인공지능 컴퓨팅 환경 확보 방안 및 전략

2016. 08. 25.
2016 정보과학회 HPC연구회 하계 워크샵

추 형 석

소프트웨어정책연구소 선임연구원
신기술확산연구팀

소프트웨어 중심사회의 Think Tank SPRi Software Policy & Research Institute

목 차

1. 연구 목적

연구 목적

● 배경 및 필요성

- 컴퓨팅 환경 확보는 인공지능 연구를 위해 선결되어야 하는 과제
- 인공지능 연구에 왜 “컴퓨팅 파워”가 중요한지에 대한 논리적 근거 마련
  - 최신 컴퓨팅 하드웨어에 대한 현황과 분석
  - 구체적인 인공지능 성공사례 분석 (딥러닝)

● 인공지능 컴퓨팅 환경 확보 전략 연구

- 중소기업, 스타트업, 대학의 인공지능 연구 활성화를 위한 컴퓨팅 환경 확보 방안
- 국내외 클라우드 GPU instance 분석

2. 컴퓨팅 파워와 병렬 컴퓨팅

계산성능의 척도 - 부동소수점 연산수

- **부동소수점 연산수 (floating-point operations, flop)**
  - **알고리즘을 실제로 구현했을 때 필요한 연산수를 나타냄**
  - **한 개의 연산은 일반적으로 덧셈, 곱셉, 비교로 간주하나, 계산자원 구조에 따라 덧셈과 곱셈을 하나의 연산으로 보기도 함**
    - FMA(Fuzed Multiply and Add)는 곱셈과 덧셈을 한 번에 처리하는 유닛
  - **유효숫자(Precision)에 따른 성능차이가 존재**
    - 32-bit 부동소수점(float, 유효숫자 7자리)과 64-bit 부동소수점(double, 유효숫자 16자리)에 대한 연산성능이 다름 (일반적으로 float에 대한 성능이 높음)

- **초당 부동소수점 연산수 (floating-point operations per second, FLOPS or FLOP/s)**
  - **연산처리장치의 연산능력을 표현하는 지표로 슈퍼컴퓨터의 성능비교 등에 사용됨**
  - **2016년 6월 세계에서 가장 빠른 슈퍼컴퓨터의 성능은 93 PetaFLOP/s**
    - (중국 국립 슈퍼컴퓨터센터) 선웨이 타이후라이트 : 10,649,600 cores

※ (한국 기상청, 36위) 누리 : 2.4PetaFLOPS, 69,600 cores

AP와 슈퍼컴퓨터

Samsung Galaxy S7

Processing Power
AP : Exynos 8890 (4+4 core, 2.3+1.6GHz)
GPU : Mali-T880 MP12 (265.2 GFLOPS)

Top500.org Performance Development

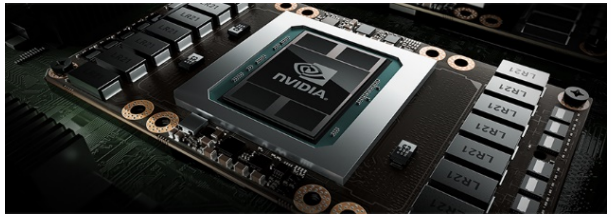
1996/06 1st supercomputer
University of Tokyo SR2201/1024 ($50M)
Peak Performance (220.4 GFLOPS)

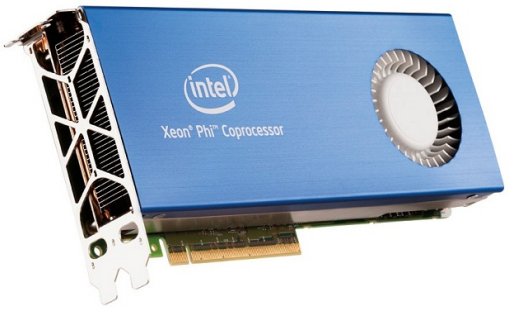
계산자원의 종류

CPU
(Central Processing Unit)

GPU

Accelerator





Intel Xeon Processor E5-2699 v4

22 cores (hyper-threading 44cores)
Clock Speed : 2.2GHz(Turbo 3.6GHz)
Price : $4,115
Performance : 1549 GFLOPS (single)
744 GFLOPS (double)

NVIDIA Tesla P100

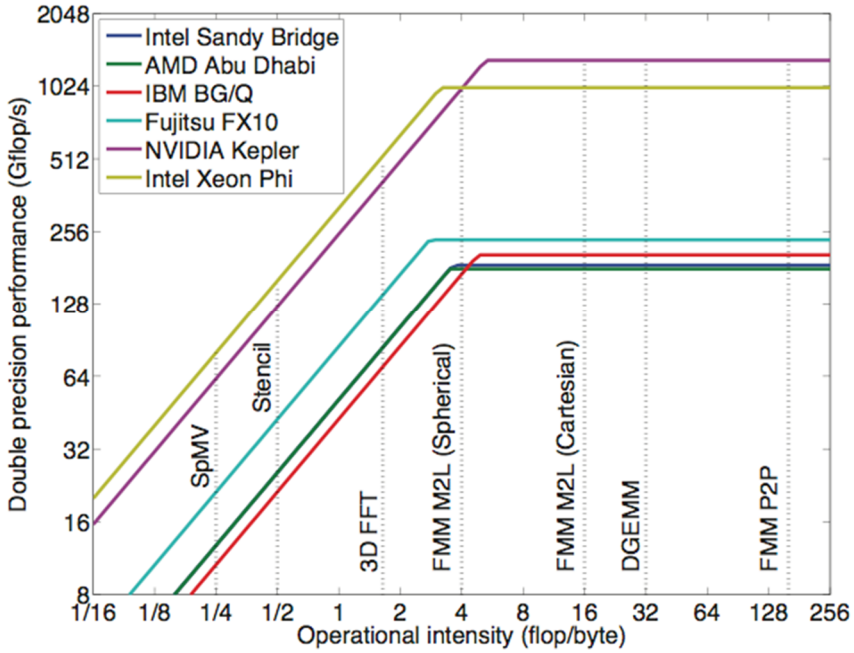
3584 cores
Clock Speed : 1.3GHz (Turbo 1.4GHz)
Price : TBA (around $5000)
Performance : 10608 GFLOPS (single)
5304 GFLOPS (double)
* NVlink : 160 GB/s (CPU-GPU)

Intel Xeon Phi 7120P

61 cores (244 threads)
Clock Speed : 1.3GHz
Price : $4,129
Performance : 2416 GFLOPS (single)
1208 GFLOPS (double)

GPU의 성능

- **Roofline model**
  - 계산량과 메모리전송량을 대비하여 달성할 수 있는 성능을 나타냄
  - 일반적으로 계산성능(clock speed)이 메모리 대역폭(memory bandwidth)보다 높음
  - 계산 강도(arithmetic intensity)가 높을수록 효율이 증대



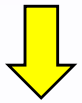
- **알고리즘의 병렬화**
  - 병렬화가 불가능한 알고리즘은 many-core 기반 계산자원에서 성능이 급격히 저하
  - HW 아키텍쳐에 따라 병렬화의 효율이 결정

- **이론성능은 달성하기 어려움**

알고리즘의 병렬화

- for loop의 병렬화

```
// Simple saxpy operation
for(j = 0 ; j < n ; j++)
    y[j] = alpha * x[j] + y[j];
```



Parallelization

```
// idx is assigned randomly from 0 to n
idx = myid;
y[idx] = alpha * x[idx] + y[idx];
```

Memory : 4*(2*n+1) byte
Computation : n operations
Arithmetic Intensity : 1/8

- 피보나치 수열의 경우 병렬화?

```
// Fibonacci series
for(j = 2 ; j < n-1 ; j++)
    x[j] = x[j-2] + x[j-1];
```

행렬 곱 (Matrix Multiplication)

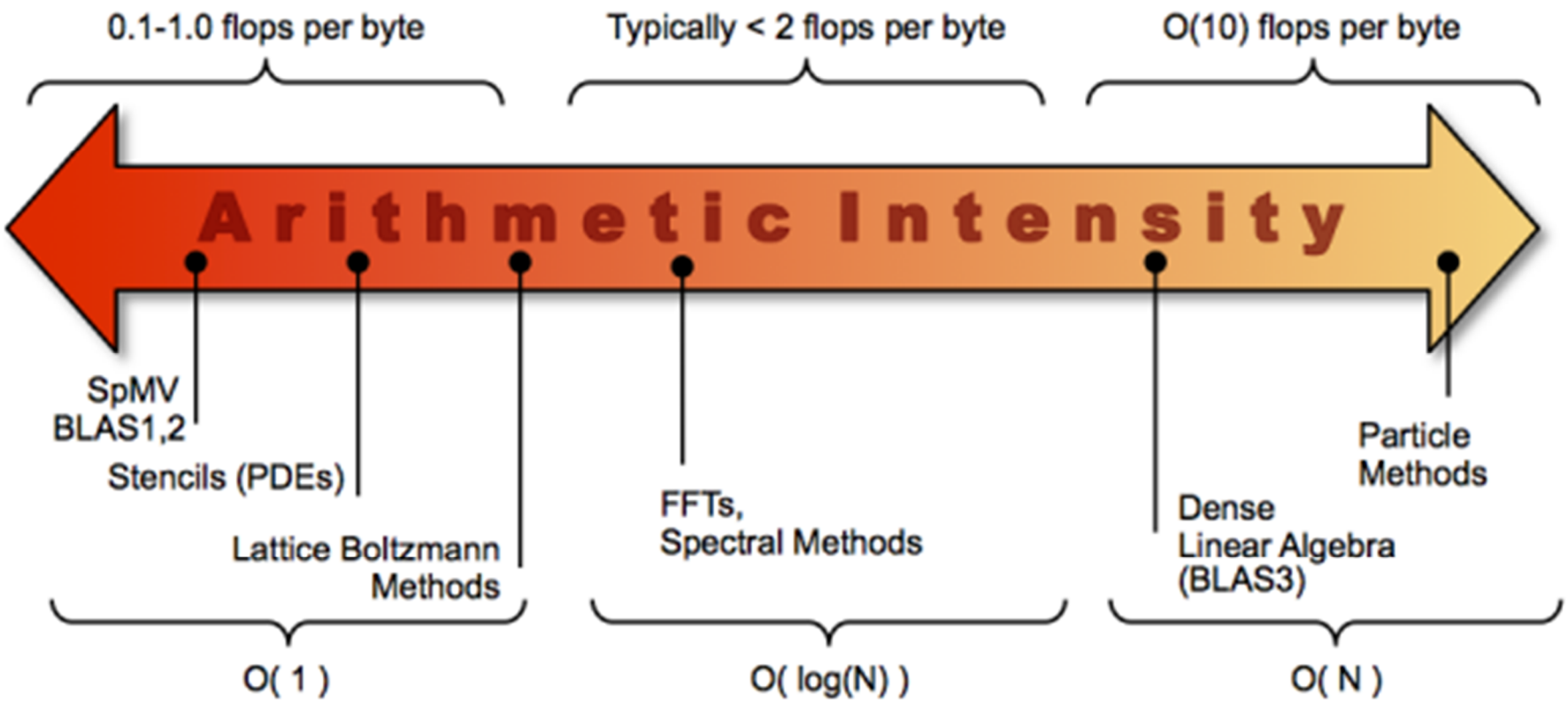
- **BLAS 3 – SGEMM**

```
// C = A*B where each matrix is n by n matrix

for(j = 0 ; j < n ; j++)
   for(k = 0 ; k < n ; k++)
      for(l = 0 ; l < n ; l++)
         C[j][k] += A[j][l] * b[l][k]
```

- **Arithmetic Intensity**
  - 메모리 전송량 : $4 \times 4byte \times n^2 = 12n^2$
  - 계산량 : n^3
  - Arithmetic Intensity : $n/12$
  - 행렬이 커질수록 이론성능에 가까워짐

알고리즘 별 Arithmetic Intensity



Source : https://crd.lbl.gov/departments/computer-science/PAR/research/roofline/

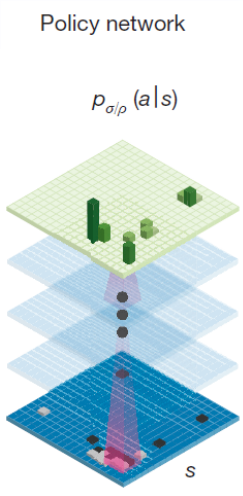
3. AlphaGo 계산량 분석

AlphaGo의 인공지능 알고리즘 분석

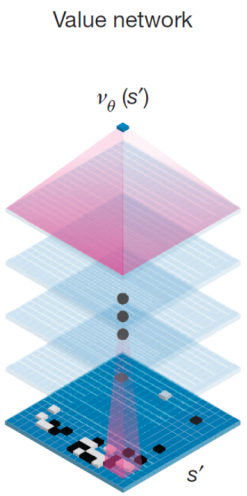
Source : https://spri.kr/post/14725

AlphaGo (1/2)

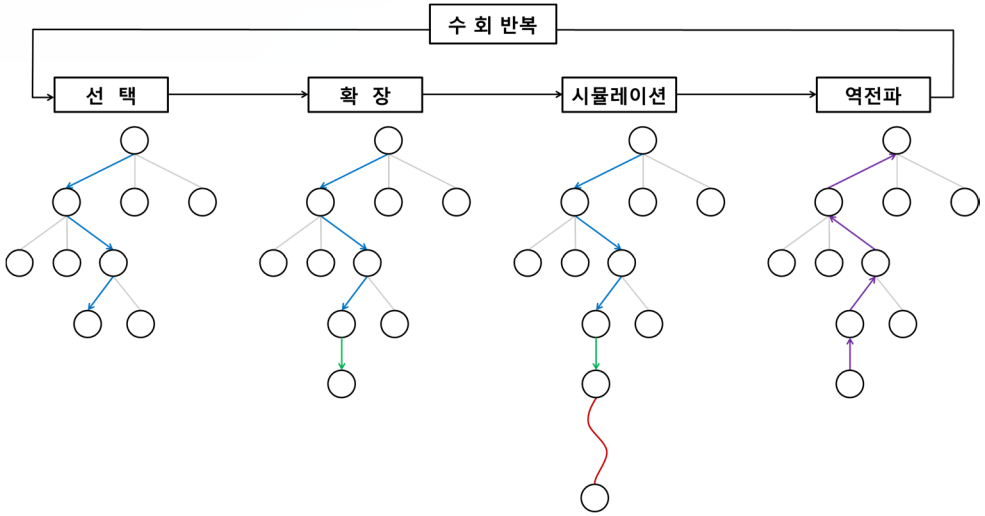
- **바둑 세계챔피온을 꺾은 최초의 인공지능 바둑프로그램**
  - **딥러닝으로 바둑 프로기사의 기보 16만 개를 학습**
  - **무한대에 가까운 바둑의 경우의 수를 프로바둑기사의 관점으로 좁힘**
    - 정책 네트워크 : 프로바둑기사들의 착수 선호도 + 스스로 대국하여 튜닝
    - 가치 네트워크 : 현재 바둑판 상태의 승률을 근사
  - **정책과 가치네트워크를 활용한 경로 탐색으로 최적의 수를 결정**
    - 몬테카를로 트리 탐색(MCTS) 알고리즘 활용



정책 네트워크



가치 네트워크

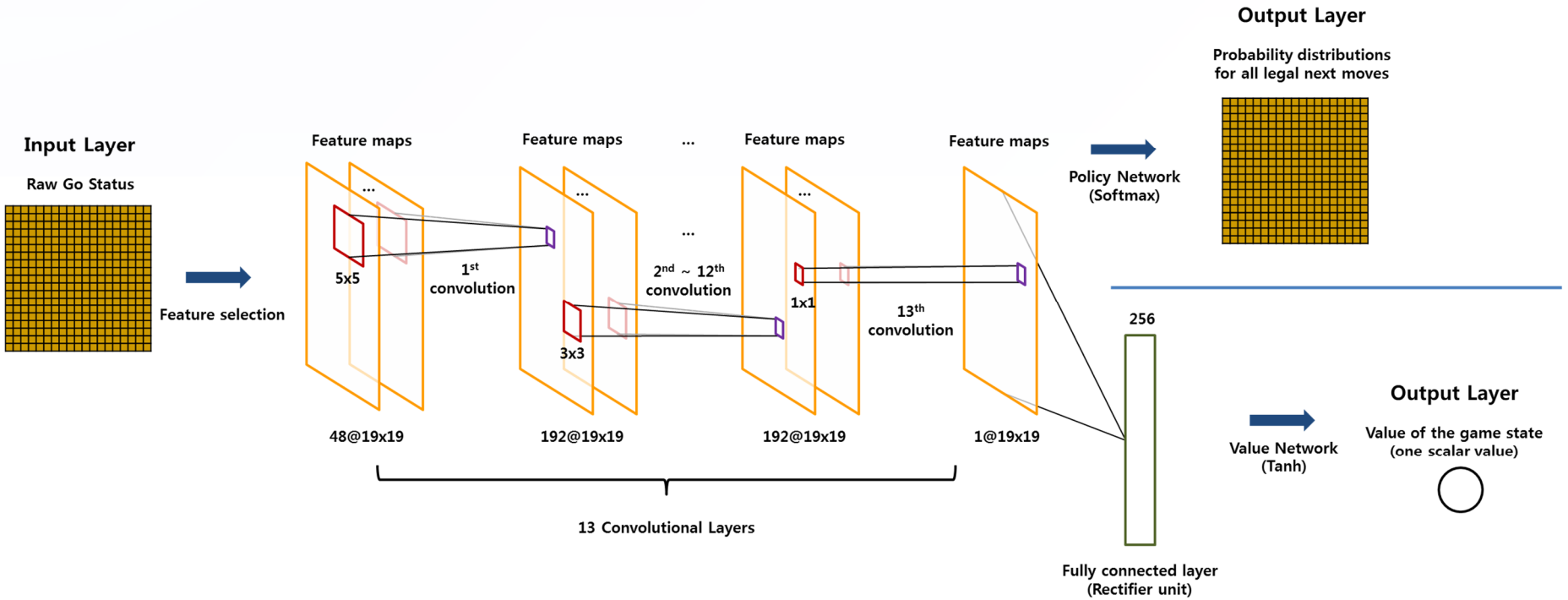


MCTS 알고리즘

Source: Mastering the game of Go with deep neural networks and tree search, Nature

AlphaGo (2/2)

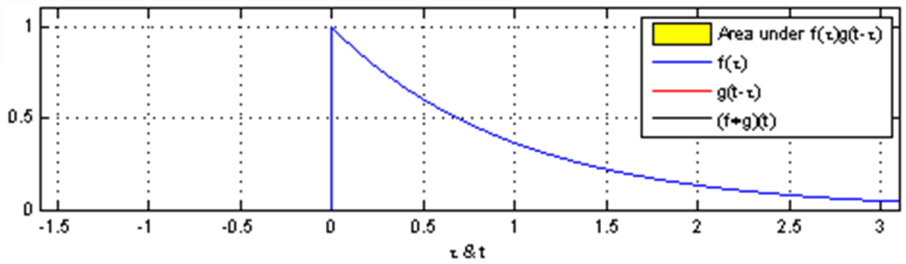
- **AlphaGo의 딥러닝 구조 – 콘볼루션 뉴럴네트워크**
  - **콘볼루션 뉴럴네트워크는 이미지를 학습하는데 탁월한 성능을 가짐**
    - 이미지의 국지적인 패턴을 인식하여 전체를 재구성
    - AlphaGo에서는 바둑판 상태를 48가지 특징 맵으로 전환하여 국지적 형세를 판단함
  - **13층의 콘볼루션층을 활용하여 프로기사들의 기보를 성공적으로 학습**



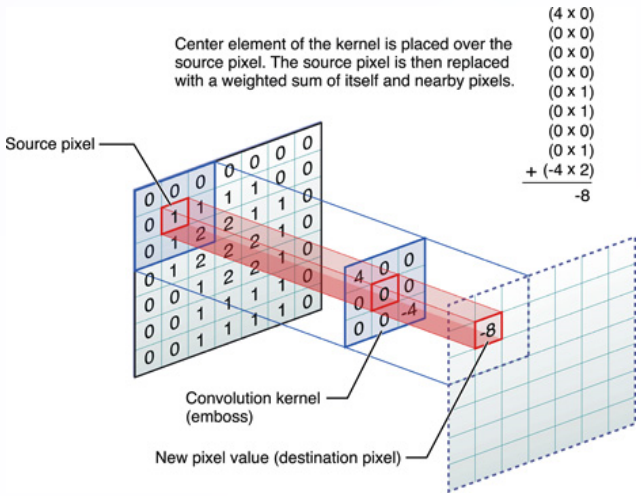
컨볼루션 신경망
(Convolutional Neural Network, CNN)

- 이미지 분석에 특화된 딥러닝 기법
  - '컨볼루션 필터'를 학습
- 컨볼루션 이란?

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$



Source: https://en.wikipedia.org/wiki/Convolution

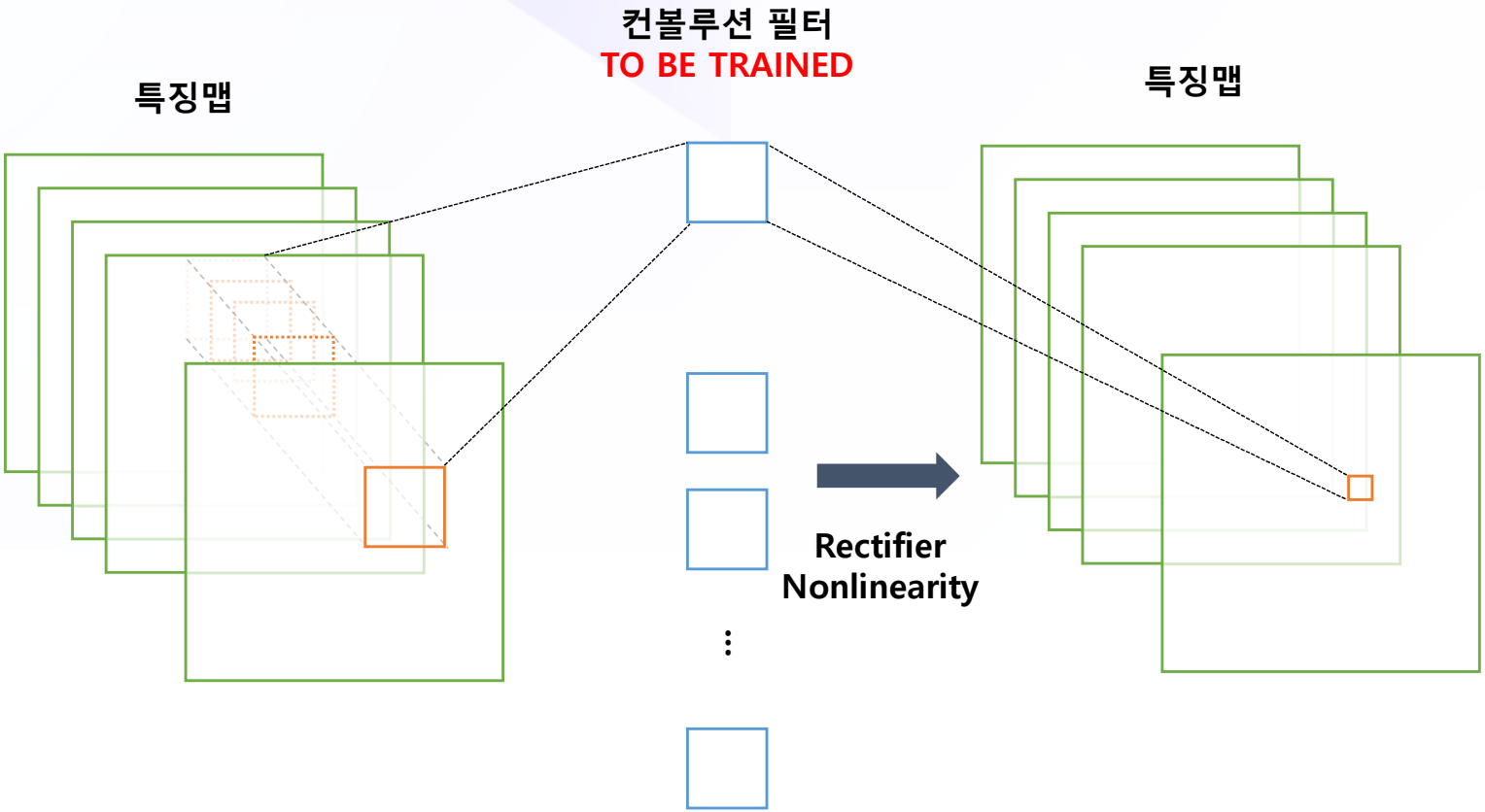


Source : iOS Developer Library – vImage Programming Guide
https://developer.apple.com/library/ios/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html

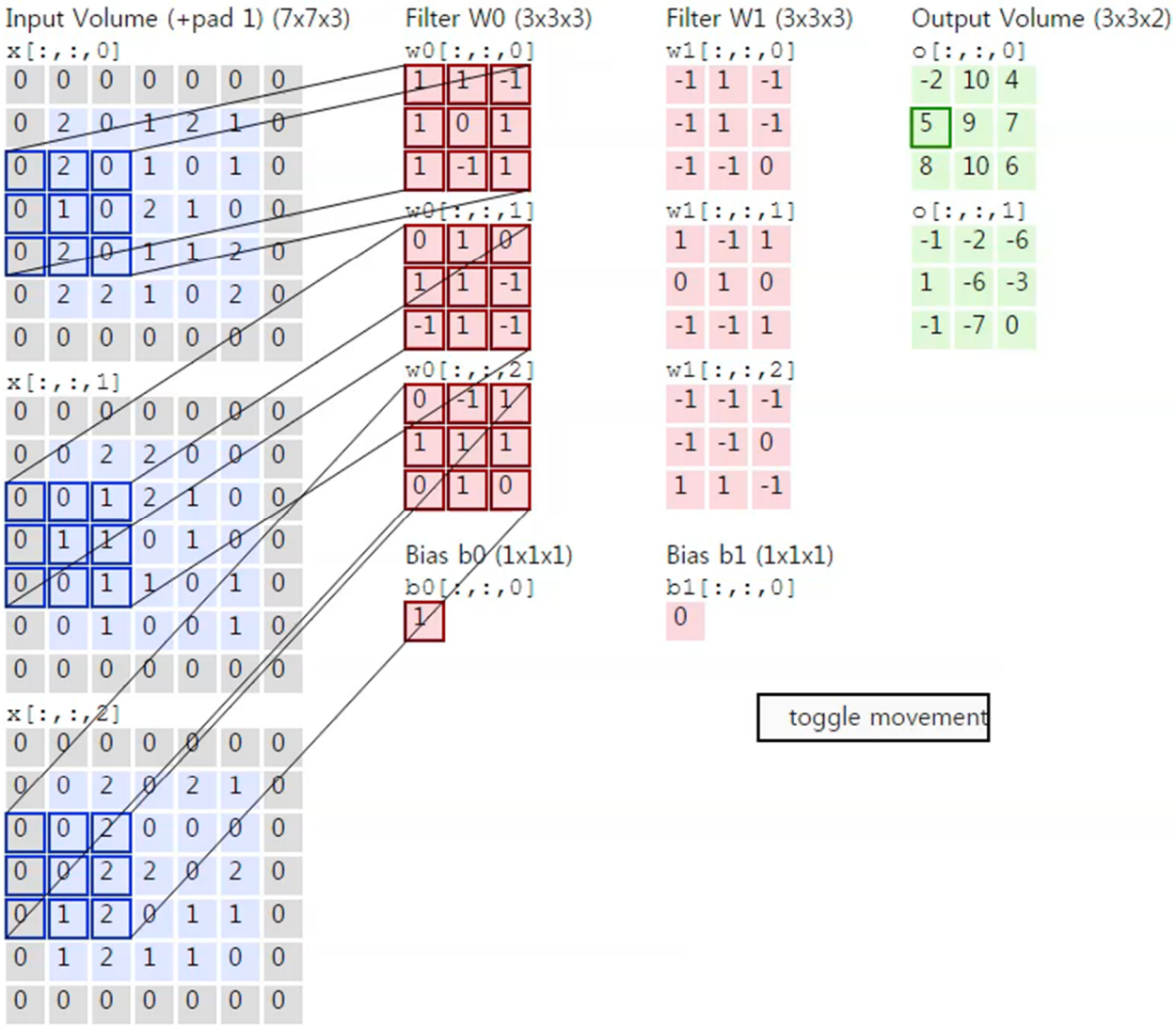
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	

Source: https://en.wikipedia.org/wiki/Kernel_(image_processing)

컨볼루션 신경망 – 컨볼루션 층

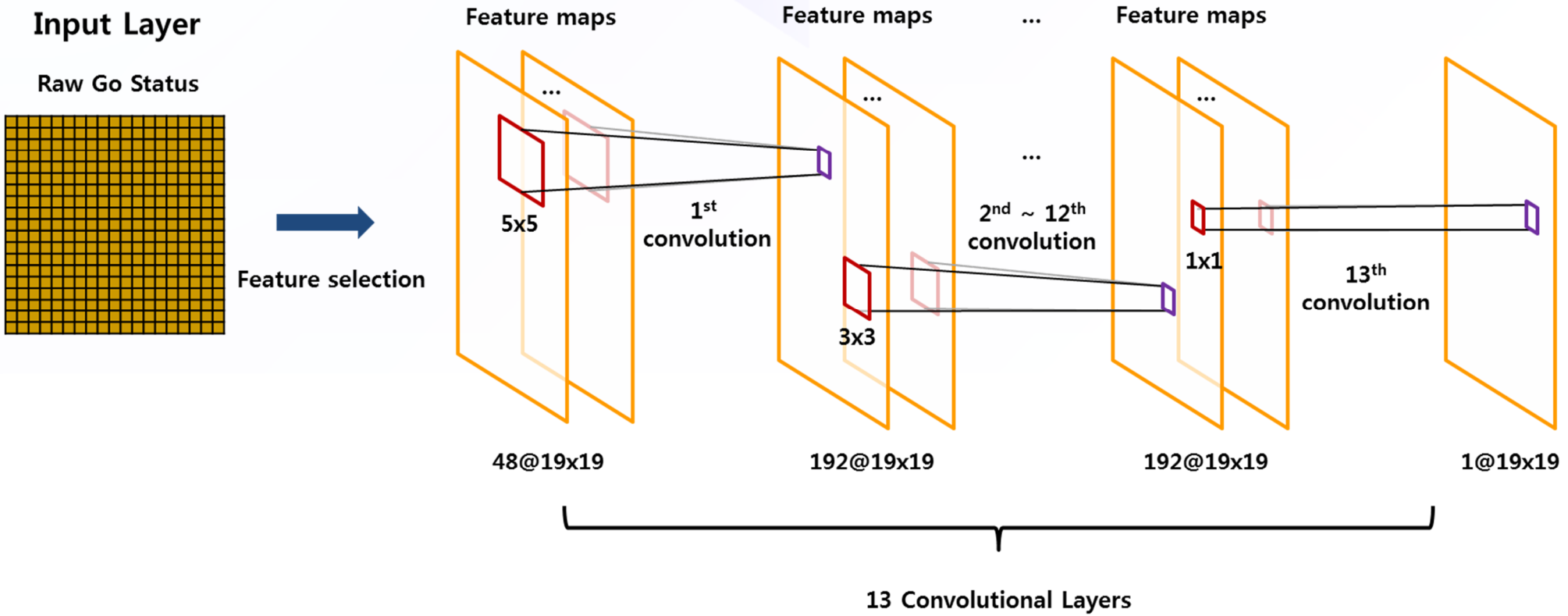


컨볼루션의 과정

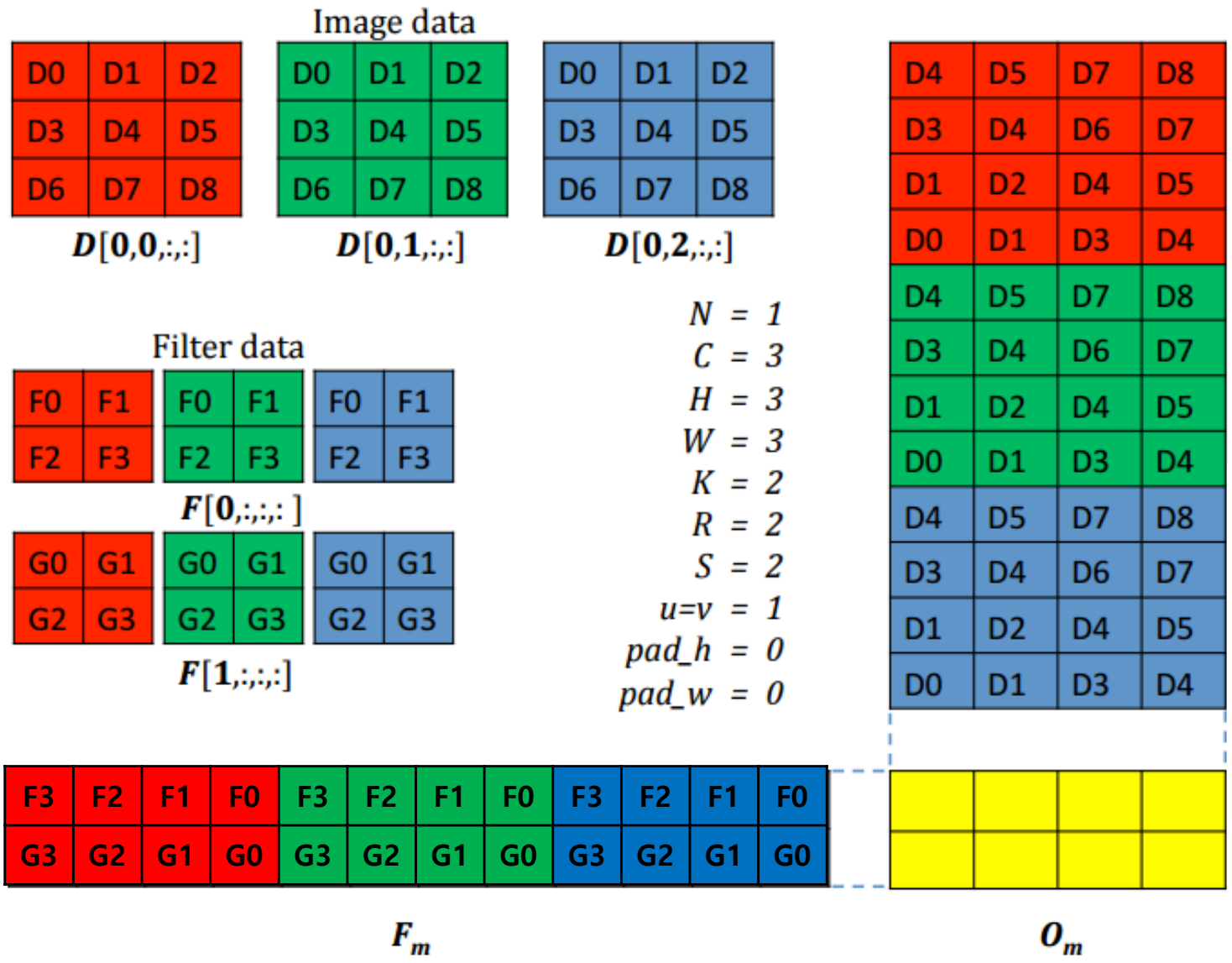


Source : http://cs231n.github.io/convolutional-networks/

AlphaGo의 컨볼루션 신경망 구조

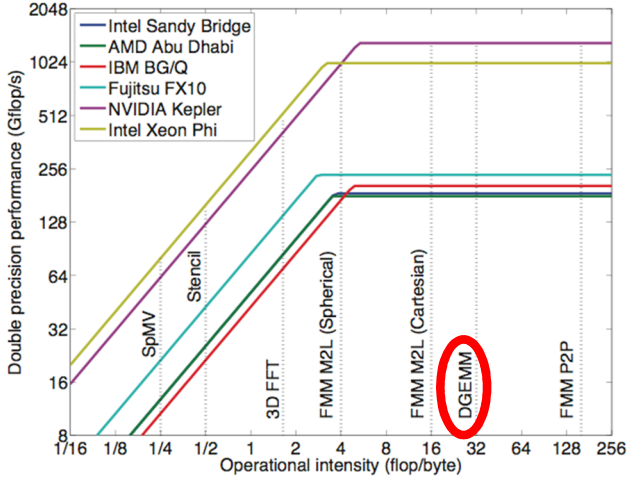


GPU와 컨볼루션 (GEMM)



AlphaGo

Fm : 192 x 9
D : 9 x 69312



AlphaGo 인공신경망 계산량

- **ConvNet Inference (estimation)**
  - 1층 : 19x19 바둑판 * 48개 특징맵 * 5x5 콘볼루션 * 25 덧셈 * 192개 필터 * 2개 연산 (활성함수 계산) = 4.159 GFLOP
  - 2~13층 : 19x19 바둑판 * 192개 특징맵 * 3x3 콘볼루션 * 9 덧셈 * 192개 필터 * 2개 연산 (활성함수 계산) * 11층 = 23.715 GFLOP
  - 약 30 GFLOP (학습 시는 반복 한 번에 필요한 계산량)
  - NVIDIA K40 GPU의 cuDNN(딥러닝 패키지)의 성능은 1.2 TFLOP/s
  - GPU 한 개당 1초에 약 40번의 inference 가능

- **Memory**
  - 16만 개의 기보 : 1.85 Tbyte(single), 58 Gbyte(boolean)
  - 13층의 ConvNet weights : 약 3Mbyte

AlphaGo의 (테스트) 계산자원

CPU

- **Intel Xeon CPU E5-2643 v2 @ 3.5 GHz**

Source : http://www.amazon.com/HP-712775-L21-E5-2643-3-5GHz-Processor/dp/B00PYTVVWI

- 코어수/스레드수 : 6 cores / 12 threads
- 성능 : 66.61 GFLOP/s
- 최대 CPU 구성 : 2
- 가격 : $ 1552
- 발매일 : Q3′ 2013

GPU

- **GeForce GTX Titan Black**

Source : http://www.nvidia.co.kr/gtx-700-graphics-cards/gtx-titan-black/

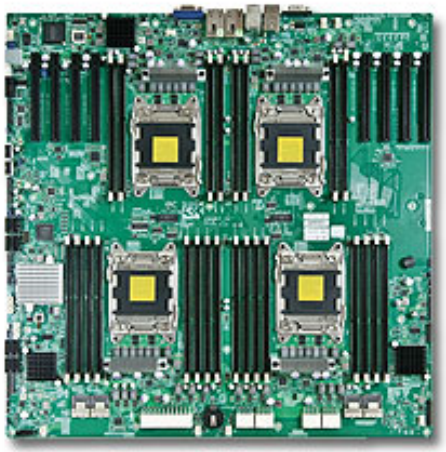
- 코어수 : 2880 cores
- 성능 : 5.1 Tera FLOP/s (single),
  1.7 Tera FLOP/s (double)
- 가격 : $ 999
- 발매일 : March 25, 2014

Source : Maddison, Chris J., et al. "Move evaluation in go using deep convolutional neural networks." *arXiv preprint arXiv:1412.6564* (2014).
CPU Performance, https://setiathome.berkeley.edu/cpu_list.php
List of NVIDIA Graphics Processing Units, https://en.wikipedia.org/wiki/List_of_Nvidia_graphics_processing_units

AlphaGo의 계산자원 (추정)

● 싱글 머신

- **CPU cores : 48 개**
  - 12cores(with HTT) x 4 CPUs, or 8cores x 6 CPUs
- **GPU 개수 : 8 개**
- **노드 구성은 (4 CPU sockets + 8 PCIes)를 탑재한 고성능 계산서버로 추정**
  - or (6 CPU sockets + 8 PCIes)
- **가격은 약 5만불 정도이고 시간당 소비전력은 2500 Watt 수준**



Supermicro MB
4CPUs + 4PCIe + (4PCIe)
$1,278

● 분산 머신

- **CPU cores : 1202 개 (최대1920)**
- **GPU 개수 : 176 개 (최대280)**
- **약 40대 내외의 싱글머신으로 구성**
  - 한화 약 **22 ~ 25억 원**

8 VGAs 예시

인공지능과 컴퓨팅 파워

● 인공지능 연구에 왜 컴퓨팅 파워가 중요한가?

- **인공신경망 학습에 필요한 계산량이 막대함**
  - 경험적으로 추정 가능한 hyper-parameter 존재 (여러 번 시도하는 것이 최상책)
- **AlphaGo의 인공신경망 학습은 50개의 GPU를 사용하여 3주 동안 학습함**
  - 약 5MWh, 가정집에서 소모할 경우 누진세가 적용되어 약 330만 원의 전기요금 소요

● GPU는 인공지능 연구에 최적화된 장비

- **인공신경망 학습 방법인 오류역전파법(Error Backpropagation Method)는 기본적인 선형대수루틴(Basic Linear Algebra Subroutines)으로 이루어짐**
- **BLAS는 GPU에 최적화된 라이브러리 중 하나**

● 국내 인공지능 연구 활성화를 위한 컴퓨팅 환경 확보 전략

- **가성비가 좋은 GPU 컴퓨팅 환경을 확보할 필요성이 있음**

4. 결 론

클라우드 GPU instance 조사 및 분석

● 인공지능 연구를 위한 컴퓨팅 환경 조성 : 클라우드

- GPU 클러스터 구성시 하중, 전력공급, 쿨링 등 설계상의 많은 제약이 존재
- 또한 GPU의 교체주기가 매우 짧음
- 따라서 직접 구축하는 것 보다 클라우드 형태로 계산하는 것이 효율적

● 클라우드 GPU 서비스 관련 동향

- 해외 글로벌 IT 기업을 필두로 GPU instance 보급

	세부 내용
Amazon AWS G2	- CPU : Intel Xeon E5-2670 - GPU : NVIDIA GRID K520 (1,536 core, 4GB GDDR) - 클러스터 네트워킹 지원 - GRID GPU는 CAD와 같은 3D작업에 적합
SOFTLAYER IBM Cloud	- CPU : Intel Xeon E5-26xx - GPU : NVIDIA Tesla M60, K80 지원
Aliyun	- CPU : Intel Xeon E5-26xx - GPU : NVIDIA Tesla M40, K40 지원 * 중국내 서비스만 가능

- 국내에는 SK C&C가 SOFTLAYER측과 협력하여 데이터센터 오픈예정 ('16.9월 중)

향 후 계 획

- **컴퓨팅 환경 확보 관련 현황 조사**
  - Compute Canada
  - 기타 해외 선진 사례 분석
  - 국내 중소기업 및 스타트업의 수요조사

- **클라우드 GPU instance 조사 및 분석 (계속)**
  - 국내외 현황 파악

- **인공지능 컴퓨팅 환경 확보방안 제시**