

---

CS442 Mobile Computing, Networking & Applications  
Essay #4  
**Deus EM Machina: On-Touch Contextual Functionality for  
Smart IoT Appliances**

20160042 Inyong Koo

---

We are living in a world of an ever-growing ecosystem of connected and computationally-enhanced appliances. We have to keep pace with the revolution of the "Internet of Things" (IoT). Hence, the number of researches on environmental sensing and human-computer interface (HCI) is rising rapidly. The motivation of this paper is to enrich UI controls for handling multiple IoT devices.

The paper presents a novel system that allows instant recognition via EM signature with tap-to-device interaction. Also, the authors contribute a new UI mechanism what they call *contextual charm*, that provides small widgets relevant to the touched appliance upon currently running smartphone application. I should first address my gratitude towards the authors for the demonstrative video. It successfully shows the results and the concept of the paper and helped me immensely with understanding.

There have been many approaches for recognizing appliances from a controller (usually a smartphone), and the authors used EM sensing. Results with high accuracy suggest that we can utilize this approach, but I don't think this is the best approach. The paper acknowledges its limitations, like disability for detecting powered off objects<sup>1</sup> and distinguishing multiple instances of a particular appliance. But my main concern does not come from how they implemented but what.

I think tap-to-device interaction is not very practical because it requires us to go to the appliance physically. If we cannot utilize machines remotely, then what's the point of IoT? I mean, why should we use an additional device to control an appliance when we can reach its buttons? Of course, controlling with another device can support better UI and extra functionalities, but still. I think it is rather ineffective. The authors compared their system with related works, arguing the strength that their approach does not require additional instruments like tags or sensors. However, they still have to augment an EM sensor to currently available smartphones for their methods too.

I rather liked the approaches of Mayer et al.<sup>2</sup> or Snap-to-It<sup>3</sup>. I didn't read the papers thoroughly, but I believe with the advance of computer vision technology will enable a novel system for object recognition in IoT soon. We do not need extra instruments, and we can still recognize objects remotely.

Nevertheless, the tap-to-device interaction method still has its worth. We won't be able to use a computer vision approach in a dark environment. Also, a visually impaired person would benefit more from this admirable work. If the authors appealed this point, then the paper would have been much more convincing.

Also, I think contextual charms are revolutionary. It is a remarkable idea, supporting appropriate functionalities contextually, communicating with the currently running application. There can be a whole bunch of new researches for designing contextual charms for various appliances. Most of all, I genuinely agree with the necessity of more open APIs.

In summary, I may have different Ideas with the approach the paper has taken, but I share its vision for better human-computer interaction. I admire the contribution and hope to see follow-up researches.

---

<sup>1</sup>In this sense, I'd like to highly appreciate the approach from next paper, T. Gong's "Knocker: Vibroacoustic-based Object Recognition with Smartphones"(P10) for being able to detect even non-electronic objects

<sup>2</sup>S. Mayer, M. Schalch, M. George, and G. Sörös. "Device recognition for intuitive interaction with the web of things," in Proceedings of the ACM conference on Pervasive and ubiquitous computing adjunct publication (UbiComp '13 Adjunct), 2013, 239-242.

<sup>3</sup>Adrian A. de Freitas et al. "Snap-To-It: A User-Inspired Platform for Opportunistic Device Interactions," in Proceedings of the SIGCHI conference on Human Factors in Computing Systems (CHI'16), 2016, 5909-5920.