

LED 기반 텐지블 프로그래밍 도구 개발 및 적용

심재권[†] · 이원규^{††} · 권대용^{†††}

요 약

텐지블 프로그래밍 도구는 그래픽 인터페이스 기반의 프로그래밍 방식과는 달리 컴퓨터를 사용하지 않고 손으로 조작하여 프로그래밍 할 수 있는 도구이다. 어린학생을 대상으로 알고리즘을 설계하고 시뮬레이션 하는 경험을 제공하여 논리적 사고 능력을 향상시킬 수 있다. 본 연구에서는 기존에 개발된 텐지블 프로그래밍 도구를 분석하여 도출된 문제점을 수정·보완하기 위해 다수의 명령어를 하나의 블록에 맵핑, 즉시적인 결과 확인, 다양한 수준의 알고리즘 표현이 가능하도록 설계 후 B-Bricks를 개발하였다. 개발한 B-Bricks를 초등학교 4-6학년 29명을 대상으로 적용한 결과, 문항의 난이도가 상승할수록 정답률이 감소하고 문제풀이시간이 증가하는 것으로 분석되었다.

주제어 : 텐지블 프로그래밍 도구, 텐지블 인터페이스, 프로그래밍 교육

Development and Application of Tangible Programming Tool based on Dotmatrix LED

JaeKwoun Shim[†] · WonGyu Lee^{††} · DaiYoung Kwon^{†††}

ABSTRACT

Tangible programming tool based on a graphic interface platform is unlike programming on a computer due to the fact that young students can utilize their hands and tools without an on-line interface. This is effective in enhancing student understanding about difficult programming algorithms. B-Bricks was constructed to consolidate multiple commands onto one block, having immediate feedback about the result, and other algorithms based on assessing the problems that arose from the previous tangible block model. Upon experimenting on 29 students of grades 4th~6th, it was observed that as problems became more difficult, the accuracy of the answers derived diminished and their problem-solving time increased.

Keywords : Tangible Programming Tool, Tangible Interface, Programming Education

† 정 회 원: 고려대학교 컴퓨터교육학과 박사수료
 †† 종신회원: 고려대학교 컴퓨터학과 교수
 ††† 종신회원: 고려대학교 컴퓨터교육과 연구교수(교신저자)
 논문접수: 2014년 10월 17일, 심사완료: 2014년 12월 10일, 게재확정: 2014년 12월 31일
 * 본 논문은 2013년 한국연구재단의 지원을 받아 연구되었음(NRF-2013R1A1A2061352)

1. 서론

텐지블 인터페이스는 사용자가 신체를 사용하여 컴퓨팅 환경과 상호작용하는 인터페이스이다 [1]. 텐지블 인터페이스는 키보드 마우스와 같이 보편적인 입력 도구와 달리 사용하고자 하는 컴퓨팅 환경과 사용자의 목적에 최적화 되도록 상호작용 중심의 직관적인 인터페이스를 제공한다 [2]. 따라서 텐지블 인터페이스는 컴퓨팅 환경이 생활과 밀접하게 연계되는 고도화된 정보사회에 효과적인 인터페이스이다.

텐지블 인터페이스는 예술, 의료, 교육, 게임 등 다양한 분야에서 활용에 대한 연구가 진행되고 있다. 예술분야에서는 사용자가 텐지블 인터페이스에 기반한 블록들을 조합하여 다양한 음악과 사운드를 쉽게 만들 수 있는 환경을 개발하거나 [3] 복잡한 동영상 편집 프로그램을 사용하기 어려운 사람을 대상으로 블록에 동영상 클립을 저장하고, 블록들의 조합과 편집블록들의 간단한 조작을 통해 동영상을 편집하고 제작할 수 있는 환경이 개발되고 있다[4]. 의료분야에서는 주의력결핍 과잉행동장애(ADHD)의 치료를 목적으로 신체의 사용에 기반한 주의력 향상 게임을 개발하거나[5] 노인을 대상으로 기억력을 향상시키고 근력을 강화시키하고자 하는 목적으로 다양한 게임을 제작하여 활용하고 있다. 교육에서는 학생의 이해를 돕기 위해 추상적인 분자구조를 시각화하거나 직접 조작하여 화합물을 조합하는 시뮬레이션을 개발하는 등 다양한 분야에서 활용되고 있다[6].

텐지블 프로그래밍 환경은 텐지블 인터페이스를 기반의 도구를 조작하여 프로그래밍 활동을 수행할 수 있는 환경을 의미한다. 신체를 사용하여 프로그래밍 활동을 할 수 있는 환경은 키보드 마우스를 사용하는 일반적인 컴퓨터 환경과 비교하여 직관적인 조작과 물리적인 환경에서 결과를 확인할 수 있다는 장점 때문에 높은 교육적 효과를 보여 주고 있다[7]. 특히 텐지블 프로그래밍 환경은 도구 사용에 있어서 인지적 부담을 많이 느끼는 초보 학습자에게 큰 효과가 있는 것으로 나타났다[8]. 또한 교육적인 효과를 증대시키기 위해 초등학생의 흥미와 관심을 고려하여 로봇과 함께 연계되어 제공되고 있다. 교육용 로봇의 활

용은 학생이 프로그래밍한 결과를 실세계에서 로봇의 움직임을 통해 확인할 수 있고, 학생의 창의적인 아이디어나 설계를 현실에서 구현할 수 있어 프로그래밍 학습의 동기를 유발과 더불어 유의미한 경험을 제공받을 수 있다는 장점이 있어 초등학생을 대상으로 널리 사용되고 있다[9].

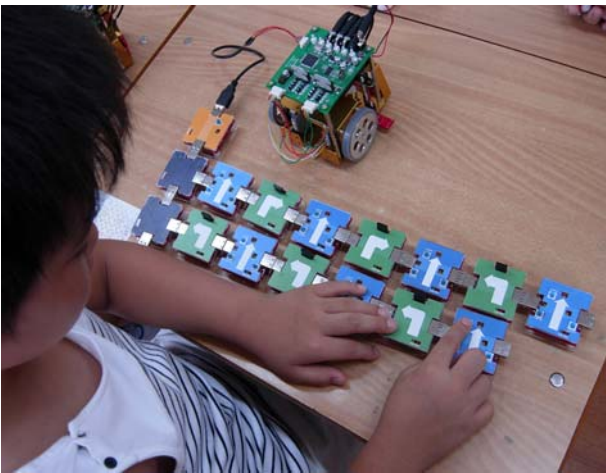
하지만 텐지블 프로그래밍 도구를 활용한 로봇 프로그래밍 활동은 프로그래밍 한 결과를 로봇으로 확인하는 시간이 많이 소요되거나 다운로드하는 절차가 복잡하고 오류가 발생하는 등의 단점이 있다. 또한 바퀴가 두 개로 이동하는 형태인 로봇의 전진, 회전하는 등 단순한 수준 움직임을 제어하는 명령어만 제공되어 난이도가 있는 문제를 풀이하기에는 부족한 점이 많다. 하나의 블록에는 하나의 명령어가 일대일로 대응되도록 설계되어, 복잡한 문제를 해결할 때에는 수많은 블록이 필요한 단점이 있다. 즉, 프로그래밍 도구의 조작적인 측면의 난이도가 감소하기는 하였지만 알고리즘의 난이도도 함께 낮아져 텐지블 프로그래밍 도구의 장점인 추상화된 개념을 쉬운 조작을 통해 학습하는 것이 아니라 단순히 개념을 그대로 텐지블 도구에 적용하여 사용하고 있는 상황이다.

본 연구에서는 텐지블 프로그래밍 환경을 개선하기 위해 하나의 텐지블 블록에 다수의 명령어를 표현하는 방안과 프로그래밍한 결과를 즉시적으로 시각화하고, 알고리즘의 난이도를 조절할 수 있는 방안을 제안하고 개발하여 초등학생에 적용하고자 한다. 연구에서 제안하는 내용은 첫째, 도트 매트릭스(Dotmatrix) LED를 텐지블 프로그래밍 도구에 적용하여 다수의 명령어를 표현하고, 표현된 명령어들 중에서 하나의 명령어만을 선택하여 사용할 수 있도록 설계하였다. 둘째, 프로그래밍한 결과를 즉시적으로 확인하기 위해 LED의 on/off를 개별로 제어하여 도트 매트릭스에 그림을 그리는 콘텐츠를 개발하였다. 마지막으로 알고리즘의 난이도를 조절하기 위해 반복과 중첩반복을 사용하는 문제를 개발하였다. 개발한 도구의 검증을 위해 초등학생 30명에게 적용하여 학생이 풀이하는 문제의 정답률과 사용성을 통해 개발한 도구에 대한 평가를 하였다.

2. 관련연구

2.1 Algorithmic Bricks (A-Bricks)

Algorithmic Bricks는 초등학생이 한 손으로 잡을 수 있는 작은 블록을 좌우로 연결하거나 쌓는 조작을 통해 로봇을 제어하는 텐지블 프로그래밍 도구로 순차, 반복, 분기, 함수, 매개변수의 개념을 구체적인 조작을 통해 프로그래밍 할 수 있도록 개발 되었다[8].



<그림 1> A-Bricks

개발된 A-Bricks를 활용하여 프로그래밍 할 수 있는 논리적 수준에 따른 로봇 과제를 개발하여 초등학교 1학년에게 적용한 결과, 프로그래밍 후에 논리적 사고력의 수준이 향상된 것으로 나타났다[10]. 또한 사용성이 4점 이상으로 나타났고, 특히 즐거움에 대한 항목이 4.5점 이상으로 나타나 A-Bricks의 활용을 재미있어 하는 것으로 나타났다.

2.2 CHERP

CHERP(Creative Hybrid Environment for Robotic Programming)은 Tufts 대학에서 개발한 텐지블/그래픽 인터페이스 기반의 프로그래밍 도구로 LEGO Mindstorms RCX 와 Lego WeDo 로봇을 제어할 수 있다[11].



<그림 2> CHERP

CHERP 블록은 사용하는 대상인 유치원과 초등학교 저학년 수준을 고려하여 나무로 제작되었고, 로봇의 전진, 후진, 좌회전, 우회전의 간단한 동작을 제어하는 명령어를 바코드로 걸 표면에 표시하였다. 학생은 주어진 로봇 문제를 해결하기 위해 블록을 순차적으로 조립하여 프로그래밍 하고, 바코드 리더기를 통해 프로그래밍 한 결과를 로봇에 전송하여 결과를 확인하는 과정을 통해 정보과학적 사고(Computational Thinking)가 향상 되는 것으로 나타났다[7].



<그림 3> 바코드리더기로 로봇에 다운로드하는 모습

기존에 개발된 텐지블 프로그래밍 도구를 분석하면, 첫째 텐지블 블록 하나에 한 가지의 명령어만 대응되도록 되어 있어 다양한 프로그래밍 활동을 위해서는 다수의 블록들이 필요하다. 둘째, 프로그래밍한 결과를 로봇을 통해 확인하는 과정에서 많은 시간이 걸리고, 기계적인 에러가 발생한다. 셋째, 프로그래밍 활동에서 요구하는 알고리즘 수준이 순차적으로 명령어를 단순하게 조합하는 수준이다.

2. B-Bricks

2.1 설계

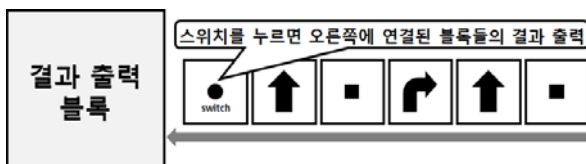
본 연구에서는 기존의 텐지블 프로그래밍 도구들의 분석을 통해 도출한 문제점을 해결하기 위해서 다음의 세 가지 요소를 고려하였다.

첫째, 하나의 블록에 다수의 명령어 사용할 수 있도록 블록에 다수의 명령어를 저장하고 조작을 통해 명령어를 변경하여 사용할 수 있도록 설계하였다.



<그림 4> 다수의 명령어를 하나의 블록에 맵핑

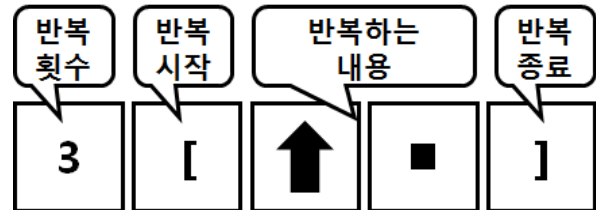
둘째, 프로그래밍 결과를 즉시적으로 확인하기 위해, 스위치를 누르면 프로그래밍한 결과를 이미지로 출력할 수 있도록 설계하였다.



<그림 5> 스위치를 누르면 즉시적으로 결과를 출력

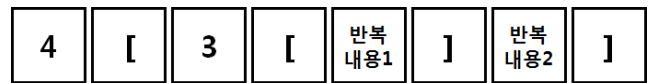
셋째, 단순한 알고리즘에서 복잡한 알고리즘까지 다양하게 표현하기 위해 순차, 반복, 중첩방법

이 가능하도록 설계하였다. 순차는 텐지블 블록을 연결한 순서대로 왼쪽에서 오른쪽으로 글을 쓰는 방향으로 실행되도록 설계하였다. 반복은 반복횟수, 반복시작, 반복내용, 반복종료의 순으로 구성하였다.



<그림 6> 반복 설계

반복을 여러 개 중첩하여 사용할 수 있도록 설계하였다. <그림 7>는 반복내용1을 3회 반복하고, 반복내용2를 1회 수행하는 것을 총 4회 반복하는 중첩반복 설계이다.



<그림 7> 중첩반복 설계

2.2 구현

텐지블 프로그래밍 블록의 구현은 이미지를 출력하는 8*8 Dotmatrix LED 기반으로 구현하였다. Dotmatrix LED는 자유로운 이미지 표현이 가능하여 직관적으로 이해할 수 있고, 학생이 현실에서 전광판이나 신호등 등을 통해 이미 쉽게 접하고 있어 보다 친숙하게 사용할 수 있다. 개발한 LED 기반 텐지블 프로그래밍 도구인 B-Bricks는 자석을 이용하여 어린학생도 쉽게 붙이고 분리할 수 있고, 스위치를 눌러 명령어를 변경하거나 전송할 수 있다.



B-Brick

오른편

왼편

<그림 8> B-Bricks

첫째, 하나의 블록에 다수의 명령어를 사용할 수 있도록 좌우, 상하 버튼이 있는 블록을 연결하여 명령어를 변경하고 저장하도록 구현하였다. 둘째, 프로그래밍한 결과를 즉시적으로 확인할 수 있도록 스위치를 누르면 오른쪽에 연결된 블록의 결과가 이미지로 출력될 수 있도록 구현하였다. 셋째, 다양한 알고리즘을 표현하기 위해 반복과 중첩반복이 가능하도록 구현하였다. <그림 9>의 모습은 왼쪽 상단에서 시작하여 한칸씩 전진하며 점찍기를 7회 반복하고, 이를 오른쪽으로 돌면서 4회 반복하여 사각형을 그리는 중첩반복 프로그래밍 예이다.

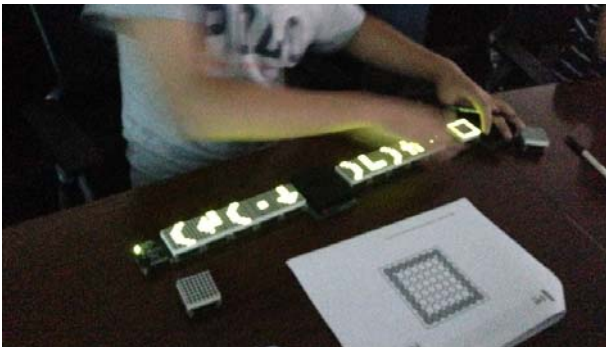


<그림 9> 중첩반복 결과 출력예시

3. 연구방법

3.1 연구대상

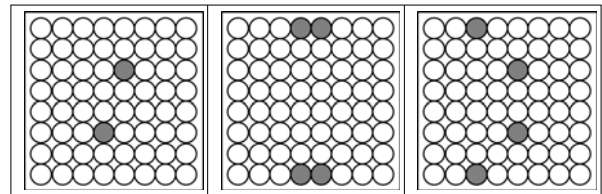
초등학교 4-6학년 29명(남학생:19, 여학생:10)을 대상으로 3시간 동안 수업을 하였다. 1시간 동안은 텐지블 프로그래밍 도구를 사용하는 방법과 예시문제를 풀이하였고, 나머지 2시간 동안 상중하 각 수준별로 3문제씩 40분 동안 풀이하였다. 문제풀이의 순서는 하 수준부터 상 수준으로 난이도가 올라가도록 풀이하였다.



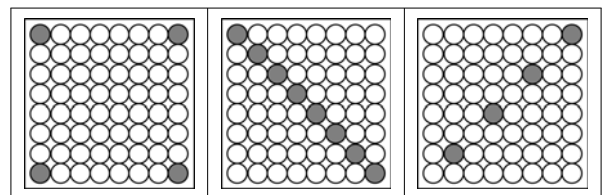
<그림 10> B-Bricks을 사용한 초등학교 4학년의 프로그래밍 활동 모습

3.2 프로그래밍 문제

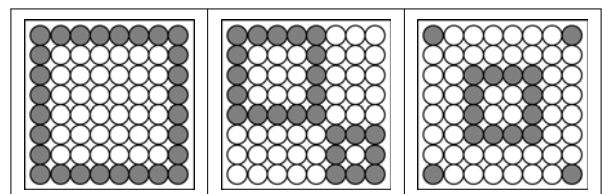
프로그래밍 문제는 결과블록에 문제로 제시하는 그림과 동일하게 LED를 출력하는 문제이다. 프로그래밍 학습의 단계를 고려하여 난이도를 3단계로 구분하였다. 난이도는 문제를 해결과정인 단순한 절차적, 순차적인 처리를 통해 해결하는 경우의 문제는 난이도 하, 문제를 분석하여 반복적인 패턴을 발견하고 조건을 설정하여 반복적인 처리를 통해 문제를 해결하는 형태로 구조화 하는 문제는 난이도 중, 반복을 중첩하여 문제를 해결하는 형태로 구조화 하는 문제는 난이도 상으로 설정하였다[12].



<그림 11> 난이도 하 수준의 프로그래밍 문제



<그림 12> 난이도 중 수준의 프로그래밍 문제



<그림 13> 난이도 상 수준의 프로그래밍 문제

3.3 사용성 평가도구

사용성 평가도구는 ISO/IEC 25010:2011 과 ISO 9241-11:1998 에서 Usability 항목 중 Understandability, Learnability, Operability, Attractiveness, Usability Compliance 을 참고하여, 초등학교 수준에 적합하도록 수정하였다[13]. 개발한 검사도구의 Cronbach α 계수는 .91으로 나타났다.

4. 연구결과

4.1 프로그래밍 결과

순차, 반복, 중첩반복에 따른 수준을 구분한 문제의 정답률과 풀이시간을 분석한 결과는 다음과 같다. 정답률은 명령어의 순차적인 연결을 통해 해결이 가능한 하 수준의 문제는 83.3%의 정답률을 보였고, 반복을 사용하여 문제를 해결하는 중 수준의 문제는 78.7%의 정답률을 나타냈었다. 중첩반복을 사용하는 상 수준의 문제는 37.0%의 정답률을 나타내어, 수준이 높아질수록 정답률은 하락하는 것으로 나타났다.

<표 1> 난이도에 따른 정답률

| 구분 | 하 수준 | 중 수준 | 상 수준 |
|--------|-------|-------|-------|
| 정답률(%) | 83.3% | 78.7% | 37.0% |

문제풀이 시간은 결과블록에 문제에서 제시하는 그림이 정확하게 나타난 경우에만 시간을 측정하였고, 풀이를 하지 못한 경우에는 제외하였다. 풀이시간은 반복을 사용하여 문제를 풀이하는 중 수준의 풀이시간이 평균 6분 정도로 가장 짧은 것으로 나타났고, 반복을 중첩하여 풀이하는 상 수준의 문제의 풀이시간이 평균 15분 정도로 가장 긴 것으로 나타났다.

<표 2> 난이도에 따른 문제풀이 시간

| 구분 | 하 수준 | 중 수준 | 상 수준 |
|---------|-------|-------|-------|
| 풀이시간(초) | 437.4 | 351.9 | 914.0 |

학생이 풀이하는 모습을 관찰한 결과, 하 수준의 문제가 반복구조를 이용하기 보다는 다수의

블록을 연결하여 풀이하도록 되어 있어 적은 수의 블록으로 문제를 풀이하는 중 수준의 문제들에 비해 블록을 조작하는 시간이 더 소요하는 것을 나타냈다. 상 수준의 경우는 도구의 조작에 시간을 사용하기보다 중첩반복의 논리적인 구조를 만들어내는데 더 많은 시간이 소요되는 것으로 관찰되었다.

4.2 수준별 문제풀이에 대한 사용성

LED 기반 텐지블 프로그래밍 블록에 대한 사용성 평가는 순차(하 수준), 반복(중 수준)의 문제를 풀이한 이후에 1차 평가를 하였고, 중첩반복을 사용한 이후에 2차 평가하였다. 1차 사용성 평가에 대한 결과는 다음과 같다. 문제를 해결함에 있어 반복블록을 사용하여 반복구조를 만들어 문제를 해결하는 방법이 효율적이라고 인식(4.79점)하고 있음을 알 수 있다.

<표 3> 하 수준과 중 수준 문제풀이에 대한 사용성 결과

| 번호 | 문항 | 사용성 |
|----|-------------------------------------|----------------|
| 1 | 반복블록(괄호 모양의 그림 (,))을 사용하는 방법은 편리하다 | 4.52 (0.69) |
| 2 | 문제해결과정을 반복블록을 사용하여 만드는 것은 쉽다 | 4.14 (0.79) |
| 3 | 반복블록은 문제해결과정을 효율적으로 표현하는데 도움이 된다. | 4.79 (0.41) |
| 4 | 반복블록을 사용하는 것은 흥미롭다. | 4.62 (0.68) |
| 5 | 방향블록(화살표 모양의 그림)을 사용하는 방법은 편리하다 | 4.55 (0.91) |
| 6 | 문제해결과정을 방향블록을 사용하여 만드는 것은 쉽다 | 4.62 (0.56) |
| 7 | 방향블록은 문제해결과정을 효율적으로 표현하는데 도움이 된다. | 4.66 (0.90) |
| 8 | 방향블록을 사용하는 것은 흥미롭다. | 4.69 (0.71) |

중첩반복을 사용하는 상 수준의 문제를 풀이한 이후 사용성에 대한 평가는 다음과 같다. 중첩반복을 사용하여 문제를 해결하는 과정에 대해 단 순반복(4.14점) 보다 어렵게 생각(3.71점)하는 것으로 나타났다. 방향블록(전진, 좌우회전)의 순차적인 사용은 어렵게 생각하지 않는 것으로 분석되었다.

<표 4> 상 수준 문제풀이에 대한 사용성 결과

| 번호 | 문항 | 사용성 |
|----|--|----------------|
| 1 | 중첩(이중) 반복블록(괄호 모양의 그림 (,))을 사용하는 방법은 편리하다 | 4.21 (0.88) |
| 2 | 문제해결과정을 중첩(이중) 반복블록을 사용하여 만드는 것은 쉽다 | 3.71 (0.90) |
| 3 | 중첩(이중) 반복블록은 문제해결과정을 효율적으로 표현하는데 도움이 된다. | 4.50 (0.79) |
| 4 | 중첩(이중) 반복블록을 사용하는 것은 흥미롭다. | 4.32 (0.90) |
| 5 | 방향블록(화살표 모양의 그림)을 사용하는 방법은 편리하다 | 4.48 (1.02) |
| 6 | 문제해결과정을 방향블록을 사용하여 만드는 것은 쉽다 | 4.57 (0.63) |
| 7 | 방향블록은 문제해결과정을 효율적으로 표현하는데 도움이 된다. | 4.31 (1.20) |
| 8 | 방향블록을 사용하는 것은 흥미롭다. | 4.75 (0.52) |

4.3 도구에 대한 사용성

문제풀이가 종료된 이후, 도구에 대한 사용성 평가를 진행한 결과는 다음과 같다.

<표 5> LED 기반 텐지블 프로그래밍 도구에 대한 사용성 결과

| 번호 | 문항 | 사용성 |
|----|-----------------------------|----------------|
| 1 | 블록의 사용방법이 편리하다. | 4.64 (0.59) |
| 2 | 블록은 내가 문제를 푸는데 사용하기 쉽다. | 4.44 (0.71) |
| 3 | 블록을 사용하는 방법이 쉽다. | 4.68 (0.56) |
| 4 | 블록의 사용방법을 다른 친구에게 설명할 수 있다. | 4.64 (0.57) |
| 5 | 블록의 사용방법을 기억하기 쉽다. | 3.93 (1.77) |
| 6 | 블록을 사용하는 것은 재미있다. | 4.80 (0.41) |
| 7 | 블록은 흥미롭고, 계속 사용해 보았으면 좋겠다. | 4.00 (1.85) |

텐지블 프로그래밍 블록에 대한 사용성은 대부분 4점 이상으로 높게 분석되었고, 블록을 사용하는 것에 대한 재미가 4.80점으로 가장 높게 나타났다. 텐지블 블록을 사용하는 방법에 대한 기억성이 3.93점으로 비교적 낮은 것으로 나타

나 면담을 통해 확인한 결과, 조작방법에 대한 구체적인 설명과 블록의 외형에 on/off 와 같은 추가적인 정보에 대한 표시가 필요함을 도출하였다.

<표 6> LED 기반 텐지블 프로그래밍 도구에 대한 사용성 결과

| 번호 | 문항 | 사용성 |
|----|---|----------------|
| 8 | 명령어를 조작하여 결과블록을 통해 바로 확인하는 사용방법은 편리하다. | 4.36 (0.99) |
| 9 | 명령어를 조작하여 결과블록을 통해 바로 확인하는 것은 쉽다. | 4.52 (0.92) |
| 10 | 명령어를 조작하여 결과블록을 통해 바로 확인하는 것은 문제풀이에 도움이 된다. | 4.52 (0.65) |
| 11 | 명령어를 조작하여 결과블록을 통해 바로 확인하는 것은 재미있다. | 4.60 (0.76) |

프로그래밍 결과를 즉시적으로 확인을 하는 것에 대한 사용성을 평가한 결과, 문제풀이에 도움이 된다고 응답한 경우가 4.52점으로 대부분이 “매우 만족한다(5점)”로 높게 응답하였다. 또한 즉시적인 확인이 재미있다고 응답한 경우가 가장 높은 4.60점으로 분석되었다. 즉, 개발한 B-Bricks를 사용하여 프로그래밍 결과를 즉시적으로 확인할 수 있는 프로그래밍 교육환경은 학생의 프로그래밍 학습에 긍정적임을 알 수 있다.

5. 결론

본 연구는 기존의 텐지블 프로그래밍 도구를 분석하여 보다 쉽고, 재미있게 프로그래밍 할 수 있는 B-Bricks를 설계하고 개발하였다. 기존의 텐지블 도구를 분석한 결과 첫째, 하나의 블록에 하나의 명령어가 일대 일로 대응하도록 되어 있어 프로그래밍 활동에 다수의 명령어 블록이 필요하거나 사용하는 블록의 수를 고려하여 제한적으로 문제를 출제해야 하는 한계가 있었다. 둘째, 프로그래밍 한 결과를 로봇에 다운로드하여 로봇의 움직임을 통해 결과를 확인하는 방법은 시간이 많이 소요된다는 단점이 있었다. 셋째, 프로그래밍 활동에서 요구하는 알고리즘의 수준이 단순하여 보다 복잡한 알고리즘 활동에 제약이 있었다. 이를 해결하기 위해서, B-Bricks는 첫째, 하나의 블록에 다수의 명령어를 입력하여 학생이 프로그

래밍시 필요한 명령어를 선택하여 사용할 수 있도록 하였다. 둘째, 결과를 기존의 로봇이 아닌 LED를 사용하여 즉시적으로 표현할 수 있도록 하였다. 셋째, 반복, 중첩반복을 사용하여 난이도를 조정할 수 있도록 개발하였다.

개발한 B-Bricks를 초등학생에게 적용한 결과, 도구의 사용성은 4점 이상으로 높게 분석되었고, 프로그래밍 난이도가 높아질수록 정답률은 낮아지고, 문제풀이 시간은 늘어나는 것으로 나타났다. 또한 즉시적으로 프로그래밍한 결과를 확인하는 것에 대한 사용성 평가결과, 문제풀이에 도움이 되고, 내적인 동기를 유지하는데 도움이 되는 것으로 분석되었다. 이를 토대로 본 연구에서 개발한 B-Bricks는 초등학생을 대상으로 알고리즘적 사고 향상을 위한 프로그래밍 활동이 가능하고, 학생의 인지수준에 적합함을 알 수 있다.

이후 연구로는 본 연구를 통해 초등학생의 요구분석을 통해 텐지블 도구를 초등학생에 보다 적합하게 할 필요가 있고, 다양한 콘텐츠의 개발과 더불어 중학생 수준에서 프로그래밍 활동을 경험할 수 있는 내용의 개발이 필요하다.

참 고 문 헌

- [1] Hiroshi Ishii, Brygg Ullmer (1997). Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. *Proceedings of Conference on Human Factors in Computing Systems(CHI '97)*, 1-8.
- [2] M. J. Kim, M. L. Maher (2008). The impact of tangible user interface on spatial cognition during collaborative design. *Design Studies*, 29, 222-253.
<http://dx.doi.org/10.1016/j.destud.2007.12.006>
- [3] Bert Schiettecatte, Jean Vanderdonckt (2008). AudioCubes: a Distributed Cube Tangible Interface based on Interaction Range for Sound Design. *Proceedings of the Tangible and Embedded Interaction (TEI'08)*, 3-10.
- [4] Jamie Zigelbaum, Michael S. Horn, Orit Shaer, Robert J.K. Jacob (2007). The Tangible Video Editor: Collaborative Video Editing with Active Tokens. *Proceedings of the Tangible and Embedded Interaction (TEI'07)*, pp 43-46.
- [5] M. D. Heller, K. Roots, S. Srivastava, J. Schumann, J. Srivastava, & T. Sigi Hale (2013). A Machine Learning-Based Analysis of Game Data for Attention Deficit Hyperactivity Disorder Assessment. *Games for Health Journal*, 2(5), 291-298.
- [6] Morten F., Benedikt M. Voegtli (2002). Augmented Chemistry: An Interactive Educational Workbench, *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR'02)*, 259-321.
<http://dx.doi.org/10.1109/ISMAR.2002.1115100>
- [7] M. U. Bers, L. Flannery, E. R. Kazakoff, A. Sullivan (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computer & Education* 72, 145-157.
<http://dx.doi.org/10.1016/j.compedu.2013.10.020>
- [8] DaiYoung Kwon, HanSung Kim, JaeKwoun Shim, & WonGyu Lee. (2012). Algorithmic Bricks: A Tangible Robot Programming Tool for Elementary School Students. *IEEE transactions on Educations*, 55(4), 474-479.
<http://dx.doi.org/10.1109/TE.2012.2190071>
- [9] M. Virnes, E. Sutinen, K. Eija (2008). How Children's Individual Needs Challenge the Design of Educational Robotics. *Proceedings of the Interaction design and children(IDC'08)*, 274-281.
<http://dx.doi.org/10.1145/1463689.1463766>
- [10] 권대용(2013). 텐지블 프로그래밍 도구를 활용한 논리적 사고력기반의 초등 로봇 과제 개발 및 적용. *컴퓨터교육학회논문지*, 16(4), 13-21.
- [11] Bers, M. & Horn, M. (2010). *Tangible programming in early childhood: Revisiting developmental assumptions through new technologies*. Greenwich, CT: Information Age Publishing.

[12] Theodosios S., Stavros D., Ioannis S. (2014). *Evaluating children performance with graphical and tangible robot programming tools*, Personal and Ubiquitous Computing, online. <http://dx.doi.org/10.1007/s00779-014-0774-3>

[13] ISO (2011). <http://www.iso.org/iso/home.htm> (revised 2014. 10. 15.)



심재권

2007 경인교육대학교
컴퓨터교육과(교육학학사)

2012 고려대학교
컴퓨터교육학과(이학석사)

2012~현재 고려대학교 컴퓨터교육학과 박사수료
관심분야: 컴퓨터교육, 프로그래밍 교육
E-Mail: jaekwoun.shim@inc.korea.ac.kr



이원규

1985 고려대학교 문과대학
영어영문학과(문학사)

1989 筑波大學 大學院
理工學研究科(공학석사)

1996~현재 고려대학교 컴퓨터학과 교수
관심분야: 컴퓨터교육, 정보검색, 데이터베이스
E-Mail: lee@inc.korea.ac.kr



권대용

2003 고려대학교 사범대학
컴퓨터교육과(이학사)

2006 고려대학교 대학원
컴퓨터교육학과(이학석사)

2011 고려대학교 대학원 컴퓨터교육학과(이학박사)
2011~현재 고려대학교 컴퓨터교육과 연구교수
관심분야: 컴퓨터교육, EPL, 교육용 로봇, 학습과학, 영재교육
E-Mail: daiyoung.kwon@inc.korea.ac.kr