo por qualquer outro motivo, como por exemplo, privacidade.

Foram realizados testes de significância estatística para avaliar as mudanças nas respostas dos participantes. Como os dados são emparelhados e não possuem uma distribuição normal, foi utilizado o teste Wilcoxon [11], o qual lida com a mudança na medida posicional das amostras e se baseia postos das diferenças intrapares. Foi utilizada uma confiança de 95%, portanto a hipótese nula de que a aceitação não cresce é rejeitada quando o valor-p fica abaixo de $0,05$. Os testes para **Q-3** [2] e **Q-7** [3] resultam em valores-p de $0,051$ e $0,24$, respectivamente, ou seja, a mediana não cresce, apesar da pequena margem. Isto indica que embora os valores de aceitação tenham aumentado entre o pré-teste e o pós-teste, não é possível afirmar que tal mudança é significativa. Já considerando as questões **Q-4** [4] e **Q-8** [5] a hipótese nula é rejeitada com valores-p de $0,015$ e $0,035$, respectivamente, o que evidencia de forma significativa o crescimento na aceitação dos usuários com relação à utilidade e frequência no uso da câmera para localizar SOs.

4 considerações finais

A dissertação foi apresentada, e aprovada em maio de 2021, como requisito para conclusão do curso de mestrado em ciências da computação pela Universidade Federal do Ceará (UFC). O trabalho foi supervisionado pelos professores Dr. Windson Viana de Carvalho e Dr. José Gilvan Rodrigues Maia. Esta pesquisa contou com um artigo apresentado no WebMedia de 2021 com o titulo *Point and Control it! Using Computer Vision for Service Discovery to Control Smart Objects* obteve o prêmio de melhor artigo do evento [2] [10].

O objetivo central desta pesquisa foi projetar e avaliar a usabilidade de uma abordagem de interação com SOs do tipo *Apontar* [8], considerando requisitos da literatura e *insights* coletados por meio de *surveys* com 76 pessoas. A avaliação remota da abordagem foi realizada com 17 participantes e resultou em boa aceitação e poucos problemas de usabilidade. Muito embora aspectos negativos tenham sido apontados, a proposta de interação ViSCo foi bem aceita pela maioria dos participantes. Além disso, a intenção de uso da interação *Apontar* e do próprio uso do *smartphone* para controlar objetos aumentou após os participantes experienciarem o ViSCo. Também foram mencionadas vantagens não elicitadas antes pelos pesquisadores, tais como acessibilidade (e.g., para pessoas idosas).

Dentre as limitações identificadas neste estudo, destacam-se: (1) caso existam vários objetos na imagem, apenas uma categoria irá ser apresentada, o que poderia ser remediado por uma interface de toque na qual o usuário selecionaria o objeto de interesse; (2) o teste de avaliação foi feito de forma remota no qual as fases *View* e *Scan* eram reais mas a fase de controle era simulada. O teste completo possivelmente seria mais satisfatório; e (3) a versão atual do ViSCO se limita às plataformas openHAB e Android, muito embora a abordagem de busca e a CNN construída possam ser portadas para outras plataformas.

[2] **Q-3** "Como você julgaria a utilidade do uso do smartphone (e.g.: listagem do dispositivos, ou comandos de voz) para localizar e controlar objetos da sua casa?"

[3] **Q-7** "Avalie a sua concordância com a sentença: 'A busca de objetos utilizando a câmera é mais eficiente em comparação a busca de objetos utilizando listagem simples (uma lista de itens contendo todos os objetos disponíveis)'."

[4] **Q-4** "Como você julgaria a utilidade do uso da câmera do smartphone para localizar e controlar objetos da sua casa?";

[5] **Q-8** "Avalie a sua concordância com a sentença: 'Em um aplicativo de *Smart Home*, você usaria com maior frequência a localização do objeto usando a câmera do celular do que a busca por meio de listagem de objetos'.".

References

[1] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies* 4, 3 (2009), 114–123.

[2] Paulo Filipe Dantas, José Gilvan Rodrigues Maia, and Windson Viana. 2021. Point and Control It! Using Computer Vision for Service Discovery to Control Smart Objects *(WebMedia '21)*. Association for Computing Machinery, New York, NY, USA, 153–160. https://doi.org/10.1145/3470482.3479629

[3] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).

[4] Jung-Hwa Kim, Seung-June Choi, and Jin-Woo Jeong. 2019. Watch & Do: A smart iot interaction system with object detection and gaze estimation. *IEEE Transactions on Consumer Electronics* 65, 2 (2019), 195–204.

[5] James R Lewis. 2018. The system usability scale: past, present, and future. *International Journal of Human–Computer Interaction* 34, 7 (2018), 577–590.

[6] Jakob Nielsen. 2000. Why you only need to test with 5 users.

[7] Thomas P Novak and Donna L Hoffman. 2019. Relationship journeys in the internet of things: a new framework for understanding interactions between consumers and smart objects. *Journal of the Academy of Marketing Science* 47, 2 (2019), 216–237.

[8] Enrico Rukzio, Gregor Broll, Karin Leichtenstern, and Albrecht Schmidt. 2007. Mobile interaction with the real world: An evaluation and comparison of physical mobile interaction techniques. In *European Conference on Ambient Intelligence*. Springer, 1–18.

[9] Julian Seifert, Andreas Bayer, and Enrico Rukzio. 2013. PointerPhone: Using Mobile Phones for Direct Pointing Interactions with Remote Displays. In *Human-Computer Interaction – INTERACT 2013*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 18–35.

[10] Webmedia. 2021. Webmedia 2021 » Premiações. https://webmedia.org.br/2021/premiacoes/.

[11] Frank Wilcoxon. 1992. Individual comparisons by ranking methods. In *Breakthroughs in statistics*. Springer, 196–202.

[12] WordNet. 2020. WordNet | A Lexical Database for English. https://wordnet.princeton.edu/. (Accessed on 03/19/2020).

[13] Robert Xiao, Gierad Laput, Yang Zhang, and Chris Harrison. 2017. Deus EM Machina: On-Touch Contextual Functionality for Smart IoT Appliances. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4000–4008. https://doi.org/10.1145/3025453.3025828