



The 2nd Winter School in Imaging Science / 9-11 January 2017

<https://winterschool2017.mediviewsoft.com/>

Tutorial on Deep Learning: CNN



장재성, 김부권, 이성민 / 연세대학교 계산과학공학과 서진근교수 연구팀



MEDIVIEWSOFT, located at CSE department in Yonsei University, aims to provide networking opportunities for transparent communications on recent developments, emerging issues and challenging problems in mathematic-based medical imaging area. Mediviewsoft helps its members launch venture-ready startups. and regularly organize events and meetings to pitch member's venture idea. Haeeun Han <http://mediviewsoft.com/>



의료영상분야에서 딥러닝

서 진근 / 연세대학교 계산과학공학과

최근 수년간 딥러닝기술의 획기적인 개선은 고성능 GPU 컴퓨팅과 빅데이터의 분석/활용 기술의 급속한 발전, 그리고 구글, 페이스북, IBM 등의 적극적인 투자 때문인 듯하다. 의료영상 판독에서 딥러닝 기술은 Training data가 쌓여감에 따라 정확도가 급속히 향상되어가고 있어, 전 세계적으로 많은 인공지능기업들이 의료분야에 사업화를 시도하고 있다. 구글은 딥러닝 기술을 당뇨성 망막변증 진단에 적용하였고, IBM은 인공지능 왓슨을 암 치료에 활용했고, 애플은 헬스케어 휴대용기기와 기존의 의료시스템의 통합을 통해 지능형 의료서비스를 제공하려 하며, 삼성메디슨도 딥러닝 기술을 초음파영상의 판독에 접목하려 시도하고 있다.

딥러닝에 관한 수많은 언론보도와 논문에도 불구하고, 의료현장에서는 컴퓨터 자동 진단 시스템 (Computer Aided Diagnosis)처럼 아주 제한적인 비즈니스 활용에만 그칠 것이라 판단하는 의사가 많다. 이러한 불신은 지난 20년간 세계적인 과학자나 공학자들의 수많은 연구 결과들이 실제 의료현장에서 비즈니스 모델로 성공한 사례가 극히 드물기 때문이다. 의료영상기술은 1970-80년대에 CT, MRI, 초음파의 사업화와 함께 급속한 발전을 하였으나, 2000년 이후부터는 여타의 순수기초 학문처럼 연구자체에만 열중한 나머지 (의료현장에서의 여러 제한조건을 정확히 이해하지 못한 채) 현학적인 논문위주로 성장하는 바람에 대부분 결과가 사업화로 연결되지 못했다. 1990년대 잠깐 각광을 받은 인공지능이 급격히 열기가 식은 이유는 실전적인 개발능력을 갖추지 못한 채 연구논문과 코딩에만 집중했기 때문이다. 학문적인 신뢰를 얻기 위해서라도 실전적인 연구는 필수적이며, 이를 통해 학문적인 연구가 국내 의료산업의 발전에 기여함을 보여주어야 한다.

인간이 퓨마를 100m 경주에서 이길 수 없듯이, 의료현장에서 딥러닝은 극복하기 어려운 한계를 가지고 있다. 기계학습에 필요한 Training Data는 여러 의사들의 판독에 의해 수집되며, 기계학습능력은 다수의 판독을 반영하는 training data를 기반으로 형성되기에 근본적으로 소수의 전문의에 의해서만 발견되는 병은 무시되어진다. (※ 물론, 이러한 소수 명의의 판독을 스마트하게 반영하는 알고리즘을 데이터의 지능적인 관리를 통해 만들 수는 있지만, 일관성을 갖추기는 근본적으로 어렵다.) 따라서, 인공지능은 의료진의 시간과 노력을 절감하는 수단은 될 수 있지만, 숙련된 전문의를 대체하는데 한계가 있다.

이번 겨울학교 tutorial lecture “의료영상에서의 딥러닝”에서는 기계학습 분야를 시작하시는 연구자들을 대상으로 하여, 실전적인 비즈니스 모델을 바탕으로 기초개념을 설명하는데 중점을 두었다. 이 강연을 준비하기 위해 고생한 장재성, 김부권, 이성민 (연세대 계산과학공학과 박사과정)에게 감사 드린다.

Part 1-1 : Deep learning in medicine

Automated Fetal Biometry using Deep Learning

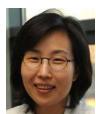
Samsung Applies Deep Learning Technology to Diagnostic Ultrasound Imaging

Korea on April 21, 2016

SHARE



서진근 (연세대학교 계산과학공학과)



Jaeseong Jang Ph.D

Bukweon Kim Ph.D

Sung Min Lee Ph.D

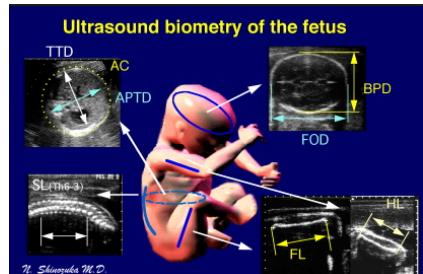
MEDIVIEWSOFT

Automated Fetal Biometry

to estimate fetal weight & fetal growth abnormalities

Goal: Improve clinical workflow

ergonomic stress 감소/업무process 간소화/ 문서작업 최소화



- Head (BPD, OFD, HC, Cephalic index)
- Thorax (Axis, Heart circumference Area, Thoracic circumference area, Head/Thorax ratio)
- Abdomen Circumference (AC),
- Bones: Femur (Femur length), Other Bones(optional)

Sub-challenges (HC, BPD, Femur length)



Challenge (AC)



현존하는 방식 : 정확도 매우 낮음

Automated or Semi-automated Fetal Biometry 기존 기술 평가

Siemens X3000 PE



Manual fetal biometric measurements by 2-dimensional (2D) sonography are an integral part of routine obstetric practice. Recently, an automated method for fetal biometry has been introduced in some sonographic equipment (*syngo Auto OB* measurements, S2000 ultrasound system; Siemens Healthcare,

2013년 논문: *syngo Auto OB* measurements 소프트웨어 사용으로 **Hc, BPD, Femur length, AC** 측정시간 절약.
....이러한 논문들은 실질적인 가치보다는 ...

Personal Opinion

- 자동화 기능에 대한 만족도가 아직은 낮은 수준.
- AC: 기존의 $|Image Gradient|$ 와 active contour을 이용한 방식으로는 실용화 어려울 듯.

IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 33, NO. 4, APRIL 2014

Automatic Fetal Head Measurements from Sonographic Images

Vikram 1996년 논문 (김용민 전 포항공대 총장이 저자)
David F Active contour 방식, $|Image Gradient|$ 가 큰 지점에서 contour stop

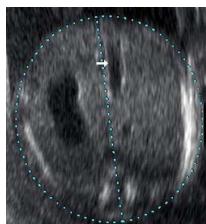
Evaluation and Comparison of Current Fetal Ultrasound Image Segmentation Methods for Biometric Measurements: A Grand Challenge

2014년 논문: **Challenge US: Biometric measurements from Fetal US images.** 이분야 전문가 5팀이 경합. 모두 상대적으로 쉬운 head 와 femur 다름. AC는 어려워 모두 포기.

Automated or Semi-automated Fetal Biometry 자동화가 어려운 이유

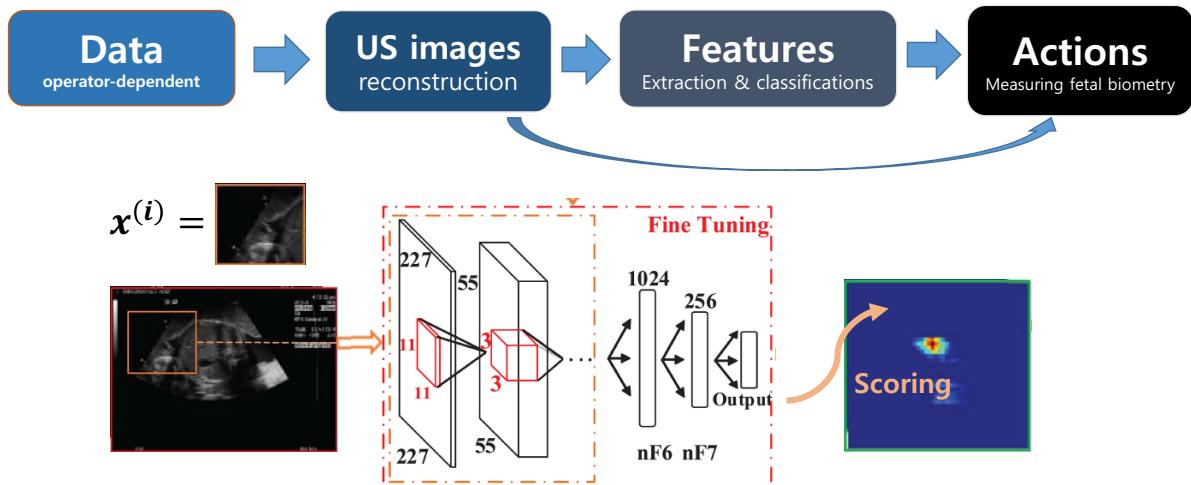
- US images are operator-dependent, patient-specific, and machine-specific.
- 특히 AC가 어려운 이유는 “low contrast against surroundings and non-uniform contrast”

CT, MRI와는 차원이 다른 어려움이 있음



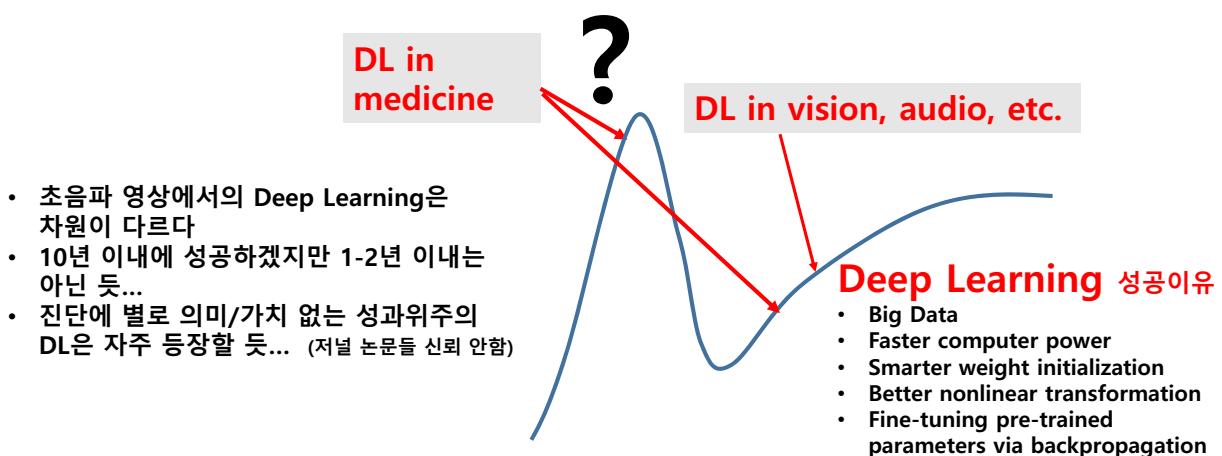
Deep Learning 방식 for Automated Fetal Biometry

딥러닝 방식이 성공하기 까지는 상당한 어려움은 있으나 결국은 성공할 듯.
기존 방식(Gradient와 active contour)으로는 어려울 듯.

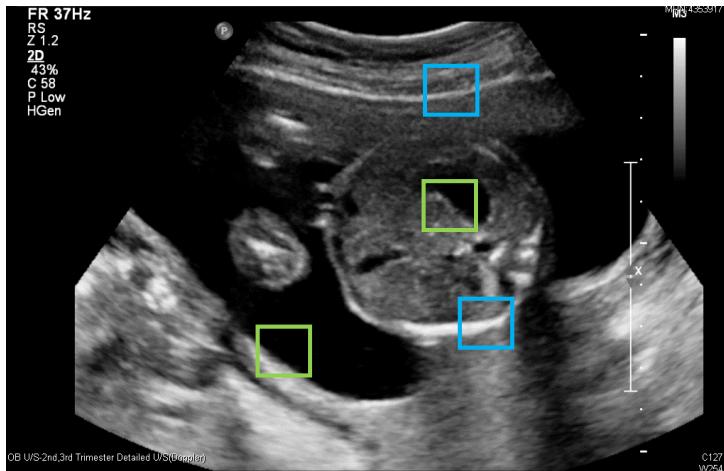


꿈에서 깨 으로

Deep Learning will not produce universal algorithm.



CNN 방식의 단순 적용은 실패 가능성 높음

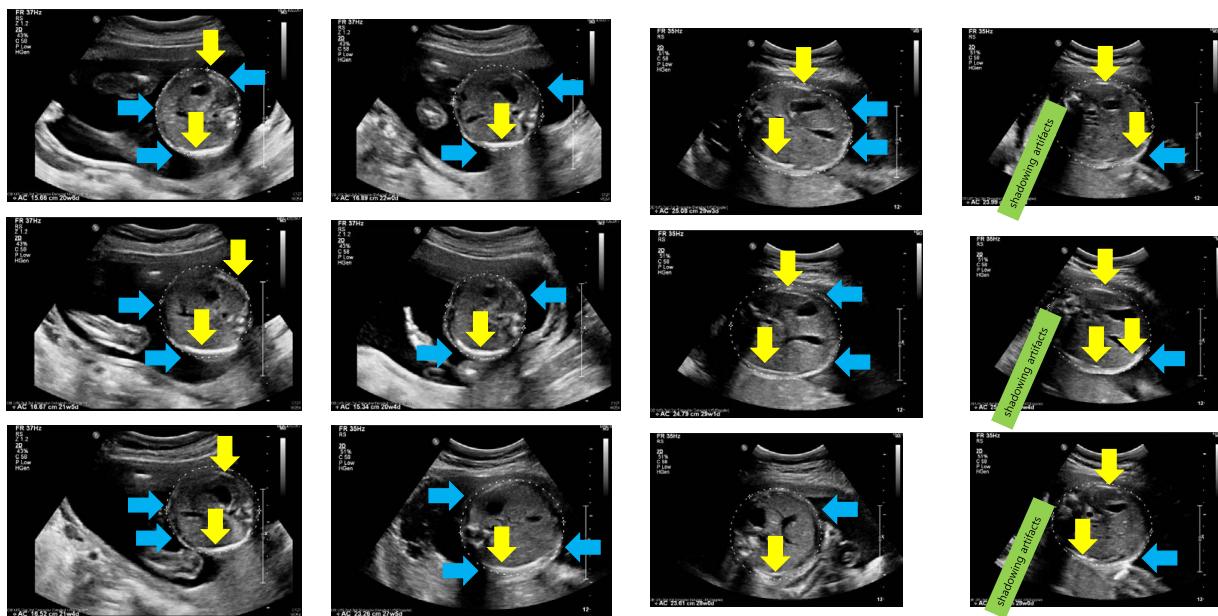


- CNN 접근 방식: 이미지 패치만을 분석하여 분류함
- ✓ 초음파 영상에 있어선 이미지 패치만을 바라보는 방법은 전문가의 판단 과정을 적절히 반영하지 않음

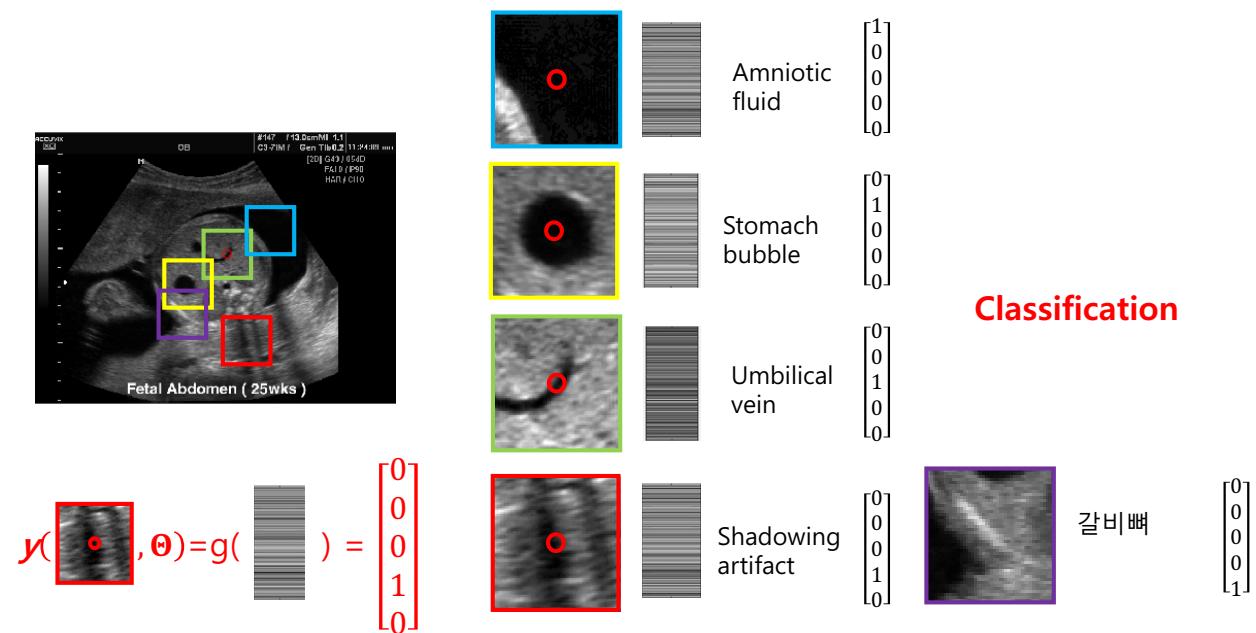
특징 분석

갈비뼈 ↓ 양수

shadowing artifacts

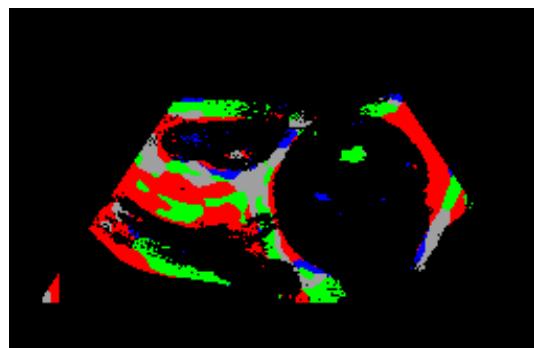


딥러닝을 이용한 해부학적 특징 추적

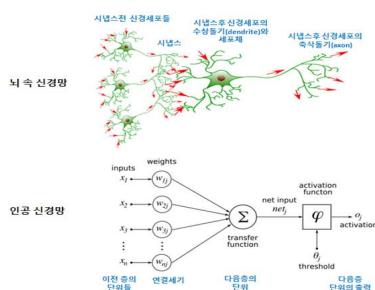


딥러닝 방식 TEST (part 5에서 딥러닝 방식 설명)

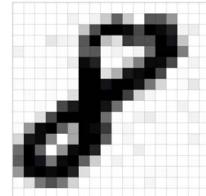
- 일반적인 Deep learning 방식이 성공하려면 (양질의) 엄청난 training data 필요. 결국 10년 이내에는 성공하기 어려울 듯.
- 이미지 특징과 초음파 진행 방향 반영한 딥러닝 방식은 (상대적으로) 적은 training data로 성공할 듯.



Part 1-2 : CNN 따라하기



What human sees



What machine sees

장재성 (연세대학교 계산과학공학과)



Jaeseong Jang Ph.D.



Bukweon Kim PhD



Sung Min Lee Ph.D.

MEDIVIEWSOFT

- CNN to classify each individual digit



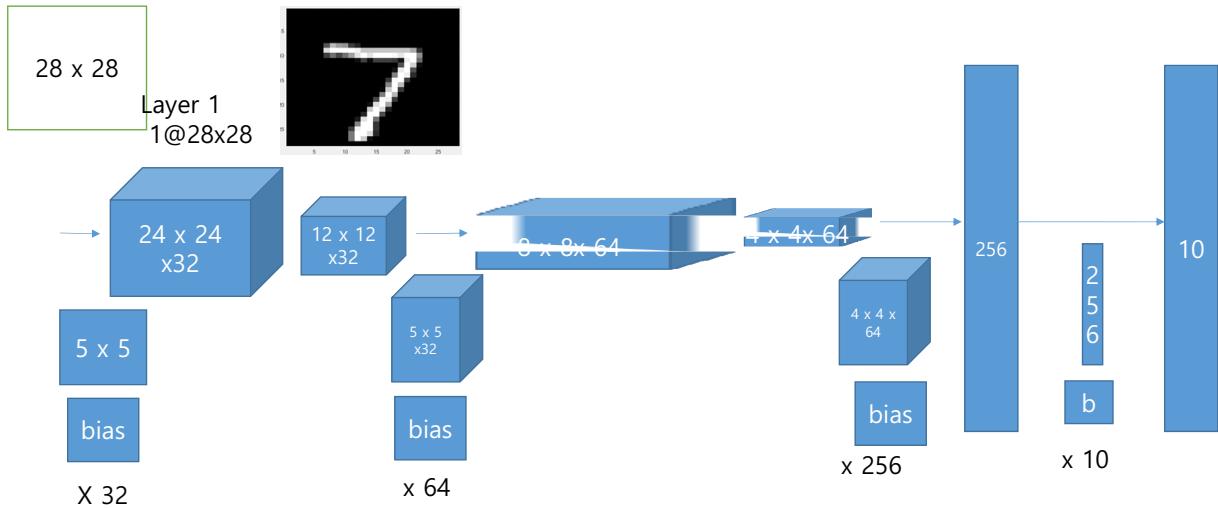
Split the problem of recognizing handwritten digits into two parts:

- [Segmentation] Break image containing many digits into a sequence of separate images, each containing a single digit.
 - [Classification] Classify each individual digit.

본 강의 노트의 이미지와 수식은 실제와 정확히 일치합니다.

- CNN to classify each individual digit

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



Feed Forward Pass

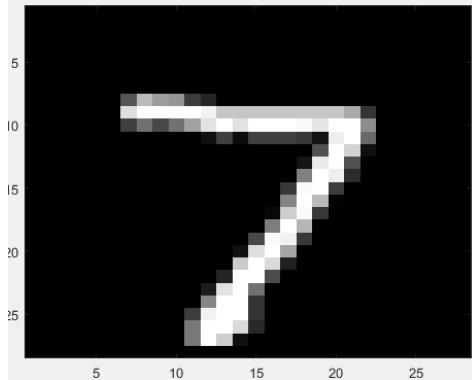
Feedforward neural networks means there are no loops in the network
 - information is always fed forward, never fed back.

- Throughout this note, we use rectified linear unit (ReLU) as the activation function $g(x)$

$$g(x) = \max(0, x).$$

Input layer

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

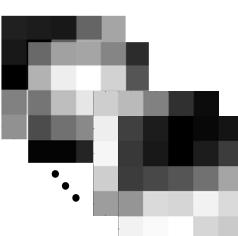
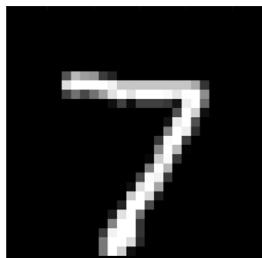


Input [1@28x28]

$$\mathbf{h}^0 = \begin{bmatrix} x_{1,1} & \cdots & x_{1,28} \\ \vdots & \ddots & \vdots \\ x_{28,1} & \cdots & x_{28,28} \end{bmatrix}$$

Layer 1 (convolutional)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



$$\mathbf{w}_i^1 = \begin{bmatrix} w_{i,(1,1)}^1 & \cdots & w_{i,(1,5)}^1 \\ \vdots & \ddots & \vdots \\ w_{i,(5,1)}^1 & \cdots & w_{i,(5,5)}^1 \end{bmatrix}$$

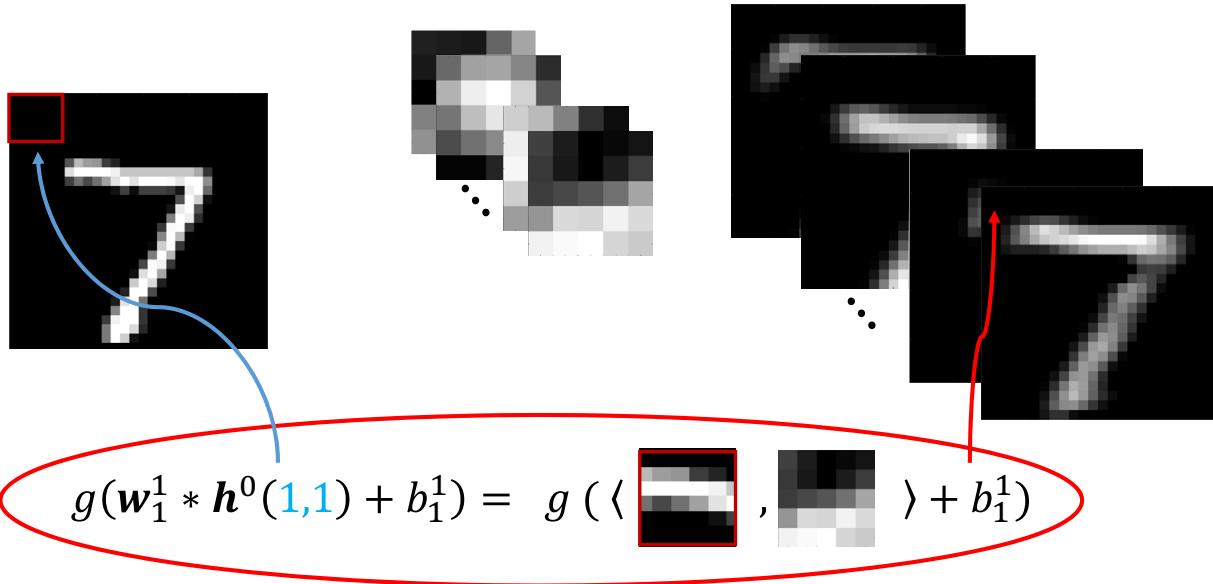
$i = 1, \dots, 32.$

$$\mathbf{h}^0(3,3) = \begin{bmatrix} h_{(1,1)}^0 & h_{(1,2)}^0 & h_{(1,3)}^0 & h_{(1,4)}^0 & h_{(1,5)}^0 \\ h_{(2,1)}^0 & h_{(2,2)}^0 & h_{(2,3)}^0 & h_{(2,4)}^0 & h_{(2,5)}^0 \\ h_{(3,1)}^0 & h_{(3,2)}^0 & h_{(3,3)}^0 & h_{(3,4)}^0 & h_{(3,5)}^0 \\ h_{(4,1)}^0 & h_{(4,2)}^0 & h_{(4,3)}^0 & h_{(4,4)}^0 & h_{(4,5)}^0 \\ h_{(5,1)}^0 & h_{(5,2)}^0 & h_{(5,3)}^0 & h_{(5,4)}^0 & h_{(5,5)}^0 \end{bmatrix}$$

$$\mathbf{h}_i^1 = \begin{bmatrix} g(\mathbf{w}_i^1 * \mathbf{h}^0(3,3) + b_i^1) & \cdots & g(\mathbf{w}_i^1 * \mathbf{h}^0(3,26) + b_i^1) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_i^1 * \mathbf{h}^0(26,3) + b_i^1) & \cdots & g(\mathbf{w}_i^1 * \mathbf{h}^0(26,26) + b_i^1) \end{bmatrix}$$

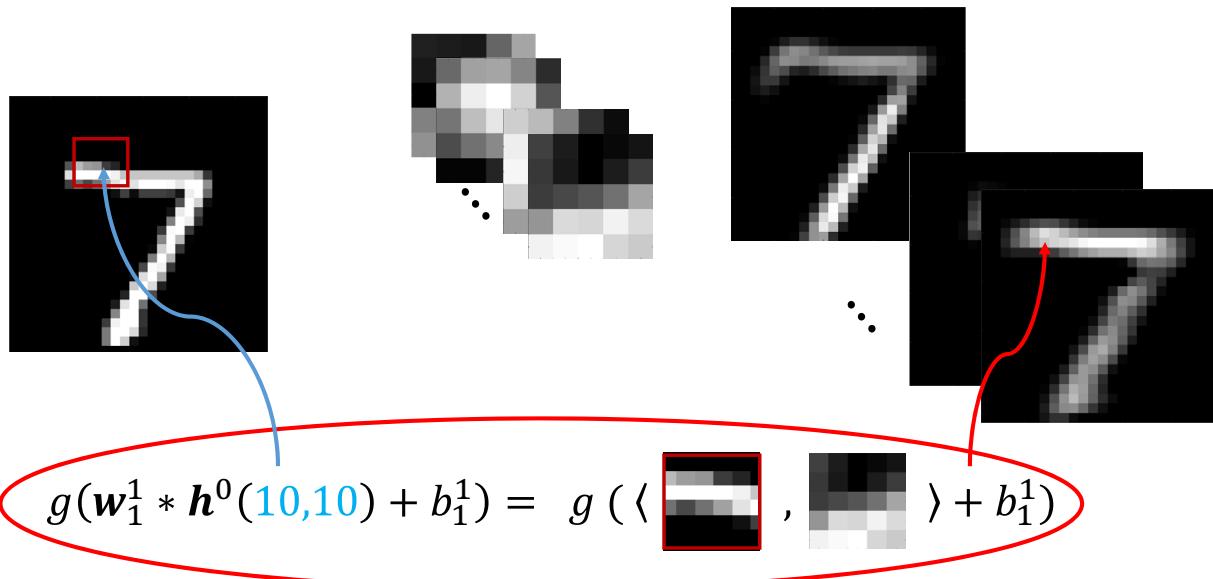
Layer 1 (convolutional)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



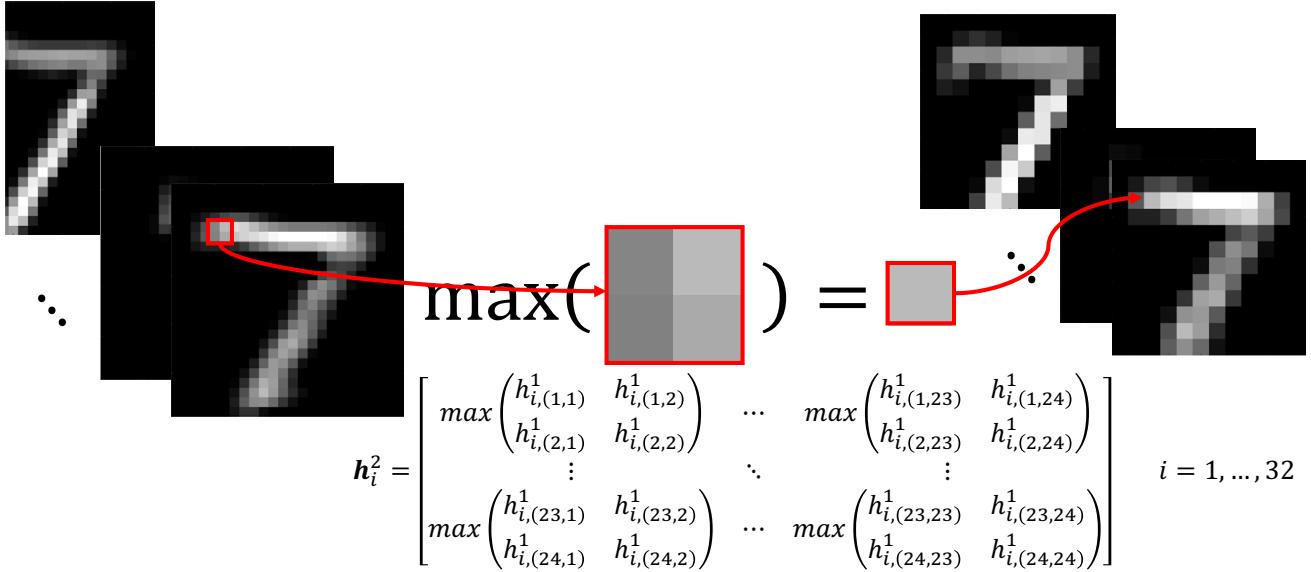
Layer 1 (convolutional)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



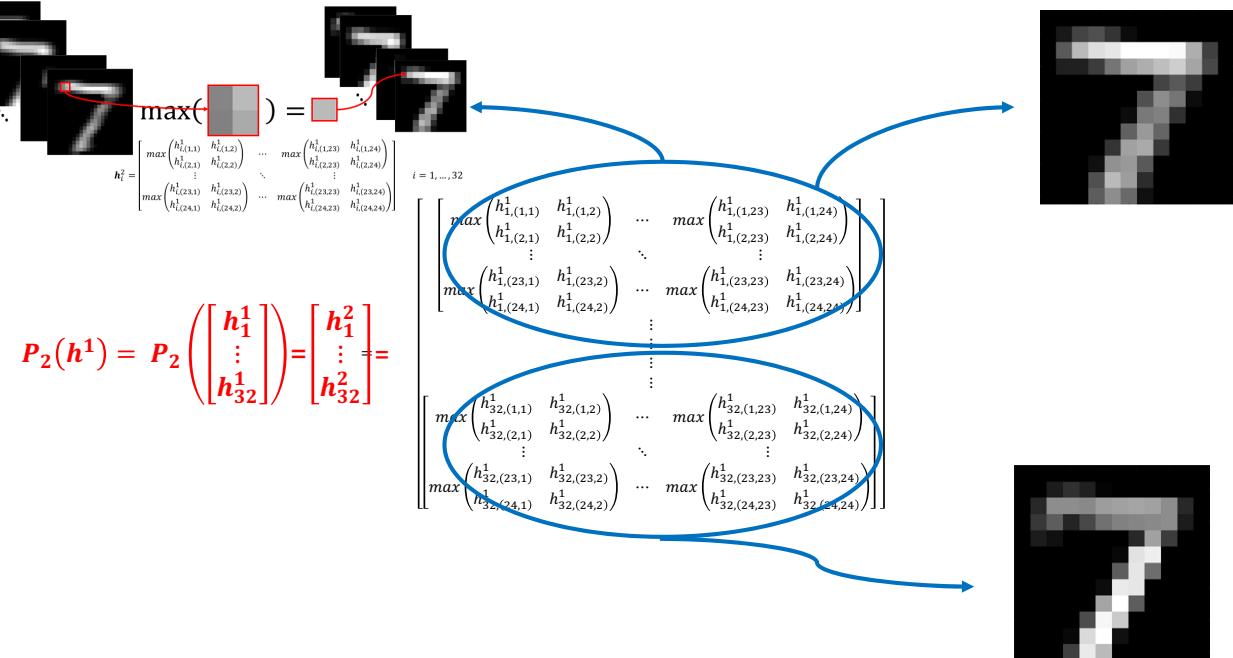
Layer 2 (pooling)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



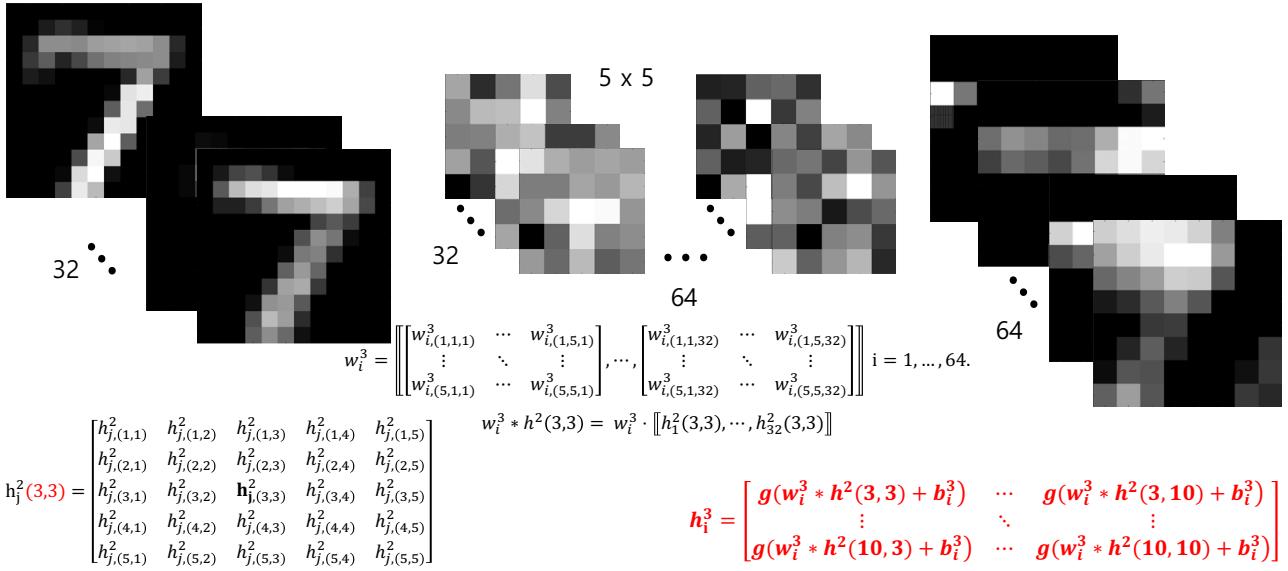
Layer 2 (pooling)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



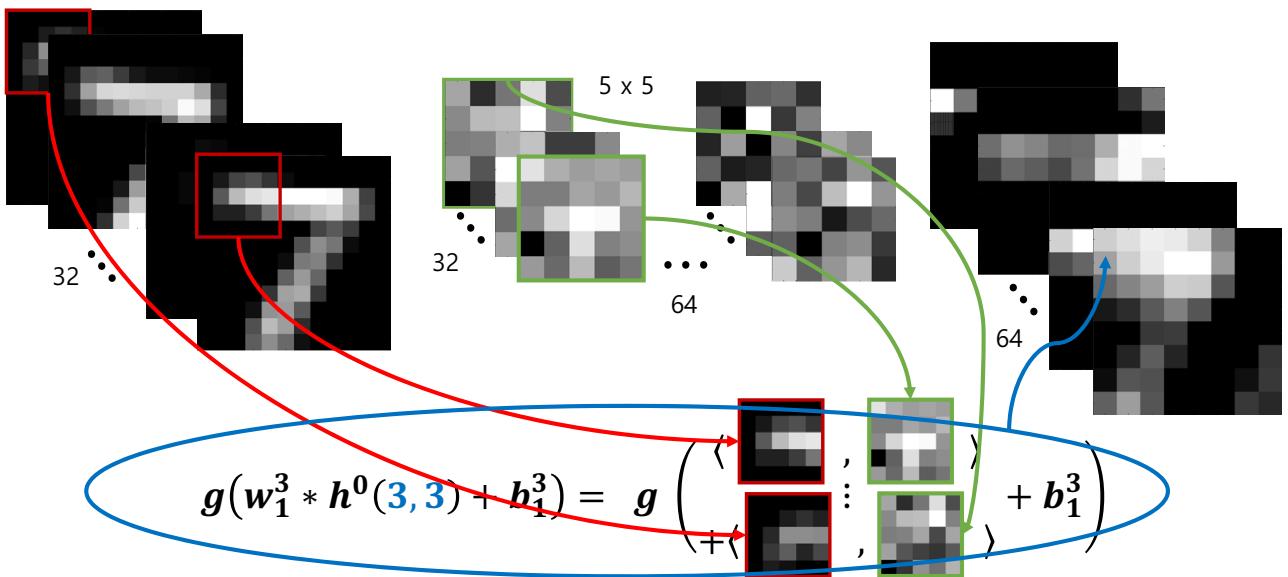
Layer 3 (convolution)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



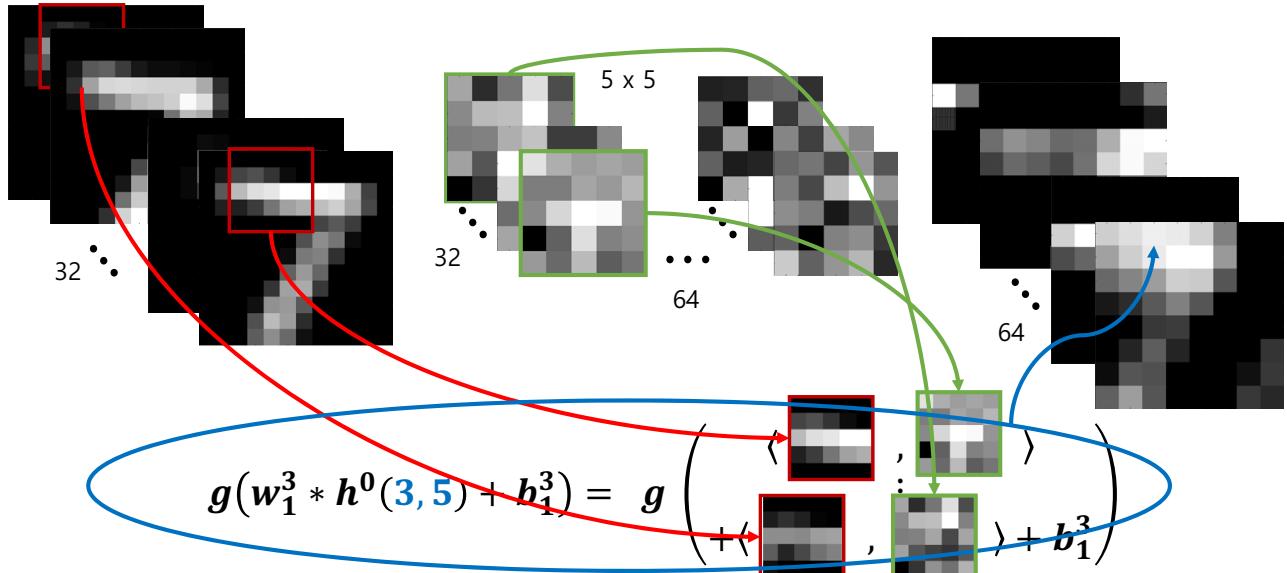
Layer 3 (convolution)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



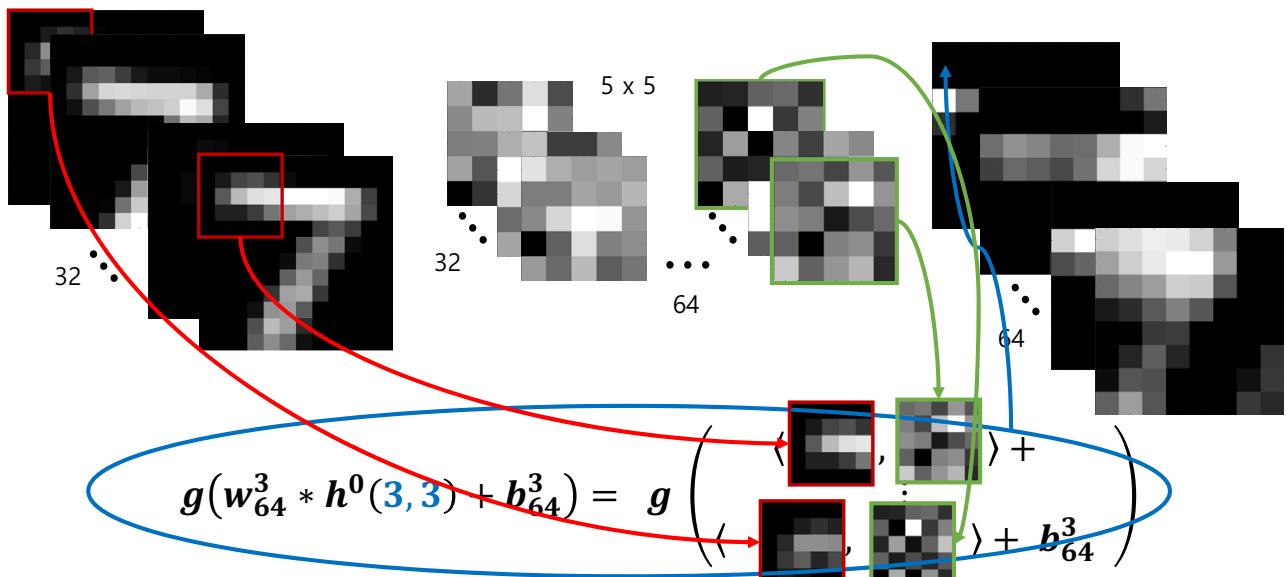
Layer 3 (convolution)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



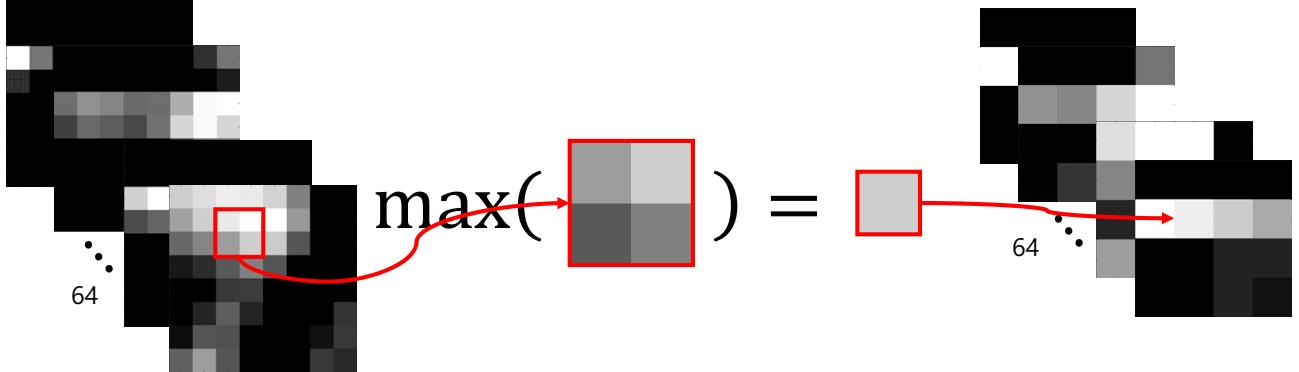
Layer 3 (convolution)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



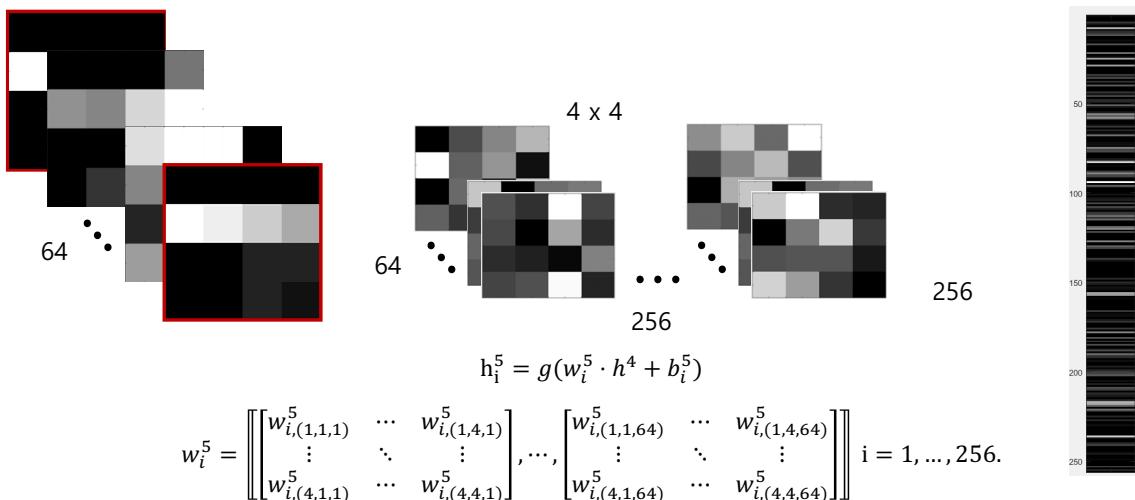
Layer 4 (pooling)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



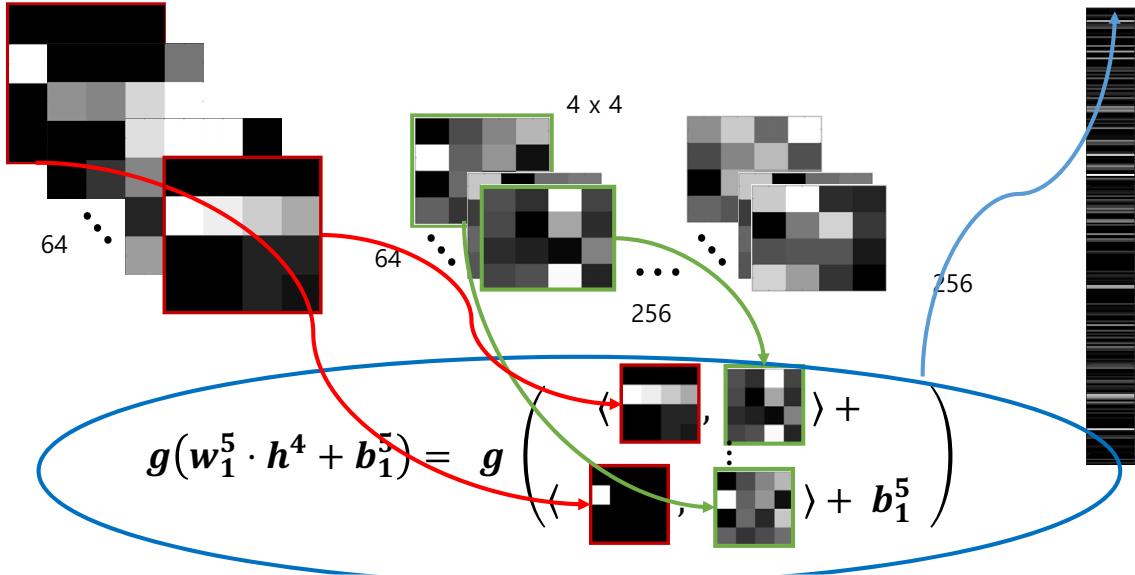
Layer 5 (fully-connected)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



Layer 5 (fully-connected)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]



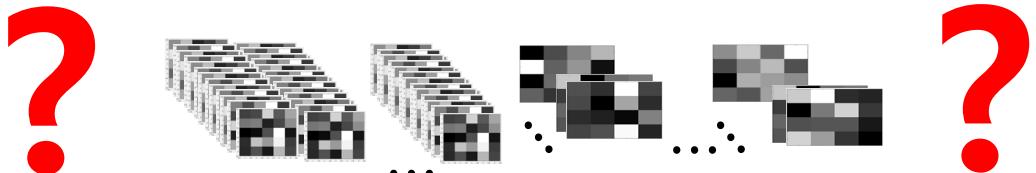
Output

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned}
 f &= \left[\begin{array}{c} \text{Input} \\ \vdots \\ \text{Input} \end{array} \right] + \left[\begin{array}{c} \text{C1} \\ \vdots \\ \text{C1} \end{array} \right] = f \\
 w_i^6 &= [w_{i,1}^6, \dots, w_{i,256}^6] \\
 W^6 &= \begin{bmatrix} w_1^6 \\ \vdots \\ w_{10}^6 \end{bmatrix} = \begin{bmatrix} w_{1,1}^6, \dots, w_{1,256}^6 \\ \vdots \\ w_{10,1}^6, \dots, w_{10,256}^6 \end{bmatrix} \\
 h^6 &= \begin{bmatrix} h_1^6 \\ \vdots \\ h_{10}^6 \end{bmatrix} = \begin{bmatrix} g(w_1^6 \cdot h^5 + b_1^6) \\ \vdots \\ g(w_{10}^6 \cdot h^5 + b_{10}^6) \end{bmatrix}
 \end{aligned}$$

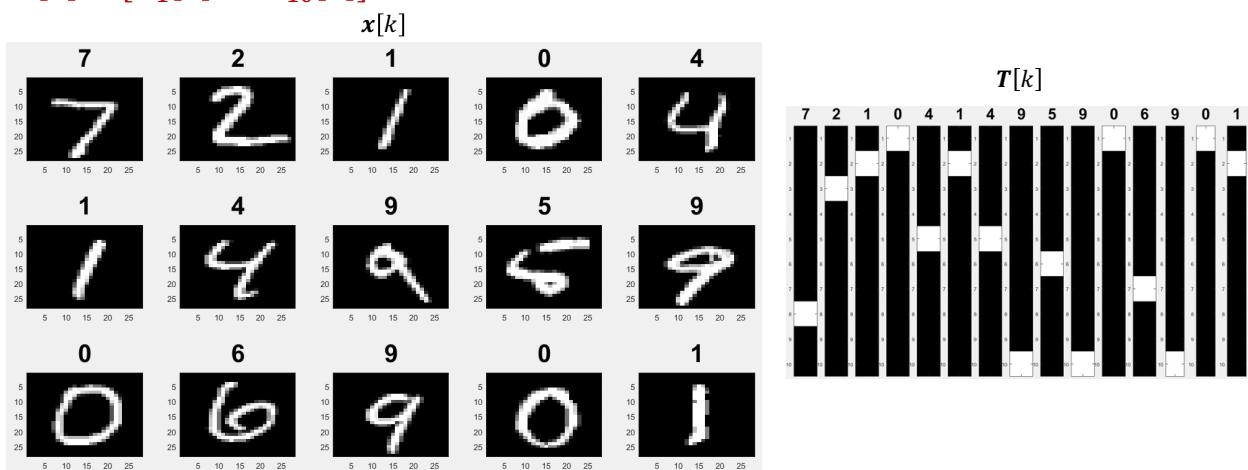
Backpropagation

The goal is to find weights and biases so that the output from the network approximates for all training inputs. To quantify how well we're achieving this goal, we define a cost function $E(W)$.



Outline of backpropagation

Suppose we are given a training set $\{(x[1], T[1]), \dots, (x[2000], T[2000])\}$. When the input pattern $x[k]$ from the training set presents to the previously mentioned network, it produced an output $O(x[k])$ different from the target $T[k] = [T_1[k], \dots, T_{10}[k]]$.



The goal is to determine

$$\mathbf{w}_i^1, \mathbf{b}_i^1 \text{ for } i = 1, \dots, 32 \Rightarrow (5 \times 5 + 1) \times 32 \text{ unknowns}$$

$$\mathbf{w}_i^3, \mathbf{b}_i^3 \text{ for } i = 1, \dots, 64 \Rightarrow (5 \times 5 \times 32 + 1) \times 64 \text{ unknowns}$$

$$\mathbf{w}_i^5, \mathbf{b}_i^5 \text{ for } i = 1, \dots, 256 \Rightarrow (4 \times 4 \times 64 + 1) \times 256 \text{ unknowns}$$

$$\mathbf{w}_i^6, \mathbf{b}_i^6 \text{ for } i = 1, \dots, 10 \Rightarrow (256 + 1) \times 10 \text{ unknowns}$$

such that $\mathbf{W} = [\mathbf{w}^1, \mathbf{b}^1, \mathbf{w}^3, \mathbf{b}^3, \mathbf{w}^5, \mathbf{b}^5, \mathbf{w}^6, \mathbf{b}^6]$ ($26 \times 32 + 801 \times 64 + 1025 \times 256 + 257 \times 10$ variables)
minimizes the following error

$$E(\mathbf{W}) = \frac{1}{2} \sum_{k=1}^{2000} \|\mathbf{o}(\mathbf{x}[k]) - \mathbf{T}[k]\|^2.$$

We update \mathbf{W} by

$$\mathbf{W} \leftarrow \mathbf{W} - \eta \nabla_{\mathbf{W}} E$$

where η represents a learning rate.

Express $E(\mathbf{W})$ in term of $\mathbf{x}[k]$

$$\begin{aligned}
 \mathbf{o}(\mathbf{x}[k]) &= f(\mathbf{h}_k^6) = f(\mathbf{W}^6 \mathbf{h}_k^5 + \mathbf{b}^6) \\
 E(\mathbf{W}) &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 \mathbf{h}_k^5 + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 \mathbf{h}_k^4 + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 P_4(\mathbf{h}_k^3) + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 P_4(g(\mathbf{W}^3 \mathbf{h}_k^2 + \mathbf{b}^3)) + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 P_4(g(\mathbf{W}^3 P_2(\mathbf{h}_k^1) + \mathbf{b}^3)) + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 P_4(g(\mathbf{W}^3 P_2(g(\mathbf{W}^1 \mathbf{x}[k] + \mathbf{b}^1)) + \mathbf{b}^3)) + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2
 \end{aligned}$$

Gradient of E(W) (F-layer)

$\mathbf{w}^6, \mathbf{b}^6$

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned}
 \frac{\partial E}{\partial \mathbf{w}_i^6} &= \sum_{k=1}^{2000} \frac{\partial E}{\partial h_{k,l}^6} \frac{\partial h_{k,l}^6}{\partial \mathbf{w}_i^6} \\
 &= \frac{\partial}{\partial \mathbf{w}_i^6} \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 \mathbf{h}_k^5 + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \sum_{k=1}^{2000} (f(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6) - T_i[k]) f'(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6) \mathbf{h}_k^5 \\
 \frac{\partial E}{\partial b_i^6} &= \frac{\partial}{\partial b_i^6} \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 \mathbf{h}_k^5 + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \sum_{k=1}^{2000} (f(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6) - T_i[k]) f'(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6)
 \end{aligned}$$

Gradient of E(W) (F-layer)

$\mathbf{w}^5, \mathbf{b}^5$

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned}
 \frac{\partial E}{\partial \mathbf{w}_i^5} &= \frac{\partial}{\partial \mathbf{w}_i^5} \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 \mathbf{h}_k^4 + \mathbf{b}^5) + \mathbf{b}^6) - \mathbf{T}[k]\|^2 \\
 &= \sum_{k=1}^{2000} \sum_l \frac{\partial E}{\partial h_{k,l}^6} \frac{\partial h_{k,l}^6}{\partial h_{k,i}^5} \frac{\partial h_{k,i}^5}{\partial \mathbf{w}_i^5} \\
 &= \sum_{k=1}^{2000} \sum_l [(f(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6) - T_i[k]) f'(\mathbf{w}_i^6 \mathbf{h}_k^5 + b_i^6) \mathbf{w}_{l,i}^6] g'(\mathbf{w}_i^5 \mathbf{h}_k^4 + b_i^5) \mathbf{h}_k^4
 \end{aligned}$$

$g'(\mathbf{w}_i^5 \mathbf{h}_k^4 + b_i^5)$ 의 의미는?
활성화된 노드를 통해서만 반응이 역전파.

$$g'(x) = \begin{cases} 1 & 0 \leq x \\ 0 & 0 > x \end{cases}$$

Gradient of E(W) (F-layer)

$\mathbf{w}^5, \mathbf{b}^5$

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned}\frac{\partial E}{\partial b_i^5} &= \frac{\partial}{\partial b_i^5} \frac{1}{2} \sum_{k=1}^{2000} \|f(\mathbf{W}^6 g(\mathbf{W}^5 \mathbf{h}_k^4 + \mathbf{b}^5) + \mathbf{b}^6) - T[k]\|^2 \\ &= \sum_{k=1}^{2000} \sum_l \frac{\partial E}{\partial h_{k,l}^6} \frac{\partial h_{k,l}^6}{\partial h_{k,i}^5} \frac{\partial h_{k,i}^5}{\partial w_{k,i}^5} \\ &= \sum_{k=1}^{2000} \sum_l [(f(\mathbf{w}_l^6 \mathbf{h}_k^5 + b_l^6) - T_l[k]) f'(\mathbf{w}_l^6 \mathbf{h}_k^5 + b_l^6) \mathbf{w}_{l,i}^6] g'(\mathbf{w}_l^5 \mathbf{h}_k^4 + b_l^5)\end{aligned}$$

$g'(\mathbf{w}_l^5 \mathbf{h}_k^4 + b_l^5)$ 의 의미는?

활성화된 노드를 통해서만 반응이 역전파.

$$g'(x) = \begin{cases} 1 & 0 \leq x \\ 0 & 0 > x \end{cases}$$

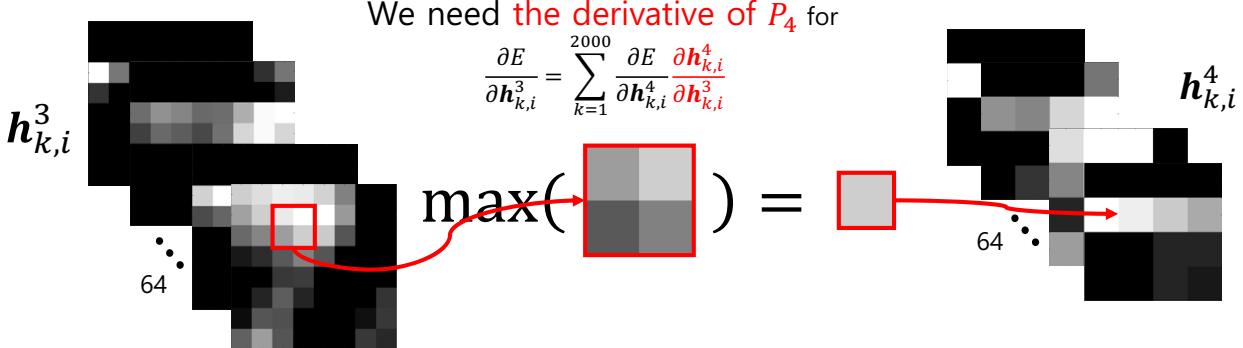
더 나아가기 전에.. (P-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned}\frac{\partial E}{\partial \mathbf{h}_{k,i}^4} &= \sum_{k=1}^{2000} \sum_l \frac{\partial E}{\partial h_{k,l}^6} \frac{\partial h_{k,l}^6}{\partial h_{k,i}^5} \frac{\partial h_{k,i}^5}{\partial \mathbf{h}_{k,i}^4} \\ &= \sum_{k=1}^{2000} \sum_l [(f(\mathbf{w}_l^6 \mathbf{h}_k^5 + b_l^6) - T_l[k]) f'(\mathbf{w}_l^6 \mathbf{h}_k^5 + b_l^6) \mathbf{w}_{l,i}^6] g'(\mathbf{w}_l^5 \mathbf{h}_k^4 + b_l^5) \mathbf{w}_l^5\end{aligned}$$

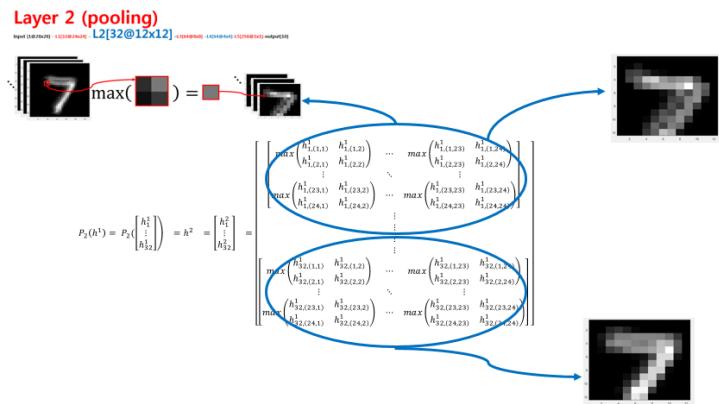
We need the derivative of P_4 for

$$\frac{\partial E}{\partial \mathbf{h}_{k,i}^3} = \sum_{k=1}^{2000} \frac{\partial E}{\partial \mathbf{h}_{k,i}^4} \frac{\partial \mathbf{h}_{k,i}^4}{\partial \mathbf{h}_{k,i}^3}$$



Derivative of \max function

$$\frac{\partial}{\partial h_{i,(a,b)}^1} \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = \begin{cases} 1 & \text{if } h_{i,(a,b)}^1 = \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix}, \\ 0 & \text{otherwise} \end{cases}$$



Feed forward pass 시 저장!

For example, if $\max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = h_{i,(1,1)}^1$,

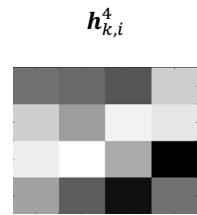
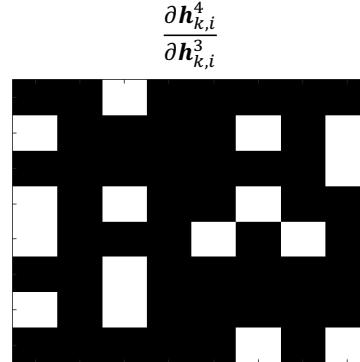
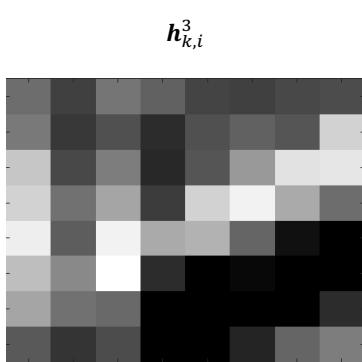
$$\frac{\partial}{\partial h_{i,(1,1)}^1} \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = 1, \quad \frac{\partial}{\partial h_{i,(1,2)}^1} \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = 0,$$

$$\frac{\partial}{\partial h_{i,(2,1)}^1} \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = 0, \quad \frac{\partial}{\partial h_{i,(2,2)}^1} \max \begin{pmatrix} h_{i,(1,1)}^1 & h_{i,(1,2)}^1 \\ h_{i,(2,1)}^1 & h_{i,(2,2)}^1 \end{pmatrix} = 0.$$

- 같은 철학
Activate된 노드로만 반응이 역전파.

Gradient of E(W) (P-layer)

Input [1@28x28] – L1[32@24x24] – L2[32@12x12] – L3[64@8x8] – L4[64@4x4] – L5[256@1x1] – output[10]



Gradient of E(W) (P-layer)

Input [1@28x28] – L1[32@24x24] – L2[32@12x12] – L3[64@8x8] – L4[64@4x4] – L5[256@1x1] – output[10]

$$\frac{\partial E}{\partial \mathbf{h}_{k,i}^4}$$

$$\frac{\partial \mathbf{h}_{k,i}^4}{\partial \mathbf{h}_{k,i}^3}$$

$$\frac{\partial E}{\partial \mathbf{h}_{k,i}^3}$$

$$\frac{\partial \mathbf{h}_{k,i}^3}{\partial \mathbf{h}_{k,i}^2}$$

$$\frac{\partial E}{\partial \mathbf{h}_{k,i}^2}$$

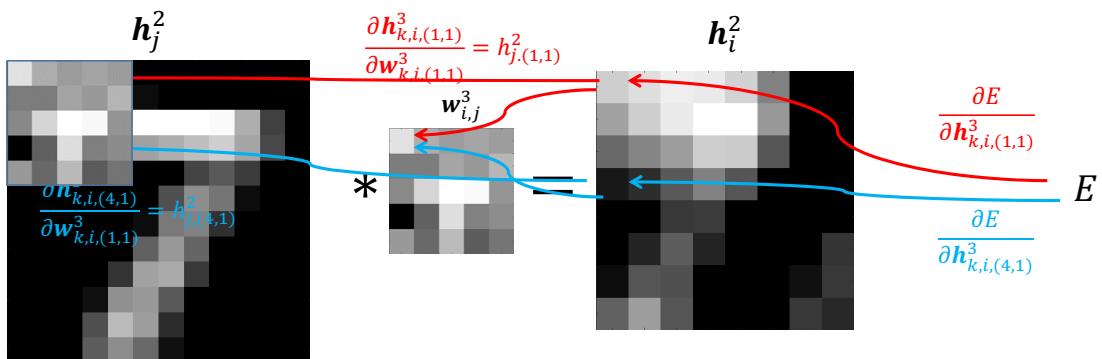
$$\frac{\partial \mathbf{h}_{k,i}^2}{\partial \mathbf{h}_{k,i}^1}$$

$$\frac{\partial E}{\partial \mathbf{h}_{k,i}^1}$$

Gradient of E(W) (C-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

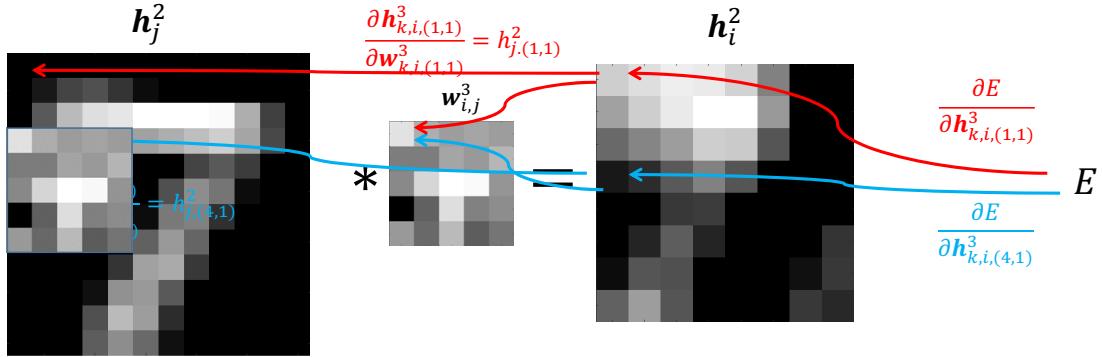
$$\frac{\partial E}{\partial \mathbf{w}_{i,j,(1,1)}^3} = \sum_k \sum_{1 \leq m, n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \frac{\partial \mathbf{h}_{k,i,(m,n)}^3}{\partial \mathbf{w}_{k,i,(1,1)}^3} = \sum_k \sum_{1 \leq m, n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} h_{j,(m,n)}^2$$



Gradient of E(W) (C-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

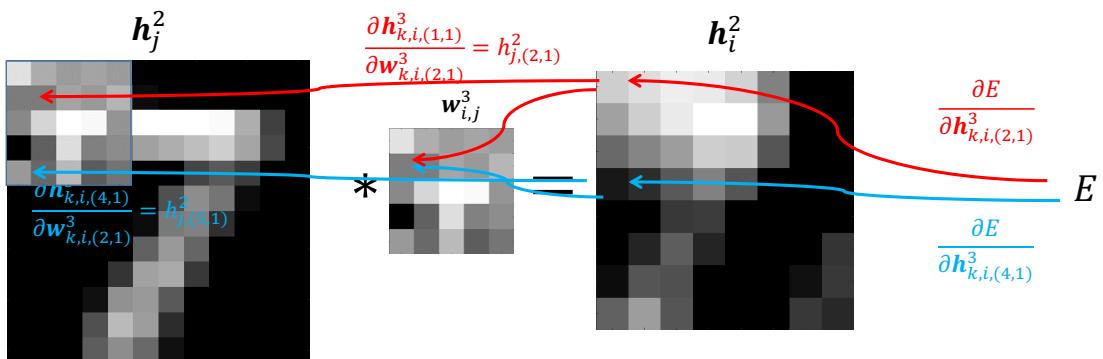
$$\frac{\partial E}{\partial \mathbf{w}_{i,j,(1,1)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \frac{\partial \mathbf{h}_{k,i,(m,n)}^3}{\partial \mathbf{w}_{k,i,(1,1)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} h_{j,(m,n)}^2$$



Gradient of E(W) (C-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

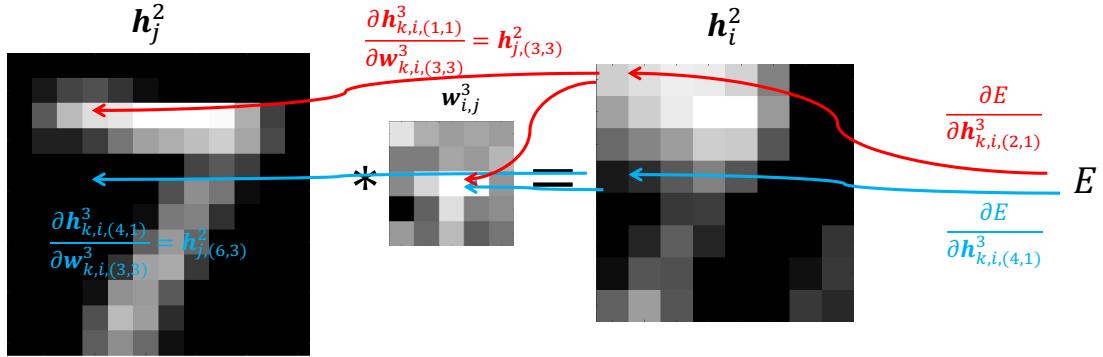
$$\frac{\partial E}{\partial \mathbf{w}_{i,j,(2,1)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \frac{\partial \mathbf{h}_{k,i,(m,n)}^3}{\partial \mathbf{w}_{k,i,(2,1)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} h_{j,(m+1,n)}^2$$



Gradient of E(W) (C-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

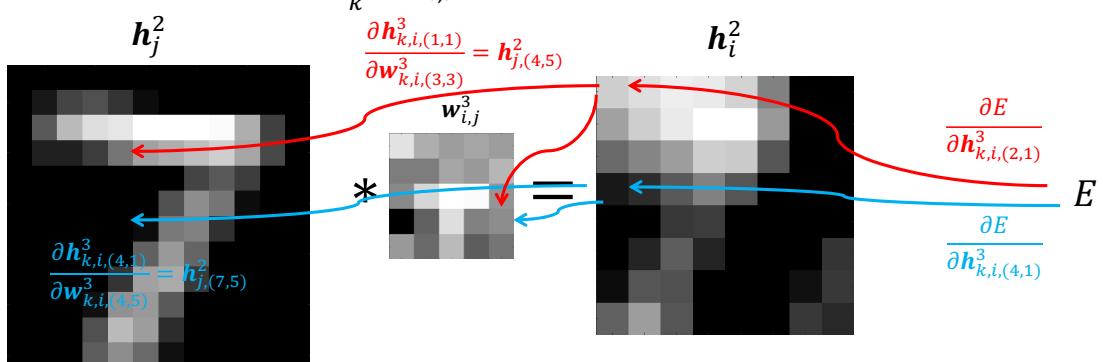
$$\frac{\partial E}{\partial \mathbf{w}_{i,j,(3,3)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \frac{\partial \mathbf{h}_{k,i,(m,n)}^3}{\partial \mathbf{w}_{k,i,(3,3)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \mathbf{h}_{j,(m+2,n+2)}^2$$



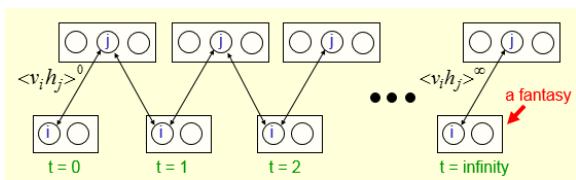
Gradient of E(W) (C-layer)

Input [1@28x28] - C1[32@24x24] - P2[32@12x12] - C3[64@8x8] - P4[64@4x4] - F5[256@1x1] - output[10]

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{w}_{i,j,(k,l)}^3} &= \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \frac{\partial \mathbf{h}_{k,i,(m,n)}^3}{\partial \mathbf{w}_{k,i,(k,l)}^3} = \sum_k \sum_{1 \leq m,n \leq 8} \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} \mathbf{h}_{j,(m+k-1,n+l-1)}^2 \\ &= \sum_k \frac{\partial E}{\partial \mathbf{h}_{k,i,(m,n)}^3} *_{m,n} \mathbf{h}_j^2(m, n) \end{aligned}$$



Part 1-3 : RBM (Restricted Boltzmann Machine)



binary state of visible unit i
 binary state of hidden unit j
 $E(v, h) = - \sum_{i,j} v_i h_j w_{ij}$
 Energy with configuration v on the visible units and h on the hidden units
 $-\frac{\partial E(v, h)}{\partial w_{ij}} = v_i h_j$
 weight between units i and j

김부권 (연세대학교 계산과학공학과)



Jaeseong Jang Ph.D.



Bukweon Kim Ph.D.

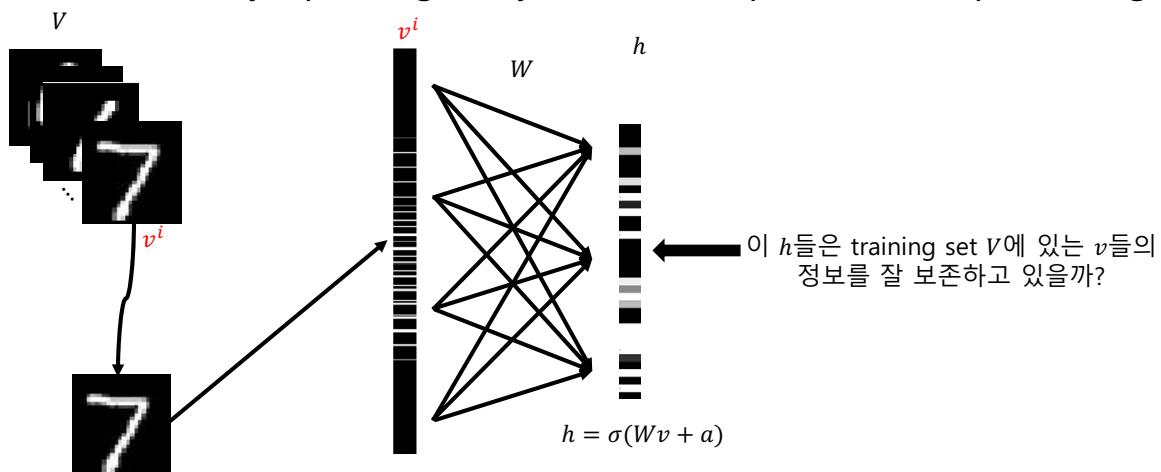


Sung Min Lee Ph.D.

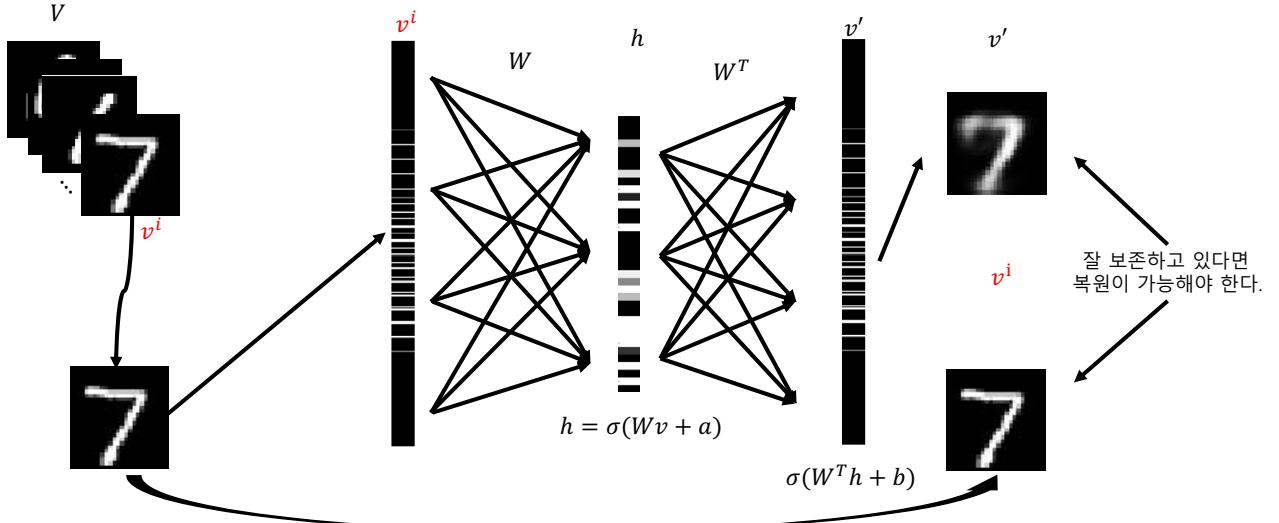
MEDIVIEWSOFT

Restricted Boltzmann Machine

- Recursively updating 2-layered non-supervised deep learning.



Restricted Boltzmann Machine



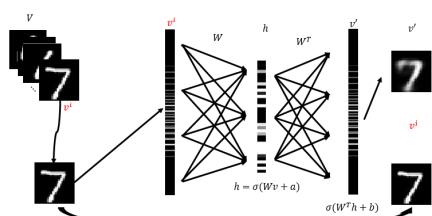
Likelihood Function

- Likelihood function

$$L((W, a, b)|V) = \prod_{v^i \in V} P(v^i|(W, a, b))$$

- Probability of having V as output for determined (W, a, b)
- Often maximized after applying log

$$\log(L((W, a, b)|V)) = \sum_{v \in V} \log(L((W, a, b)|v))$$



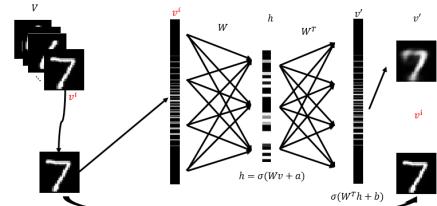
Gradient of Likelihood

- Let $E(v, h) = -(a^T v + b^T h + v^T W h)$
- Then if we suppose (v, h) follows Gibbs distribution

$$P(v, h) = \frac{1}{Z} e^{-E(v, h)}$$

- Then we take derivative of $\log(L((W, a, b)|V))$ by each W to update W . Then we, get for training set $V = \{v^1, v^2, \dots, v^l\}$

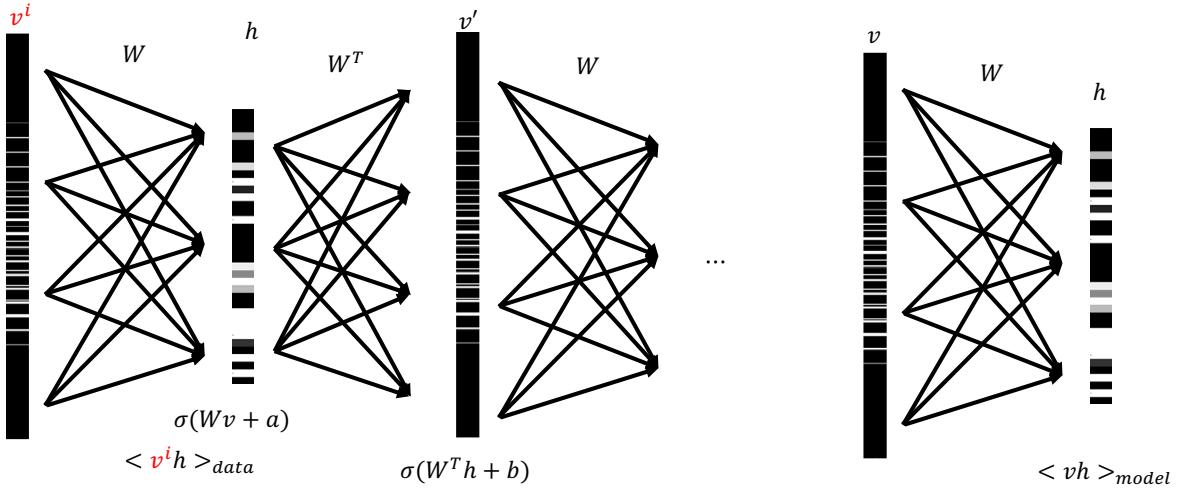
$$\frac{\partial \log(L((W, a, b)|V))}{\partial W} = \frac{1}{l} \sum_{i=1}^l \frac{\partial \log(L((W, a, b)|v^i))}{\partial W}$$



$$\frac{\partial \log(L((W, a, b)|v^i))}{\partial W} = \langle v^i h \rangle_{data} - \langle vh \rangle_{model}$$

$$\begin{aligned} \frac{\partial \log(L((W, a, b)|v^i))}{\partial W} &= \frac{\partial}{\partial W} \left(\log \left(\sum_h e^{-E(v^i, h)} \right) - \log \left(\sum_v \sum_h e^{-E(v, h)} \right) \right) \\ &= \frac{1}{\sum_h e^{-E(v^i, h)}} \left(\sum_h e^{-E(v^i, h)} v^i h \right) - \frac{1}{\sum_v \sum_h e^{-E(v, h)}} \left(\sum_v \sum_h e^{-E(v, h)} v h \right) \\ &= \sum_h P(h|v^i) v^i h - \sum_v \sum_h P(v, h) v h \\ &= \langle v^i h \rangle_{data} - \langle vh \rangle_{model} \end{aligned}$$

Maximum likelihood



RBM Algorithm

- 실제 알고리즘은 Gradient Decent가 아닌 다음과 같은 알고리즘을 사용한다.

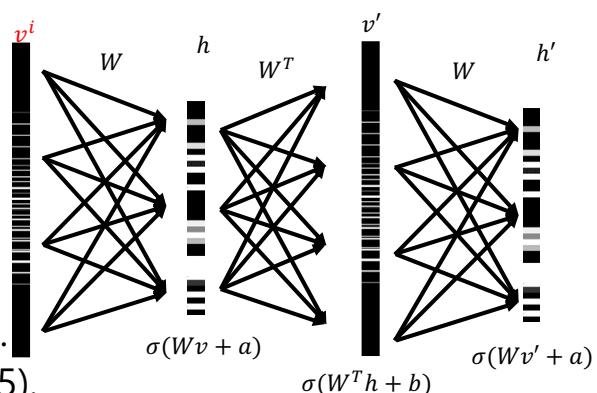
- Update W , a and b by

$$\begin{aligned} W_{n+1} &= W_n + \Delta W \\ a_{n+1} &= a_n + \Delta a \\ b_{n+1} &= b_n + \Delta b \end{aligned}$$

- Where

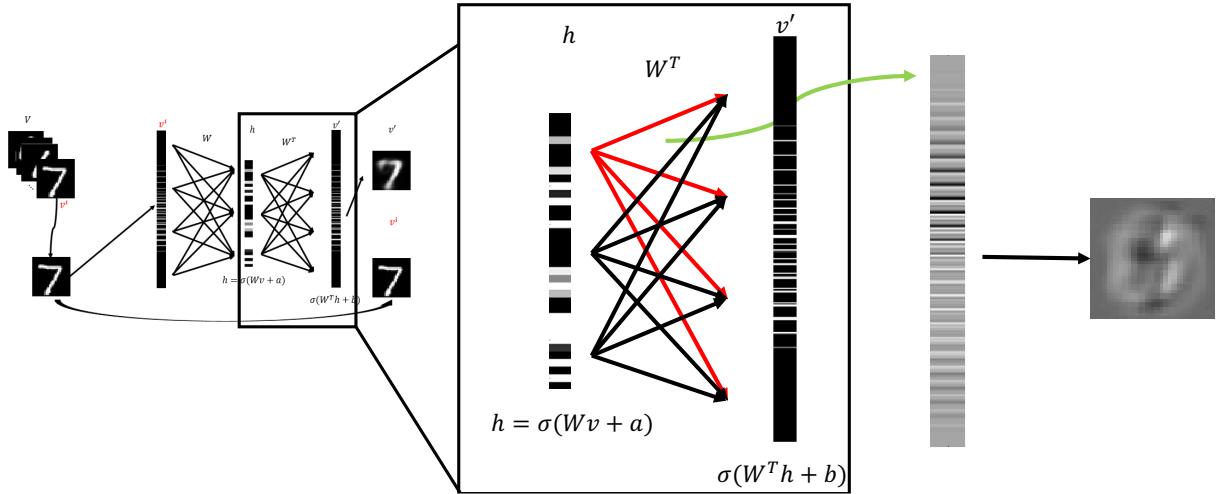
$$\begin{aligned} \Delta W &= \epsilon(vh^T - v'h'^T) \\ \Delta b &= \epsilon(h - h') \\ \Delta a &= \epsilon(v - v') \end{aligned}$$

- 이 방식으로도 최소로 가게 된다. (Carreira-Perpinan & Hinton, 2005).



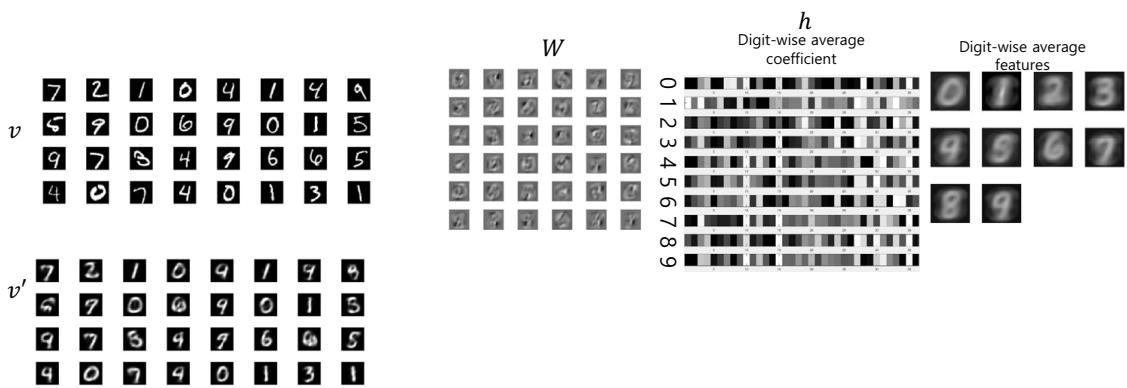
W

- W 의 각 vector들은 다음과 같이 data image를 이루는 bases로 볼 수도 있다.



Good Initial(Test)

- At 10000 iteration with learning coefficient 0.01, RBM showed 59.27% accuracy of digit classification.



Part 1-4 : AdaBoost

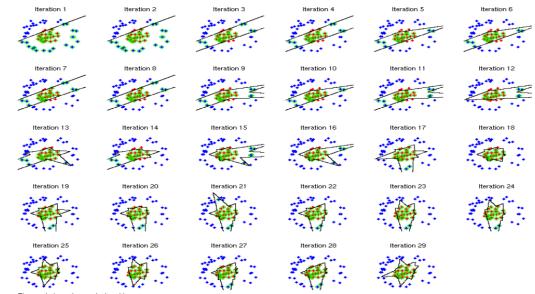


Figure 1. Iterations of algorithm

이성민 (연세대학교 계산과학공학과)



Jaeseong Jang Ph.D



Bukweon Kim Ph.D



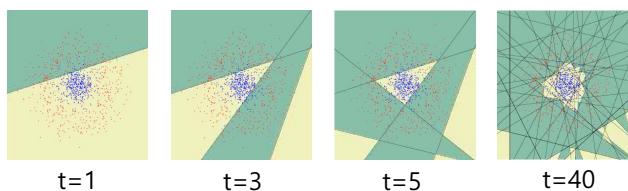
Sung Min Lee Ph.D

MEDIVIEWSOFT

AdaBoost

Constructing a "strong" classifier as linear combination of "weak" classifiers $h_t(x)$

$$f(x) = \sum_{t=1}^T \alpha_t h_t(x)$$



AdaBoost algorithm

Given : $(x_1, y_1), \dots, (x_m, y_m)$ where $x_i \in X, y_i \in Y = \{-1, +1\}$

Initialize $D_1(x_i) = \frac{1}{m}$.

For $t = 1, \dots, T$:

- Train weak learner using distribution D_t .
- Get weak hypothesis $h_t : X \rightarrow \{-1, +1\}$ with the low weighted error

$$\epsilon_t = \sum_{i=1}^m D_t(x_i)[h_t(x_i) \neq y_i].$$

- Choose $\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$.

- Update:

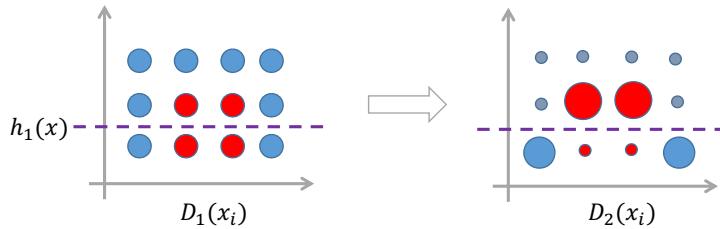
$$D_{t+1}(x_i) = \frac{D_t(x_i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } h_t(x_i) = y_i \\ e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \end{cases}$$

$$= \frac{D_t(x_i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

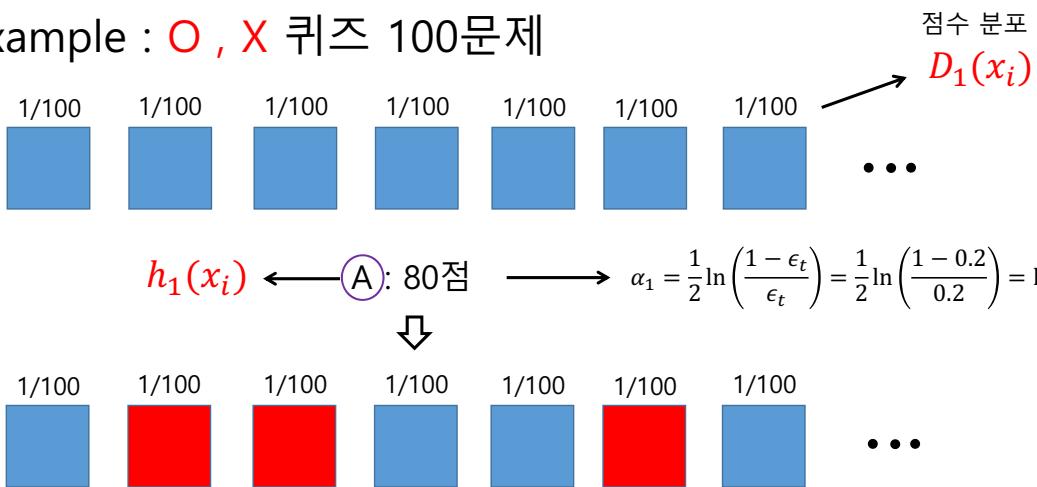
Where Z_t is a normalization factor (chosen so that D_{t+1} will be a distribution).

Output the final hypothesis:

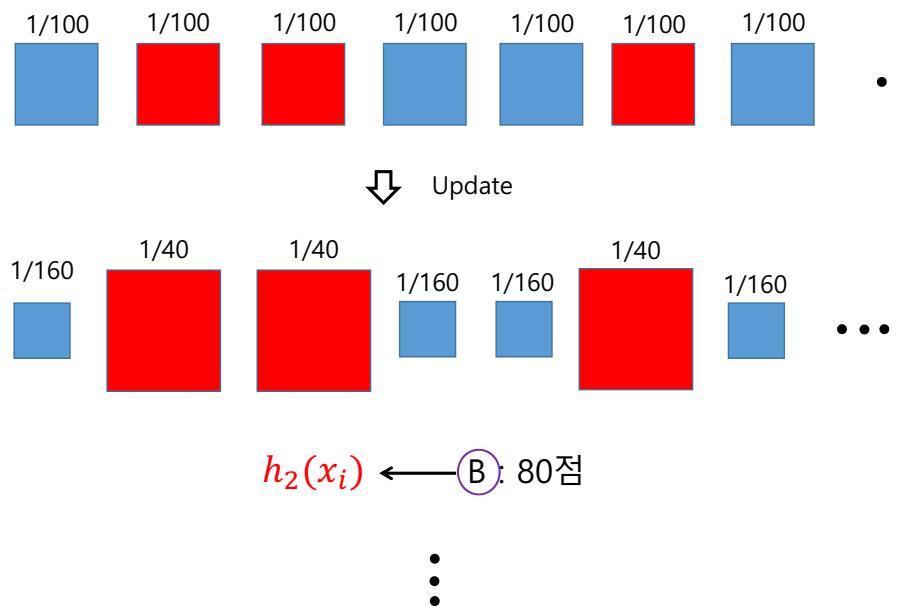
$$H(x) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(x)).$$



Example : O , X 퀴즈 100문제



$$D_2(x_i) = \frac{D_1(x_i)}{Z_1} \times \begin{cases} e^{-\alpha_t} & \text{if } h_t(x_i) = y_i \\ e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \end{cases} = \frac{1}{Z_1} \times \begin{cases} 1/200 & \text{if } h_t(x_i) = y_i \\ 1/50 & \text{if } h_t(x_i) \neq y_i \end{cases}$$



Part 1-5 : Practical application automated fetal biometry

장재성 (연세대학교 계산과학공학과)



Jaeseong Jang Ph.D



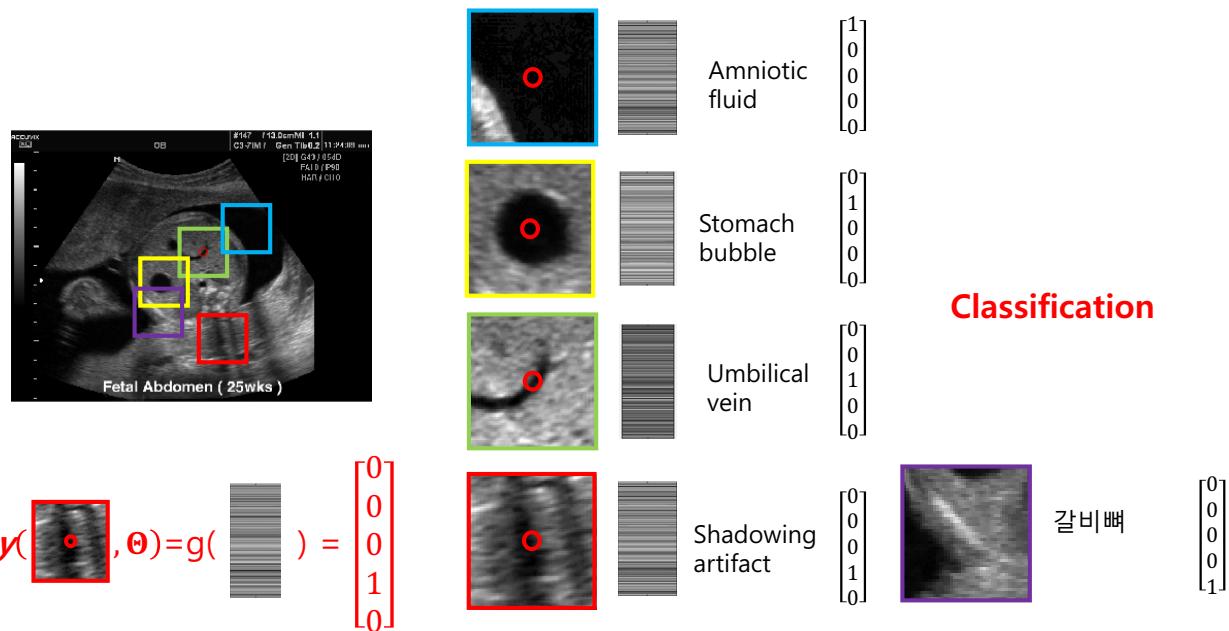
Bukweon Kim Ph.D



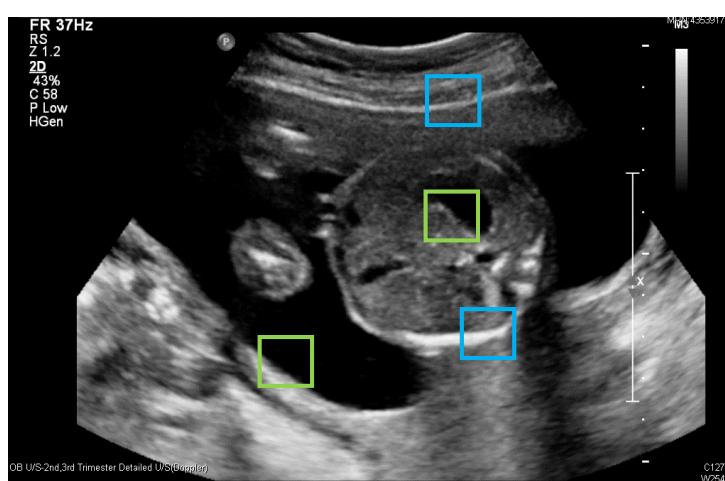
Sung Min Lee Ph.D

MEDIVIEWSOFT

Goal : 딥러닝을 이용한 해부학적 특징 추적

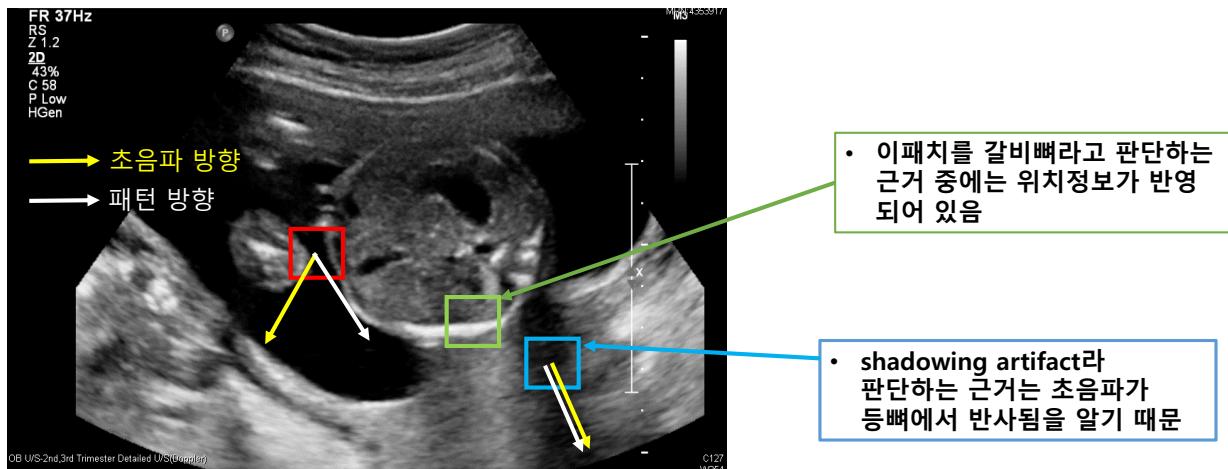


기존 CNN 방식의 단순 적용은 실패 가능성 높음

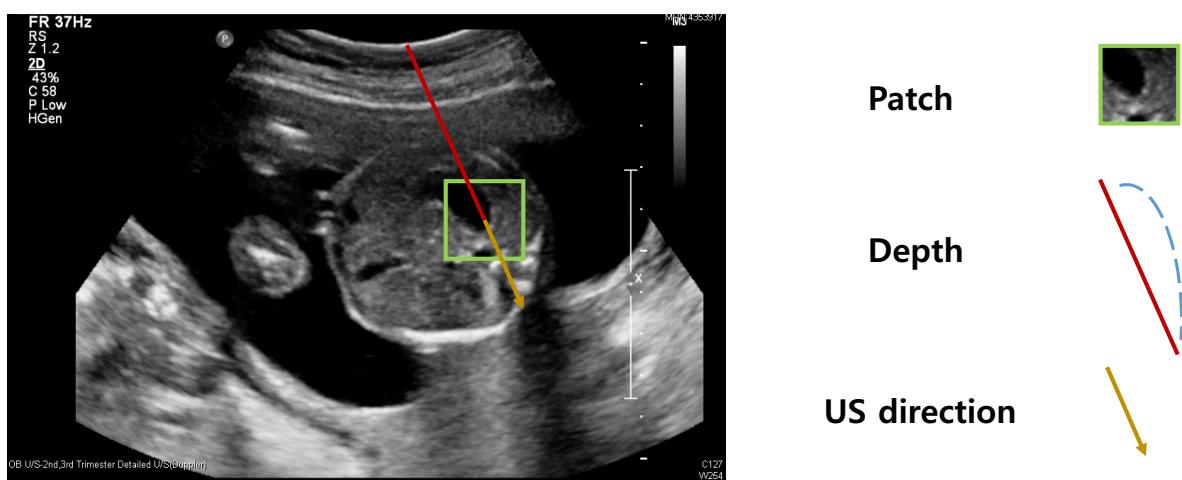


- 기존 CNN 접근 방식: 이미지 패치만을 분석하여 분류함
- ✓ 초음파 영상에 있어선 이미지 패치만을 바라보는 방법은 전문가의 판단 과정을 적절히 반영하지 않음

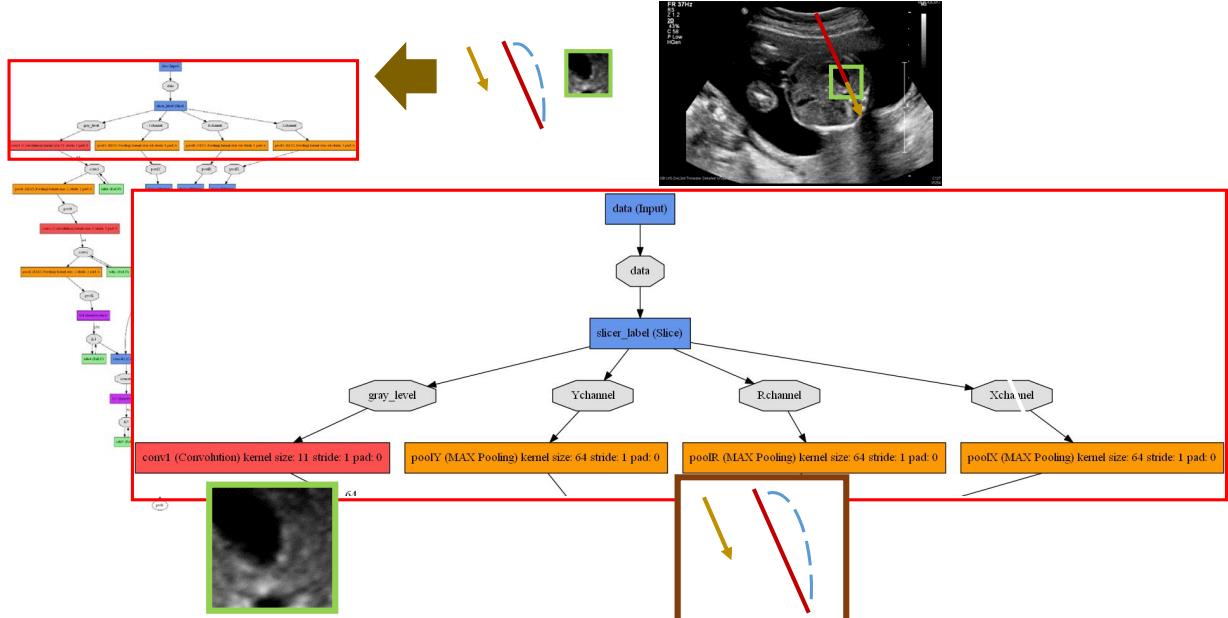
초음파의 진행 방향, 모델의 특성, 영상왜곡을 모두 반영하는 딥러닝 방식 개발 필요



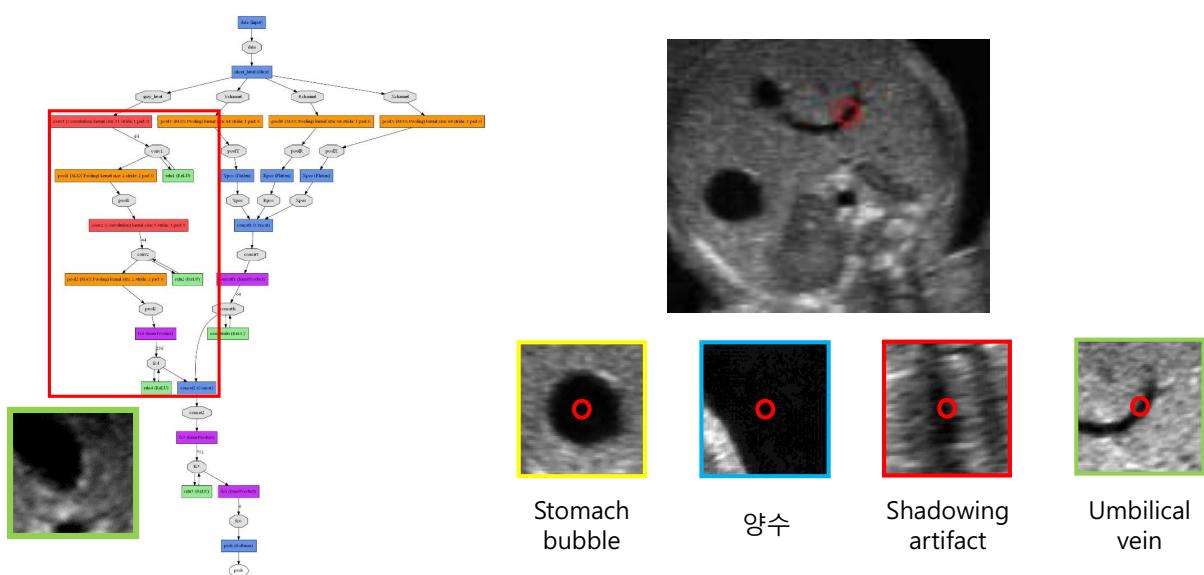
Input data



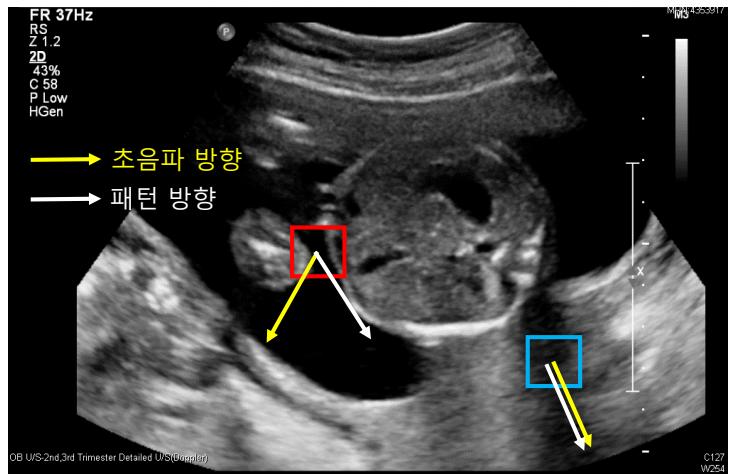
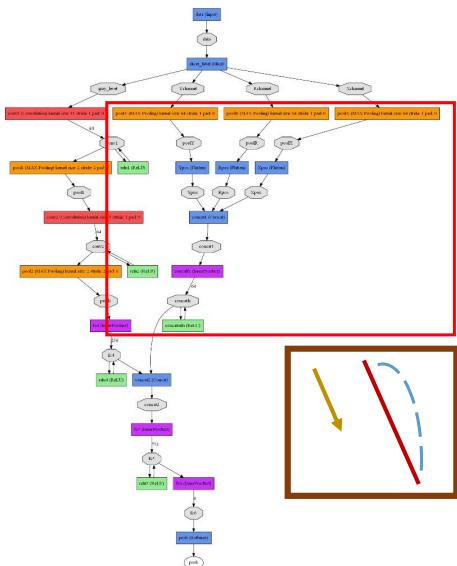
제안 1: 딥러닝에 이미지 특징과 초음파 진행 방향 반영



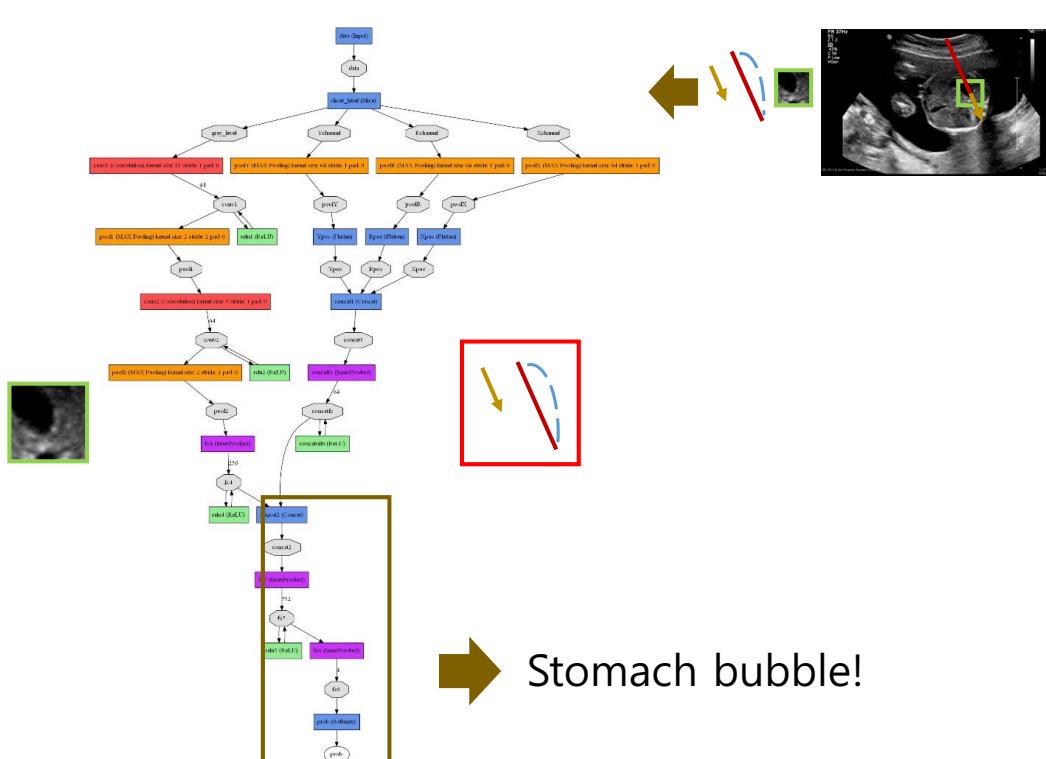
제안 1: 딥러닝에 이미지 특징과 초음파 진행 방향 반영



제안 1: 딥러닝에 이미지 특징과 초음파 진행 방향 반영

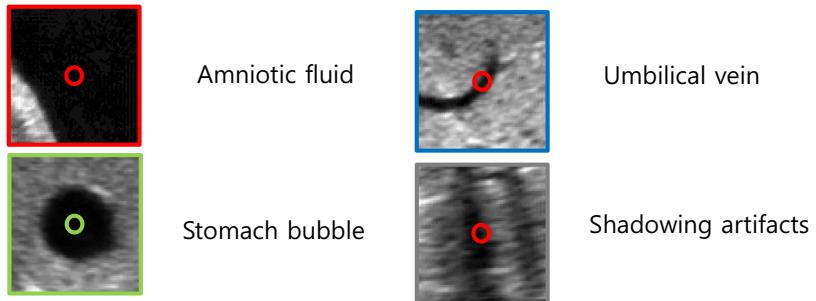


분류



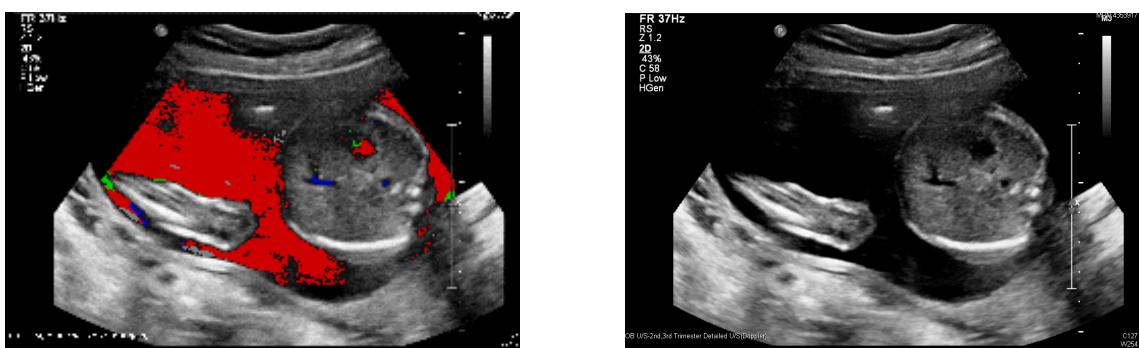
트레이닝 환경

- Caffe
- Ultrasound data from Severance hospital
- # of training data ~ 4000 (아래와 같은 4가지)
- ADAM for optimization of softmax loss function



Results

- Color : Amniotic fluid, Stomach bubble (SB), Umbilical vein (UV),
Shadowing artifact



연세대학교 계산과학공학과

<http://cse.yonsei.ac.kr/>

계산과학공학 대학원 프로그램

Graduate Program in Computational Science and Engineering

연세대 계산과학공학과는 공학, 의료영상, 자연과학 및 산업분야에서 제기되는 문제를 수리모델링–시뮬레이션–시각화과정을 통하여 예측/분석, 사용자에게 효율적인 프로그램 개발, 산업화에 이르기까지 다단계 연구를 종합적으로 수행하는 계산과학중심의 대학원 과정입니다. (BK21플러스 선정 학과 : 글로벌 인재양성형 계산과학공학사업단)

CSE tries to provide multidisciplinary areas combining the traditional fields of engineering – electrical, mechanical, biomedical, and material sciences.

Our programs focus on modeling–computer simulation–visualization, based on applied mathematics.

- Ph.D/MS CSE – Mathematics
- Ph.D/MS CSE – Mechanical/Electrical Engineering
- MS CSE – Industrial Mathematics



MEDIVIEWSOFT, located at CSE department in Yonsei University, aims to provide networking opportunities for transparent communications on recent developments, emerging issues and challenging problems in mathematic-based medical imaging area. Mediviewsoft helps its members launch venture-ready startups. and regularly organize events and meetings to pitch member's venture idea. Haeeun Han <http://mediviewsoft.com/>