Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2023, gehouden op donderdag 30 maart 2023 jl. en georganiseerd door NHL Stenden Hogeschool). Bij elkaar bijna 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op donderdag 27 maart 2025 in Zwolle en wordt dan georganiseerd door Hogeschool Windesheim. Kijk op www.nioc2025.nl voor meer informatie.

Wil je op de hoogte blijven van de ontwikkeling rond Stichting NIOC en de NIOC kennisbank, schrijf je dan in op de nieuwsbrief via

www.nioc.nl/nioc-kennisbank/aanmelden nieuwsbrief

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Physical Computing in Computer Science Education: Creative Learning with Interactive Objects

Introduction

Computer science in the form of interactive and embedded computing systems dominate our lives in many areas: we take automatically operated trains, trust the car-wash not to crash our vehicle, have goals in the FIFA World Cup given by goal-line technology, lend books from the library without talking to anyone and let robots vacuum clean our houses. Pervasive and ubiquitous computing, as they were envisioned by Weiser (1993) more than twenty years ago, have become reality: *"Ubiquitous computing enhances computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user."* The same way that interactive computing systems gain more relevance in our society, the development of easily programmable micro-controllers offers a new way of creatively designing computing systems at school. Physical computing is a phenomenon that has become increasingly popular throughout the last years, mostly among the so-called maker community, but also in arts and design contexts.

Current situation

As a new form of creatively designing computing systems, physical computing also gains relevance in school contexts, particularly in computer science education. However, when looking into computer science education in Germany, especially in the areas of Berlin and Brandenburg, the trend has not yet come to schools. In a survey based on self-administered questionnaires conducted with 115 students from different age groups and school types, the participants' associations with computer science in every day life (free response tasks) and their perception of computer science classes (four-point Likert items) was investigated with respect to constructionist learning and creativity (Przybylla & Romeike, 2014b). There, it was found that embedded, physical or interactive computing systems are not in students' focus, none of the students had encountered physical computing activities in computer science lessons, only few students learned in creativity-rich lessons or constructionist learning environments and students' interests in physical computing activities vary a lot depending on e.g. gender.

Physical computing as a means to enriched computer science classes

A possible way to overcome those issues is to integrate physical computing activities in computer science education. Physical computing is the development of interactive objects or installations that communicate with their environment through sensors and actuators. This way, physical computing blends the virtual and the physical world and makes artifacts of learning visible, tangible and shareable. It takes the focus from pure software development and includes aspects of hardware design. Physical computing further promotes learning in computer science in a creative and practical fashion and perfectly matches with the ideas of constructionist learning, which has the creation of personally relevant artifacts in its core (Papert & Harel, 1991, p.1).

Three pillars of physical computing

With the aim of understanding physical computing in the context of computer science education, relevant literature and course descriptions were analyzed. Those analyses and own experiences with physical computing in classroom settings have shown that there are three "pillars" that practitioners need to take care of when designing classroom activities (Przybylla & Romeike, 2014a). Those aspects are interwoven and influential on each other:

What are the (conceptual) characteristics of the resulting products?

Typical products of physical computing are interactive jewelry and clothes, intelligent toy pets, interactive mood lamps, room filling installation arts or useful everyday-devices. Technically, all those objects have in common that they make use of transducers, are not transformational, run continuously, interact steadily with their environment and can form networks of interactive installations.

How is the creative process to be organized?

In physical computing, there is a clear focus on the ideas underlying projects. Creative learning thus also means that students might not even know which tools they use when brainstorming their project ideas – this ensures, that the focus does not shift to the limitation of available tools. In this creative process, learners quickly develop working prototypes of their interactive objects – something that is highly valuable in the context of constructionist learning, because this way they create meaningful artifacts of learning that they can present to and share with others.

What are the right tools for the intended purpose?

In order to make interactive objects with physical computing, suitable tools are needed. There is a large variety of tools available, ranging from programmable toys over bricks and microcontrollers to miniature computers. In order to decide for the right tool, teachers have to think about their project specific requirements. For example, with "My Interactive Garden", a constructionist learning environment for physical computing (designing, crafting, and programming interactive objects), the following requirements were to be met (cf. Przybylla & Romeike, 2012):

- *Simplicity:* Use preassembled sensors and actuators (plug and play)
- *Flexibility and Extensibility:* Sensors and Actuators can easily be added, removed, exchanged
- *Black/whiteboxing:* Only relevant components are visible, the black box can be opened
- *Emphasize computing principles:* Underlying computing principles are visualized (e.g. IPO model)

Based on those requirements, it was decided that a microcontroller and pre-assembled sensors and actuators would best fit the purpose. Another aspect of the choice of tools is on the software side: the possibilities seem unlimited, anything from Scratch-like drag-and-drop environments to advanced programming languages and editors can be used – of course depending on the hardware chosen.

Creative learning with My Interactive Garden

There are different approaches how physical computing can enrich computer science education in letting students learn with and about interactive computing systems. For instance, in our department's learning lab we have designed lesson plans for topics such as coding with bar codes, pixel and vector graphics, Nassi–Shneiderman diagrams or introduction to programming. "My Interactive Garden" (Przybylla & Romeike, 2013) is an award-winning physical computing teaching concept that was tested and evaluated in different contexts (afternoon club, university courses, computer science lessons in school, teacher training workshops) and fosters the latter topic. MyIG and the didactical concepts that underlie this idea, give concrete examples for implementing physical computing in computer science lessons of lower secondary education. It is accompanied by a website with detailed descriptions and all the materials ready for download: http://tangible-cs.de (currently in German only). The main idea of MyIG is to provide learners with a context in which they create concrete tangible products of the real world that arise from their own imagination. The aim of a MyIG project is to collaboratively work on an exhibition of a futuristic interactive garden. This way, learners are encouraged to not just copy or rebuild systems, which they already know, but to use their imagination and creativity in order to develop personally relevant interactive objects that can be used in interactive installations.

Experiences

Observations and evaluations of MyIG show that learning experiences are described as self-determined, creative and collaborative. Students emphasized that they liked working together on a larger project while at the same time creating their own interactive objects. Based on their interests, they developed concrete tangible products in constructionist learning environments. They transferred the concepts learned to everyday objects, e.g. parking sensors. This way, abstract concepts became concrete and practical, e.g. analog and digital data, sensing and acting technologies, data and information, programming concepts and control structures.

Conclusion

Summarizing, it can be concluded that physical computing triggers creativity, broadens students' conceptions of computer science, makes them aware of embedded and interactive systems in their environment and gives them the opportunity to create something meaningful that they then can take home and use in their private environment. With the Internet of things, cyber physical systems or smart objects as technologies of our present and future, new contents become relevant for computer science education - contents that can be learnt with physical computing.

References

Papert, S., & Harel, I. (1991). Situating Constructionism. In S. Papert & I. Harel (Eds.), *Constructionism*. Norwood: Ablex Publishing Corporation.

Przybylla, M., & Romeike, R. (2012). My Interactive Garden – A Constructionist Approach to Creative Learning with Interactive Installations in Computing Education. In C. Kynigos, J. E. Clayson, & N.

Yiannoutsou (Eds.), *Constructionism: Theory, Practice and Impact. Proceedings of Constructionism 2012* (pp. 395–404).

Przybylla, M., & Romeike, R. (2013). *Physical Computing mit „My Interactive Garden"*. (N. Breier, P. Stechert, & T. Wilke, Eds.)*15. GI-Fachtagung „Informatik und Schule" - Praxisband*. Department of Computer Science, CAU Kiel.

Przybylla, M., & Romeike, R. (2014a). Key Competences with Physical Computing. In *Proceedings of Key Competencies in Informatics and ICT 2014*. Potsdam: Universitätsverlag Potsdam.

Przybylla, M., & Romeike, R. (2014b). Overcoming Issues with Students ' Perceptions of Informatics in Everyday Life and Education with Physical Computing - Suggestions for the Enrichment of Computer Science Classes. In *ISSEP 2014 - 7th Conference on Informatics in Schools: Situation, Evolution and Perspectives*.

Weiser, M. (1993). Some Computer Science Issues in Ubiquitous Computing. *Communications of the ACM*, *XXXVI*(7), 75–84.