SVG를 이용한 컨벌루션 연산의 시각화

김영미† · 강의성††

요 약

본 논문에서는 디지털 신호처리 및 영상처리 분야 등에서의 기본 연산이라고 할 수 있는 컨벌루션(convolution) 연산 과정을 웹을 위한 벡터 그래픽인 SVG(scalable vector graphics)를 이용하여 시각화하였다. 그런데, 컨벌루션 연산의 개념과 연산 과정은 주로 수학적인 표현에 의존하므로 그 개념을 직관적으로 이해하기는 다소 어려운 점이 있다. 본 논문에서는 SVG를 이용하여 컨벌루션의 연산 과정을 시각화하고 그에 대한 활용 예를 제시함으로써 학습자가 컨벌루션 연산을 보다 직관적으로 이해할 수 있도록 하였다. 본 논문에서 제시한 컨벌루션 연산의 시각화를 학습자들에게 적용한 결과를 면담을 통하여 분석한 결과, 학습자들이 컨벌루션 연산을 흥미를 가지고 학습에 임할 뿐만 아니라, 직관적으로 컨벌루션 연산을 이해할 수 있는 가능성을 볼 수 있었다.

키워드 : 컨벌루션, SVG, 시각화

Visualization of Convolution Operation Using Scalable Vector Graphics

Yeong-Mi Kim† · Eui-Sung Kang††

ABSTRACT

In this paper, visualization of convolution operation is presented, which is implemented by scalable vector graphics (SVG). Convolution operation is one of the basic essential concepts in the area of signal and image processing. However, it is difficult for students to intuitively understand the operation of convolution since it is mainly based on mathematical representation. We present the visualization of convolution operation and its applications which are implemented by SVG. The effects of the proposed approach have been analyzed by interviews. It has been seen that the proposed visualization of convolution operation could be effectively applied to learn the convolution operation and its applications.

Keywords : Convolution, SVG, Visualization

1. 서 론

디지털 영상처리는 컴퓨터를 이용하여 영상을 획득하여 영상을 해석하고 인식하는 등의 다양한 처리 기법과 관련된 분야를 의미한다. 이러한 영상처리 기법에는 다른 영상과 비교·분석하여 특징을 찾는 영상 인식(analysis), 영상의 정보는 그대로 유지하면서 데이터 양의 크기를 줄이기 위한 영상 압축(compression), 영상에 잡음이 섞여있는 경우 영상 내의 잡음을 제거하기 위한 영상 필터링(filtering) 등이 있다[1][2][3][4]. 컨벌루션은 이와 같은 대부분의 영상처리 기법에서 기

† 정 회 원: 동광양중학교 교사
†† 정 회 원: 순천대학교 컴퓨터교육과 교수(교신저자)
논문접수: 2007년 1월 9일, 심사완료: 2007년 1월 17일
* 본 논문은 2005년 순천대학교 학술연구비 공모과제로 연구되었음

본적으로 사용되는데, 영상을 의도적으로 뿌옇게 만드는 블러링(blurring), 영상의 세세한 부분을 더욱 두드러지게 하는 샤프닝(sharpening), 윤곽선 부분을 추출해 내는 에지(edge) 검출, 원하지 않는 신호를 제거하기 위한 잡음 제거 등과 직접적으로 관련이 있다.

중등학교 컴퓨터 그래픽스 교과인 '웹디자인' 교과의 필터 효과에서 위와 같은 블러링, 샤프닝, 에지 검출 등을 다루고 있으며[5], 대표적인 이미지 편집 프로그램에서 기본적인 기능으로 제공되고 있다. 컨벌루션에 대한 개념과 연산 과정에 대한 이해가 선행된다면 블러링, 샤프닝, 에지 검출 등, 이미지 편집 과정에서의 다양한 필터링 기능에 대한 근본적인 원리를 이해하여 보다 효과적으로 활용할 수 있다. 컨벌루션의 연산 과정은 고등학교 수학 수준 정도로 충분히 이해할 수 있는 연산이지만, 컨벌루션 이해를 위한 배경 지식이 다소 필요하고 수학적인 연산만으로 그 의미를 직관적으로 이해하기는 어렵다.

영상처리를 효율적으로 교육하고 연구를 하기 위한 비주얼 프로그래밍 환경[15]이나, 신호처리 및 영상처리를 교육하기 위한 라이브러리 기반 소프트웨어 환경[16]이 개발되어 교육에 이용되고 있다. 그런데, 이러한 시도들은 새로운 영상처리 응용을 개발하는 능력을 배양하거나 연구에 사용하기 위한 것으로서 컨벌루션 연산 과정과 같은 근본적인 개념을 다루지는 않는다.

본 논문에서는 컨벌루션의 개념과 연산 과정을 시각적이고 직관적으로 이해할 수 있도록 컨벌루션에 대한 시각화를 구현하였다. 또한, 블러링, 샤프닝, 에지 검출 등 컨벌루션이 이용되는 응용 예를 웹을 통하여 상호작용적으로 학습자가 이미지에 직접 적용해 보고 그에 대한 결과를 확인해 볼 수 있도록 구현하였다.

본 논문의 구성은 다음과 같다. 2장에서는 컨벌루션과 관련된 중등 교육과정과 컨벌루션 연산을 소개하고, SVG(scalable vector graphics)와 플래시(Flash)에 대해서 비교하였다. 3장에서는 컨벌루션 연산의 시각화 및 활용 예시의 구현 내용에 대해서 설명한다. 4장에서는 본 연구에서 구현한 내용을 학습자들에게 적용한 결과를 면담을 통하여 분석한 내용을 제시하고, 5장에 결론을 제시하였다.

2. 이론적 배경

2.1 중등과정에서의 컨벌루션 관련 내용

컨벌루션은 중등 교육과정에서 직접적으로 다루지는 않는다. 그러나, 그래픽이나 영상의 처리 과정이나 그로 인한 효과를 이해하는 배경 지식이 된다. 예를 들어, 컴퓨터 그래픽스 교과목 웹디자인 교과[5] 내용 중 '포토샵으로 이미지 제작하기' 단원을 위해서는 컨벌루션의 개념을 알면 여러 가지 필터 효과를 적용하거나 그 효과를 이해하는데 도움이 된다.

효과적인 이미지를 제작하기 위해서 레이어를 활용하거나 필터기능을 적용하여 다양한 효과를 적용한다. 포토샵과 같은 영상 편집 프로그램에서 필터는 이미지 전체나 선택 영역 내부에 있는 픽셀의 컬러를 변화시키거나 픽셀을 이동시킨다. 가령 샤픈 필터는 픽셀과 픽셀 간의 콘트라스트를 증가시켜 이미지를 날카롭게 보이게 하고, 블러 필터는 콘트라스트를 감소시켜 뿌옇게 만든다[6]. 이외에도 많은 여러 필터 효과들은 이미지를 보정하거나, 여러 가지 특수 효과를 얻는데 사용된다. 이와 같은 여러 가지 필터 효과들 중에는 컨벌루션 연산을 이용하여 구현되는 경우가 많이 있다.

2.2 컨벌루션 연산

입력 신호 x(n)이 임펄스 응답이 h(n)인 시스템에 입력될 때, 다음과 같은 식으로 나타나는 출력신호 y(n)을 x(n)과 h(n)의 컨벌루션이라고 한다.

$$y(n)= x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \qquad (1)$$

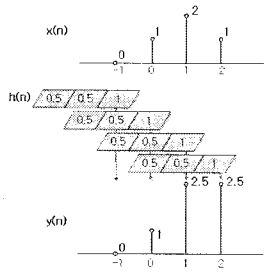
식 (1)을 풀어 쓰면 다음과 나타낼 수 있다.

$$y(n)=\cdots+x(-1)h(n+1)+x(0)h(n)+\cdots \qquad (2)$$

디지털 신호처리나 영상처리 응용에서 임펄스 응답 h(n)의 각 항 $h(-\infty),\cdots,h(\infty)$들의 대부분은 0이며, 몇 개의 항만이 0이 아닌 값을 갖는데, 이러한 h(n)을 흔히, 컨벌루션 마스크(mask), 커널

(kernel), 등으로 불린다[3]. 컨벌루션 마스크가 가지는 값에 따라 블러링, 샤프닝, 에지 검출 등 다양한 영상처리 효과를 가져 올 수 있다.

<그림 1>은 h(n)={1, 0.5, 0.5}일 때, 즉, n이 0, 1, 2일 때만 h(n) 값이 존재하고 나머지는 0인 경우의 컨벌루션 연산 과정을 보여 준다. 컨벌루션 연산을 위해서는 컨벌루션 마스크를 h(-k)와 같이 세로축에 대해서 반전한 후, x(k)와 겹치는 것끼리 곱한 후 더한다. 이러한 과정을 매 샘플마다 수행하면 출력 y(n)을 얻을 수 있다[7].



<그림 1> 1차원 신호의 컨벌루션 연산

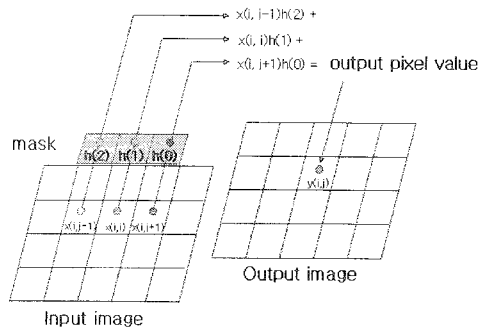
영상처리에서의 입력 신호는 x(n1, n2)와 같이 2차원 신호이다. 따라서, 영상 처리를 위한 컨벌루션은 식 (1)을 다음과 같이 2차원으로 확장한 것이다.

$$y(n1,n2)=\sum_{k1=-\infty}^{\infty}\sum_{k2=-\infty}^{\infty}x(k1,k2)h(n1-k1,n2-k2) \quad (3)$$

2차원 신호를 위한 컨벌루션 마스크는 응용 분야에 따라서 1*3과 같은 벡터 형태인 경우도 있고, 3*3과 같은 행렬 형태의 경우도 있다.

<그림 2>는 컨벌루션을 위해서 1*3 컨벌루션 마스크를 적용한 예로, (i, j)에서의 출력값 y(i, j)는 입력 영상의 (i, j)와 그 이웃에 위치한 픽셀값과 각 픽셀과 겹치는 컨벌루션 마스크값을 각각 곱한 후, 이를 더함으로써 얻을 수 있다[1].

입력 영상의 모든 (i, j)에 대해서 래스터 방향으로 컨벌루션 마스크를 이동하면서 위와 동일한 과정을 수행하면, 2차원 컨벌루션 연산의 결과인 y(n1, n2)를 얻을 수 있다[7].



<그림 2> 2차원 신호의 컨벌루션

2.3 SVG와 플래시의 비교

플래시는 애니메이션 기능, 뛰어난 상호작용성 등으로 인해 교육용 컨텐츠를 위한 저작도구로서 널리 쓰이고 있는데, 게임 형식의 교육용 컨텐츠, 집합 영역이나 도형의 닮음을 비교, 이해하기 위한 코스웨어[8], 체육교과의 세단뛰기 학습용 컨텐츠[9], 시뮬레이션 형태의 마이크로프로세서 실습용 컨텐츠 등 여러 교과를 위한 학습용 컨텐츠[10]를 제작하는데 사용되고 있다.

플래시가 지니고 있는 여러 가지 장점에도 불구하고, 플래시를 활용한 교육용 컨텐츠 개발에는 다음과 같은 단점을 가지고 있다.

첫째, 플래시는 상용 프로그램으로서 컨텐츠를 제작하기 위해서는 교사들을 비롯한 컨텐츠 제작자들이 고가의 프로그램을 구입해야 한다. 최근의 상용프로그램들은 빈번하게 업그레이드되므로 인해서 소프트웨어 유지 비용이 많이 든다.

둘째, 플래시를 이용하여 제작된 컨텐츠들은 일반적으로 바이너리 파일 형태로 변환되어 제공되기 때문에 컨텐츠를 수정하거나 활용하여 더 나은 형태의 컨텐츠를 제작하는데 제약이 많다.

셋째, 다양한 웹 표준 언어와의 연계성이 제한적이다. 플래시는 상호작용, 애니메이션, 그래픽 객체의 제어 등을 위해서 ActionScript라는 자체의 스크립트를 사용하기 때문에, HTML, XML, MathML, SMIL, Javascript, Java 등 인터넷을 위해서 유용한 마크업 언어나 스크립트 언어와의 연동에 제약이 많다. 플래시도 하이퍼링크를 통하여 기존의 웹 컨텐츠를 단순히 접근하거나, 웹 문서 내에 HTML 태그나 Javascript를 이용하여

플래시 객체를 삽입할 수 있지만 플래시 프로그램의 폐쇄성으로 인해서 상호 연동 측면에 제약이 많다.

반면에, SVG는 웹 그래픽스를 위해서 W3C(World Wide Web Consortium)에 의해 제안된 XML 기반 그래픽 표준으로서, 비표준포맷인 플래시와 달리 XML의 개방성, 상호운용성 등의 장점을 모두 수용하며 SMIL, GML, MathML 등 다른 XML 언어들과 결합하여 다양한 웹 어플리케이션을 생성할 수 있다[11][14].

또한, SVG 역시 벡터 그래픽 방식이기 때문에 벡터 그래픽의 장점을 모두 갖는다. 그리고, XML에 기반을 두고 있어서 간단한 텍스트에디터만으로 그래픽을 생성할 수 있다. 그리고, 손쉽게 SVG 기반 그래픽을 그릴 수 있는 다양한 종류의 SVG 전용 편집기가 공개되어 있다[12].

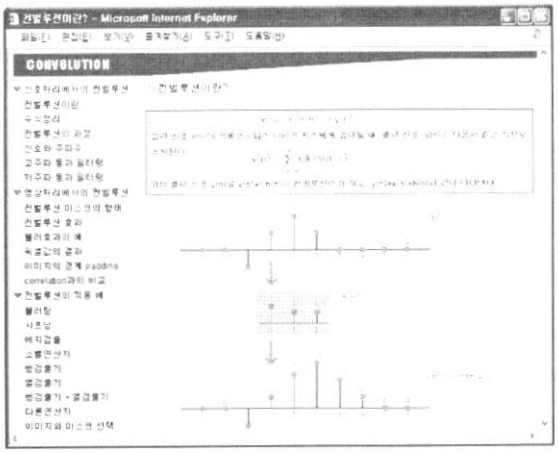
SVG의 개방적인 장점은 다른 마크업 언어와의 연동에서 잘 나타난다. 수학교과에서 플래시를 이용하여 교육용 컨텐츠를 제작할 경우[13], 수식에 필요한 문자나 기호를 입력한 후, 마우스를 이용하여 배치해야 하는 번거로운 과정을 거쳐야 하거나, 수식을 이미지로 제작하여 삽입해야 한다. 그러나, SVG에서는 웹상에서 수식을 표현하고 처리하기 위해서 제안된 MathML과의 연동을 통하여 수식을 효과적으로 나타낼 수 있다.

위와 같은 SVG의 장점으로 인해 SVG 기반 교육용 컨텐츠 개발에 대한 시도가 증가하고 있다[17]-[19].

3. SVG에 의한 컨벌루션 연산의 시각화 및 컨벌루션 활용 예시의 구현

3.1 전체적인 화면 구성

전체적인 화면 구성은 <그림 3>과 같이 왼쪽에는 신호처리에서의 컨벌루션, 영상처리에서의 컨벌루션, 컨벌루션의 적용 예 등 세 개의 주 메뉴로 구성하여 펼침 버튼을 클릭하면 세부 항목들을 보여주도록 하였으며, 오른쪽에는 각 항목을 클릭했을 때에 이에 해당하는 내용을 보여주도록 하였다.



<그림 3> 전체적인 화면 구성

3.2 1차원 신호처리에서의 컨벌루션 연산의 시각화

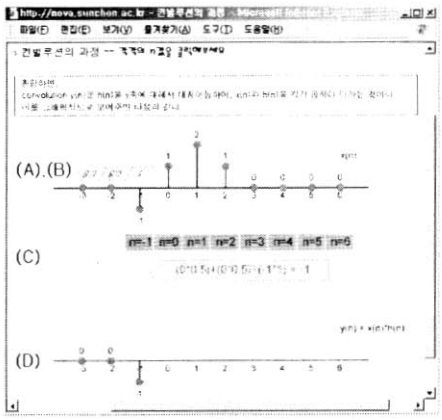
SVG는 점, 선, 면, 원, 타원, 사각형, 베지어 곡선, 채우기, 패턴 등 그래픽 객체들이 HTML에서의 태그와 속성 형태로 제공된다. <표 1>은 SVG의 그래픽 요소의 예이다. 이외에도 그룹핑, 이동, 회전, 크기변환 등 컴퓨터 그래픽스에서 그래픽을 표현하기 위해서 필요한 대부분의 요소들이 제공된다.

<표 1> SVG의 그래픽 요소의 예

객체	요소명	속성
원	<circle>	cx="<중심점 x좌표>" cy="<중심점 y좌표>" r="<반지름>"
타원	<ellipse>	cx="<중심점 x좌표>" cy="<중심점 y좌표>" rx="<x축 반경>" ry="<y축 반경>"
사각형	<rect>	x="<좌표값>" y="<좌표값>" width="<크기>" height="<크기>"
선	<line>	x1="<시작점 좌표값>" y1="<시작점 좌표값>" x2="<끝점 좌표값>" y2="<끝점 좌표값>"
다각형	<polygon>	points="<점들의 리스트>"
연결선	polyline	points="<점들의 리스트>"

<그림 4>는 1차원 신호에 대한 컨벌루션 연산 과정을 그래픽을 이용하여 시각화한 화면이다.

컨벌루션 연산이 이루어지는 과정은 다음과 같다. SVG 파일이 로딩이 되면, 애니메이션을 통하여 마스크가 반전되어 입력 신호 위에 위치하는 과정이 진행되고(A), 사용자가 입력 신호 밑의 버튼을 차례대로 클릭할 때마다 마스크의 위치가 이동되고(B), 해당 위치에서 마스크값과 겹치는 입력 신호가 서로 곱해서 더해지는 계산 과정이 타이핑이 쳐지는 효과로 제시되며(C), 그에 따른 결과가 출력으로 나타난다(D).



<그림 4> 컨벌루션 연산의 시각화

<그림 4>를 구현한 SVG 코드를 <그림 5>에 제시하였다. 실제 구현한 내용 중에서 주요 부분을 제시하였으며 사용된 핵심 요소와 속성을 볼드체로 표시하였다.

<그림 5>의 (A)는 컨벌루션 연산식 (1)에서 컨벌루션 마스크 $h(k)$가 세로축에 대해서 반전되어 $h(-k)$가 되는 것을 구현한 것으로서, <animateTransform> 요소와 그에 따르는 속성을 이용하였다. SVG 파일이 로딩이 되면 <그림 4>에서 볼 수 있는 것과 같이 마스크가 반전되어 입력 신호 위에 위치하게 된다.

(B)는 학습자가 (C)에 제시된 버튼을 클릭할 때마다 마스크의 위치가 이동되는 것을 구현한 내용을 보여준다.

(C)는 <그림 4>에서 "n=-1"으로 표시된 버튼을 구현한 것을 나타내는데, 버튼이 클릭된 후의 애니메이션 제어를 위해서 이 버튼에 button-1이라는 아이디를 할당하였다. 학습자가 버튼을 클릭하면 자바스크립트 함수 typing(evt,-1)가 호출된다. typing(evt,n) 함수는 해당 버튼을 클릭됐을 때, n의 위치에 있는 반전된 마스크값과 겹치는 입력 신호 $x(n)$들을 서로 곱하고, 곱한 결과를 더하는 계산 과정이 처리되는 함수이다.

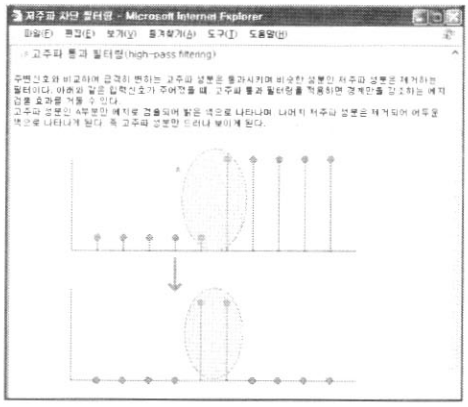
(D)는 수치로 계산된 결과를 그래프로 그려 주는 부분이다. 사용자가 버튼을 차례대로 클릭하면서 (B)-(D) 과정이 반복적으로 진행되면 컨벌루션 연산이 수행되는 과정과 그에 따른 결과 $y(n)=x(n)*h(n)$를 볼 수 있다.

```
<svg>
...
  <animateTransform attributeName="transform"
      attributeType="XML" type="rotate" from="0"
      to="180" additive="sum" begin="3s" dur="2s" />          (A)
  ...
    ...
<use xlink:href="#mask" transform="translate(150,-30)
      skewX(-30)">
  <animate attributeName="x"
      attributeType="XML" from="0" to="-40"
      begin="button-1.click" dur="1s" fill="freeze"/>         (B)
  ...
    ...
<a><g onClick="typing(evt,-1)">
      <rect x="160" y="260" width="38" height="22"
        rx="3" ry="3" id="button1"/>
      <text id="button-1" x="165" y="276">n=-1</text>         (C)
</g></a>
  ...
    ...
<use xlink:href="#result(-1)" style="visibility:hidden">
  <set attributeName="visibility" to="visible"
      begin="button-1.click+2s" dur="1s" />                   (D)
</use>
...
</svg>
```

<그림 5> 컨벌루션 연산 과정에 대한 SVG 코드

이밖에 1차원 신호처리에서의 컨벌루션 연산이 적용되는 분야를 이해하기 쉽도록 <그림 6>과 같은 고주파 및 저주파 통과 필터링에 대한 개념과 예를 간단한 그림을 통하여 제시하였으며, 식 (1)과 이를 식 (2)와 같이 전개한 수식을 포함한 화면 등을 제시하였다.

이후 제시된 그림에서의 애니메이션, 그래픽은 SVG, Javascript, HTML 등을 이용하여 <그림 5>에 제시된 것처럼 구현되었는데 구체적인 코드는 지면상 생략한다.

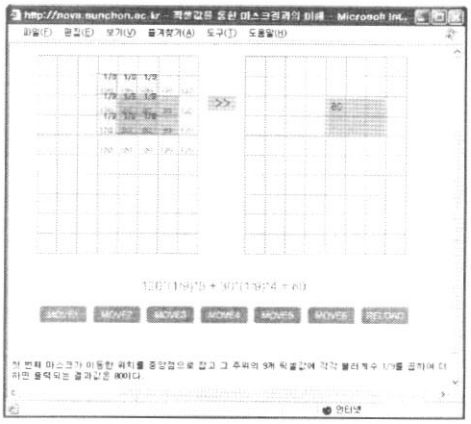


<그림 6> 고주파 통과 필터링 개념의 이해

3.3 영상에서의 컨벌루션 연산의 시각화

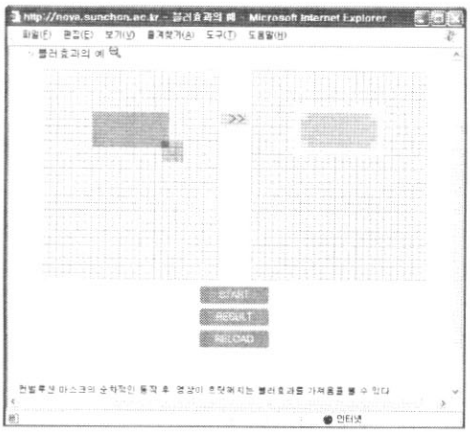
영상에 대한 컨벌루션은 입력 신호가 2차원이다. <그림 7>은 2.2절에서 언급한 영상에서의 컨벌루션 연산을 시각화한 화면을 나타낸다.

1차원 신호를 위한 컨벌루션 연산의 시각화에서와 유사하게 SVG 파일이 로딩되면 마스크가 반전이 되어 입력 영상의 픽셀값에 위치한다. 그리고, 사용자가 버튼을 클릭하면 마스크가 래스터 스캔 방향으로 이동하고, 애니메이션을 이용하여 연산 결과가 타이핑 쳐지는 것처럼 제시하여 학습자의 관심을 끌 수 있도록 하였다.



<그림 7> 영상에 대한 컨벌루션 연산

그리고, 컨벌루션 연산 과정에 대한 이해를 돕기 위하여 <그림 8>과 같은 블러 효과에 대한 예를 제시하였다. 사용자가 START 버튼을 클릭하면 그림에서 좌측에 제시된 입력 영상에 대해서 래스터 스캔 방향으로 마스크가 픽셀 단위로 이동하고, 그에 따른 결과가 우측에 제시된 결과 화면에 실제의 픽셀값 변화가 나타나는 것을 확인할 수 있도록 하였다.

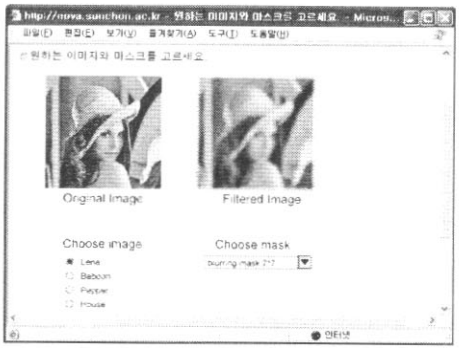


<그림 8> 블러 효과의 예

또한, 컨벌루션 마스크의 종류를 소개하는 화면과 영상의 가장자리 부분의 처리를 위한 간단한 패딩(padding) 방법을 그림을 통하여 쉽게 이해할 수 있도록 제시하였다.

3.4 컨벌루션 적용 예시의 구현

컨벌루션이 쓰이는 분야와 적용될 나타나는 효과를 직관적으로 학습할 수 있도록 사용자가 영상과 컨벌루션 마스크를 선택하여 적용하고, 결과를 볼 수 있도록 구현하였다. <그림 9>에서 학습자는 영상과 적용하고자 하는 컨벌루션 마스크를 직접 선택할 수 있으며, 마스크의 종류와 크기에 달라짐에 따라서 블러링, 샤프닝, 에지 검출 등의 결과를 확인해 볼 수 있다.



<그림 9> 컨벌루션의 응용과 효과

4. 컨벌루션 시각화의 적용 및 면담 분석

4.1 학습자 및 학습 절차

컨벌루션 연산의 시각화를 웹 화면을 통해 학습자가 학습하였을 경우, 이에 대한 효과를 알아보기 위하여 경기도 소재 실업계 S 고등학교 2학년 학생 중 자발적인 참여를 원하는 남학생, 여학생 1명씩을 선정하여 개별 학습을 한 후, 이에 대한 면담을 실시하였다.

수업은 두 번으로 나누어 진행하였는데 학습자들이 수업 시작부터 수식에 대한 거부감을 갖는 것을 피하기 위해서 그래픽 화면이 상대적으로 많이 제공되는 영상에 대한 컨벌루션을 먼저 진행하고, 수식이 포함되어 있는 1차원 신호처리에 대한 컨벌루션에 대한 수업을 진행하였다.

면담 또한 2차에 걸쳐서 진행하였는데, 1차 면담은 영상처리에 대해 공부를 진행 후 실시하였다. 그리고, 신호처리에 대해 학습을 진행한 후 2차 면담을 실시하였다. 면담은 모두 녹음한 후 전사(transcription)하여 문서화하였다. 학습자들과의 대화 내용은 가급적 그대로 옮겼다. 그리고, 면담을 실제 학습자의 실제 이름 대신 소은, 정우를 사용하였다.

4.2 영상에서의 컨벌루션 학습 후의 면담

영상에서의 컨벌루션을 학습한 후 1차면담에서의 반응을 정리해보면 다음과 같다. 소은과 정우는 <그림 6> 화면에서 컨벌루션 마스크가 이동하면서 영상 위에 마스크가 겹쳐져서 마스크값과 픽셀값이 곱해지는 애니메이션을 통해 컨벌루션 결과를 쉽게 이해할 수 있다고 답했다.

연구자 : 화면에서 픽셀 값 변화에 대한 이해가 잘 돼?
소은: 값을 정해놓고 그 변화되는 결과를 보여주는 게 앞에서의 다른 화면보다 더 잘 이해가 돼요. 이게 제일 맘에 들어요. 특히나 마스크가 한 픽셀 이동하면 아래의 이미지의 값과 곱해지는 게 이해하기 가장 쉬워요.
정우: 네. 값의 분명한 차이를 뭉개주는 효과를 말하는 거죠? 차이가 조금 줄어든 것처럼 보여요. 그리고 실제 값을 곱해서 더해보니 결과도 맞는 거 같아요. 그냥 설명글만 있는 것보다는 애니메이션으로 아래의 값과 겹쳐서 보여주는 게 가장 이해하기 쉬운 거 같아요.

<그림 8>의 블러 효과 화면에서 컨벌루션의 입력에 해당하는 픽셀을 이해하고 있는지를 알아보기 위한 질문에 대해서 소은은 픽셀을 이미 알고 있어서인지 알고 있다고 쉽게 대답하였다. 반면에 정우는 다소 막연한 반응을 보였다. 그런데, 소은의 경우 컨벌루션 마스크를 첫 번째 픽셀부터 진행해야 되지 않느냐는 적극적인 반문을 하면서 흥미를 나타내었다.

연구자 : 이미지에 픽셀을 가상하여 그린 화면은 픽셀에 대해 말로 하는 설명보다 더 좋은 거 같아?
소은 : 네. 픽셀은 예전에 선생님이 몇 번 설명해주셨잖아요. 그런데, 이미지의 첫 번째 픽셀부터 마스크를 씌워야하지 않나요?
연구자 : 소라가 급하군. 첫 번째 픽셀과 같은 가장자리는 조금 있다 설명할게에요.
소은 : 갑자기 궁금했어요. (얼굴에 미소가 가득함)
정우 : 그냥 픽셀인가보다 막연히 그렇게 보이는데요.

블러링과 샤프닝을 설명하는 화면에서는 소은은 이미 이전 화면을 통해 마스크가 이동하며 컨벌루션을 수행하는 과정을 이해했기 때문에 더 이상 애니메이션의 필요를 느끼지 않는다고 하였다. 이는 컨벌루션 연산이 영상 내의 모든 픽셀에 대해서 마스크가 움직이며 계산된다는 것을 이해하고 있음을 엿볼 수 있는 대목이다. 정우는 두 화면을 서로 비교해서 컨벌루션 결과를 보여주어 이해가 잘 되지만 설명글은 지루하다는 반응을 나타냈다.

연구자 : (블러링)과 (샤프닝)화면은 애니메이션 효과가 없어서 재미없지?
소은 : 마스크가 움직이며 계산하는 과정은 같을 텐데요. 뭘... 마스크 안에 값이 있는 걸 보니 수학시간에 배운 행렬이 생각나요.
정우 : 블러링 효과와 샤프닝 효과가 두드러지게 보여서 좋은데요. 설명글은 지루해요.

이는 식 (2)의 영상에 대한 컨벌루션 연산의 시각화를 통하여 직관적으로 컨벌루션 연산 과정을 이해하는 것이 가능하다는 것을 나타낸다.

4.3 1차원 신호에서의 컨벌루션 학습 후의 면담

1차원 신호에서의 컨벌루션 수업에서는 컨벌루션 시각화 화면과 함께 식 (1)의 컨벌루션 수식

에 대한 학습을 시도하였다. 학습 후 2차 면담에서의 반응을 정리해보면 다음과 같다. 두 학생 모두 수식을 전혀 사용하지 않은 영상에서의 컨벌루션 수업과 달리 수식이 있는 1차원 신호에서의 컨벌루션을 부담스러워함을 볼 수 있었다.

연구자 : 영상처리와 신호처리를 비교해보면 어때?
소은 : 신호처리에서 영상처리보다 이해가 더 잘되는 화면이 있었고, 반면에 수식은 조금 까다로웠어요.
정우 : 수식 말고는 쉬웠어요. 그리고 아.... 이런 원리들이 있구나 하고 막연히 생각이 들어요.

식 (1)의 컨벌루션 수식 학습 후 소은은 수식을 연습장에 풀어보며 이해하려는 노력을 나타낸 반면, 정우는 조금은 수식을 더 들여다보다 더 이상 매달리지 않았다. 그리고, 두 학생 모두 시각화를 통한 컨벌루션과 수식과의 관계를 연결시키는데 어려워 하였다.

연구자 : 수식이 영상처리에서 배운 컨벌루션과는 어떤 관계인 거 같아? x는 원래 이미지고, h는 마스크고, 이미지에 마스크가 덧씌워져 출력결과 y가 얻어진다는 걸 수식으로 표현한건데.
소은 : 아무 상관이 없는 관계요. 수식은 잘 이해가 안되고, 1차원인 평면적으로 움직이는 화면(<그림 11>)은 조금 이해가 되는데요.
연구자 : 컨벌루션의 과정이 바로 수식을 나타낸 건데... 수식에서 x(k)와 뒤집어진 마스크 h(-k)가 곱하여지는 것을 애니메이션으로 보여주는 게 (컨벌루션의 과정)인데...
정우 : 수식과는 별개로 보여지는데요.

상대적으로 수업에 적극적으로 참여한 소은이도 수식에 대해서 어려움을 호소하였다. 정우의 경우 수식이 들어 있는 두 번째의 수업 후에는 새로운 학습을 시도할 마음이 없다고 하였다.

연구자 : 지난번 선생님하고 공부 끝나고 집에 가서 생각나는 거 없었어?
소은 : 아무 생각 없었어요. 그냥 오늘도 재미있을 거라고 기대했는데 갑자기 수식을 만나니 어려워요. 그리고 이해가 안돼서 짜증이 나요. 식만 없었더라면 100% 이해했을 거 같은 욕심이 나는데....
연구자 : 다음에 또 이렇게 새로운 원리를 설명해주는 게 있다면 공부해볼 마음은 있어?
소은 : 네. 다음에도 가르쳐주세요.
정우 : 아뇨. 이번으로 끝내죠.

그리고, <그림 6>의 고주파 통과 필터링과 이와 유사한 형태로 제시된 저주파 통과 필터링화면에서 1차원적인 막대의 변화를 통해 컨벌루션의 결과를 쉽게 이해할 수 있다고 답했다.

연구자 : 가장 쉽게 설명하고 있는 건 뭐지?
소은 : (고주파 통과 필터링)과 (저주파 통과 필터링)요. 1차원으로 보여주니 더 잘 이해가 되고 막대의 변화를 비교해보면 차이를 알 수 있어요.
연구자 : 가장 이해하기 쉬운 화면은 어디야?
정우 : (고주파 통과 필터링)과 (저주파 통과 필터링)요.
연구자 : 이유가 뭐지?
정우 : 처음엔 저주파, 고주파라는 용어도 어려웠지만 두 화면이 비교되어서 보여주니 결과를 쉽게 이해할 수 있겠어요.

면담에서의 반응을 종합해 보면, 컨벌루션 연산을 전개한 수식은 단순히 곱셈과 덧셈만으로 이루어져 있으며, 연구자가 충분히 이해시키려고 노력하였음에도 불구하고 어려워함을 볼 수 있었다. 고주파 통과 필터링의 경우는 컨벌루션 전개 수식에 비해서 오히려 학습자들에게 어려울 수 있는 개념임에도 불구하고 오히려 이해하기 쉽다고 느끼고 있음을 볼 수 있었다. 학습자들은 컨벌루션 마스크가 시각적으로 움직이면서 컨벌루션 결과가 얻어지는 과정에서는 흥미를 보이며, 짧은 수업 시간에도 불구하고 직관적으로 컨벌루션 연산을 어느 정도 이해하고 있음을 볼 수 있었다.

5. 결 론

본 논문에서는 웹을 위한 벡터 그래픽 표준인 SVG를 이용하여 신호처리 및 영상처리 분야 등에서 기본 연산이라고 할 수 있는 컨벌루션 연산을 그래픽 요소와 애니메이션을 활용하여 시각화하여 학습자의 직관적인 이해를 돕고자 하였다. 이를 위하여 학습자가 직접 버튼을 조작해보며 컨벌루션 마스크의 이동과 그 결과를 시각적이고도 직관적으로 알 수 있도록 제공하여 학습자가 보다 쉽고 흥미롭게 학습할 수 있도록 하였다.

컨벌루션의 개념과 연산 과정을 학습 후 학습자가 원하는 영상과 컨벌루션 마스크를 직접 선택하여 컨벌루션 결과가 학습한 내용과 일치하는지를 확인해봄으로써 학습자의 흥미를 유발시키며, 학습에 자신감과 만족감을 주도록 제작하였다. 중등교육 과정의 그래픽 교과에서 단순히 암기나 반복적인 숙련에 의해서 익히던 다양한 영상처리 기법을 근본적으로 이해하는데 도움이 될 것으로 예상된다.

본 연구에서는 특정한 저작 도구에 의존하지

않고 SVG, Javascript, HTML, MathML 등 공개된 웹 언어만을 이용하여 컨벌루션에 대한 시각화를 구현하였다. SVG가 가지는 벡터 그래픽 기반의 장점, 풍부한 상호작용성, 소스 코드의 공개성, 그리고 자료수정 및 가공이 용이한 장점 등으로 인하여 향후 SVG를 이용하여 다양한 웹 기반 교육용 컨텐츠 제작을 위한 시도들이 많이 이루어질 것으로 기대된다.

참 고 문 헌

[1] R.C.Gonalez · R.E.Woods(1992). Digital Image Processing. Prentice Hall.

[2] 천인국 · 이강승(2004). 영상처리 기초편. 기한재.

[3] 최형일 · 이근수 · 이양원(1999). 영상처리 이론과 실제. 홍릉과학출판사.

[4] 최윤철 · 고견 · 임순범(1998). 멀티미디어 배움터. 생능출판사.

[5] 신찬호 · 조성분 · 김해용 · 홍주화(2005). 고등학교 교과서 웹디자인. 삼양미디어.

[6] 김길수(2005). 할수있다! PhotoShop 7 기본+활용 쉽게 배우기. 영진닷컴.

[7] 이진이(2004). 디지털 신호처리. 기한재.

[8] 교원캠퍼스(2004). 플래시를 이용한 수학 학습 자료 작성. 교원캠퍼스.

[9] 이경금 · 이광호(2005). Flash를 이용한 체육과 웹 코스웨어 설계 및 구현. 한국컴퓨터교육학회 동계학술대회논문집.

[10] 오애경 · 장은영 · 배영권 · 이태욱(2005). 마이크로프로세서 실습을 위한 시뮬레이션형 웹 코스웨어 설계 및 구현. 한국컴퓨터교육학회 동계학술대회논문집.

[11] 나방현 · 심규찬 · 이종연(2001). XML 그래픽 입문. 21세기사.

[12] W3C. SVG Implementation and Resource Directory. www.svgi.org.

[13] 교육소프트연구소. www.math2000.co.kr.

[14] J.D.Eisenberg(2002). SVG Essentials. O'REILLY.

[15] 이정헌 · 허훈 · 채옥삼(2005). 영상처리에서 효율적인 교육과 연구를 위한 비주얼 프로그래밍 환경 개발. 정보과학회 논문지. 32(1)

[16] J.Campbell, F.Murtaph, and M.Kokuer (2001). DataLab-J: A signal and image processing laboratory for teaching and research. IEEE Transaction on Education. 44(4).

[17] 김희원(2003). SVG를 이용한 지리정보 활용에 대한 연구. 서울대학교 석사논문.

[18] 김현아(2003). SVG를 이용한 웹 기반 교육의 설계 및 구현. 숙명여자대학교 석사논문.

[19] 박경아 · 강의성(2004). 베지어 곡선 학습을 위한 SVG기반 코스웨어. 한국컴퓨터교육학회 하계학술대회논문집.

김 영 미

1990 순천대학교 컴퓨터교육과 (경영학사)

2001 순천대학교 컴퓨터교육과 (이학사)

2001 순천대학교 교육대학원 컴퓨터교육전공 (교육학석사)

관심분야: 컴퓨터교육

E-Mail: wealthy3@hanmail.net

강 의 성

1991 고려대학교 전자전산공학과(공학사)

1995 고려대학교 전자공학과(공학석사)

1999 고려대학교 전자공학과(공학박사)

1999~2000 고려대학교 전자공학과 Post-Doc.

2001~현재 순천대학교 컴퓨터교육과 조교수

관심분야: 컴퓨터교육, 영상신호처리, 멀티미디어통신, 바이오인포매틱스

E-Mail: magasa@sunchon.ac.kr