

시간 할당 방식의 양 3차회선 스케일러 하드웨어 구현

Hardware Implementation of Bicubic Scaler of Time Allocation Method

정 세 민^{*+}, 한 시 연^{*+}, 응 오 닷^{**★}, 강 봉 순^{*}

Semin Jung^{*+}, Si-Yeon Han^{*+}, Dat Ngo^{**★}, and Bongsoon Kang^{*}

Abstract

The growth of the multimedia industry has led to the advent of digital devices utilizing images of varying resolutions. Accordingly, it is essential to adapt the resolution of the source image to match with the resolutions of the target digital device, while ensuring that the quality of the input image is preserved throughout this process. In this paper, we propose the implementation of a hardware system that performs image scaling using interpolation algorithms. The bicubic interpolation algorithm is employed as the interpolation algorithm, as it produces the highest quality image among the three most common methods. Each interpolated axis in the image can be operated independently to achieve asymmetric scaling in any desired ratio. The hardware of the proposed scaler satisfies real-time processing and is implemented with less memory resources than the previously studied hardware using the time allocation method, showing that it is a suitable structure for the hardware of the scaler.

요 약

멀티미디어 산업의 발전으로 다양한 해상도의 이미지를 사용하는 디지털 기기가 등장하고 있다. 따라서 동일한 입력 이미지라도 사용되는 디지털 기기에 따라 해상도를 변환하는 과정이 필요하며, 이와 같은 과정에서 입력 이미지의 일관된 품질을 유지하는 것이 중요하다. 본 논문에서는 보간 알고리즘을 이용해 이미지 스케일링을 진행하는 하드웨어를 구현한다. 사용되는 보간 알고리즘은 대표적인 3가지 방식 중 가장 높은 품질의 이미지를 생성하는 양 3차회선 보간 알고리즘을 사용한다. 이때 이미지에서 보간되는 각 축은 독립적으로 동작해 원하는 비율로 비대칭 확대 및 축소가 가능하다. 제한하는 스케일러의 하드웨어는 실시간 처리를 만족하며, 시간 할당 방식을 사용해 기존 연구된 하드웨어보다 적은 메모리 자원으로 구현되어 스케일러의 하드웨어에 적합한 구조임을 보인다.

Key words : Verilog, Xilinx Up-scaling, Down-scaling, Interpolation, Real-time processing

* Dept. of Electronics Engineering, Dong-A University

** Dept. of Computer Engineering, Korea National University of Transportation

★ Corresponding author

E-mail : datngo@ut.ac.kr, Tel. : +82-43-841-5347

+ These authors contributed equally to this work.

※ Acknowledgment

This was supported by Korea National University of Transportation in 2024.

Manuscript received Aug. 16, 2024; revised Sep. 2, 2024; accepted Sep. 9, 2024.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

현재 멀티미디어 산업의 발전으로 다양한 해상도를 가지는 디지털 기기가 등장하였다. 따라서 디지털 기기에서 사용되는 입력 이미지는 디지털 기기의 해상도에 맞춰 변환하는 과정(Up-scaling 또는 Down-scaling)이 필요하다. 이와 같은 과정에서 입력된 이미지의 일관된 품질을 유지하는 것이 중요하며, 해상도 변환은 스케일링 과정을 통해 진행된다. 이미지의 해상도 변환은 크게 디퍼닝 기반 방식[1]과 보간(Interpolation) 알고리즘을 사용하는 방식[2]으로 구분할 수 있다.

본 논문에서는 실시간 처리에 적합한 보간 알고리즘을 사용하며, 대표적인 보간 방식 중 가장 높은 품질의 이미

지를 생성하는 양 3차회선 보간(bicubic interpolation) 알고리즘을 스케일러의 하드웨어에 적용한다[3]. 하드웨어는 각 축으로 독립적인 스케일 비율을 사용해 0.2~2.0 배의 비대칭 또는 대칭 확대 및 축소가 가능하다.

스케일러의 하드웨어는 사용하는 메모리의 종류와 개수에 따라 다양한 구조로 설계가 가능하다[4]. 하드웨어의 구조는 크게 dual-clock을 이용하는 주파수 변환 방식과 single-clock을 이용한 시간 할당 방식 두 가지 구조로 설계가 가능하다.

주파수 변환 방식은 입력 이미지 데이터를 받아들일 때 사용되는 클럭(clock)보다 높은 주파수를 가지는 클럭과 dual-port SRAM(DPSRAM)을 이용해 입력 데이터가 처리되는 주파수를 변환시키는 방식이다[5]. 이와 같은 방식은 확대 동작에서 증가한 출력 데이터가 생성되는 시간을 보장할 수 있지만, 추가로 높은 주파수 클럭과 하드웨어 구현 시 큰 면적을 차지하는 DPSRAM이 필요하다는 단점이 있다.

본 논문에서 사용하는 시간 할당 방식은 출력 데이터가 증가하는 경우 증가한 데이터의 크기만큼 시간을 할당하는 방식이다. 시간 할당 방식의 경우 주파수 변환 방식과 비교하였을 때 설계에 필요한 메모리 자원을 줄일 수 있으며, 두 가지 이상의 클럭 도메인을 사용할 때 발생할 수 있는 문제인 clock domain crossing(CDC) 위험[6] 문제에서 벗어난다는 장점이 있다.

설계되는 스케일러에 적용되는 양 3차회선 보간은 다른 보간 알고리즘에 비해 비교적 많은 곱셈 연산으로 회로의 동작 주파수가 낮아져 처리되는 영상의 frame per seconds(fps)가 낮아지는 문제가 존재한다. 따라서 파이프라이닝 기법[7]을 활용해 QHD 표준 이미지(2560×1440)를 각 축으로 구현되는 하드웨어의 최대 확대 비율인 2.0배까지 확대하더라도 영상을 보는 것과 같은 실시간 처리를 만족할 수 있도록 field programmable gate array(FPGA) 환경에서 구현한다. 실시간 처리의 기준은 30fps를 기준으로 한다.

II. 본론

본 장에서는 설계된 스케일러 하드웨어에 적용되는 양 3차회선 보간 알고리즘에 대해 설명한다. 또한 스케일러 성능 비교를 목적으로 기존에 연구된 스케일러[8]와 peak signal-to-noise ratio(PSNR)[9] 수치를 활용해 정량적 평가를 수행한다. 마지막으로 스케일러 하드웨어 설계에 사용되는 주파수 변환 방식[5]과 본 논문에서 제안하는

시간 할당 방식을 소개한다.

1. 양 3차회선 보간 알고리즘

양 3차회선 보간 알고리즘의 보간 과정은 유효한 픽셀, 유효한 픽셀의 위치에서 보간되는 축의 -1, +1 그리고 +2번째 위치한 4개의 인접한 픽셀을 이용해 3차 다항식을 구한 후 3차 다항식을 이용해 보간되는 픽셀의 값을 구한다.

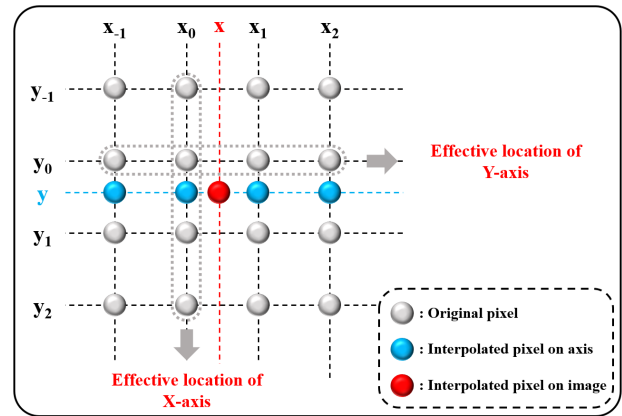


Fig. 1. Bicubic interpolation.

그림 1. 양 3차회선 보간

그림 1은 양 3차회선 보간이 진행되는 과정을 나타낸다. 보간되는 위치를 구하기 위해 스케일 비율(scale ratio)에 따른 증감값(increment)을 구해 보간되는 축에 증감값을 가산하는 방식을 사용한다[10]. Y축으로 보간되는 픽셀의 값을 구했다고 가정할 때 구해진 픽셀(푸른 원)을 이용해 X축으로 보간된 픽셀(붉은 원)을 구할 수 있다. X축 보간 과정을 예로 보간 방식을 설명하면, (x, y) 의 위치가 보간되는 픽셀 위치일 때 (x_0, y) 에 위치한 픽셀이 유효한 픽셀이다. 이때, 보간되는 픽셀의 값은 (x_{-1}, y) , (x_0, y) , (x_1, y) 그리고 (x_2, y) 를 지나는 3차 다항식을 이용해 구한다. 참조되는 4개의 픽셀을 지나는 3차 다항식은 아래 식 (1)과 같이 표현할 수 있다 [10].

$$P(x, y) = ax^3 + bx^2 + cx + d \quad (1)$$

식 (1)의 $P(x, y)$ 는 (x, y) 위치의 픽셀 값을 의미하며, a , b , c 그리고 d 는 다항식 계수를, x 는 유효한 픽셀과 보간되는 픽셀 사이의 거리로, 0과 1사이의 값을 가지는 가중치를 의미한다. 그림 2는 인접한 픽셀을 지나는 3차 다항식을 이용해 보간이 진행되는 모습을 나타낸다.

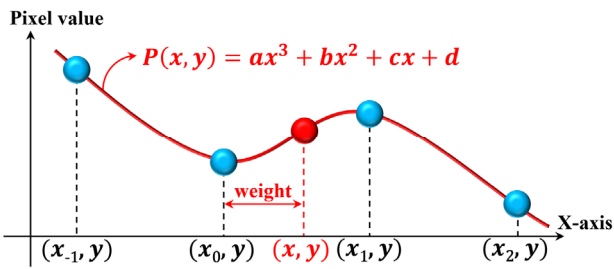


Fig. 2. Cube polynomials for interpolation.
그림 2. 보간을 위한 3차 다항식

그림 2의 다항식 계수 a, b, c, d 는 다음과 같은 방식으로 구할 수 있다. (x_0, y) 픽셀 X축의 위치가 0이라면, $P(x_0, y) = d$ 이고 (x_1, y) 에서 X축 위치가 1일 때 $P(x_1, y) = a + b + c + d$ 임을 알 수 있다. 또한 같은 방식으로, 식 (1)의 도함수를 이용해 $P'(x_0, y) = c$ 와 $P'(x_1, y) = 3a + 2b + c$ 를 구할 수 있다. 앞서 구한 식들로 다항식의 계수를 구할 수 있으며, 아래 식 (2)와 같이 표현할 수 있다[10].

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P(x_{-1}, y) \\ P(x_0, y) \\ P(x_1, y) \\ P(x_2, y) \end{bmatrix} \quad (2)$$

식 (2)에 인접한 픽셀 값을 대입하면, $P(x, y)$ 를 구할 수 있다. 이와 같은 방식으로 스케일러는 그림 1에 나타난 것처럼 Y축 보간을 우선적으로 수행해 y 에 위치한 4개의 픽셀(푸른 원)의 값을 구하고 X축 보간을 진행해 (x, y) 에 위치한 픽셀(붉은 원)의 값을 구할 수 있다. 즉, 양 3차회선 보간은 16개의 인접한 픽셀 값을 참조하는 것을 알 수 있다.

스케일러 하드웨어에 앞서 설명한 양 3차회선 보간 알고리즘을 적용할 때 하드웨어용 알고리즘으로 변환하는 작업

이 필요하다. 그 이유는, 부동 소수점(floating point)으로 표현된 소프트웨어용 알고리즘은 신호를 표현하는 bit 수가 증가해 하드웨어 복잡도가 증가하기 때문이다. 따라서 고정 소수점(fixed point)을 사용하는 하드웨어용 알고리즘을 이용해 필요한 bit만을 신호에 할당한다.

본 논문에서 설계되는 하드웨어는 부동 소수점 알고리즘 결과와 비교하였을 때, 8-bit 기준 ± 0.1 이내의 오차를 가지는 하드웨어용 알고리즘을 적용하였다.

2. 알고리즘 성능 평가

양 3차회선 보간 알고리즘을 하드웨어에서 구현하게 된다면, Y축 보간을 진행할 때 인접한 4개의 라인 데이터를 참조한다. 하드웨어에서 Y축 보간은 라인메모리를 이용해 입력되는 라인의 데이터를 저장하는 과정이 필요하다. 이러한 방식으로 양 3차회선 보간은 다른 보간 알고리즘과 비교했을 때 하드웨어를 구현하기 위해 더 많은 메모리가 필요하다.

스케일러 하드웨어에서 사용되는 메모리 자원을 줄이기 위해 양 3차회선 보간 알고리즘을 변형한 알고리즘의 연구가 진행되고 있다[8], [11]. 그중 Han[8]은 Y축 보간에는 인접한 2개의 라인을 참조하는 선형 보간을 사용하고, X축 보간은 3차회선 보간을 사용하는 조합 보간(combinatorial interpolation) 알고리즘을 제안했다. 조합 보간은 하드웨어 구현 시 Y축 보간 과정에서 2개의 라인 데이터를 참조하기에 양 3차회선 보간과 비교했을 때 사용되는 메모리를 줄일 수 있다. 그러나 보간에 사용되는 원본 픽셀이 줄어들어 출력 이미지의 품질이 낮아지는 단점이 있다.

표 1은 그림 3의 실험 이미지를 Han[8]의 조합 보간과 양 3차회선 보간을 이용해 1.8배 확대 후 축소한 이미지의 PSNR 수치를 측정해 본 것이다[8]. 표 1을 통해 양 3차회선 보간을 사용한 경우 PSNR 수치가 전체적으로

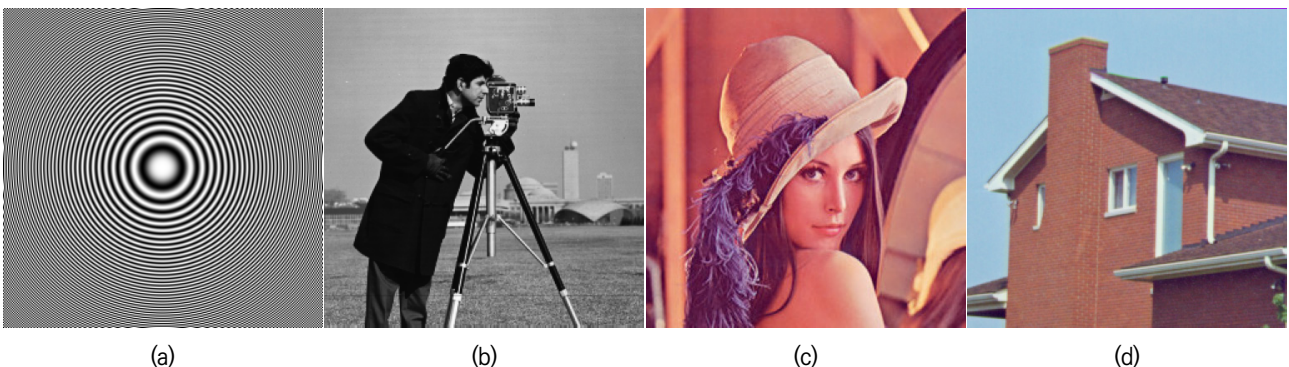


Fig. 3. Experimental images : (a) CZP, (b) Camera, (c) Lena, (d) House.
그림 3. 실험 이미지 : (a) CZP, (b) Camera, (c) Lena, (d) House

조합 보간보다 높음을 알 수 있다. PSNR 수치는 원본 이미지 정보의 손실 정도를 판단하며, 높은 값을 가질수록 보간 과정에서 원본 이미지 정보를 많이 보존했음을 의미한다.

Table 1. Comparison of PSNR with previous study.
표 1. 선행 연구와의 PSNR 수치 비교[dB]

Image \ Method	Han [8] (Combinatorial)	Proposed (Bicubic)
CZP	20.51	28.88
Camera Man	39.40	46.39
Lena	44.25	49.21
House	42.72	49.39

3. 주파수 변환 방식의 하드웨어 구조

스케일러의 하드웨어는 이미지 확대 동작에서 입력 데이터보다 많은 데이터가 출력되기 때문에 출력 데이터를 생성하기 위한 시간이 필요하다. 주파수 변환 방식은 입력 데이터를 받아들일 때 사용되는 클록보다 높은 주파수의 클록을 이용해 출력 데이터를 생성하는 시간을 할당하게 되는데 이때, 추가적인 클록의 주파수는 아래와 식 (3)을 이용해 설정할 수 있다.

$$f = f_{in} (XSR_{max} \times YSR_{max}) \tag{3}$$

f 는 추가로 사용되는 클록의 주파수를 나타내며, f_{in} 은 입력 데이터를 받아오는 클록의 주파수를 나타낸다. XSR_{max} 그리고 YSR_{max} 는 각 X축의 최대 scale ratio 그리고 Y축의 최대 scale ratio를 의미한다. 즉, 최대 확대 비율이 커질수록 높은 주파수의 클록이 필요함을 알 수 있다.

주파수 변환 과정은 DPSRAM에 입력 데이터를 저장한 후 높은 주파수 클록을 이용해 읽어내는 과정으로 진행된다. 이러한 방식을 통해 낮은 주파수로 처리되는 입력 데이터가 높은 주파수 클록을 통해 처리된다.

그림 4는 양 3차회선 보간 스케일러의 하드웨어에서 각 축의 최대 확대 비율이 2.0배인 경우 주파수 변환하는 과정을 나타낸다. 식 (3)을 통해 입력 데이터(i_data)를 받아오는 클록(clk)보다 4배 높은 주파수의 클록 (clk_4)이 필요함을 알 수 있다. 입력 데이터는 첫 번째 DPSRAM에 저장될 때 clk 의 도메인에서 동작되며, 데이터를 읽고 라인메모리에 전달하는 과정 그리고 보간을 진행할 때 clk_4 의 도메인에서 동작한다. 이와 같은 방

식으로 최대 확대 동작의 경우 한 라인의 데이터가 입력되는 시간 동안 2배 증가한 두 라인의 데이터를 생성할 수 있다.

주파수 변환 방식은 앞서 설명된 방식으로 동작하여 스케일러의 동작을 구현할 수 있지만, 스케일 비율에 따른 높은 주파수 클록 그리고 메모리를 사용한다는 점은 큰 단점으로 볼 수 있다.

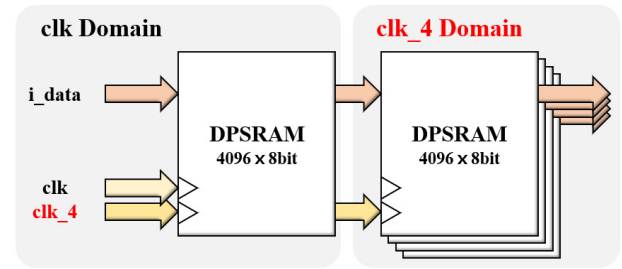


Fig. 4. Frequency conversion method.
그림 4. 주파수 변환 방식

4. 시간 할당 방식의 하드웨어 구조

시간 할당 방식은 단일 클록만을 이용해 스케일러의 하드웨어를 구현하는 방식이다. 주파수 변환 방식과 달리 시간 할당 방식은 입력 이미지 데이터를 받아들일 때 라인과 라인 사이의 구분을 위한 공백 구간(horizontal

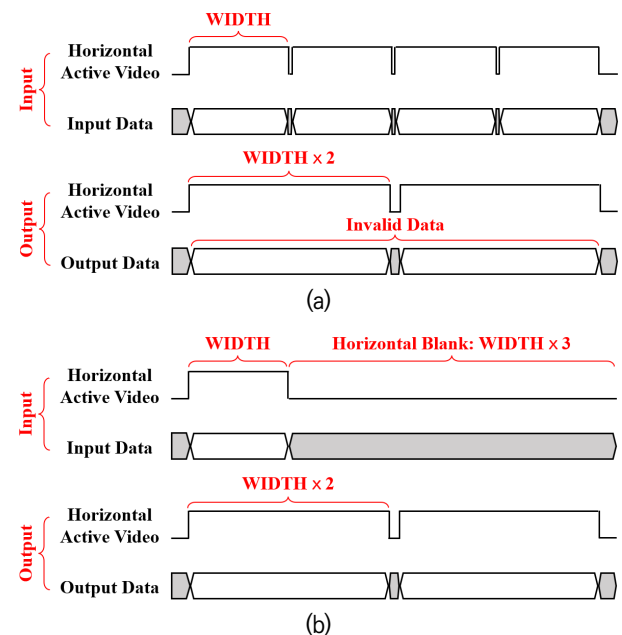


Fig. 5. Timing diagram of scaler hardware, (a) Case of insufficient time for computation, (b) Case of time allocation method.

그림 5. 스케일러 하드웨어의 타이밍도, (a) 공백 구간이 부족한 경우, (b) 시간 할당 방식의 경우

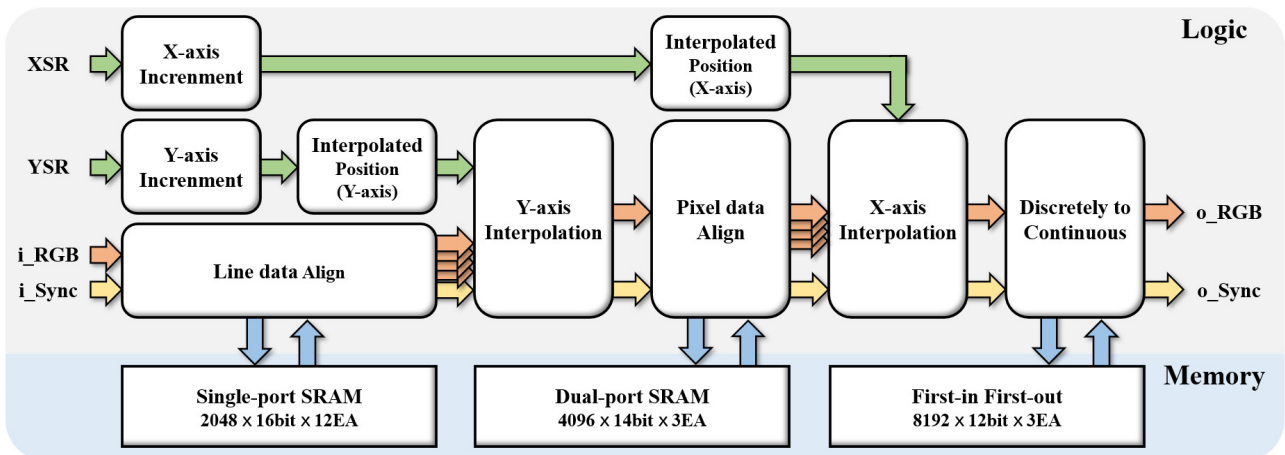


Fig. 6. Architecture of proposed scaler hardware.
그림 6. 제안하는 스케일러 하드웨어 구조

blank)을 확대 동작에서 증가된 출력 데이터의 크기만큼 할당해 출력 데이터를 생성하는 시간을 확보한다.

그림 5는 단일 클록을 사용하는 스케일러의 하드웨어에서 각 축으로 최대 2.0배까지 확대가 가능한 경우, 시간 할당 유무에 따른 타이밍도를 나타낸다. (a)는 출력 데이터 생성을 위한 충분한 시간이 부여되지 않은 상황, (b)는 시간 할당을 통해 출력 데이터를 생성할 때 필요한 시간이 보장된 상황을 나타낸다. 스케일러의 하드웨어는 확대 및 축소 동작에서 인접한 라인 혹은 픽셀 데이터를 저장한 후 보간에 활용한다. 그러나 (a)의 경우에는 보간이 진행되는 도중에 새로운 라인 데이터 및 픽셀 데이터가 입력되는 문제가 발생해 참조하는 데이터가 유효하지 않게 되어, 출력 데이터 생성에 문제가 발생한다. 이에 반해, (b)에서는 최대 확대의 경우를 기준으로 증가된 출력 데이터를 처리할 수 있는 충분한 시간을 할당함으로써 보간에 필요한 유효한 데이터를 확보할 수 있다.

III. 제안하는 스케일러 하드웨어의 구조

1. 스케일러의 하드웨어 구조

양 3차회선 보간 알고리즘을 사용하는 스케일러의 하드웨어 구현을 위해서는 각 축에서 인접한 4개의 라인 그리고 픽셀 데이터가 필요하다.

스케일러의 하드웨어는 각 축의 스케일 비율, 입력 이미지 데이터(red-green-blue, RGB) 그리고 vertical active video와 horizontal active video 신호를 포함하는 동기 신호를 입력으로 받아온다. 각 축의 스케일 비율을 나타내는 XSR, YSR은 증감값을 구할 때 사용되며,

증감값은 보간되는 라인 또는 픽셀의 위치를 구할 때 사용된다. 입력 이미지 데이터는 Y축 보간을 위해 라인메모리에 저장되며, 앞서 계산된 보간되는 라인 위치에 따라 보간을 진행한다. 이때, 채널별로 라인 데이터를 저장하기에 12개의 라인메모리를 사용한다. 보간된 라인 데이터는 다시 라인메모리에 저장되는데, 이는 X축 보간을 위해 인접한 픽셀 데이터를 저장하기 위함이다. 저장된 데이터는 보간되는 픽셀의 위치에 따라 X축 보간을 진행한다. 마지막으로, 결과 데이터는 스케일 비율에 따라 불연속적인 데이터가 생성되는 경우가 발생하기 때문에 first-in first-out을 이용해 연속적인 라인 데이터로 변환하는 과정을 진행한다.

주파수 변환 방식의 경우 입력 이미지 데이터를 받아 오는 경우 추가로 DPSRAM에 저장하는 과정이 필요하다. 따라서 제안하는 하드웨어는 주파수 변환 방식과 비교해 3개의 DPSRAM 사용을 줄일 수 있다는 장점을 가진다.

2. Pipelining 기법

양 3차회선 보간 알고리즘은 비교적 많은 곱셈 연산으로 실시간 처리를 만족하기 위한 동작 주파수를 가지는 클록의 동작에서 타이밍 위반(timing violation)이 발생한다. 따라서 타이밍 위반이 발생하는 경로에 pipeline을 추가하는 방식으로 회로의 최대 동작 주파수를 높인다[7].

그림 7은 pipeline을 추가하는 예시를 보인다. flip-flop(FF) 사이의 조합 논리회로의 연산량이 많은 경우 연산 과정을 분할하며, FF를 추가해 더 높은 동작 주파수에서 동작이 가능하게 한다. 이와 같은 pipelining 기

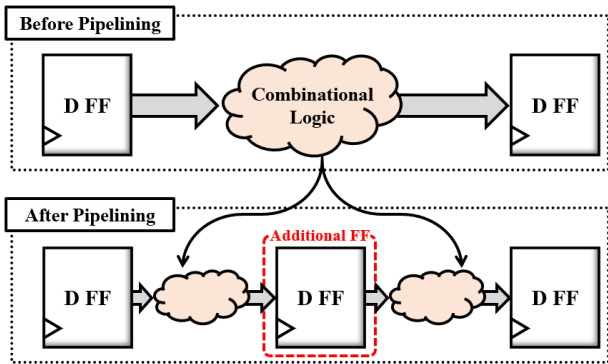


Fig. 7. Before and after applying the pipelining technique. 그림 7. 파이프라이징 기법 적용 전과 후

법을 활용해 보간 과정에서 발생하는 타이밍 위반을 방지하고 실시간 처리가 가능한 동작 주파수에서 동작이 가능하도록 하드웨어를 구현한다.

IV. 하드웨어 구현 및 성능 평가

본 논문에서 구현한 스케일러의 하드웨어는 하드웨어 기술 언어인 Verilog를 이용해 register transfer level (RTL)로 설계하였다. 설계된 RTL은 Xilinx사의 Vivado v2023.1을 이용해 FPGA 환경에서 ZCU106을 목표 보드로 구현한다[12].

표 2는 양 3차회선 보간을 사용하는 주파수 변환 방식 [5]과 이 논문에서 제안한 시간 할당 방식의 하드웨어 합성 결과를 비교한다. 표에서는 Gain을 통해 FPGA 내에서 하드웨어의 자원 사용 효율성을 평가하고 있다. Gain은 주파수 변환 방식[5]에 사용된 자원 수치를 제안된 하드웨어에서 사용된 자원 수치로 나눈 값을 나타낸다. 제안하는 시간 할당 방식은 상대적으로 적은 면적을 차지하는 자원인 Register와 LUT를 약 1%와 2% 정도 더 사용함을 볼 수 있다. 반면에, 제안한 방식이 상대적으로 큰 면적을 차지하는 자원인 block RAM을 11%씩이나 작게 사용함을 볼 수 있다. 하드웨어의 최대 동작 주파수는 두 방식 모두 약 450MHz로 거의 비슷한 수준이다. 또한 표 2를 통해 두 방식의 하드웨어가 사용하는 클록의 수를 확인할 수 있다. 주파수 변환 방식은 주파수 변환을 위한 더 빠른 클록이 필요하여 총 두 개의 클록을 사용하는 단점이 있다. 하지만, 제안한 방식은 이 단점을 극복하기 위해 시간 할당 방식을 사용하여 단일 클록으로 하드웨어를 구현하였다. 이를 통해, 다중 클록 시스템에서 발생할 수 있는 CDC 위반 문제를 해결할 수 있었다[6].

Table 2. Comparison of synthesis results by hardware architecture.

표 2. 하드웨어 구조 별 합성 결과 비교

Xilinx Vivado v2023.1						
Device	xczu7ev-ffvc1156-2					
Logic Utilization	Avaliable	Jung [5]		Proposed		Gain
		Used	Util.	Used	Util.	
CLB Register(#)	460,800	8,563	1.86%	8,649	1.88%	×0.99
CLB LUTs(#)	230,400	9,645	4.19%	9,808	4.26%	×0.98
Block RAMs(#)	312	30	9.62%	27	8.65%	×1.11
Min. period(ns)	-	2.233		2.226		-
Max. Freq.(MHz)	-	447.828		449.236		-
Used Clocks(#)	-	2		1		-

스케일러 하드웨어의 이미지 처리 속도를 의미하는 maximum processing speed (MPS)의 수치는 하드웨어의 최대 동작 주파수와 이미지를 처리할 때 요구되는 클록 사이클을 이용해 아래 식 (4)를 통해 구할 수 있다 [13].

$$MPS = \frac{f_{imax}}{(H + B_V)(W + B_H)} \tag{4}$$

식 (4)의 f_{imax} 는 입력 데이터를 처리할 수 있는 클록의 최대 동작 주파수이다. 표 2에 하드웨어 최대 동작 주파수가 표기되어 있는데, 이는 출력 데이터 기준의 동작 주파수를 나타낸다. 주파수 변환 방식은 두 개 클록을 사용하여 수평과 수직 방향으로 각각 최대 2배 확대가 가능하기에 입력 동작 주파수 기준으로 환산하면, f_{imax} 는 최대 동작 주파수의 1/4배인 111.957MHz가 된다. 제안한 방식은 단일 클록을 사용하기에 f_{imax} 는 최대 동작 주파수와 동일한 449.236MHz이다. 분모는 이미지를 처리하는 데 필요한 클록 사이클을 나타낸다. H (height)는 이미지의 수직 라인 수이고, B_V (vertical blank)는 연속적인 이미지와 이미지를 구분하기 위한 수직 공백

Table 3. Maximum processing speed for various standard images in fps.

표 3. 다양한 표준 이미지의 최대 처리 속도 (fps)

Video Resolution	Frame Size	Processing Speed (Jung [5])	Processing Speed (Proposed)
HD	1280×720	121.22fps	121.65fps
FHD	1920×1080	53.91fps	54.10fps
QHD	2560×1440	30.34fps	30.44fps
DCI 4K	4096×2160	12.65fps	12.69fps

구간이다. W (width)는 이미지의 수평 구간 픽셀 수이고, B_H (horizontal blank)는 연속적인 라인과 라인 사이를 구분하기 위한 수평 공백 구간이다.

표 3은 수식 (4)를 이용해 주파수 변환 방식과 제안한 방식을 하드웨어로 구현하였을 때, 네 종류의 표준 이미지 처리 가능 속도를 보여준다. QHD 표준 이미지의 처리 속도는 두 종류 하드웨어 모두 W 는 2560, H 는 1440 그리고 B_V 는 1이다. 그러나 B_H 는 주파수 변환 방식에서는 1이지만, 그림 4(b)에 나타난 것처럼 제안한 방식에서 B_H 는 최대 확대의 경우 출력 데이터를 생성하기 위한 시간과 증가된 라인 데이터를 구분하기 위한 공백으로 3배수 W 값에 2를 더한 값인 7682이다. 이의 결과를 표 3에서 볼 수 있는데, 두 방식 모두 QHD 표준 이미지를 실시간 처리 조건인 30fps 이상을 만족하는 것을 알 수 있다[14].

V. 결론

본 논문에서는 시간 할당 방식을 적용한 스케일러 하드웨어를 ZCU106 FPGA 보드에서 구현하였다. 설계된 하드웨어는 양 3차회선 보간 알고리즘을 사용하여, 하드웨어 크기 최적화를 고려한 조합 보간 알고리즘과 PSNR 수치를 비교하였을 때 전체적으로 더 높은 수치를 나타냈다. 또한 제안한 방식은 단일 클럭을 사용하여 주파수 변환에 필요한 추가적인 클럭과 큰 용량의 라인 메모리를 사용하지 않고 하드웨어의 구현이 가능하다. QHD 표준 이미지에 관한 실험 결과, 제안된 하드웨어는 실시간 처리 기준을 충족하였으며, 주파수 변환 방식과 유사하거나 소폭 더 높은 처리 속도를 나타내었다. 이러한 결과를 통해 제안된 스케일러 하드웨어의 우수성을 입증하였다.

References

[1] J. Dan, Z. Qu, X. Wang, and J. Gu, "A Disparity Feature Alignment Module for Stereo Image Super-Resolution," *IEEE Signal Processing Letters*, vol.28, pp.1285-1289, 2021.
DOI: 10.1109/LSP.2021.3088050

[2] A. N. A. Rahim, S. N. Yaakob, R. Ngadiran and M. W. Nasruddin, "An analysis of interpolation methods for super resolution images," *2015 IEEE*

Student Conference on Research and Development, pp.72-77, 2015.

DOI: 10.1109/SCORED.2015.7449432

[3] Y. Li, F. Qi and Y. Wan, "Improvements On Bicubic Image Interpolation," *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp.1316-1320, 2019.

DOI: 10.1109/IAEAC47372.2019.8997600

[4] S. Han, S. Jung, and B. Kang, "Analysis of the Image Processing Speed by Line-Memory Type," *Journal of IKEEE*, vol.27, no.4, pp.127-133, 2023.
DOI: 10.7471/ikeee.2023.27.4.494

[5] S. Jung, S. Han, and B. Kang, "Hardware Design of Bicubic Scaler Using Frequency Conversion Method," *Journal of KIIT*, vol.22, no.7, pp.75-81, 2024. DOI: 10.14801/jkiit.2024.22.7.75

[6] Y. Lee, N. Kim, J. Kim, and B. Min, "Millions to thousands issues through knowledge based SoC CDC Verification," in *IEIE ISOC*, pp.569-572, 2012. DOI: 10.1109/ISOC.2012.6407123

[7] D. Ngo, and B. Kang, "Taylor-Series-Based Reconfigurability of Gamma Correction in Hardware Designs," *Electronics*, vol.10, no.16, 2021.
DOI: 10.3390/electronics10161959

[8] S. Han, S. Jung, J. Son, and B. Kang, "Hardware Design for Real-Time Processing of a Combinatorial Interpolation Scaler width Asymmetric Down-scaling and Up-scaling," *Journal of IKEEE*, vol.28, no.1, pp.26-32. 2024.

[9] U. Sara, M. Akter, and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR-A Comparative Study", *Journal of Computer and Communications*, vol.07, no.03, pp.8-18, Mar. 2019. DOI: 10.4236/jcc.2019.73002

[10] S. Han and B. Kang, "A Study of the Combinatorial Interpolation Algorithm for Scaler Hardware Design," *Journal of IKEEE*, vol.27, no.3, pp.296-302, 2023.
DOI: 10.7471/ikeee.2023.27.3.296

[11] H. Moon and S. Ban, "VLSI Architecture of Digital Image Scaler Combining Linear Interpolation and Cubic Convolution Interpolation", *Journal of*

Korea Multimedia Society, vol.51, no.3, pp.112-118, Mar. 2014. DOI: 10.5573/ieie.2014.51.3.112

[12] Xilinx, "ZCU106 Evaluation Board: User Guide (UG1244)," <https://docs.xilinx.com/v/u/en-US/ug1244-zcu106>

[13] Ngo. D, S. Lee, U. J. Kang, T. M. Ngo, G. D. Lee, and B. Kang, "Adapting a Dehazing System to Haze Conditions by Piece-Wisely Linearizing a Depth Estimator," *Sensors*, vol.22, no.5, pp.1957-1981, 2022. DOI: 10.3390/s22051957

[14] K. Jack, *Video Demystified: A Handbook for the Digital Engineer*, 4th edition, Newnes, 2005.

BIOGRAPHY

Semin Jung (Member)



2023 : BS degree in Electronics Engineering, Dong-A University.
2023~present : MS degree course in Electronics Engineering, Dong-A University.

Si-Yeon Han (Member)



2023 : BS degree in Electronics Engineering, Dong-A University.
2023~present : MS degree course in Electronics Engineering, Dong-A University.

Dat Ngo (Member)



2016 : BS degree in Computing Engineering, Danang University of Science and Technology.
2018 : MS degree in Electronics Engineering, Dong-A University.
2022 : Ph.D degree in Electronics Engineering, Dong-A University.

2022~2023 : Postdoctoral Researcher, Dept. of Electronics Engineering, Dong-A University.

2023~2024 : Research Professor, Dept. of Biomedical Informatics, CHA University.

2024~present : Assistant Professor, Dept. Computer Engineering, Korea National University of Transportation.

Bongsoon Kang (Member)



1985 : BS degree in Electronics Engineering, Yonsei University.
1987 : MS degree in Electrical Engineering, University of Pennsylvania.
1990 : Ph.D degree in Electrical Engineering, Drexel University.

1989~1999 : Senior Staff Researcher, Samsung Electronics.

1999~present : Prof. of Dept. Electronics Engineering, Dong-A University.