Open DMQA Seminar

# Various Normalization Techniques for Deep Learning

**김정인**

Data Mining & Quality Analytics Lab

2022.05.13(금)

고려대학교 KOREA UNIVERSITY

Data Mining Quality Analytics

# 발표자 소개

- **김정인 (Jungin Kim)**
  - ✓ 고려대학교 산업경영공학과
  - ✓ Data Mining & Quality Analytics Lab. (김성범 교수님)
  - ✓ 석박통합과정(2021 09~)
- **관심분야**
  - ✓ Machine learning / Deep learning Algorithms
  - ✓ Self-Supervised Learning
- **이메일**
  - ✓ jungin_kim23@korea.ac.kr

# 목차

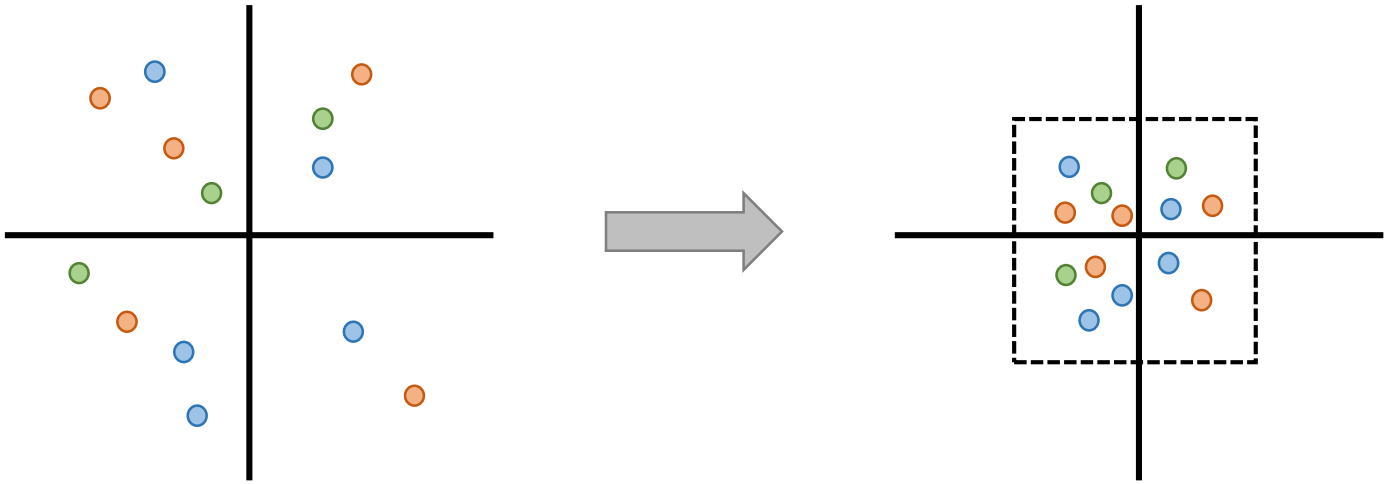

1. **Introduction**
    - What is Feature Scaling?
    - Normalization vs Standardization
2. **Methods**
    - Batch Normalization
    - Layer Normalization
    - Instance Normalization
    - Group Normalization
3. **Conclusion**
    - Summary
4. **Appendix**


# Introduction

What is Feature Scaling?

❖ **Feature Scaling 이란?**

- Feature Scaling : 서로 다른 변수의 값에 대한 범위를 일정한 수준으로 맞추는 작업
- 대표적인 방법으로 Normalization과 Standardization 존재



# Introduction

What is Feature Scaling?

❖ **Feature Scaling 이란?**

- Feature Scaling : 서로 다른 변수의 값에 대한 범위를 일정한 수준으로 맞추는 작업

| 이름 | 키(ft) | 몸무게(lbs) | 상의 사이즈 |
|---|---|---|---|
| 박진혁 | 4.7 | 120 | S |
| 정진용 | 7.1 | 145 | ? |
| 김창현 | 5.9 | 180 | L |
| 정재윤 | 6.0 | 185 | L |

**직관적으로 창현, 재윤에 비해 몸무게는 작지만 키가 더 크기 때문에 "L"**

# Introduction

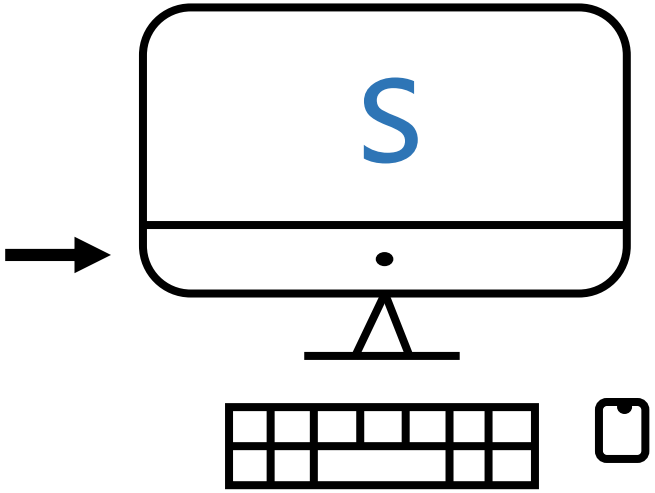
What is Feature Scaling?

❖ **Feature Scaling 이란?**

- Feature Scaling : 서로 다른 변수의 값에 대한 범위를 일정한 수준으로 맞추는 작업

**컴퓨터는 어떻게 예측할까?**

| 이름 | 키(ft) | 몸무게(lbs) | 상의 사이즈 | 키+몸무게 |
|---|---|---|---|---|
| 박진혁 | 5.2 | 115 | S | 124.7 |
| 정진용 | 6.1 | 140 | ? | 152.1 |
| 김창현 | 5.9 | 175 | L | 185.9 |
| 정재윤 | 6.0 | 174 | L | 191.0 |



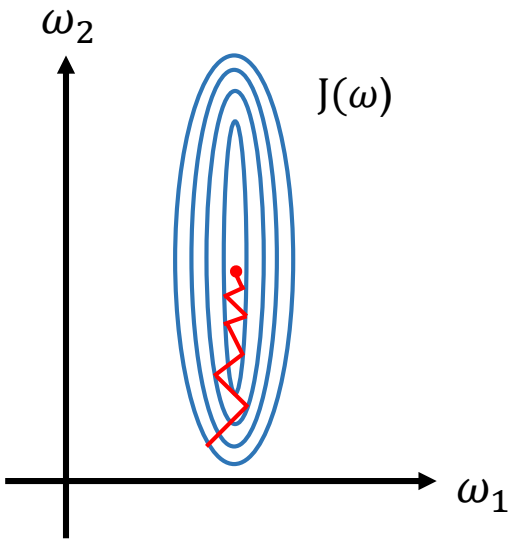

**정확한 분석을 위해 Feature를 서로 정규화 시켜 주는 작업 필요 → Feature Scaling**

# Introduction

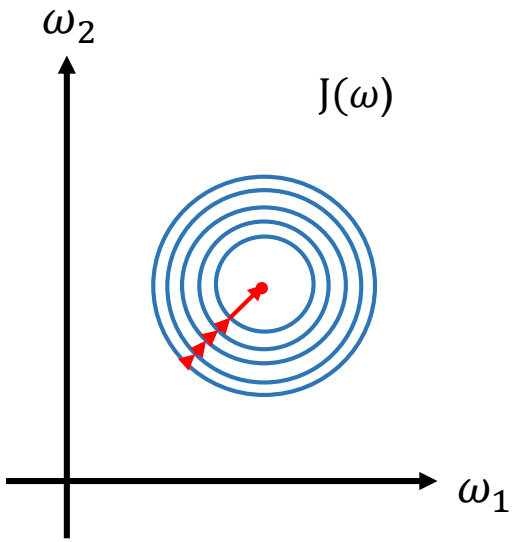
What is Feature Scaling?

❖ **Feature Scaling 장점**

- 인공신경망의 경사하강법 알고리즘의 수렴 속도를 빠르게 함
  - ✓ 스케일링을 하지 않는 경우 알고리즘이 목표로 하는 지점까지 찾아가는데 오래 걸림
  - ✓ 스케일링을 하게 되면 목표 지점까지 좀 더 빠르게 도달 가능




Scaling하지 않은 경사하강법




Scaling을 사용한 경사하강법

# Introduction

Normalization vs Standardization

❖ **Normalization (정규화) – MinMax Scaling**

- 입력된 데이터들을 모두 0~1 사이의 값으로 변환하는 방법

| 이름 | 키(ft) | 몸무게(lbs) |
|---|---|---|
| 김정인 | 5.2 | 115 |
| 정진용 | 6.1 | 140 |
| 김창현 | 5.9 | 175 |
| 정재윤 | 6.0 | 174 |

$$\longrightarrow X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}} \longrightarrow$$

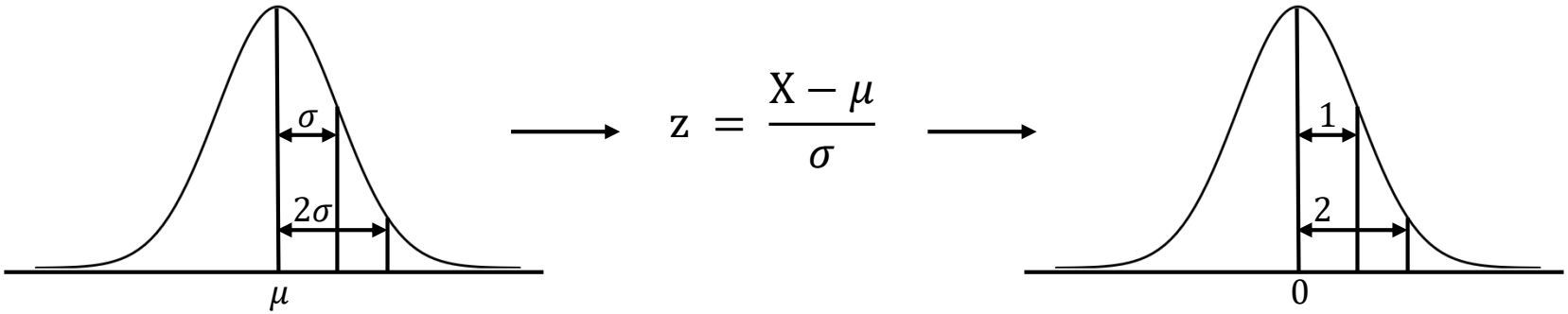
| 이름 | 키(ft) | 몸무게(lbs) |
|---|---|---|
| 김정인 | 0 | 0 |
| 정진용 | 1 | 0.42 |
| 김창현 | 0.78 | 1 |
| 정재윤 | 0.89 | 0.98 |

# Introduction

Normalization vs Standardization

❖ **Standardization (표준화)**

- 입력된 데이터들의 정규 분포를 평균이 0 이고 분산이 1 인 표준 정규 분포로 변환하는 방법
- 데이터들이 정규 분포를 따른다고 가정




기존 데이터 표준화된 데이터

# Methods

## Batch Normalization

❖ **Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift**

- Google에서 연구한 논문이며 2022년 05월 13일 기준 36407회 인용됨
- Gradient Vanishing/Exploding 문제를 방지하기 위한 획기적인 방법
- 학습 속도를 비약적으로 향상시킨 방법

---

**Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift**

---

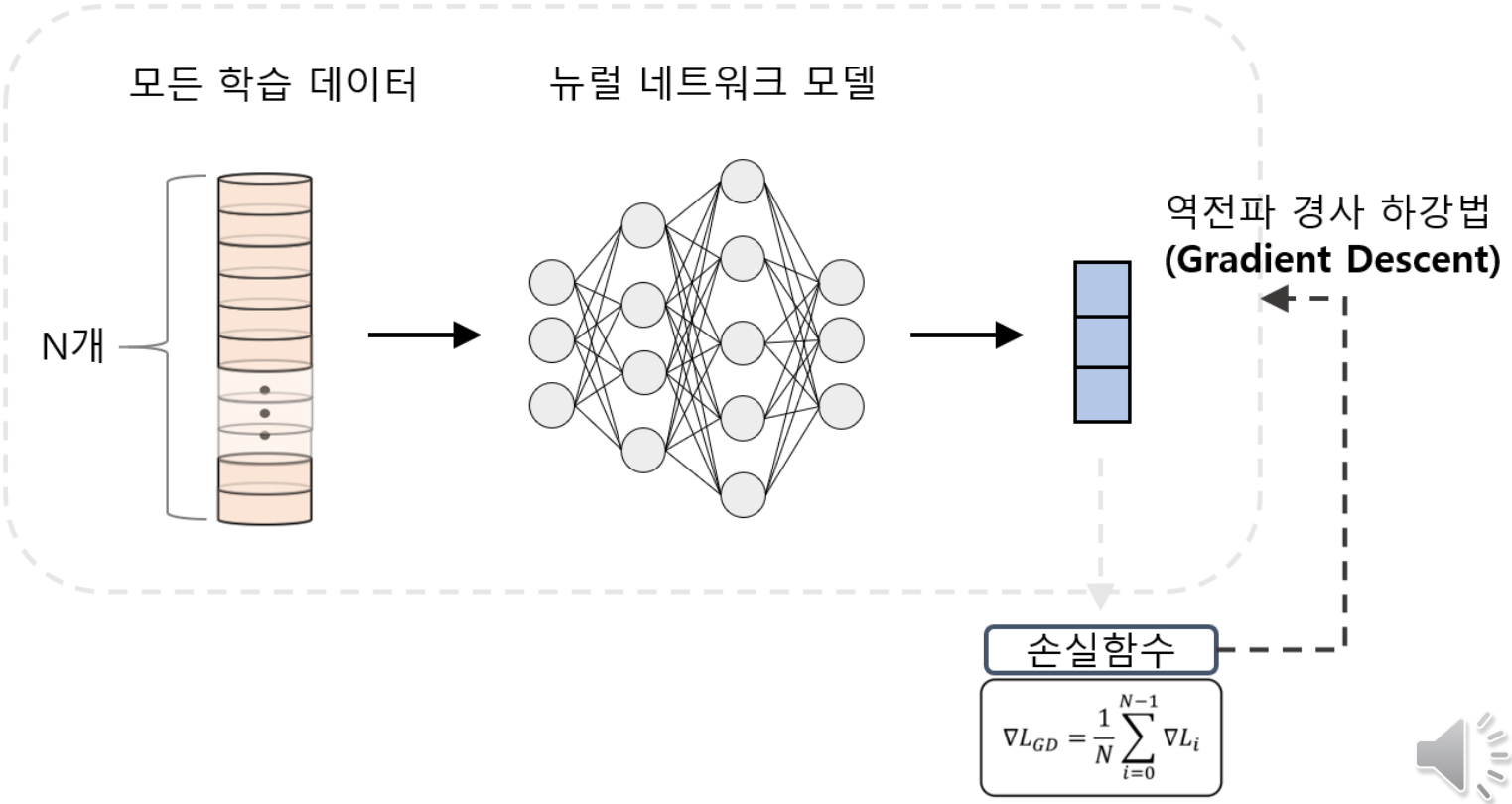
**Sergey Ioffe** SIOFFE@GOOGLE.COM
**Christian Szegedy** SZEGEDY@GOOGLE.COM
Google, 1600 Amphitheatre Pkwy, Mountain View, CA 94043

# Methods

## Batch Normalization

### ❖ Gradient Descent

- 일반적인 gradient descent에서는 gradient를 한 번 업데이트 하기 위해 모든 학습 데이터 사용
- 대용량의 데이터를 한 번에 처리하지 못하기 때문에 데이터를 batch 단위로 나눠서 학습

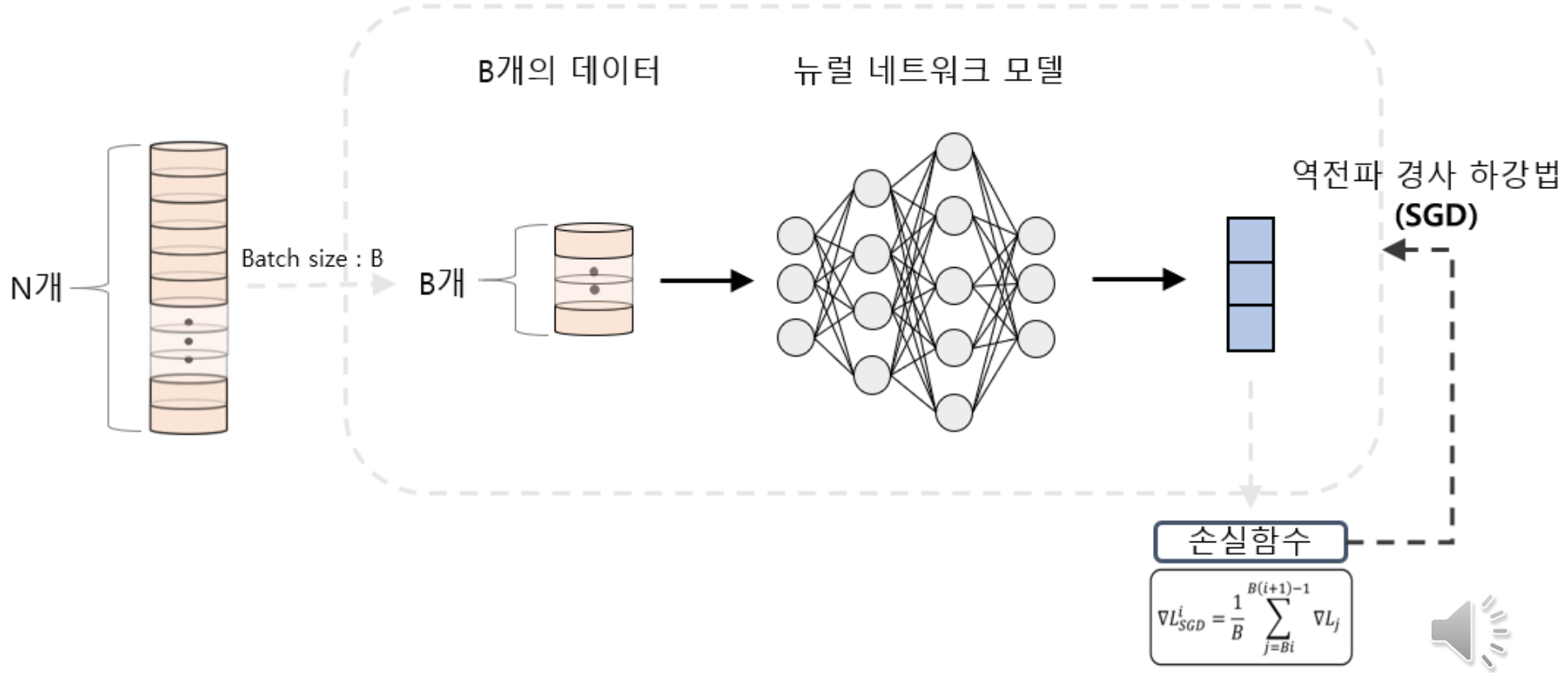


# Methods

## Batch Normalization

❖ **Stochastic Gradient Descent**

- Gradient를 업데이트 하기 위하여 일부의 데이터만을 사용 → Batch size 만큼 사용
- 한 번 업데이트 하는데 B개의 데이터를 사용하기 때문에 평균을 낼 때에도 B로 나눔

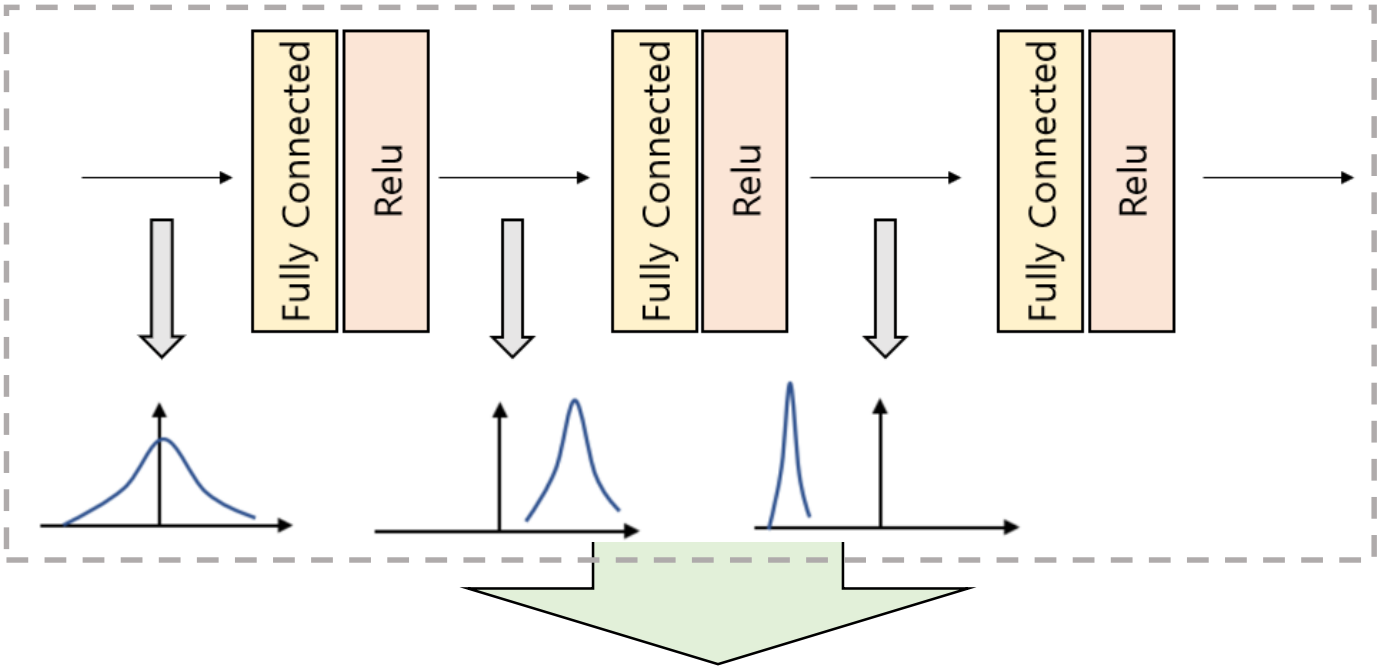


# Methods

Batch Normalization

❖ **Internal Covariant Shift**

- Internal Covariant Shift : 학습 과정에서 계층 별로 입력의 데이터 분포가 달라지는 현상
- 각 연산을 적용한 뒤 전과 후로 데이터 간의 분포가 달라질 수 있음
- SGD를 이용하여 Batch 단위로 학습하게 되면 Batch 간의 데이터 분포가 상이해지는 문제 발생

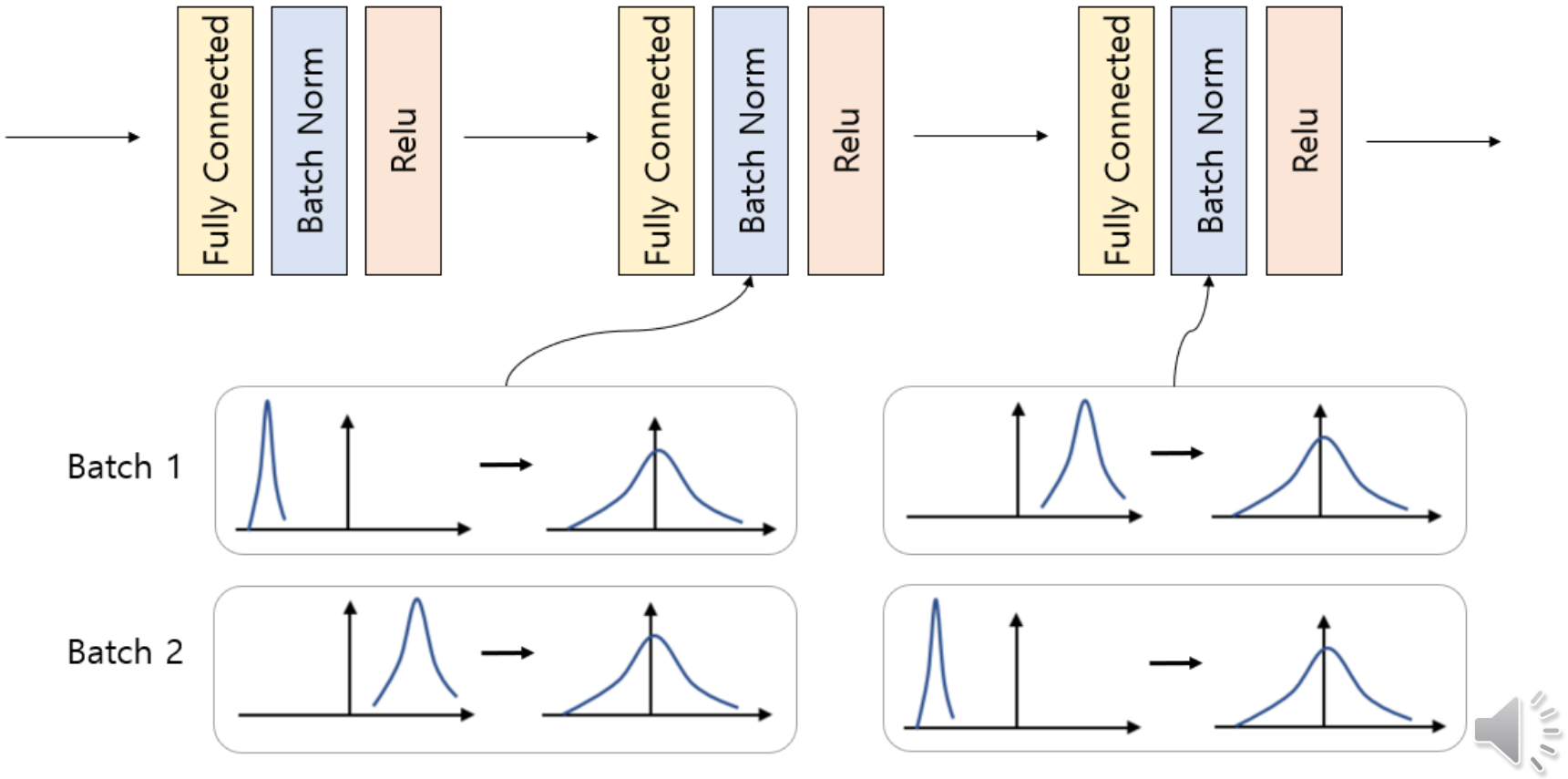


# Methods

Batch Normalization

❖ **Batch Normalization**

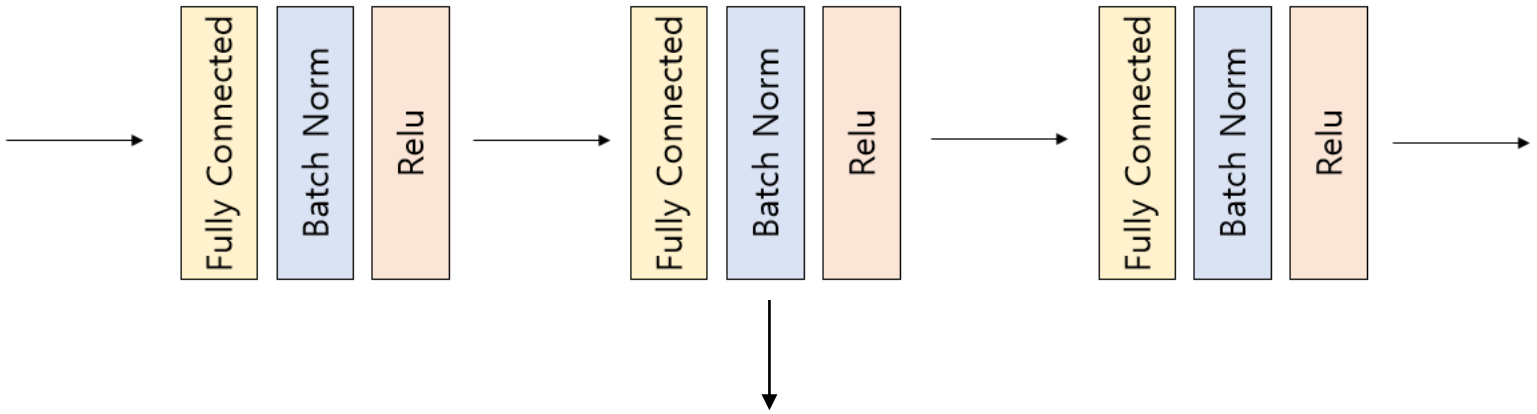
- Batch Normalization : 각 배치별로 평균과 분산을 이용해 정규화하는 것을 뜻함



# Methods

## Batch Normalization

❖ **Batch Normalization - Training**

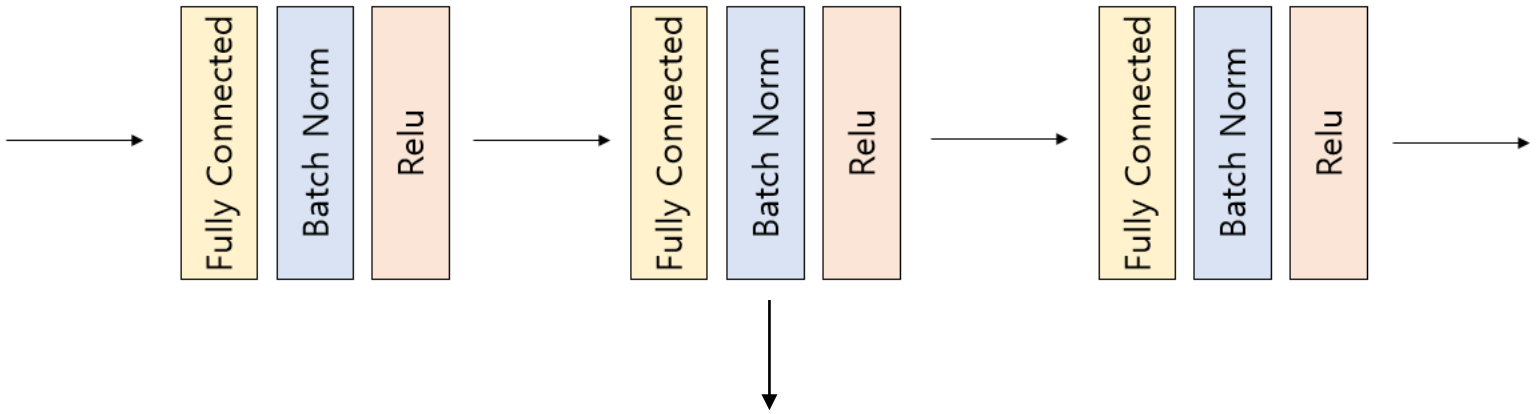



1. $\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i$
2. $\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2$
3. $Normalization: \hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$
4. $y_i = \gamma\hat{x}_i + \beta, \quad \gamma, \beta \rightarrow parameter$

# Methods

## Batch Normalization

❖ **Batch Normalization - Inference**



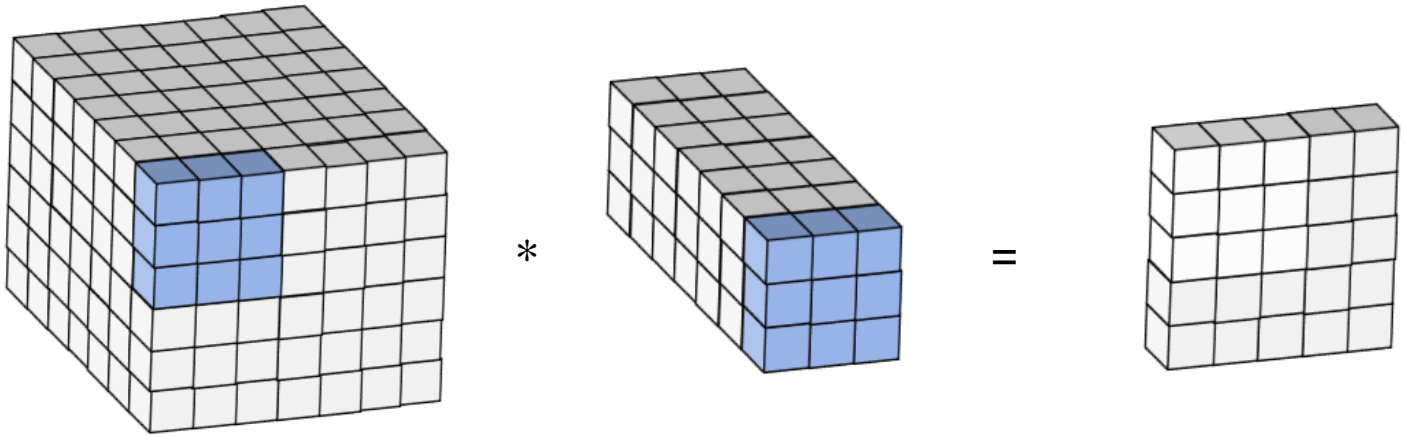

1. $\boldsymbol{\mu_B = E[\mu_B]}$
2. $\boldsymbol{\sigma_B^2 = E[\sigma_B^2] \times \frac{m}{m-1}}$
3. $Normalization : \widehat{x_i} = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$
4. $y_i = \gamma \widehat{x_i} + \beta, \ \ \gamma, \beta \ \rightarrow parameter$

# Methods

Batch Normalization

❖ **Batch Normalization**

**Convolution layer에서의 Batch Normalization?**




Input
(7x7x7)

Filter
(3x3x7)

Feature map

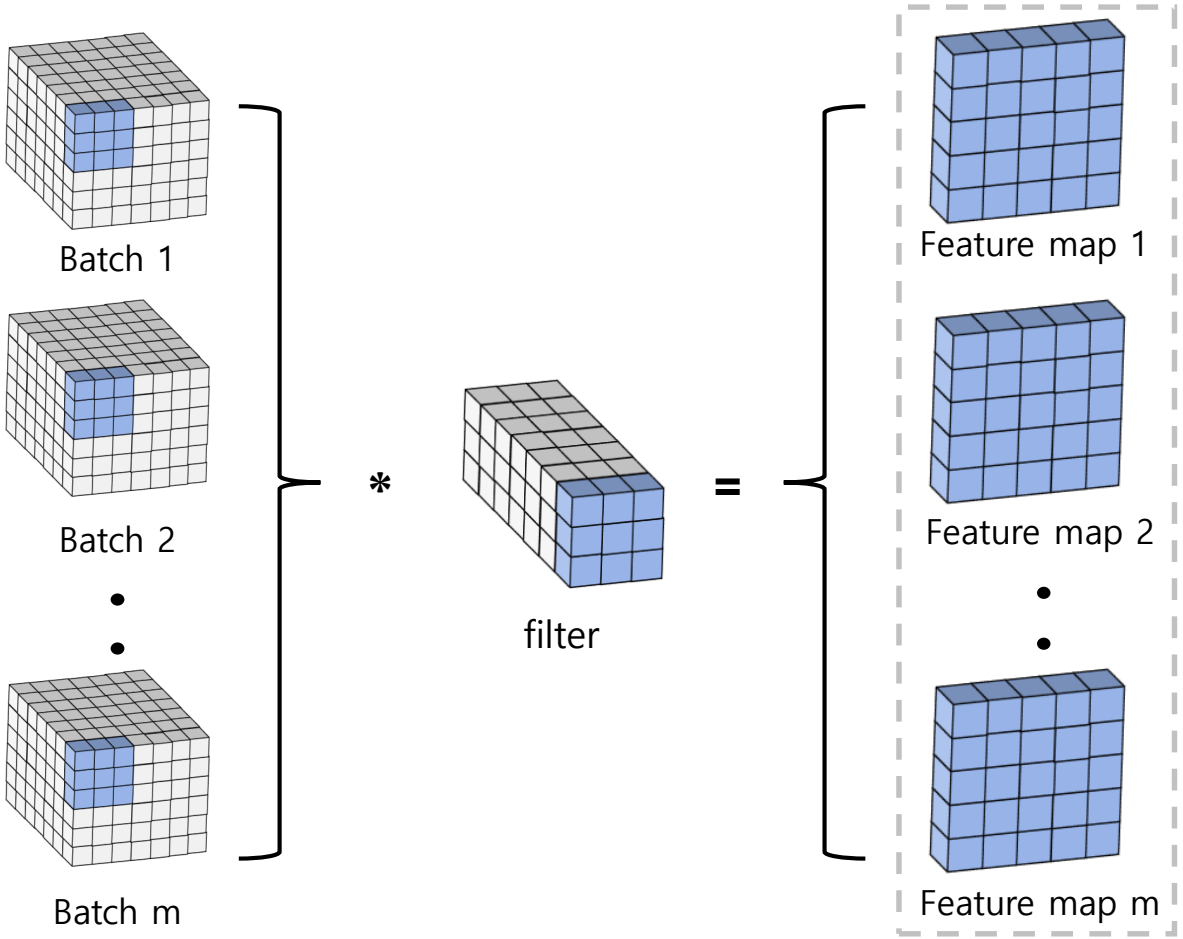
**m개의 입력 배치에 대해 m개의 feature map 도출**

# Methods

## Batch Normalization

❖ **Batch Normalization**

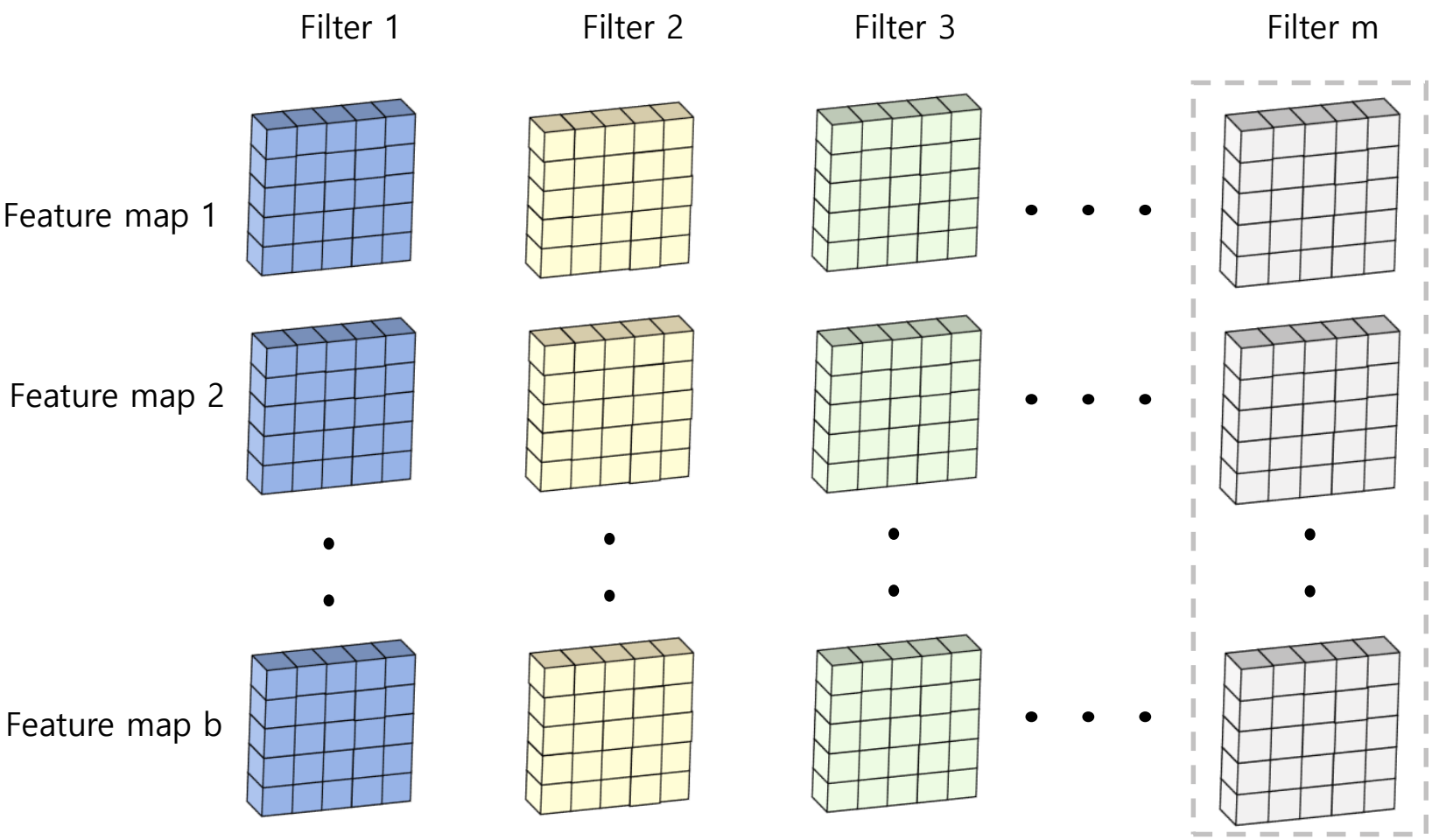
- m개의 입력 배치에 대해 m개의 feature map 도출



# Methods

## Batch Normalization

❖ **Batch Normalization**

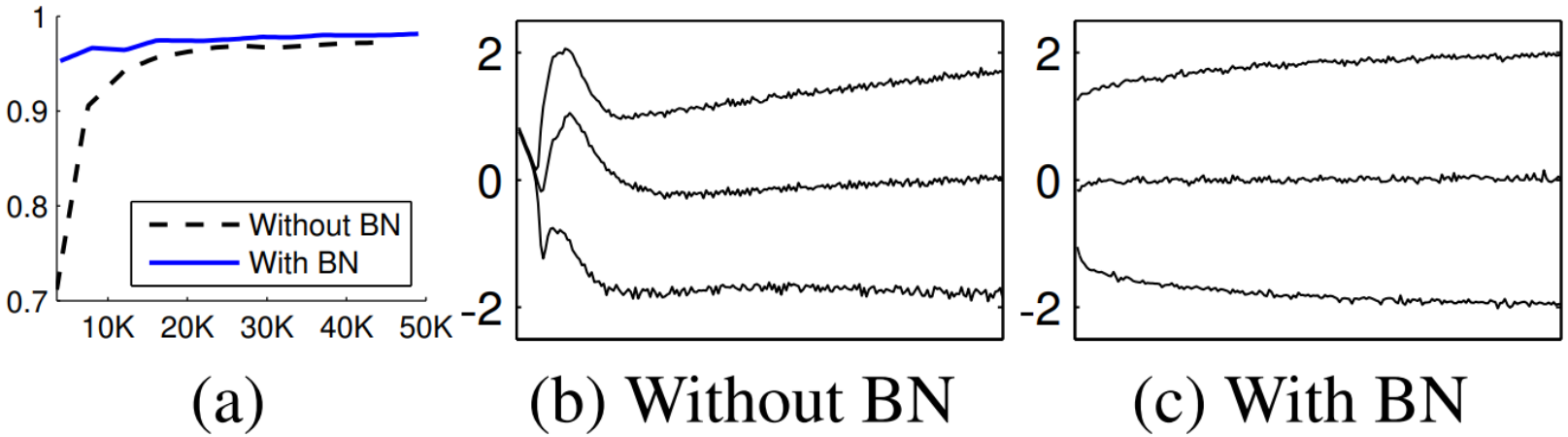


# Methods

## Batch Normalization

❖ **Experiment - Mnist**

- (a) : Batch Normalization(BN)을 사용했을 때 더 빨리 수렴하고 accuracy도 높다는 것을 확인
- (b) : BN을 사용하지 않았을 때 sigmoid function의 입력 분포가 불안정적임을 확인
- (c) : BN을 사용했을 때 sigmoid function의 입력 분포가 안정적임을 확인



# Methods

Layer Normalization

❖ **Layer Normalization**

- 토론토 대학에서 연구한 논문이며 2022년 05월 13일 기준으로 5019회 인용
- Batch Normalization의 단점을 보완한 방법론
  - ✓ Sequence data 적용의 어려움
  - ✓ 배치 사이즈가 작다면 전체 데이터 셋 표현에 어려움
  - ✓ 시퀀스의 길이에 따라 입력이 종종 달라지므로 RNN에 적용하기 어려움

→ Batch에 대한 의존성이 높기 때문에 발생하는 문제

**Layer Normalization**

**Jimmy Lei Ba**
University of Toronto
jimmy@psi.toronto.edu

**Jamie Ryan Kiros**
University of Toronto
rkiros@cs.toronto.edu

**Geoffrey E. Hinton**
University of Toronto
and Google Inc.
hinton@cs.toronto.edu

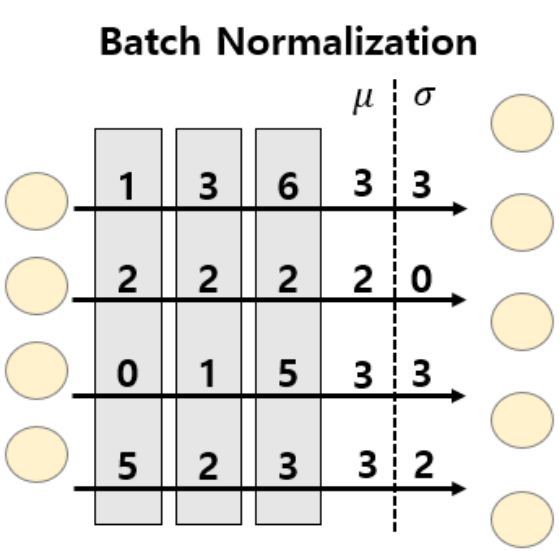
# Methods

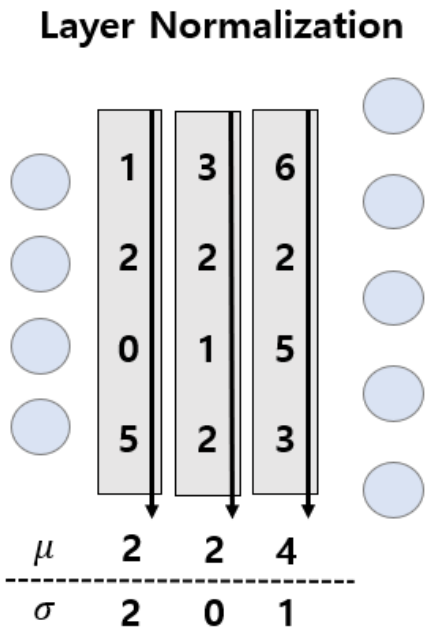
Layer Normalization

❖ **Layer Normalization**

- Batch에 대한 의존도를 제거함
- Batch가 아닌 Layer에 기반하여 Normalization 수행

→ Batch 기반해서 계산하지 않기 때문에 training, testing 할 때 동일하게 계산



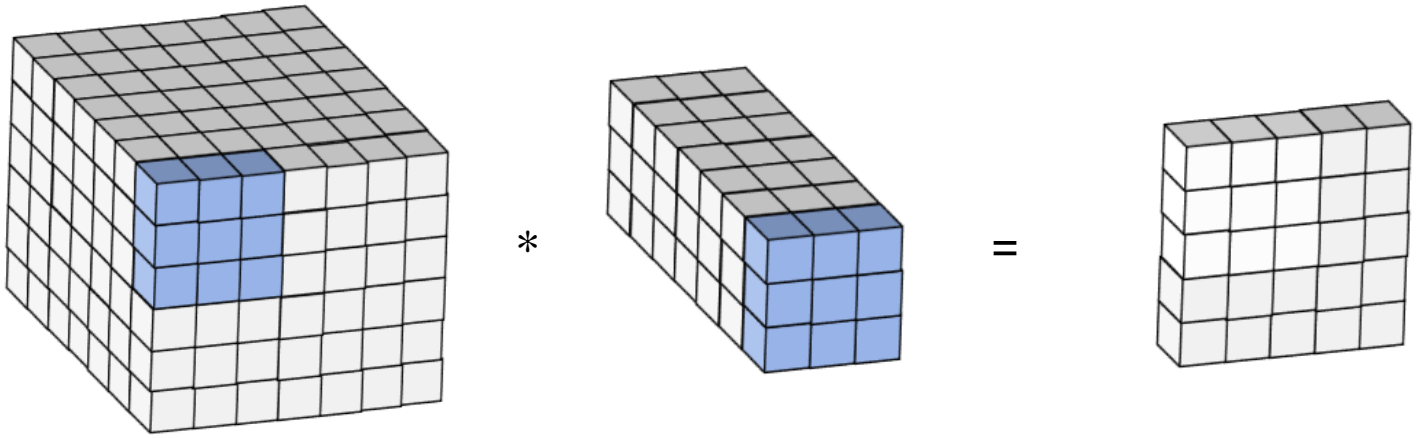



# Methods

Layer Normalization

❖ **Layer Normalization**

**Convolution layer에서의 Layer Normalization?**




Input
(7x7x7)

Filter
(3x3x7)

Feature map

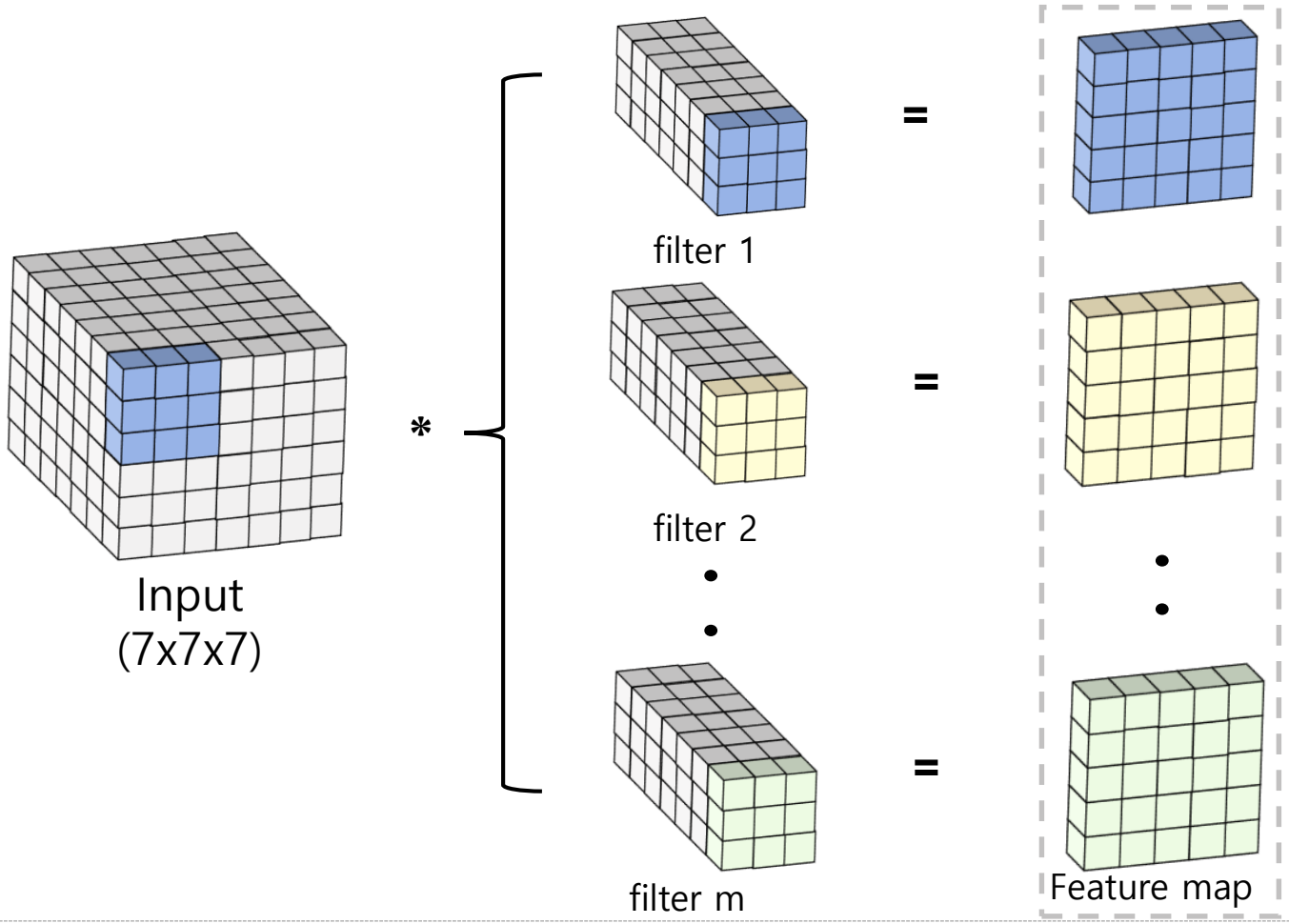
**m개의 filter가 있다면 m개의 feature map이 결과로 도출**

# Methods

## Layer Normalization

❖ **Layer Normalization**

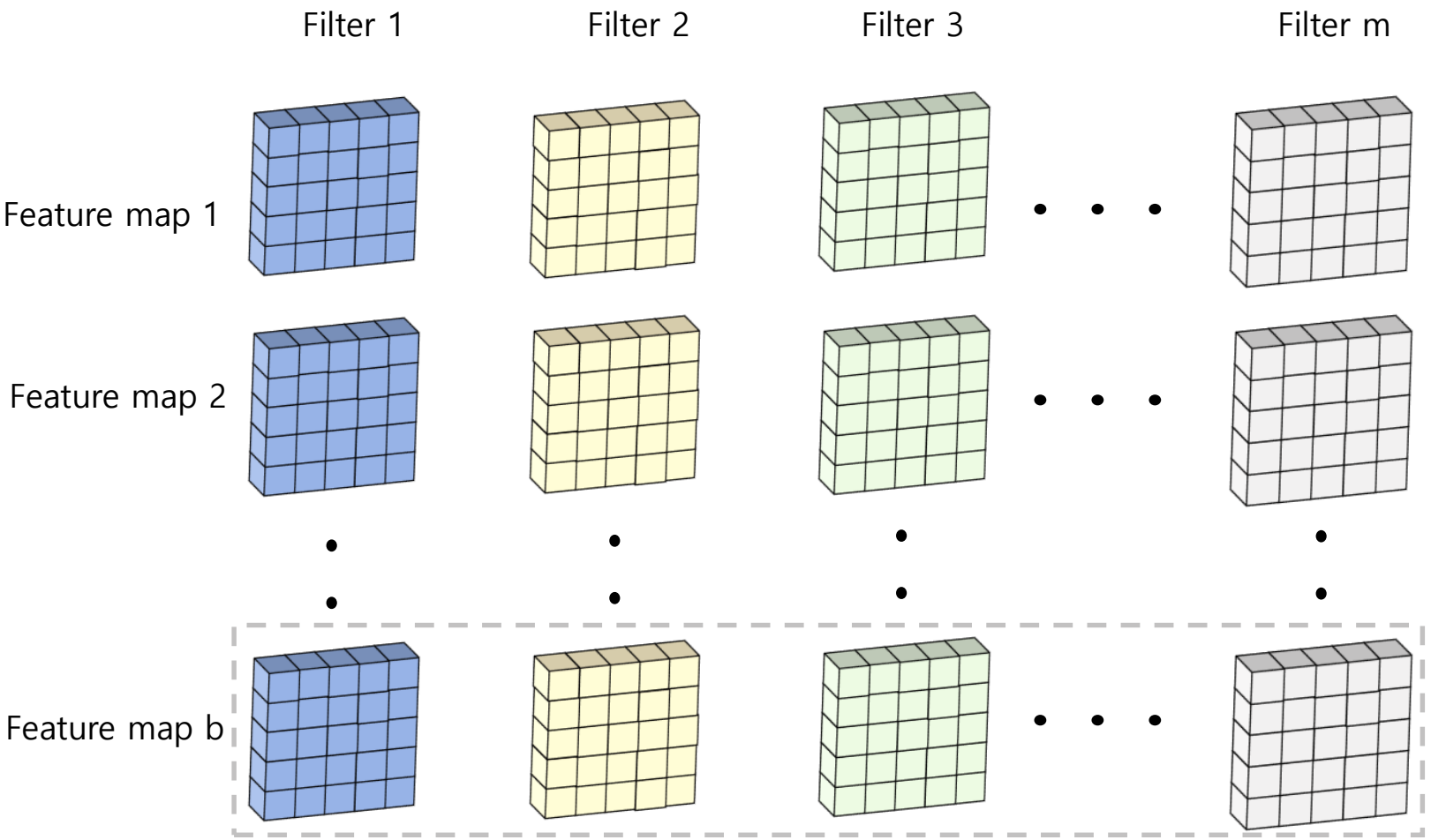
- m개의 filter가 있다면 m개의 feature map이 결과로 도출



# Methods

## Layer Normalization

❖ **Layer Normalization**

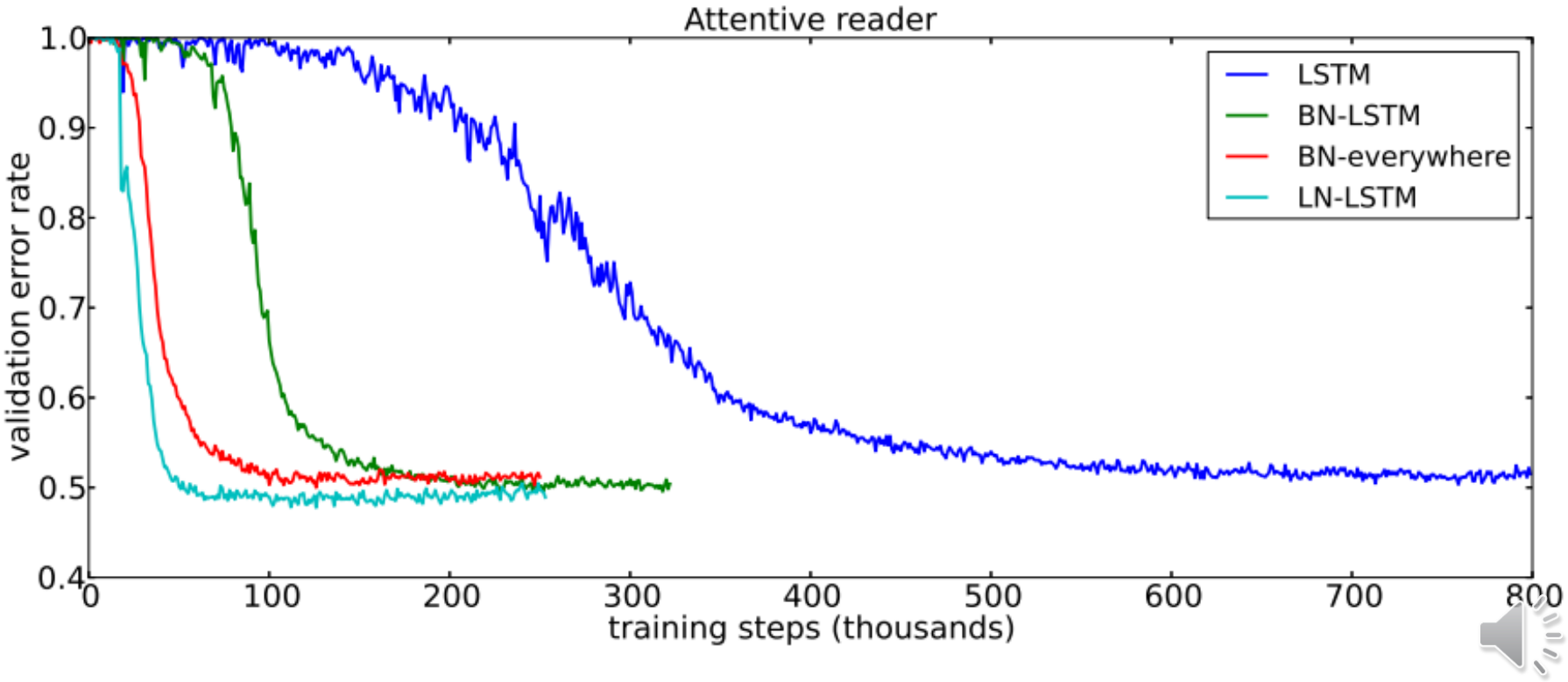


# Methods

Layer Normalization

❖ **Experiment**

- 질문에 포함된 빈칸에 대한 알맞은 단어를 예측하는 실험
- BN : scale parameter initialization 에 큰 영향
- LN : scale parameter initialization에 큰 영향 받지 않음



# Methods

Instance Normalization

❖ **Instance Normalization : The Missing Ingredient for Fast Stylization**

- 러시아 사립 연구소에서 연구했으며 2022년 05월 13일 기준 2333회 인용
- Image Style Transfer의 성능을 올리기 위해 Instance Normalization 등장
- batch normalization보다 instance normalization을 사용했을 때 성능 향상

**Instance Normalization:**
**The Missing Ingredient for Fast Stylization**


**Dmitry Ulyanov**
Computer Vision Group
Skoltech & Yandex
Russia
dmitry.ulyanov@skoltech.ru

**Andrea Vedaldi**
Visual Geometry Group
University of Oxford
United Kingdom
vedaldi@robots.ox.ac.uk

**Victor Lempitsky**
Computer Vision Group
Skoltech
Russia
lempitsky@skoltech.ru

# Methods

## Instance Normalization

- **Style transfer**

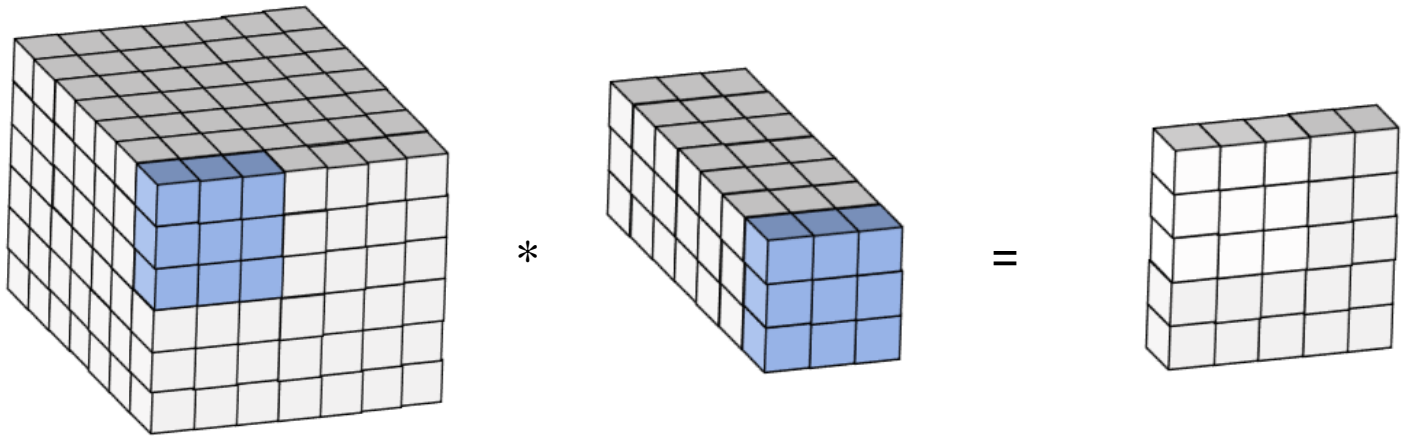
# Methods

Instance Normalization

❖ **Instance Normalization**

- 배치 단위가 아닌 각 이미지에 대해 개별적으로 normalization 진행

**Convolution layer에서의 Instance Normalization?**




Input
(7x7x7)

Filter
(3x3x7)

Feature map

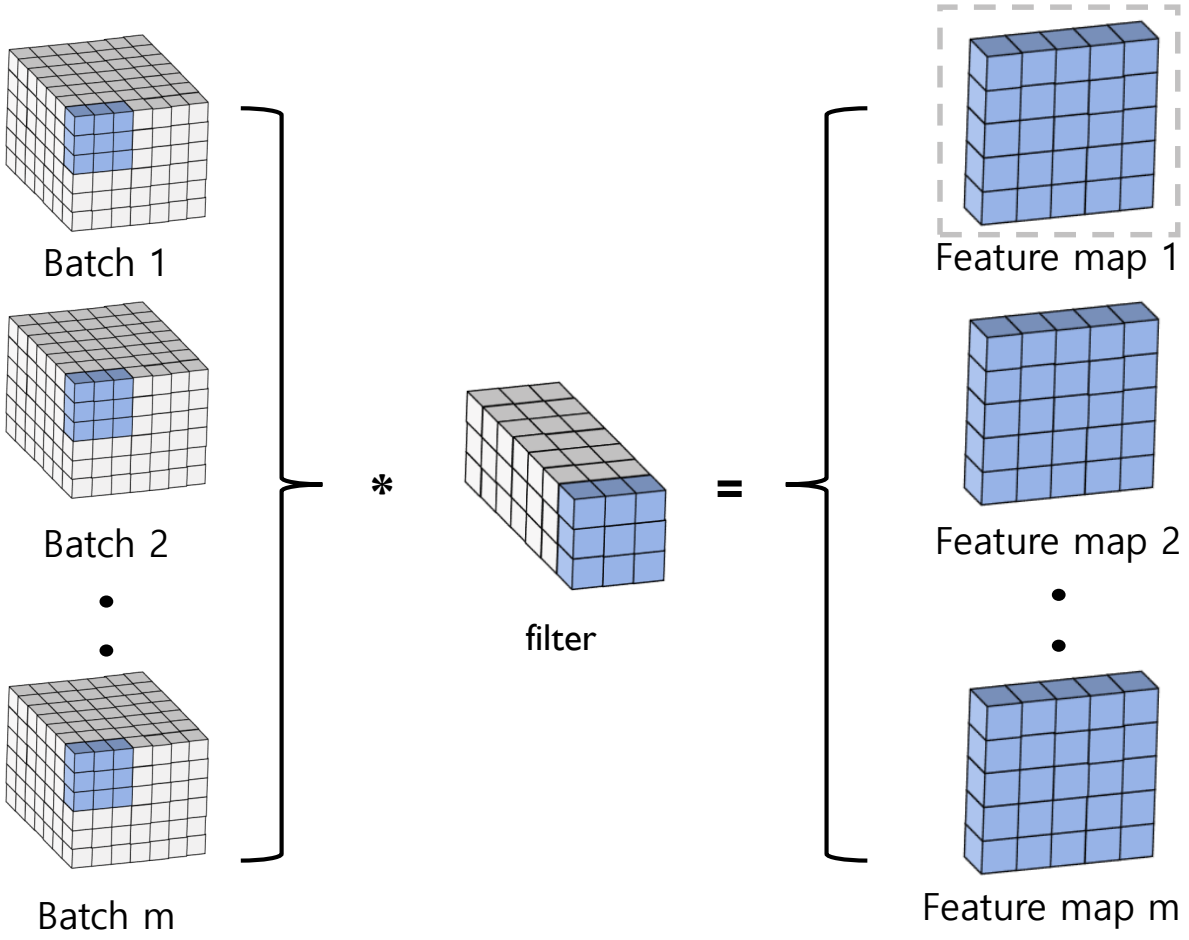
**m개의 입력 배치에 대해 m개의 feature map 도출**

# Methods

## Instance Normalization

❖ **Instance Normalization**

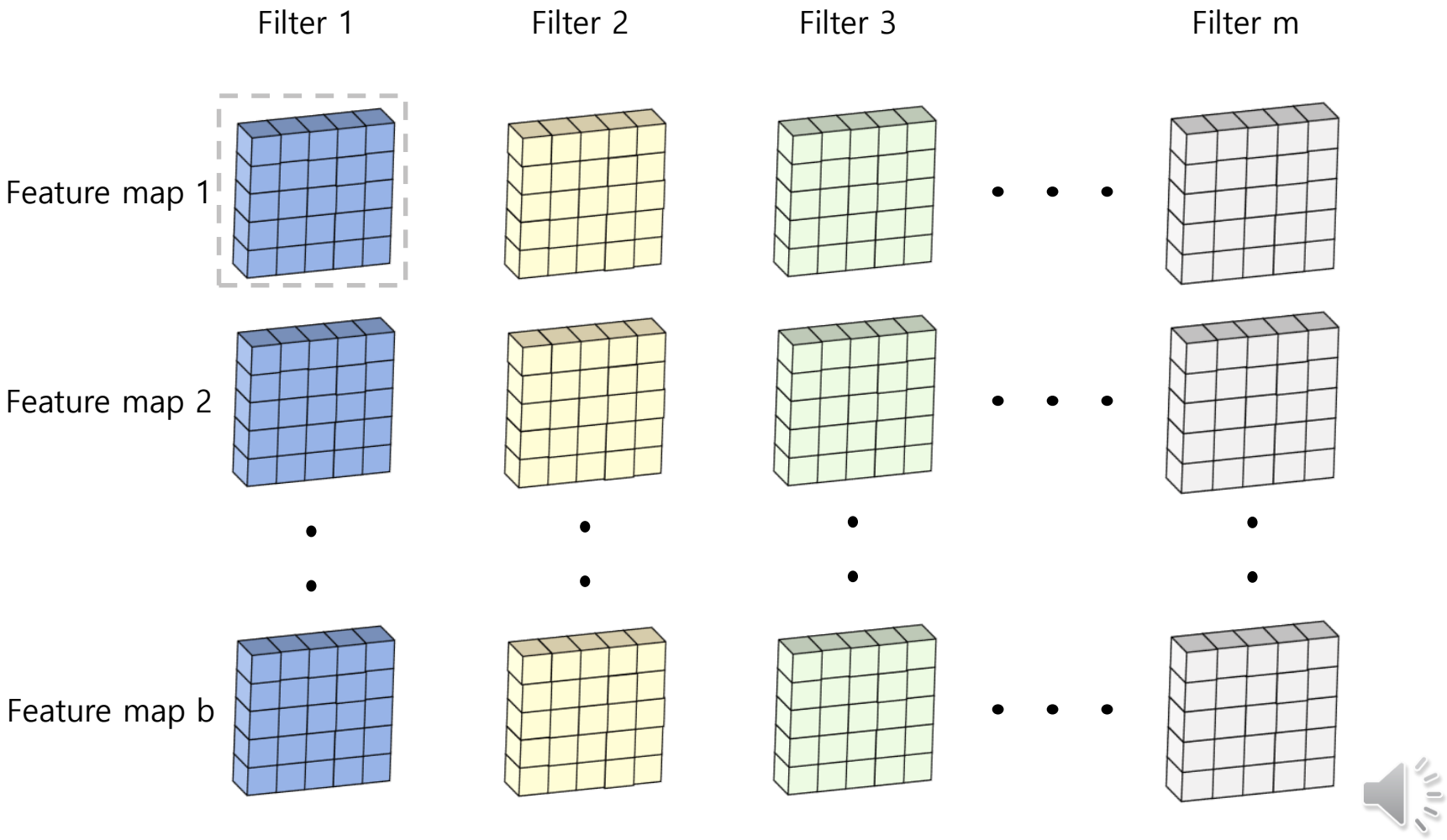
- m개의 입력 배치에 대해 m개의 feature map 도출



# Methods

## Instance Normalization

❖ **Instance Normalization**



# Methods

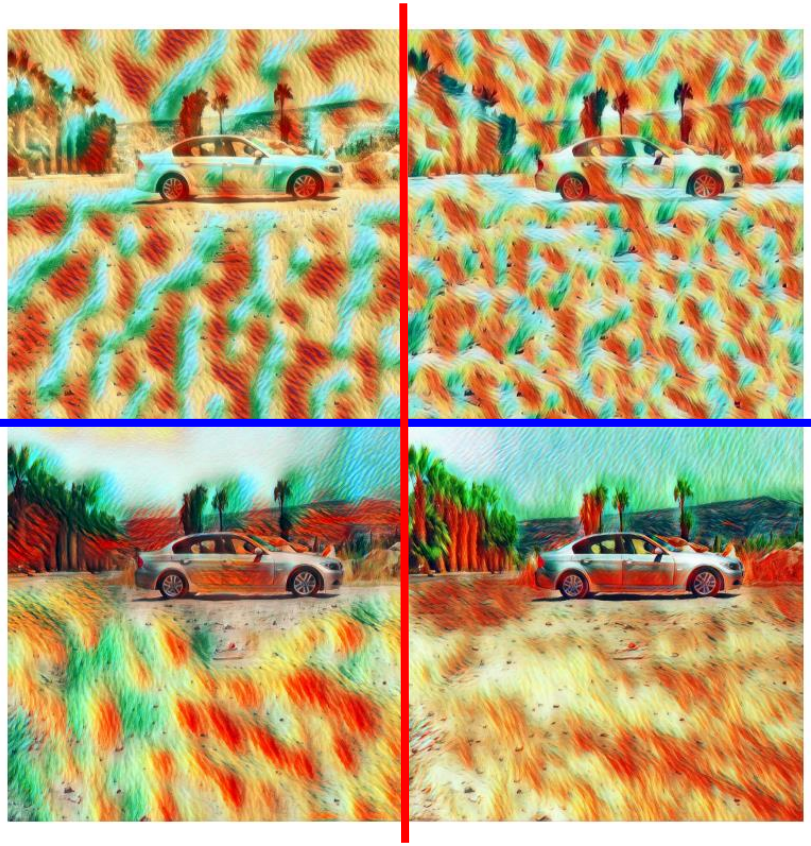
Instance Normalization

❖ **Experiment**

- 서로 다른 generator network를 사용해 BN과 IN의 성능 비교
- IN을 사용했을 때 두 개의 generator network의 성능이 더 개선됨

BN 사용

IN 사용

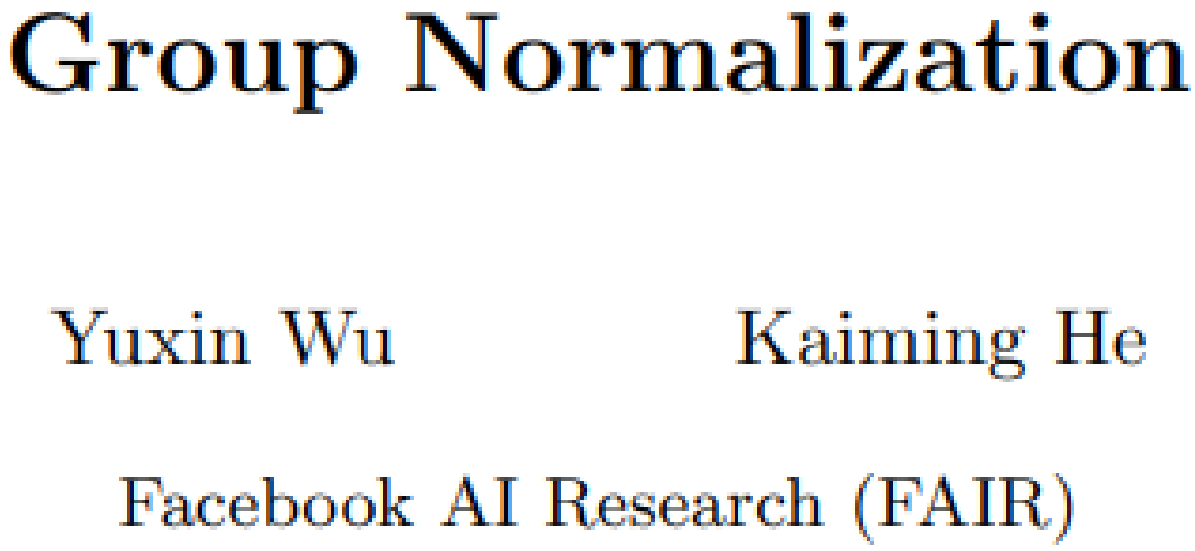


# Methods

## Group Normalization

❖ **Group Normalization**

- Facebook AI research(FAIR)에서 연구했으며 2022년 05월 13일 기준 2004회 인용
- Batch Normalization 단점을 개선하기 위해 제안된 방법론
  - ✓ Batch size(1 or 2)를 작게 설정한 경우 상당한 성능 약화 문제 발생
  - ✓ Object Detection or Segmentation 같은 task에서 고해상도 이미지 사용
- Batch 크기가 극도로 작은 상황에서 batch normalization 대신 사용하면 좋은 normalization 방법론 제안

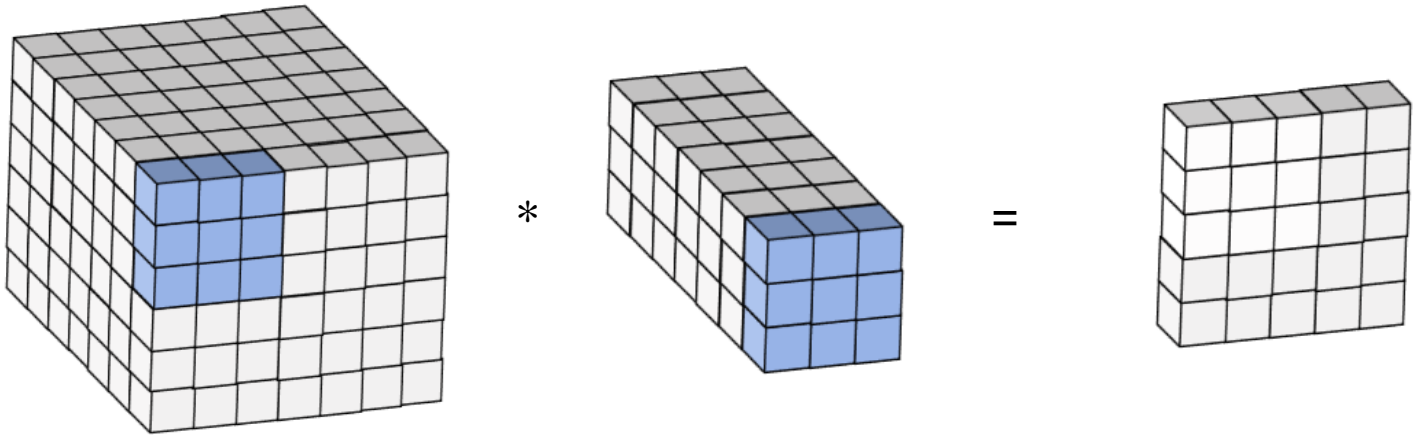


# Methods

Group Normalization

❖ **Group Normalization**

**Convolution layer에서의 Group Normalization?**




Input
(7x7x7)

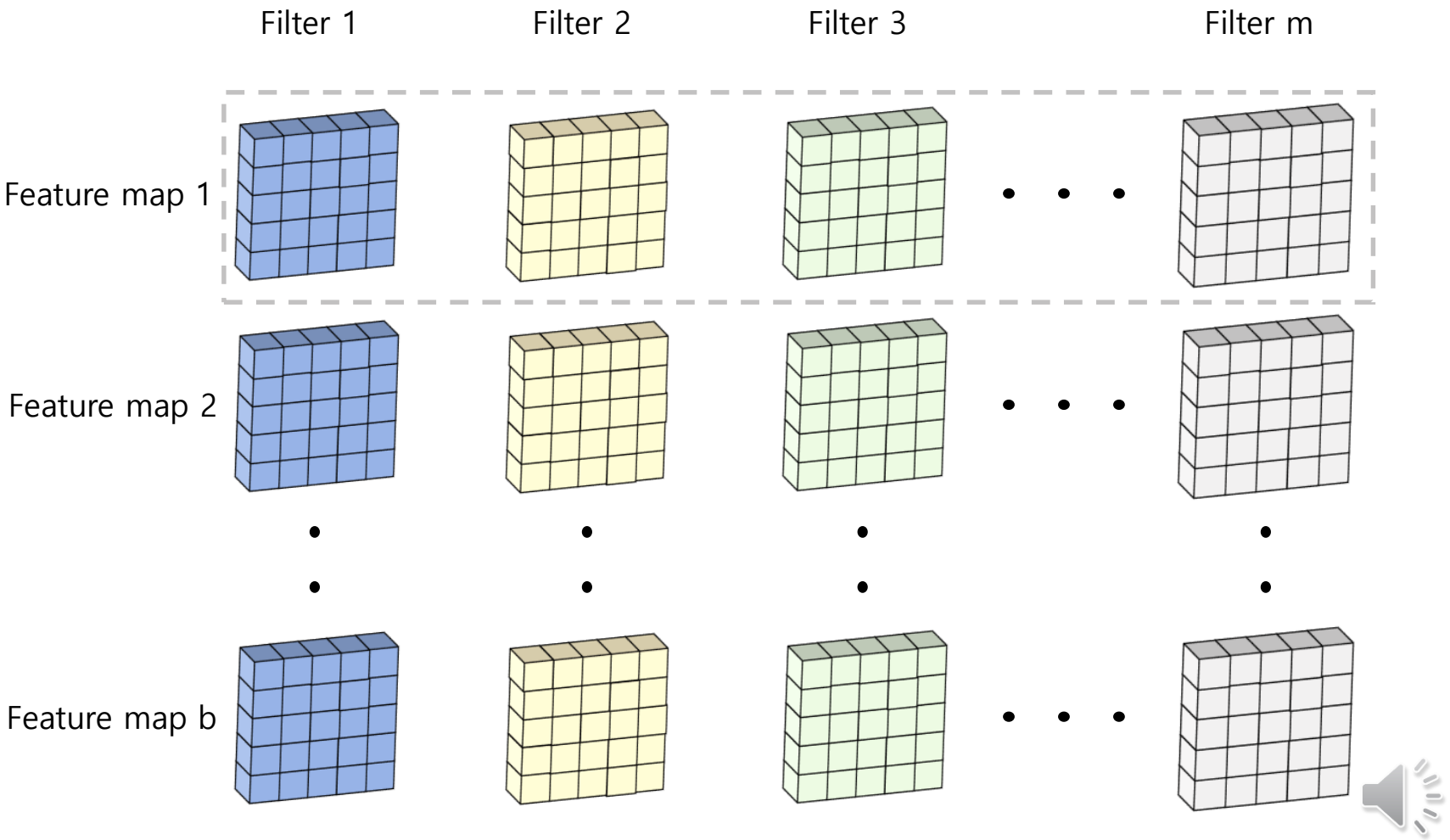
Filter
(3x3x7)

Feature map

# Methods

## Group Normalization

❖ **Group Normalization**

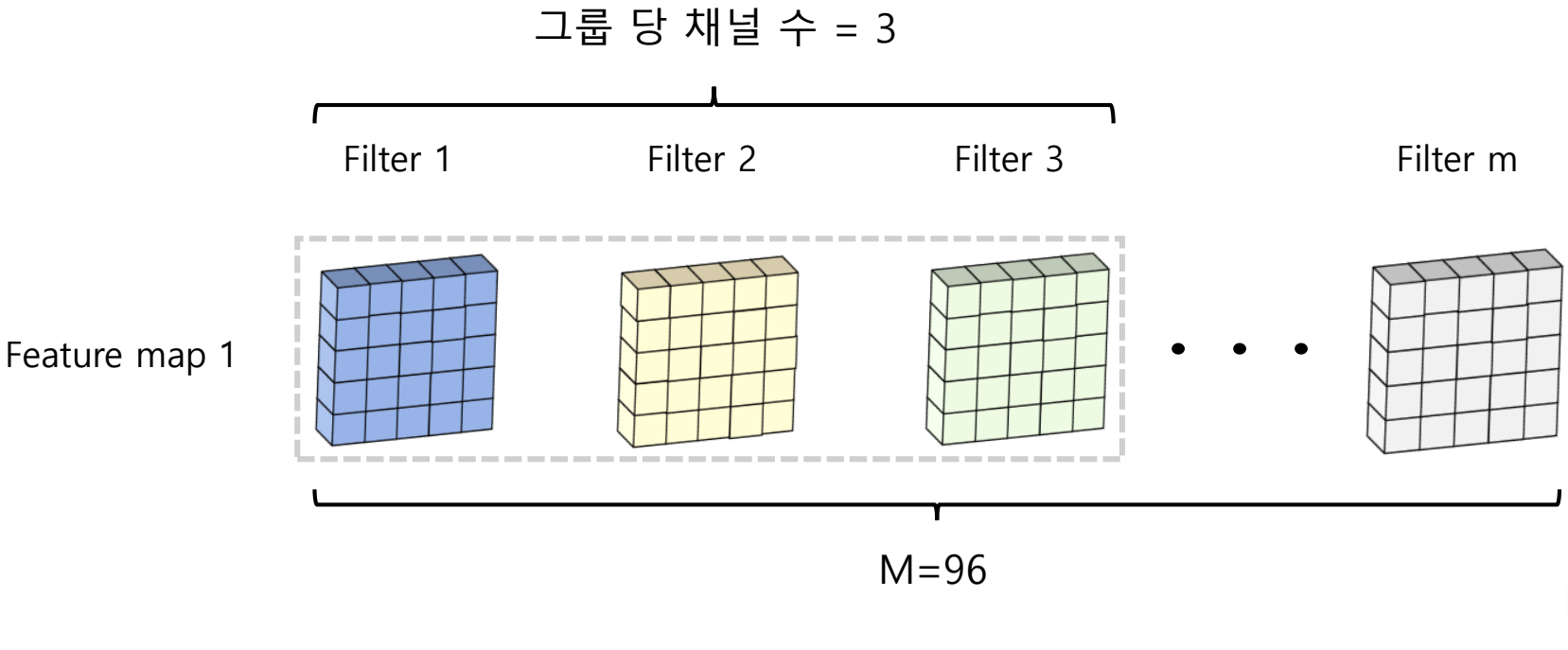


# Methods

## Group Normalization

❖ **Group Normalization**

- 평균과 분산은 각 그룹에 대해 별도로 계산
- Normalization도 각 그룹에 개별적으로 진행
- 그룹 수 = 32 → 그룹 당 채널의 수 = 96 / 32 = 3

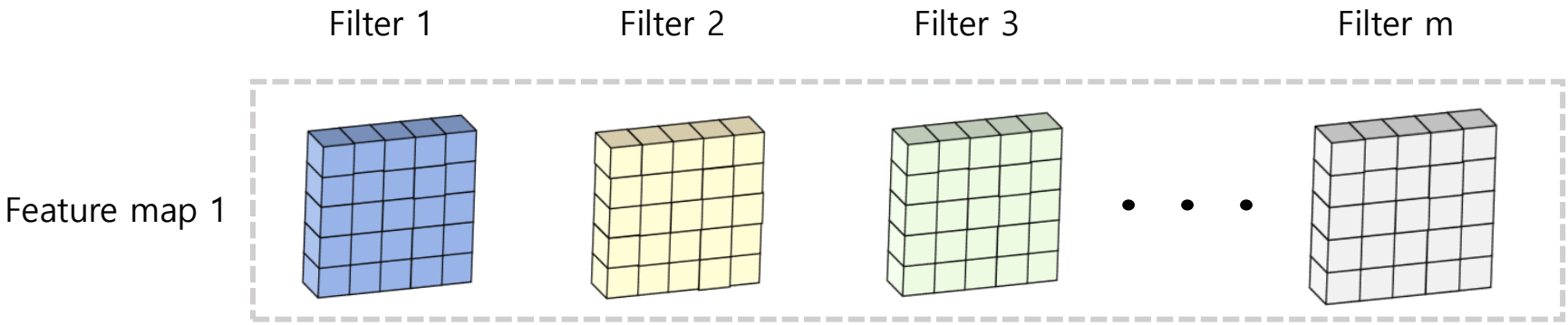


# Methods

## Group Normalization

❖ **Group Normalization**

- 그룹의 수가 1이라면 Layer Normalization과 동일
- 그룹의 수가 m이라면, 그룹 당 채널의 수는 1이 되고 Instance Normalization과 동일

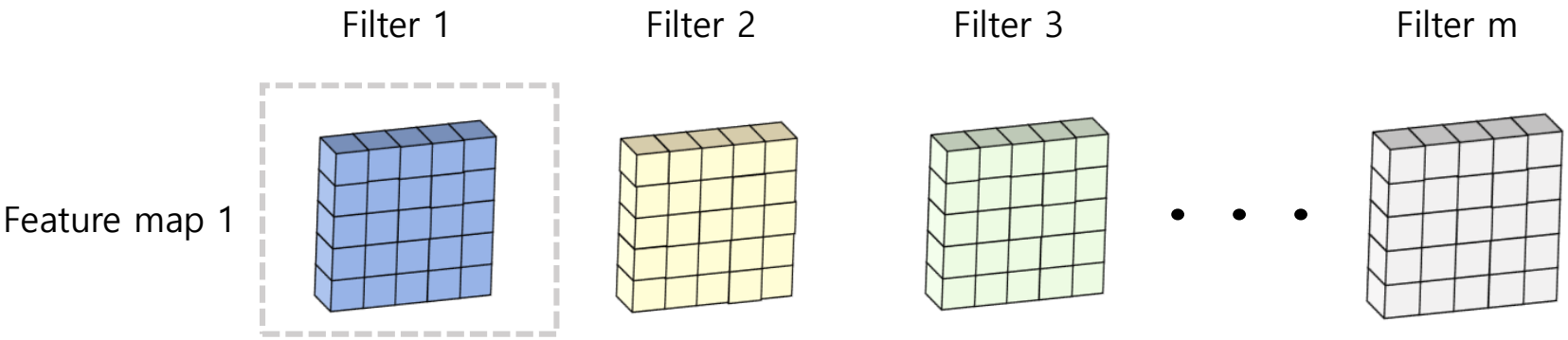


# Methods

## Group Normalization

❖ **Group Normalization**

- 그룹의 수가 1이라면 Layer Normalization과 동일
- 그룹의 수가 m이라면, 그룹 당 채널의 수는 1이 되고 Instance Normalization과 동일



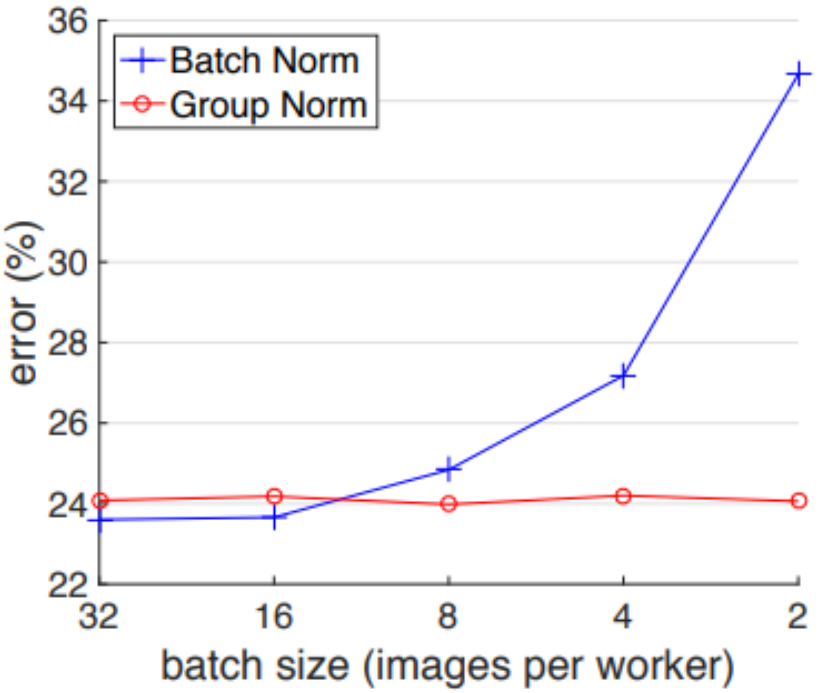
# Methods

| batch size | 32 | 16 | 8 | 4 | 2 |
|---|---|---|---|---|---|
| BN | **23.6** | **23.7** | 24.8 | 27.3 | 34.7 |
| GN | 24.1 | 24.2 | **24.0** | **24.2** | **24.1** |
| △ | *0.5* | *0.5* | *-0.8* | *-3.1* | *-10.6* |

Group Normalization

❖ **Experiment**

- GN이 batch size의 변화에도 불구하고 안정적인 error를 보임
- Batch size가 2일 때 BN과 GN 사이에 가장 큰 성능 차이를 확인 가능



# Conclusion

Various Normalization Techniques for Deep Learning

❖ **Summary**

- 데이터의 정확한 분석을 위해 Feature scaling이 꼭 필요
- Feature Scaling의 대표적인 방법 Normalization(MinMax Scaling) & Standardization
  - ✓ Standardization과 논문에서 Normalization은 같은 의미로 취급
- Batch Normalization : Vanishing/Exploding gradient 문제 해결을 위한 획기적인 방법
- Layer Normalization : 시퀀스의 길이에 따라 달라지는 입력 때문에 RNN에 적용하기 어려운 BN의 단점 보완
- Instance Normalization : Image Style Transfer의 성능을 올리기 위해 Instance Normalization 등장
- Group Normalization : Batch 크기가 극도로 작은 상황에서 사용하면 좋은 normalization 방법론 제안

# Appendix


- ✓ Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). PMLR.

- ✓ Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

- ✓ Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2016). Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*.

- ✓ Wu, Y., & He, K. (2018). Group normalization. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 3-19).

- ✓ Santurkar, S., Tsipras, D., Ilyas, A., & Madry, A. (2018). How does batch normalization help optimization?. Advances in neural information processing systems, 31.


# Thank you

E-mail : jungin_kim23@korea.ac.kr