



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

2021년 2월
석사학위논문

딥러닝 모델을 이용한 Al-Si 주조 합금의 미세조직 생성과 분석

조선대학교 대학원

첨단소재공학과

황 인 규

딥러닝 모델을 이용한 Al-Si 주조 합금의 미세조직 생성과 분석

Generation and Analysis of Microstructure of Al-Si Casting
Alloy Using Deep Learning Model

2021년 2월 25일

조선대학교 대학원

첨단소재공학과

황 인 규

딥러닝 모델을 이용한 Al-Si 주조 합금의 미세조직 생성과 분석

지도교수 김 희 수

이 논문을 공학 석사학위신청 논문으로 제출함.

2020년 10월

조선대학교 대학원

첨단소재공학과

황 인 규

황인규의 석사학위 논문을 인준함

위원장 조선대학교 교수 장희진 (인)

위 원 조선대학교 교수 김선중 (인)

위 원 조선대학교 교수 김희수 (인)

2020년 11월

조선대학교 대 학 원

목 차

LIST OF TABLES	iii
LIST OF FIGURES	iv
ABSTRACT	viii
제 1 장 서 론	1
제 2 장 이론적 배경	3
제 1 절 관련연구	3
1. 인공신경망	3
2. 합성곱 신경망	6
2.1 LeNet	14
2.2 VGGNet	15
2.3 GoogLeNet	16
3. 데이터 생성에 관한 연구	18
3.1 데이터 증강 기법	20
3.2 오버 샘플링 & 언더 샘플링	22
3.3 오토인코더	25
3.4 생성적 적대 신경망	27

3.5 Conditional GAN	30
3.6 Cycle GAN	31
4. Al-Si alloy	32
제 3 장 모델링 설계 및 구현	34
3.1 DCGAN	34
3.2 알고리즘 설계	35
제 4 장 실험방법	37
4.1 실험환경	37
4.2 데이터 전처리 및 설계	38
4.3 데이터 검증 방법	40
제 5 장 실험결과	42
5.1 Al-Si 합금 미세조직 이미지 생성	42
5.2 Inception V3 알고리즘을 이용한 유사도 평가	48
5.3 합성 데이터 생성	52
제 6 장 결론 및 향후 연구	53
참 고 문 헌	55

LIST OF TABLES

Table 2.1 VGGNet	15
Table 4.1 Computer Specifications	37

LIST OF FIGURES

Fig. 2.1 Linear Classification(OR, AND), Non-Linear Classification(XOR)	4
Fig. 2.2 Single Layer Perceptron	5
Fig. 2.3 Multi-Layer Perceptron	5
Fig. 2.4 Convolutional Neural Network Architecture	6
Fig. 2.5 Feature Map process of one channel image	7
Fig. 2.6 Feature Map process of three channel image	8
Fig. 2.7 Padding	9
Fig. 2.8 Pooling	10
Fig. 2.9 Convolutional Neural Network output process	10
Fig. 2.10 Activation Function(Sigmoid, ReLU, Leaky ReLU)	11
Fig. 2.11 Overfitting	12
Fig. 2.12 Drop Out	13
Fig. 2.13 Architecture of LeNet-5	14
Fig. 2.14 Inception Module	16

Fig. 2.15 Architecture of GooLeNet 17

Fig. 2.16 Supervised Learning and Unsupervised Learning 18

Fig. 2.17 Crop 20

Fig. 2.18 Add Noise 20

Fig. 2.19 Data Augmentation Application Example 21

Fig. 2.20 Over-Sampling and Under-Sampling 22

Fig. 2.21 SMOTE 23

Fig. 2.22 Structure of Autoencoder 25

Fig. 2.23 Overcomplete Autoencoder 26

Fig. 2.24 Generator Training Process 27

Fig. 2.25 Discriminator Training Process 28

Fig. 2.26 GAN Training Process 28

Fig. 2.27 CGAN Training Process 30

Fig. 2.28 Cycle GAN Training Process 31

Fig. 2.29 Phase diagram of the Al-Si binary alloy system 33

Fig. 3.1 Structure of DCGAN Generator 35

Fig. 3.2 Structure of DCGAN Discriminator 36

Fig. 4.1 Microstructure Image Preprocessing 39

Fig. 4.2 Inception V3 Classification Process 40

Fig. 4.3 Three Method of Verification 40

Fig. 5.1 Fake images generated early in the Epoch 42

Fig. 5.2 Al-6wt%Si Fake Images 43

Fig. 5.3 Al-9wt%Si Fake Images 44

Fig. 5.4 Al-12wt%Si Fake Images 45

Fig. 5.5 Al-15wt%Si Fake Images 46

Fig. 5.6 The changes of discriminator loss function value in training process 47

Fig. 5.7 Similarity bar graph when fake image and non-DB image are put in Inception V3 algorithm that trained real data 48

Fig. 5.8 Similarity bar graph when real image and Non-DB image are put in Inception V3 algorithm that trained fake data 49

Fig. 5.9 Similarity bar graph when Real image, Fake Image and Non-DB image are put in Inception V3 algorithm that trained Fake data and Real data 50

Fig. 5.10 Composite image generated by mixing Al-6wt%Si and Al-12wt%Si, Al-9wt%Si and Al-15wt%Si datasets 52

Fig. 6.1 Generate Al-12wt%Si, Al-15wt%Si fake microstructure image using 400 training data 54

Abstract

Generation and Analysis of Microstructure of Al-Si Casting Alloy Using Deep Learning Model

In-Kyu Hwang

Advisor: Prof. Hee-soo Kim, Ph. D.

Dept. of Advanced Materials Engineering

Graduate School of Chosun University

In this study, we conducted an experiment to extend the data set using the DCGAN algorithm, which is a generative model. The data used in the experiment are microstructure images of Al-6wt%Si, Al-9wt%Si, Al-12wt%Si, and Al-15wt%Si alloys. In all compositions, a fake image was generated using 70 training data and 1000 Epoch learning frequency, and then similarity was measured using Inception V3 algorithm. As a result, the similarity of the alloy composition was highest, and the generated fake image showed the structural characteristics of the real data well. However, in the case of the hypo-eutectic composition, a continuous pattern such as a dendritic phase did not appear, and in the case of the hyper-eutectic composition, Si primary crystals and acicular structures were generated collapsed. This result appeared to be an insufficient amount of training data, and as a result of generating a fake image using 400 pieces of training data, relatively similar patterns and structures could be generated. As an additional experiment, microstructure image data of different alloy compositions were mixed to generate intermediate composition data, but the similarity of the composition corresponding to the alloy was remarkably low. Therefore, it appears that the creation of new synthetic data requires a different approach.

제 1 장 서론

현재 SNS(Social Networking Service)와 스마트 앱(Smart App), 클라우드(Cloud)의 사용 증가에 따라 여러 데이터의 수집이 가능해졌고 이로 인해 딥러닝의 활용 사례가 늘어가고 있다. 딥러닝은 데이터를 스스로 학습하고 분석하는 머신러닝(Machine Learning)의 분야로 학습 데이터의 양이 많을수록 성능 또한 증가한다[4].

그러나 딥러닝을 여러 분야에 적용하기에 앞서 데이터를 확보하는 데에 많은 어려움이 따른다. 앞서 말했듯 최근 데이터의 양이 급증했다고는 하나 한 분야에 특화된 데이터는 거의 없다. 또한 데이터가 확보되더라도 이를 딥러닝에 적용하려면 데이터별 분류, 불필요한 데이터와 중복 데이터 제거 등 데이터 전처리 단계를 거쳐야 한다. 이러한 데이터 전처리 단계는 많은 시간과 비용을 소모하며 손실되는 데이터의 양 또한 증가하게 된다.

이러한 데이터 부족 문제를 해결하기 위해 생성모델을 이용한 데이터 생성에 관한 연구가 활발해지고 있다. 딥러닝을 활용한 대부분 모델은 크게 판별 모델(Discriminative Model)과 생성 모델(Generative Model)로 나눌 수 있다. 판별 모델의 경우 데이터 분포(Distribution)에 대한 클래스 간의 경계 조건(Boundary Conditions)을 학습하는 것과 달리 생성모델은 데이터 분포를 학습하며 새로운 데이터의 생성이 가능하다. 이러한 생성모델의 기능은 데이터 부족 문제뿐만 아니라 데이터 불균형 문제 또한 해결 가능할 것으로 보여진다.

현재 데이터 생성을 위한 여러 알고리즘들이 존재하나 GAN(Generative Adversarial Network)은 비지도 학습을 적용한 알고리즘으로 데이터를 능동적으로 생성하는 것이 가능하며 다양한 분야에서 활용이 가능하다[14, 21]. 그러나 GAN은 구조적 문제와 그로 인한 출력 데이터의 품질이 다소 떨어지는 단점이 존재했다. 때문에 여러 파생된 GAN 알고리즘들이 존재하는데 그 중 DCGAN(Deep Convolutional Generative Adversarial Network)은 GAN의 구조적 개선을 통해 향상된 모델로 향상된 모조 데이터의 출력이 가능하다[39]. 기존 GAN은 뉴런이 전부 연결되어 있는 완전 연결 계층(Fully Connected Layer, FC Layer) 구조로 설계되었으나 DCGAN에서는 이를 합성곱 신

경망(Convolution Neural Network, CNN)으로 대체 함으로써 성능이 개선되고 다양한 학습 데이터의 도입이 가능해졌다.[19]

본 논문에서는 이러한 DCGAN 알고리즘을 이용해 Al-Si 합금의 미세조직 이미지를 생성해 보고자 한다. 미세조직 이미지는 해당 합금의 기계적특성과 화학적특성을 유추할 수 있는 수단 중 하나로 합금의 면을 연마하여 관찰한다. 그러나 한 합금에 대해 미세조직 이미지를 수집 하기 위해서는 많은 전처리과정을 거쳐야 하기 때문에 시간과 비용을 소비하게 된다. 따라서 최소 데이터의 개수를 가지고 DCGAN 알고리즘을 이용해 부족한 데이터 셋을 확보하고자 한다. 이후, 생성된 모조 데이터가 실제로 사용할 수 있을지에 대한 검증 방법으로 Inception V3 알고리즘을 이용해 정확도를 비교해보고 서로 다른 조성을 가진 합금을 이용해 새로운 합성 데이터 또한 그려낼 수 있을지에 대한 연구 또한 진행 하고자 한다.

제 2 장 이론적 배경

제 1 절 관련연구

딥러닝은 데이터를 분류(Classification) 또는 군집화(Clustering) 하는 데 쓰이는 기술적 방법론으로 이를 기계학습에 적용함으로써 오늘날의 알파고(Alphago), 자율 주행 자동차[9], 스마트 팩토리(Smart factory) 등의 긍정적인 결과가 많이 도출되고 있다. 기계학습의 알고리즘으로는 베이저안 망(Bayesian Network)[41]], 의사결정 나무(Decision Tree)[25], 서포트 벡터 머신(Support Vector Machine)[2, 25, 45, 51], 인공신경망(Artificial Neural Network)[31] 등 많은 알고리즘들이 존재하는데 데이터의 속성과 형태 또는 사용목적에 따라 쓰임이 다르다. 그 중 인공신경망은 데이터의 범용성과 빠른 연산처리 능력 때문에 가장 많이 쓰이고 있다.

1. 인공신경망(Artificial Neural Network)

인공신경망은 사람의 신경 세포인 뉴런(neuron)을 도식화한 신경망으로 오늘날 딥러닝에 적용하여 많이 쓰이고 있다. 작동방식은 입력 신호와 출력 신호의 관계를 모델링하는 과정에서 뉴런, 가중치(Weight), 활성화 함수(Activation Function), 편향(bias) 등을 활용한다. 초기 형태의 인공신경망은 단층 퍼셉트론(Single Layer Perceptron)으로 입력층(Input Layer)과 출력층(Output Layer)으로 이루어졌으며 다수의 입력으로부터 하나의 출력값을 내는 형태였다[10]. 예를 들어, 입력 신호가 뉴런에 전달될 때는 각각 고유한 가중치가 곱해져 신호가 흐를 때는 ‘1’, 흐르지 않으면 ‘0’으로 간주한다. 그러나 이러한 단층 퍼셉트론은 선형 분류만 가능하며 XOR 게이트와 같은 비선형 분류의 문제는 표현하지 못한다는 한계점이 있었다[27].

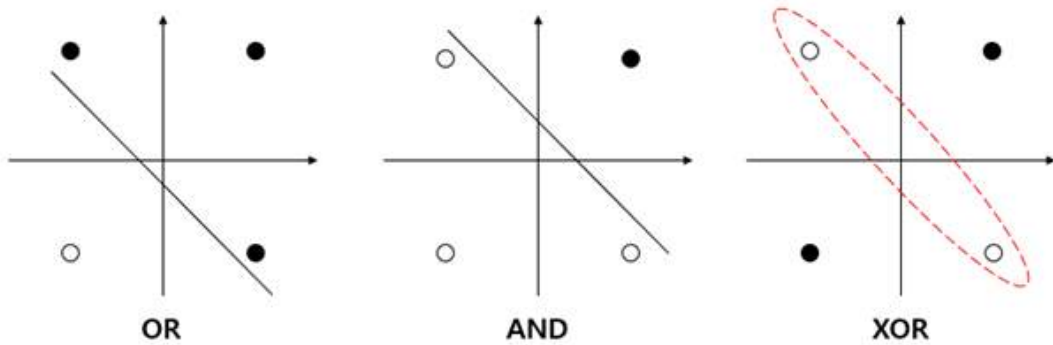


Fig. 2.1 Linear Classification(OR, AND), Non-Linear Classification(XOR)

이러한 비선형 문제를 해결하기 위해 하나의 퍼셉트론에 또 다른 퍼셉트론을 덧붙이는 다층 퍼셉트론(Multi-Layer Perceptron)의 형태로 발전했다[43, 46, 52]. 다층 퍼셉트론은 여러 개의 퍼셉트론 층을 이어붙임으로써 데이터를 보다 효과적으로 분류할 수 있게 되었다. 이 외에도 학습이 진행되는 동안 가중치의 최적화를 위해 역전파(Back-propagation)를 도입함으로써 원하는 결과의 출력이 가능해졌다.

그러나 다층 퍼셉트론은 이미지 데이터의 학습과 예측 정확도의 성능에 한계가 있었다. 동일한 종류의 이미지 데이터를 입력해도 픽셀값이 조금이라도 달라지면 매번 연산을 수행하고 공간적 정보의 손실이 발생한다. 이와 같은 문제를 보완하고 이미지 데이터의 효율적인 연산처리를 위해 합성곱 신경망이 도입되었다.

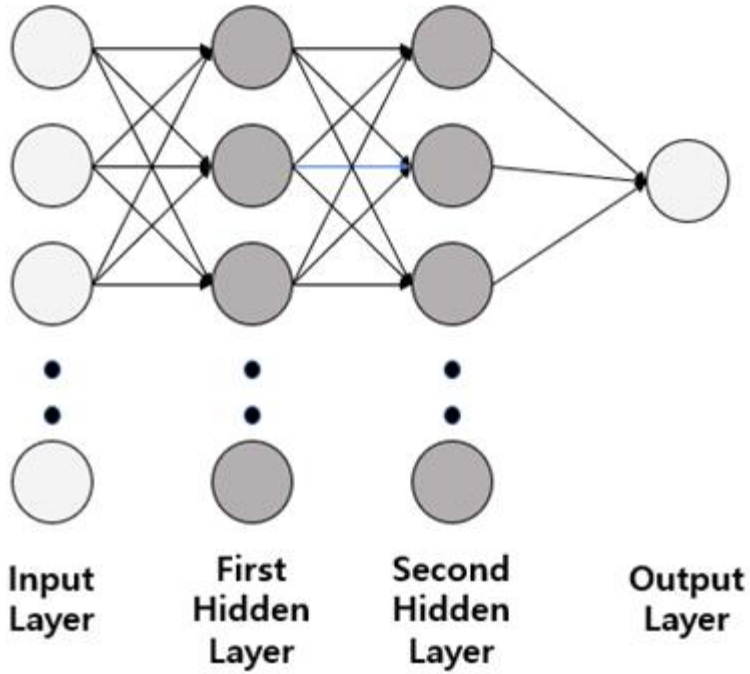


Fig. 2.2 Single Layer Perceptron

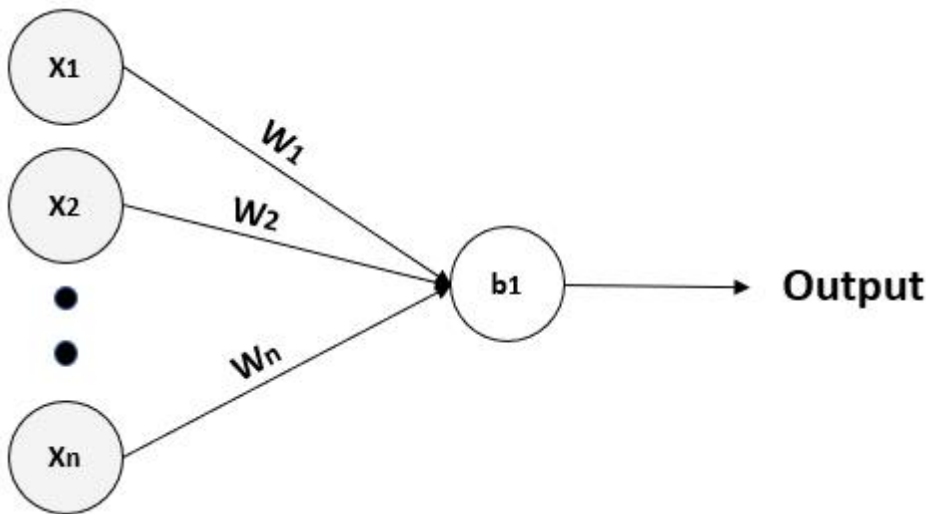


Fig. 2.3 Multi-Layer Perceptron

2. 합성곱 신경망(Convolutional Neural Network, CNN)

합성곱 신경망은 오늘날에 모든 딥러닝 분야에서 근간이 되는 알고리즘으로 자리 잡고 있다[5, 49]. 합성곱 신경망이 발표되기 전 기존 신경망은 인접하는 계층의 뉴런이 모두 연결되는 완전 연결 신경망(Fully-Connected Neural Network)을 사용하였다. 완전 연결 신경망을 이용해 분류작업을 하는 알고리즘의 경우, 3차원 데이터를 1차원 데이터를 변환하는 평탄화(Flatten) 작업을 거쳐 입력 데이터로 넣어주었다. 이러한 변환 작업은 데이터의 형상이 무시되고 공간적 정보의 손실이 내재 된다. 이러한 문제는 작은 픽셀 크기를 가지는 이미지 데이터는 학습에 무리가 없으나 데이터의 크기가 커질수록 극대화된다.

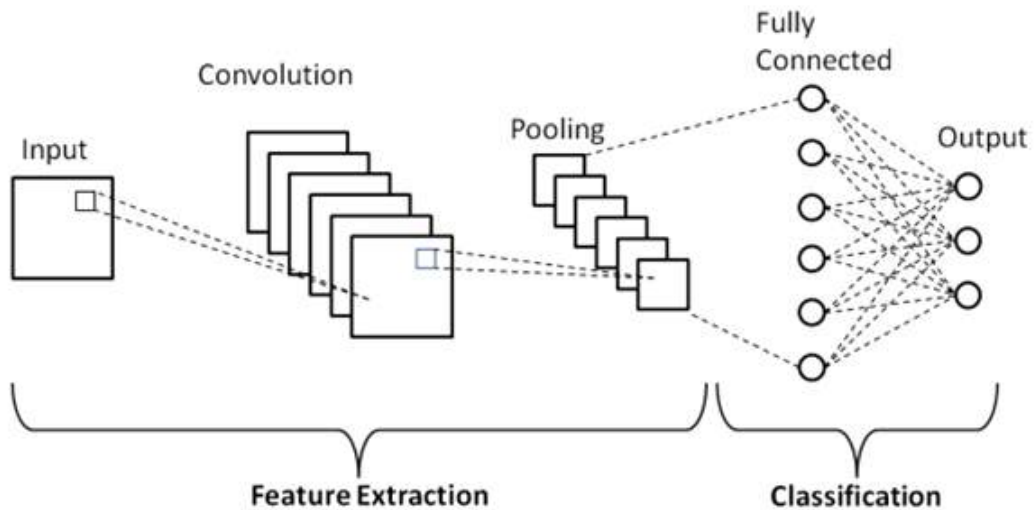


Fig. 2.4 Convolutional Neural Network Architecture

때문에 합성곱 신경망에서는 데이터의 형상을 그래도 입력받을 수 있는 형태의 모델로 발전하였으며 데이터를 보다 정확히 분석할 수 있게 되었다. 합성곱 신경망은 크게 입력 이미지의 특징(Feature)을 추출하는 합성곱 레이어(Convolution Layer)와 풀링 레이어(Pooling Layer)가 여러 겹으로 쌓여있는 형태이며 마지막 계층 이전에는 분류를 위해 완전 연결 계층을 사용한다.

합성곱 레이어는 입력받은 이미지를 합성곱 연산을 통해 이미지의 특징을 추출하는 층으로 필터(Filter)를 적용하여 특징 맵을(Feature Map)을 만든다[28, 48]. 필터는 이미지의 특징을 찾아내기 위한 하나의 파라미터(Parameter)로 커널(Kernel)이라고도 하며 입력 이미지 크기에 따라 필터의 크기를 조절할 수 있다. 보통 필터는 정사각의 행렬로 정의되며 3차원 이미지를 입력으로 넣어줬을 때는 채널(Channel) 별로 합성곱 연산을 하여 특징 맵을 만들며 필터의 적용 위치 간격은 스트라이드(Stride)를 통해 각각 다르게 적용할 수 있다.

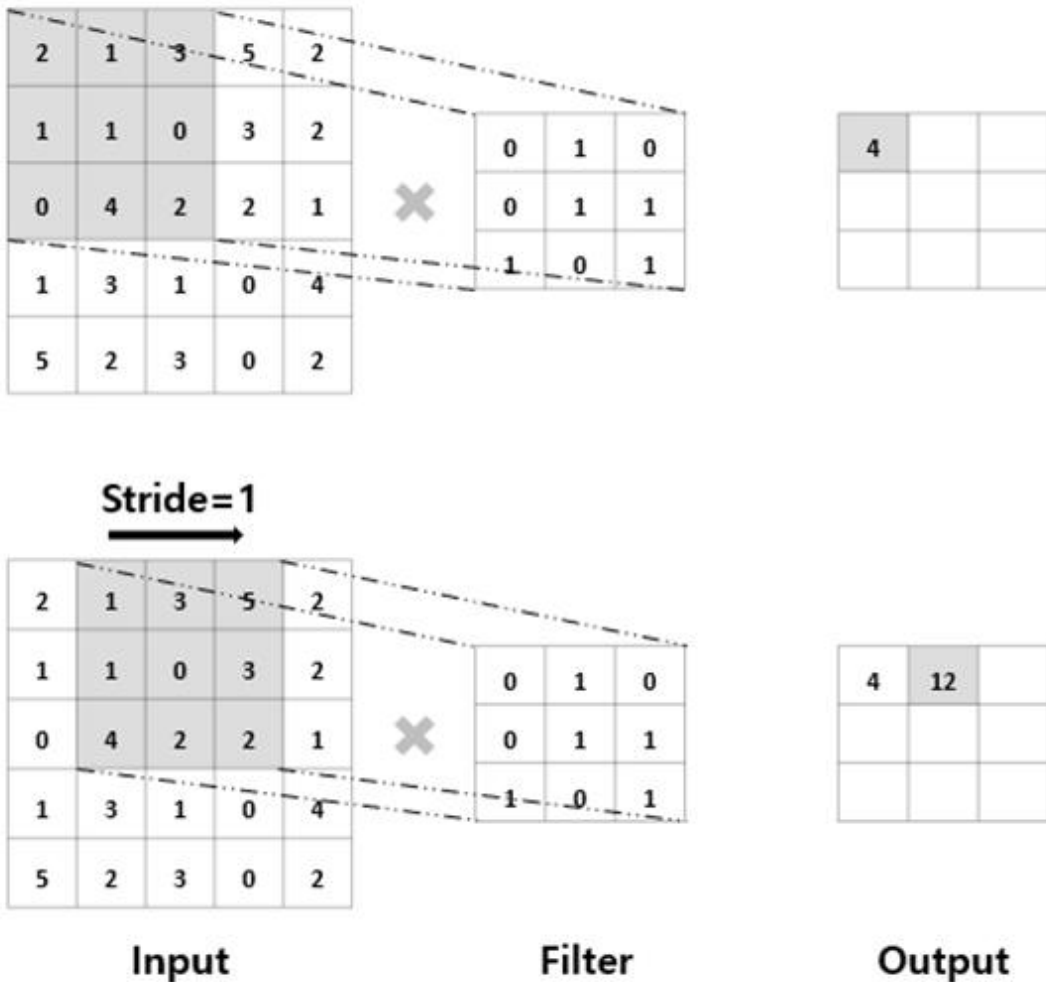


Fig. 2.5 Feature Map process of one channel image

3 Channel Image

0	1	1	1
1	0	0	1
1	1	1	0
0	1	0	1

0	1	1	1
1	0	0	1
1	1	1	0
0	1	0	1

0	1	1	1
1	0	0	1
1	1	1	0
0	1	0	1

0	1	1	1
1	0	0	1
1	1	1	0
0	1	0	1

×

0	1	1
1	1	1
1	0	0

×

1	0	1
0	0	1
1	1	1

×

1	1	0
1	0	0
1	0	1

4	4
3	4

+

4	5
3	3

+

4	3
2	2

Feature Map

12	12
8	9

Fig. 2.6 Feature Map process of three channel image

합성곱 연산을 거친 데이터는 입력 데이터의 크기보다 작은 크기를 가진다. 그러나 여러 겹의 합성곱 레이어를 가진 알고리즘이나 입력 데이터의 크기가 너무 작을 경우 결과의 출력값이 작아져 정보가 소실될 수 있다. 이를 방지하기 위해 패딩(Padding)을 적용하여 입력 데이터와 출력 데이터의 크기를 동일하게 설정할 수 있다. 일반적인 경우, 패딩은 데이터 외각을 둘러싸는 특징을 가지고 있으며 주변 값을 0으로 채우는 제로 패딩(Zero Padding)을 사용한다. 패딩은 이러한 역할과 동시에 인공지능망이 이미지의 외각을 인식하는 학습효과 또한 불러일으킬 수 있다.

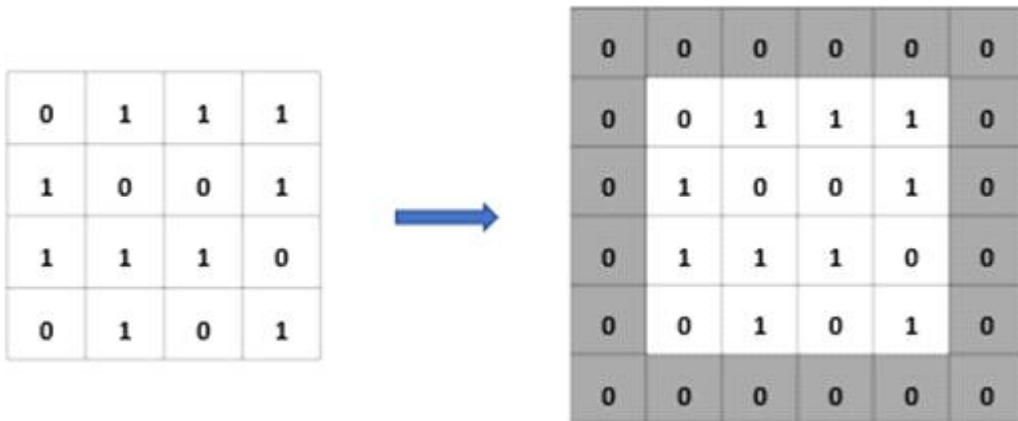


Fig. 2.7 Padding

풀링 레이어는 합성곱 연산을 거친 데이터의 차원을 축소하는 역할로 서브 샘플링(Sub Sampling)이라고도 한다. 연산 방식에 따라 최대 풀링(Max Pooling)과 평균 풀링(Average Pooling)으로 나뉜다. 최대 풀링은 해당 영역의 최댓값을 취하는 연산을 거치며 평균 풀링은 해당 영역의 평균값을 취하는 연산을 거친다. 대부분의 경우 풀링의 크기와 스트라이드의 크기를 같게 설정하여 한 번씩 연산하도록 하며 맥스 풀링을 주로 사용한다[15, 23].

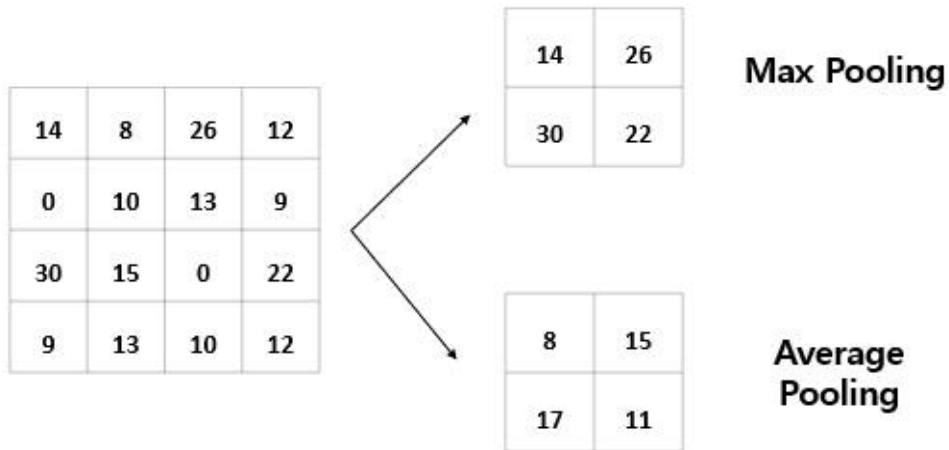


Fig. 2.8 Pooling

풀링 레이어를 거친 입력 데이터는 완전 연결 계층으로 전달된다. 앞서 추출된 특징 값들은 2차원의 데이터로 분류작업을 하기 위해서는 1차원 데이터로 변환해야 한다. 때문에 완전 연결 계층에서 Flatten 작업을 거친 후 분류하는 출력층으로 최종 입력을 거친다. 일반적으로 출력층의 활성화 함수는 소프트맥스(SoftMax)를 쓰며 전달받은 값을 0~1 사이의 값으로 정규화를 거친 후 데이터가 속하는 클래스(Class)를 예측하며 학습이 종료된다.

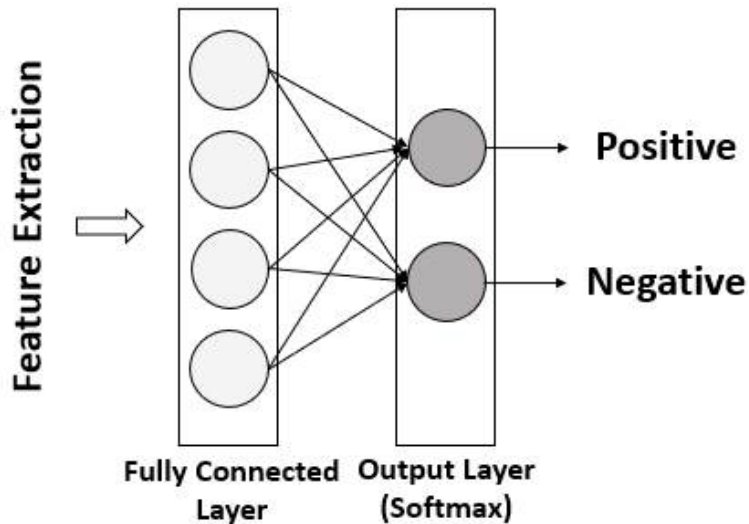


Fig. 2.9 Convolutional Neural Network output process

합성곱 신경망은 출력 결과에 따라 매개변수를 자동으로 업데이트하고 입력 데이터의 패턴인식이나 특징추출 능력에 있어 효율적이다. 이러한 특성을 이용해 최근에는 이미지 인식 분야와 음성인식, 컴퓨터 비전 등의 분야에서 많이 활용되고 있다[6, 18, 44].

그러나 합성곱 신경망을 이용한 학습과정에서 자주 발생하는 문제들이 존재한다. 대표적인 예로 기울기 소실 문제(Gradient Vanishing Problem)와 과적합(Overfitting) 문제가 있다. 기울기 소실 문제는 역전파 과정에서 입력층으로 갈수록 기울기가 점점 소실되는 문제로 이러한 현상이 반복되면 가중치들이 업데이트되지 않고 모델의 최적화가 어려워진다. 보통 활성화 함수로 시그모이드(Sigmoid)나 하이퍼볼릭 탄젠트(Hyperbolic Tangent)로 쓸 경우 많이 발생한다. 이러한 경우, 시그모이드나 하이퍼볼릭 탄젠트 대신에 ReLU 또는 Leaky ReLU와 같은 활성화 함수로 대체하면 기울기 소실 문제를 어느 정도 완화시킬 수 있다.

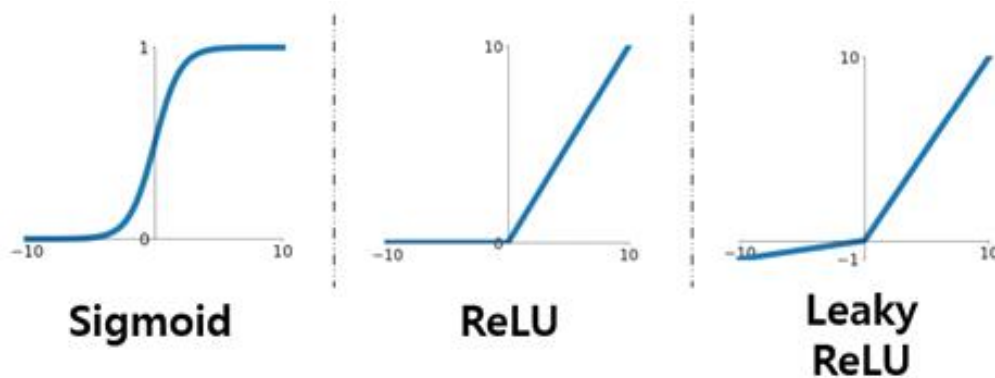


Fig. 2.10 Activation Function(Sigmoid, ReLU, Leaky ReLU)

과적합은 모델이 훈련 데이터에서는 우수한 예측 정확도를 출력하나 학습 데이터가 아닌 새로운 데이터를 적용하였을 때 성능이 급격히 저하되는 것을 말한다. 이러한 현상은 고려하는 변수가 많거나 크기가 너무 깊을 경우 또는 데이터의 양이 너무 적을 때 발생한다. 과적합을 방지하는 가장 간단한 방법은 데이터를 학습용과 검증용으로 분할하여 학습 후 검증 데이터를 이용해 테스트하거나 데이터의 양을 늘리는 방법이 있다.

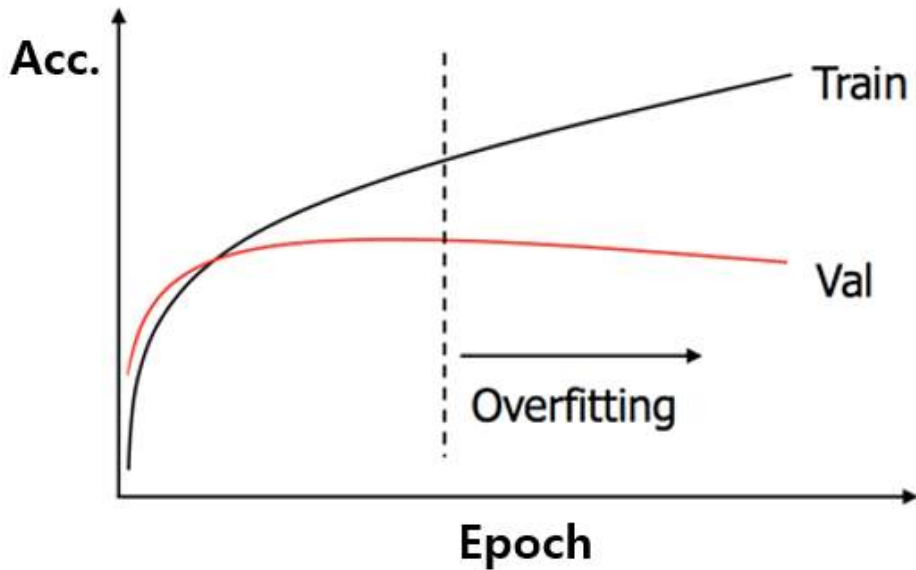


Fig. 2.11 Overfitting

때문에 과적합 현상을 방지하기 위한 다른 방법들이 제시되었다. 먼저 배치 정규화 (Batch Normalization)를 적용하는 방법이 있다. 배치 정규화는 입력을 사용자가 설정한 배치의 단위로 정규화 하는 것을 뜻한다. 이는 신경망의 각 층을 통과하기 전에 추가함으로써 입력을 분산시켜 학습 속도를 상승시키고 과적합 현상과 기울기 소실의 문제 또한 방지할 수 있다. 다음으로는 드롭아웃(Dropout) 기법이 있다. 드롭아웃은 신경망을 구성하는 은닉층의 뉴런을 무작위로 삭제시키는 방법으로 신경망의 크기와 상관없이 적용이 가능하다. 이러한 방법은 훈련 후 테스트할 때에는 사용하지 않거나 삭제한 뉴런의 비율을 곱하여 출력해야 한다. 드롭아웃을 적용했을 때는 학습이 느려지는 단점이 존재하나 배치 정규화와 같이 사용하면 이를 방지할 수 있다.

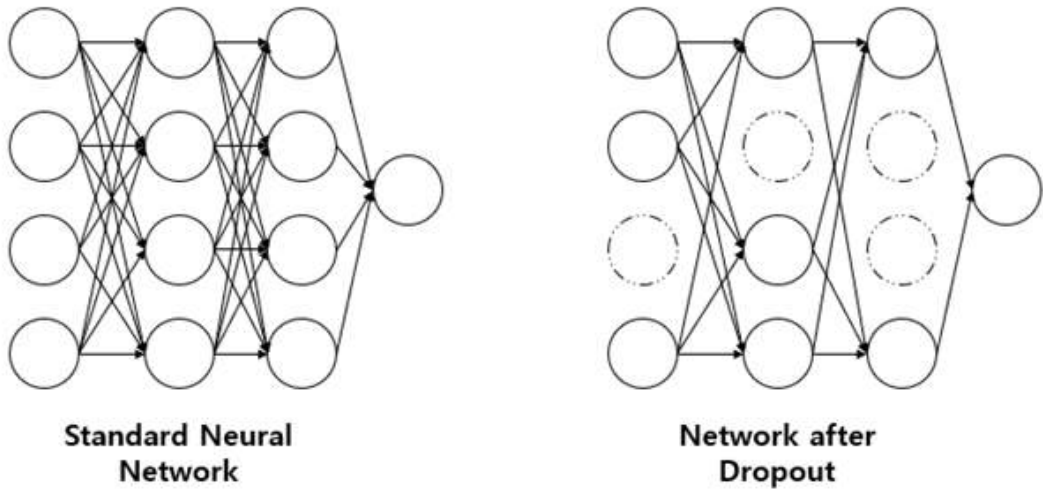


Fig. 2.12 Drop Out

이 외에도 데이터를 학습시킬 때 가중치가 큰 값에 대하여 페널티를 부여하는 가중치 감쇠(Weight Decay) 방법이 있다. 가중치 감쇠 방법은 L1 정규화(L1 Regularization)와 L2 정규화(L2 Regularization)로 나뉜다. L1 정규화는 기존 손실 함수에 파라미터의 절댓값을 더해 적용하는 방법으로 파라미터를 0에 가깝게 만들어 Feature selection이 가능하다. L2 정규화는 기존 손실 함수에 모든 파라미터의 제곱을 더한 식을 손실함수로 쓰는 기법으로 가중치의 값을 균형 있게 나누는 효과를 낸다. 위 두 가지 기법은 동시에 사용이 가능하며 데이터를 학습할 때 가중치에 혼란을 주어 모델의 범용성을 높인다.

2.1 LeNet

앞서 말했듯, 기존의 완전 연결 신경망으로는 구조적 한계가 존재했으며, 입력 데이터의 제약이 있었다. LeNet은 최초의 합성곱 신경망으로 LeNet-1부터 LeNet-5까지 점차 발전되어 왔다. 그중 LeNet-5는 신경망의 크기가 가장 크며 성능 또한 우수하다. LeNet-5는 32x32 크기의 이미지를 입력으로 받아 3개의 합성곱층과 2개의 풀링계층, 1개의 완전 연결 계층을 거쳐 결과를 출력한다[20].

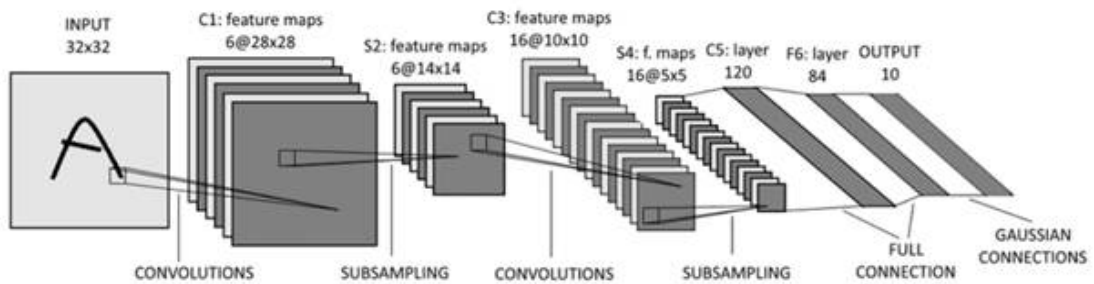


Fig. 2.13 Architecture of LeNet-5

LeNet-5의 학습과정은 다음과 같다. 첫 합성곱 계층에서는 입력 데이터의 연산을 거친 후 28x28 크기의 특징 맵 6개가 만들어지고 이후 풀링 계층에서 14x14 크기로 축소하는 과정을 거친다. 세 번째 합성곱 계층에서는 전달받은 데이터를 10x10 크기로 축소하고 네 번째 서브 풀링 계층을 거쳐 16개의 특징 맵이 생성된다. 마지막 합성곱 계층에서는 이전 계층에서 생성된 16개의 특징 맵이 다시 합성곱 연산을 거쳐 최종적으로 1x1 크기의 특징 맵이 얻어지며 완전 연결 계층으로 전달된다. 각 계층에서 사용되는 필터의 크기는 5x5 크기이며 풀링은 평균 풀링, 활성화 함수는 Sigmoid를 사용하였다.

2.2 VGGNet

VGGNet은 네트워크의 깊이가 알고리즘의 성능에 얼마나 영향을 미치는지에 대한 목적으로 서로 다른 크기를 가지는 6가지의 신경망을 만들어 비교하였으며 그 결과로 네트워크의 깊이가 깊을수록 이미지 분류의 정확도가 향상되는 것을 입증하였다.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Table. 2.1 VGGNet

이전 다른 합성곱 신경망의 경우 비교적 큰 7x7이나 11x11 크기의 필터를 적용하여 사용했는데 VGGNet에서는 3x3 크기의 필터로 대체하였다. 이는 연산 되는 파라미터의 양을 줄이고 여러 번의 필터를 적용함으로써 알고리즘의 성능향상에 기여 한다. 또한 합성곱 계층과 풀링 계층을 더 깊게 쌓을 수 있어 이미지 분류 정확도도 상승시킬 수 있다[22].

2.3 GoogLeNet

GoogLeNet 알고리즘은 LeNet 알고리즘과 같이 구조적 개선을 통해 점차 발전시켜온 알고리즘 중 하나이다. 일반적으로 합성곱 신경망의 성능을 높이는 방법 중 하나는 신경망의 크기를 깊게 만드는 것인데, 이러한 방법은 학습해야 할 파라미터가 증가하게 되면서 연산량이 많아지게 되고 기울기 소실 등의 문제 또한 초래할 수 있다. GoogLeNet 알고리즘에서는 이러한 문제들을 해결하고자 인셉션 모듈(Inception Module)을 신경망 내에 도입하였다.

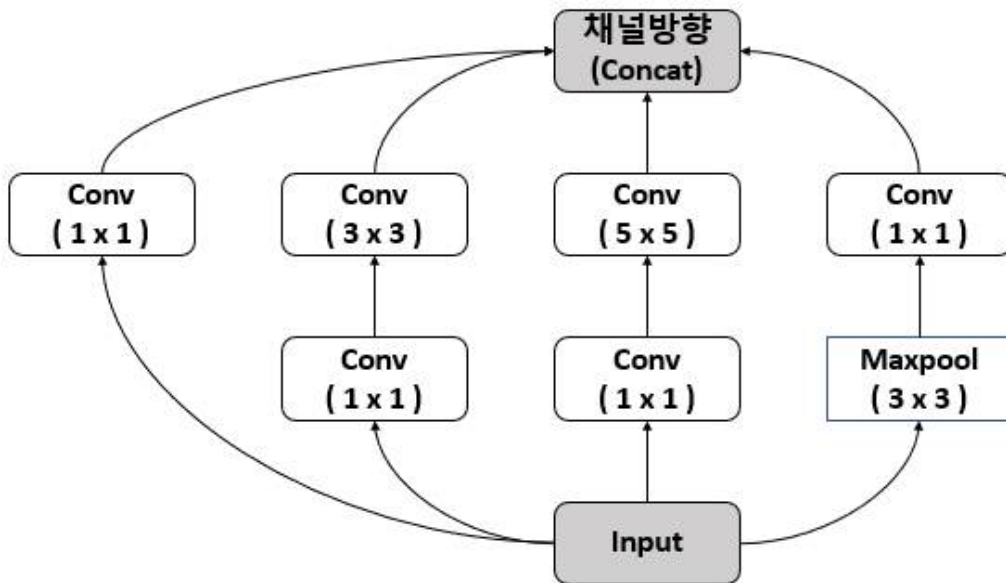


Fig. 2.14 Inception Module

인셉션 모듈은 입력 데이터에 각기 다른 총 4가지 종류의 연산을 수행하고 채널방향으로 결과의 합을 전달하는 구조를 가진다. 특히 1x1 크기의 합성곱 연산은 이전 레이어로부터 전달받은 특징 맵의 연산량을 줄임으로 인해 신경망의 크기를 더 깊게 설계할 수 있다. 예를 들어, 4개의 특징 맵이 1x1 크기의 합성곱 연산을 거치면 완전 연결 계층과 같은 역할을 하게 되는데 이로 인해 데이터의 차원이 축소되고 연산량이 감소된다. 이러한 1x1 크기의 합성곱 연산과 3x3, 5x5 크기의 서로 다른 합성곱 연산을 적용시키면 다양한 크기(Scale)의 형태로 특징을 효과적으로 추출할 수 있으며 연산량을 균형적으로 맞출 수 있다.

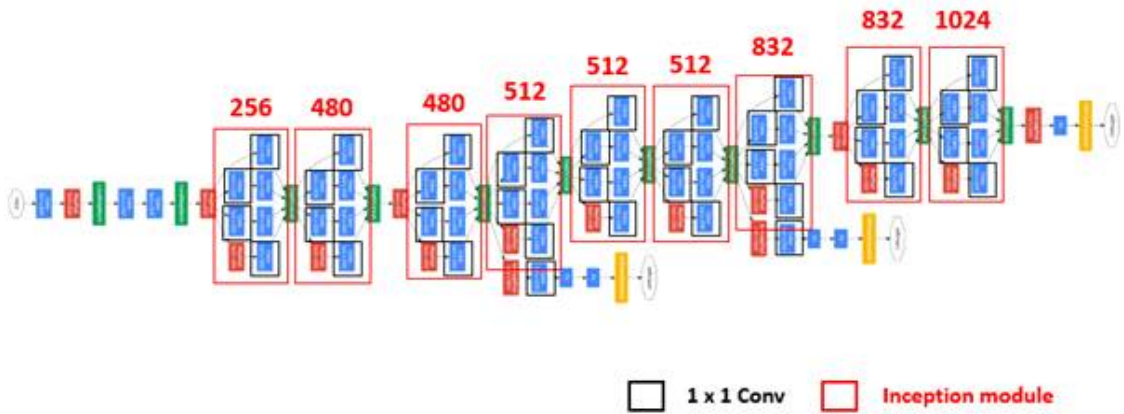


Fig. 2.15 Architecture of GoogLeNet

GoogLeNet의 자세한 구조는 Fig.2.15에 나타내었다. 앞서 언급한 인셉션 모듈 9개가 내재되어 있으며 1x1 크기의 합성곱 연산 또한 곳곳에 배치되어 있다. 이렇나 특징들 외에도, GoogLeNet 알고리즘은 Global Average Pooling을 적용하였는데 이는 학습이 진행되는 동안 가중치의 개수를 줄여 학습속도를 높여주고 파라미터의 추가발생을 방지한다. 또한 가중치의 학습을 원활하게 하기 위해 두 개의 보조 분류기(Auxiliary Classifier)를 네트워크 중간에 삽입해 두었다.

3. 데이터 생성에 관한 연구

앞서 설명한 합성곱 신경망은 VGGNet, GooLeNet 등의 알고리즘으로 발전하게 되면서 이미지 예측 측면에서 높은 정확도의 결과를 출력하고 데이터의 분류 또한 우수한 성능을 보였다. 이러한 데이터 클래스의 분류, 예측 등의 학습 방법은 지도학습(Supervised Learning)의 범주에 속한다. 지도학습은 입력 데이터와 출력 관계에 대한 모델링을 학습하는 방법으로 주로 판별 모델을 사용하여 데이터 클래스간의 경계 조건(Boundary Condition)을 학습한다. 이러한 지도학습과 반대의 학습방법론인 비지도학습(Unsupervised Learning)은 입력 데이터 간의 관계를 모델링 하는 것으로 클러스터링(Clustering), K-Means 알고리즘 등이 있으며 데이터의 숨겨진 특정 패턴이나 구조를 얻을 때 사용된다[7, 29, 36].

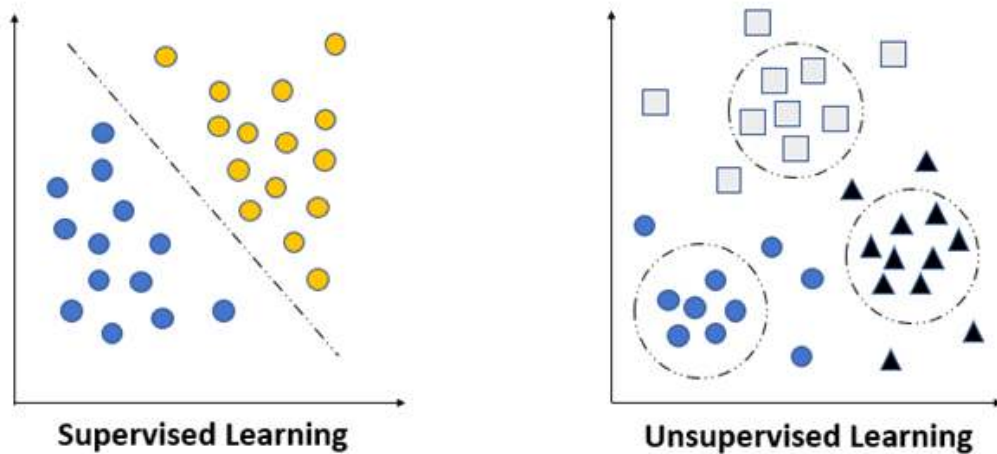


Fig. 2.16 Supervised Learning and Unsupervised Learning

이러한 방법론들을 활용하는 딥러닝 알고리즘들은 모델의 구조나 학습을 통한 매개변수의 조정 또한 중요하나 학습 데이터의 양이 충분하지 않으면 활용할 수 없다. 학습 데이터의 크기는 클수록 모델의 학습 능력향상에 크게 기여된다. 오늘날 빅데이터의 증가로 인해 데이터의 수가 급증했다고는 하나 데이터가 구조화되어 있는 정형 데이터(Structured Data)보다 비정형 데이터(Unstructured Data)의 비중이 더 크고 데이터 전처리에 드는 시간과 비용 또한 커서 적용하기에 많은 어려움이 따른다.

그래서 최근에는 생성모델(Generative Model)을 이용해 데이터 부족과 데이터 불균형(Data Imbalance) 문제를 해결하는 방안이 제시되고 있다. 딥러닝을 이용한 생성 모델은 대부분 비지도 학습의 일환으로 데이터의 패턴과 구조 분석에 용이하며 제한된 데이터 셋을 확장할 때 우수한 성능을 보이고 있다. 보통 생성 모델을 이용해 데이터 셋을 확장할 경우 이미지에 적용하는 사례가 많았으나 오늘날에는 자연어, 음악 생성 등 에도 적용하여 사용범위가 점차 넓혀지고 있다.

3.1 데이터 증강 기법(Data Augmentation)

데이터 셋의 확장을 위한 가장 간단한 방법으로 Data Augmentation 방법이 있다. Data Augmentation은 데이터에 인위적으로 변형을 가하여 불충분한 학습데이터의 양을 확보하는 방법으로 이미지 데이터와 영상 데이터에 주로 쓰인다[37, 40]. 이미지를 인위적으로 변형시키는 방법은 이동(Shift), 변환(Flip), 회전(Rotation), 크기 조정(Re-scaling), 잡음 추가(Add noise) 등의 방법이 있으며 확보한 데이터는 원본 데이터와 같이 추가되어 학습에 사용되기 때문에 모델의 성능향상에 기여 한다[42].

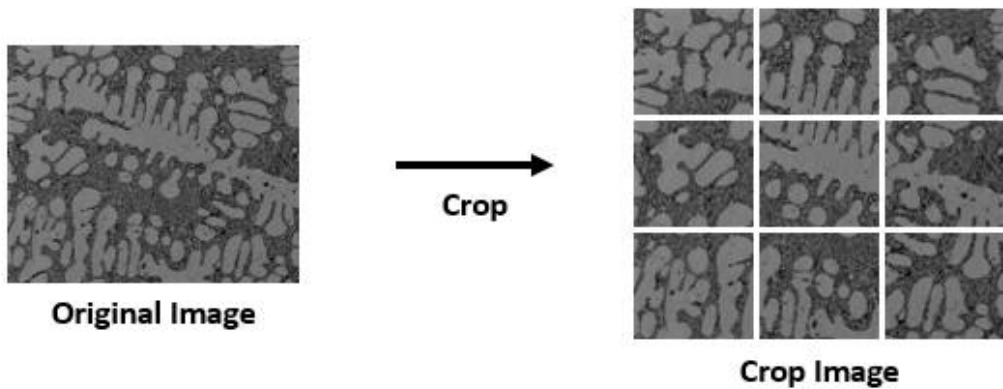


Fig. 2.17 Crop

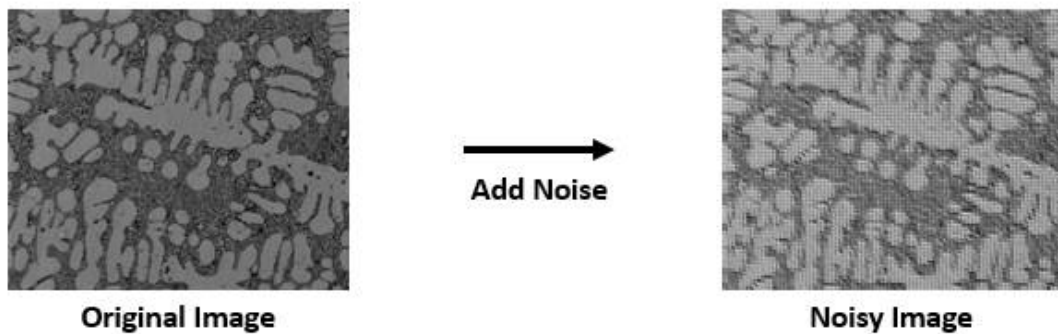


Fig. 2.18 Add Noise

Data Augmentation의 적용 과정은 Fig. 3.4에 나타내었다. 먼저 하나의 클래스에 해당하는 학습 데이터를 불러온다. 이후 데이터에 인위적인 변형을 가한 후 네트워크의 입력으로 넣어 학습 데이터로 사용한다. 이때, 데이터의 변형은 위에 설명한 방법이나 이미지의 픽셀을 한 칸씩 옮기는 방법이 될 수 있다. 픽셀을 한 칸씩 옮기는 방법은 사람의 육안으로 보았을 때 실제 데이터와 똑같이 받아들여질 수 있으나 컴퓨터의 경우 이미지를 벡터화 시켜 입력으로 받기 때문에 변형시키기 전 이미지와 다른 것으로 인식하게 된다. 때문에 학습 데이터가 증대된 것과 같은 효과를 낼 수 있으며 적용 모델이 분류기라면 분류 정확도가 상승하는 기대효과를 볼 수 있다.

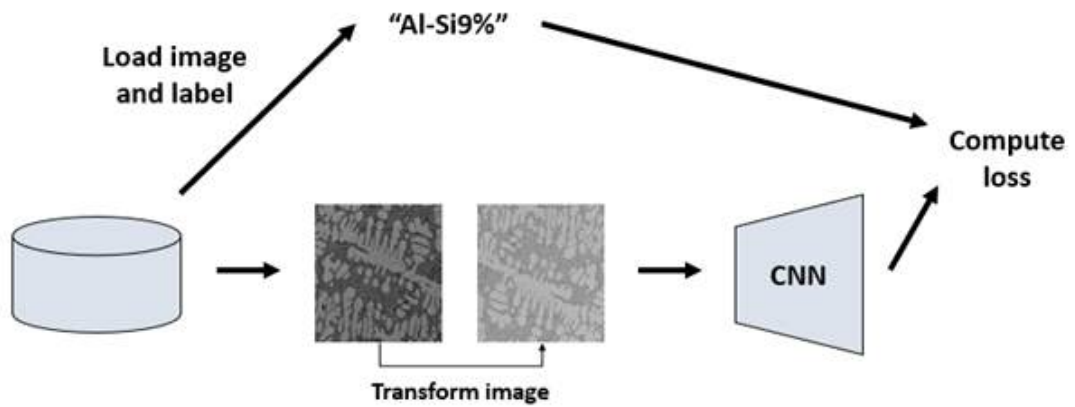


Fig. 2.19 Data Augmentation Application Example

3.2 오버 샘플링 & 언더 샘플링

딥러닝을 활용함에 있어 데이터 베이스 크기의 문제는 데이터 부족 문제뿐만 아니라 불균형 데이터 문제 또한 수반하고 있다. 불균형 데이터란 데이터 클래스 간의 비율이 균형을 이루지 않고 한쪽으로 균형이 치우친 데이터를 말한다. 예시로 공정 과정에서 정상인 제품의 데이터는 많으나 불량 제품의 데이터가 부족해 불량검출의 적용이 어렵거나 공장의 화재 예측과 예방 등 여러 분야에서 나타나는 문제이다.

이러한 데이터 불균형 문제를 해결하는 방안 중 하나로 샘플링 기반 문제 해결방법이 있다. 샘플링 기법은 데이터 클래스 별로 속해 있는 분포를 균형 있게 맞추는 방법으로, 소수 클래스 또는 다수 클래스 둘 중 어떠한 것을 중심으로 데이터 분포를 조절하느냐에 따라 오버 샘플링과 언더 샘플링 기법으로 나뉜다[17].

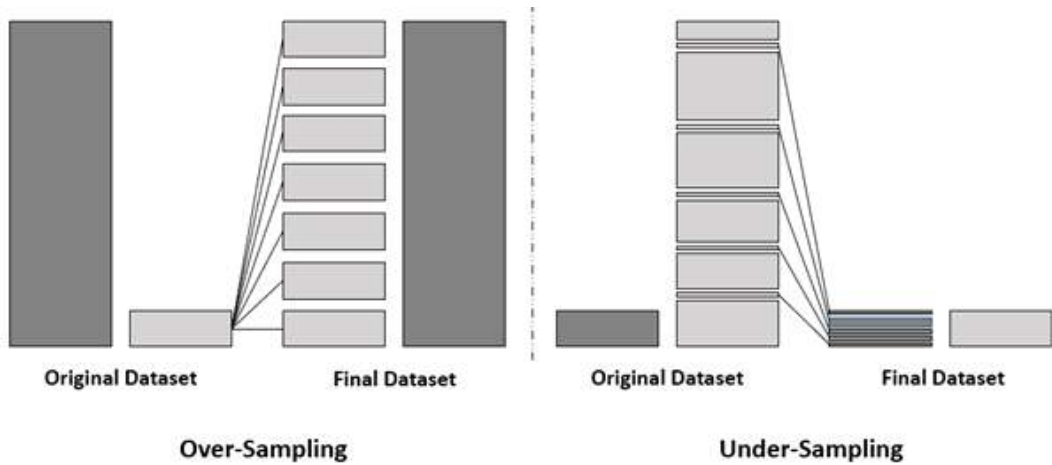


Fig. 2.20 Over-Sampling and Under-Sampling

먼저 오버 샘플링은 적은 클래스의 데이터를 다수 클래스의 데이터 크기의 분포와 유사하게 맞추는 방법으로 랜덤 오버 샘플링(Random Over-Sampling, ROS)과 SMOTE(Synthetic Minority Over-Sampling Technique) 등이 있다[3, 35]. ROS는 소수 클래스 데이터에서 무작위로 데이터를 선택해 다수 클래스의 데이터 크기와 비슷해질

때까지 반복적으로 복원 추출하는 기법으로 멀티 클래스의 적용이 가능하다. 이러한 방법은 언더 샘플링처럼 데이터의 소실이 없는 장점이 있으나 분류기 모델에 적용했을 때 과적합을 불러일으킬 가능성이 있다. SMOTE는 임의의 소수 클래스 데이터를 중심으로 최근접 이웃 k 개(k -Nearest Neighbor)를 합성해 선형의 연결 구조 사이에 새로운 데이터를 생성하는 방법이다. 소수 클래스의 데이터를 계속해서 복제하는 ROS와는 다르게 SMOTE는 새로운 데이터를 생성하기 때문에 과적합을 방지할 수 있으며 정보 손실의 문제 또한 없다. 그러나 생성 데이터는 다수 클래스의 데이터 위치를 고려하지 않기 때문에 클래스가 겹치는 데이터 생성되거나, 데이터 종류에 따른 제한적 생성, 생성 데이터가 노이즈(Noise)로 나타날 우려가 있다[11].

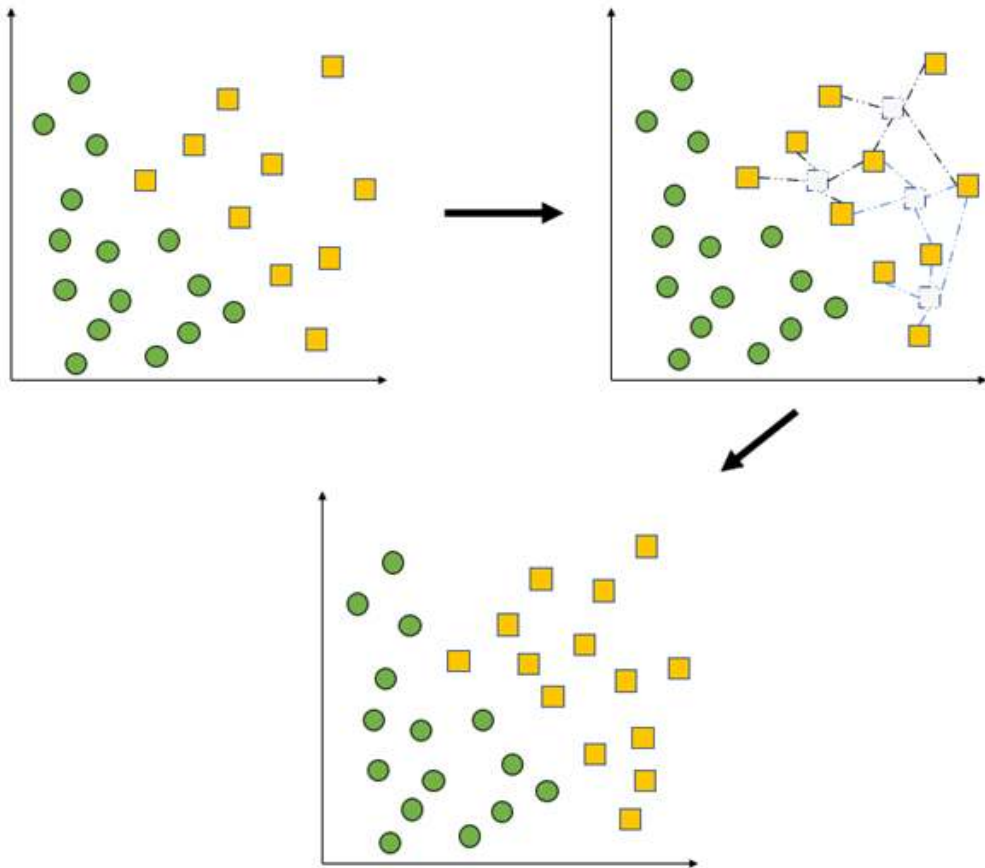


Fig. 2.21 SMOTE

언더 샘플링은 다수 클래스의 데이터를 소수 클래스의 데이터양과 일정하도록 제거하는 방법으로 Nearmiss(NM) 기법이 있으며 세 가지 방법으로 나뉜다[26]. 첫 번째 경우, 다수 클래스 데이터 값의 위치를 기반으로 가장 가까운 소수 클래스 데이터 값의 거리 계산을 통해 거리 평균이 가장 적은 것을 선택한다. 두 번째 경우, 소수 클래스 데이터 값에 가까운 다수 클래스 데이터 값을 택하는 방법으로 이 역시 거리 평균이 작은 것을 선택한다. 세 번째 경우, 소수 클래스 값에 가장 근접한 다수 클래스의 데이터를 선택한다. 위 세 가지 방법에서 소수 클래스 데이터는 세 개를 기준으로 하며 두 번째 방법이 가장 효율적인 방법으로 제시된다.

3.3 오토인코더(Autoencoder)

Autoencoder는 데이터의 특징 학습(Feature Learning)을 비지도 학습을 적용한 신경망이다[13, 38]. 모델의 구성은 입력층과 은닉층 및 출력층으로 구성되어 있으며 입력값과 출력값의 조건을 동일하게 하고 은닉층을 중심으로 좌우 대칭이 동일 구조를 가지고 있다. 여기서 은닉층의 차원 크기는 입력층과 출력층보다 낮게 설정하여야 한다. Autoencoder의 구성을 크게 보면 인코더(Encoder)와 디코더(Decoder)로 나눌 수 있다. 입력층과 은닉층 사이의 구간이 인코더이며 은닉층과 출력층 사이의 구간을 디코더라 한다.

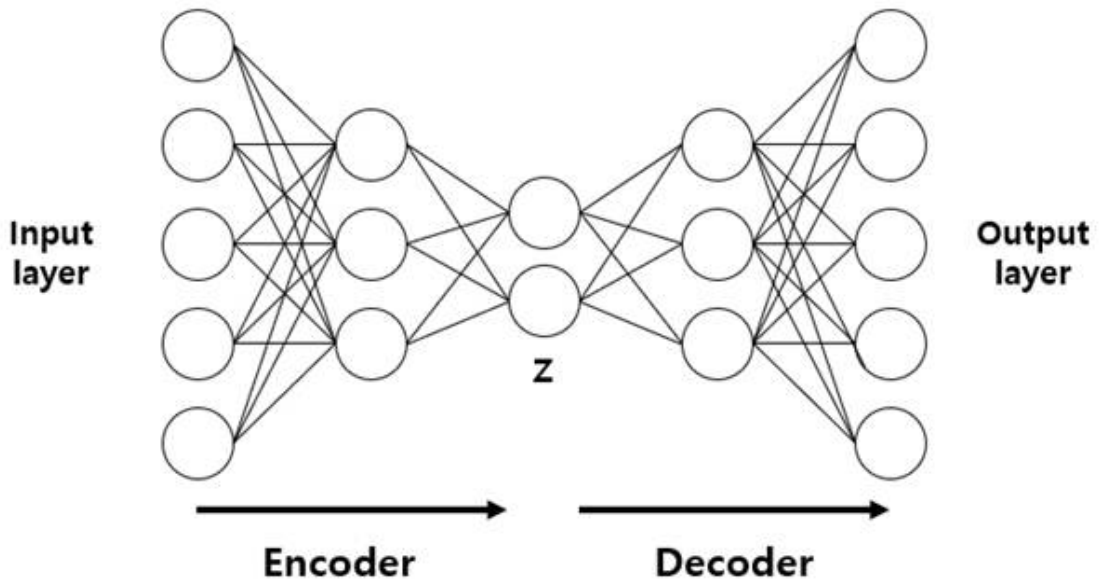


Fig. 2.22 Structure of Autoencoder

인코더는 입력 데이터의 특징을 추출하여 압축된 낮은 차원(Low-Dimensional Space)의 데이터 z 로 변환하고 이 외의 정보는 손실시킨다. 디코더는 압축된 데이터 z 를 다시 입력 데이터와 같은 고차원 데이터(High-Dimensional Space)로 재구성하여 데이터를 출력한다. 이때 출력된 데이터는 입력된 데이터를 재구성하여 생성되기 때문에 훈련 데이터와 유사하며 생성 데이터는 대체로 이미지가 흐릿한 단점이 있다. 때문에 Autoencoder는 학습 데이터를 확장하는 목적보다 입력 데이터의 특징을 추출함과 동시

에 차원 축소를 하기 위한 목적으로 사용하는 경우가 많다. 이 외에도, Autoencoder는 모델의 유용성이 좋아 인코더와 디코더가 대칭된 구조를 기반으로 은닉층의 크기를 입력층과 출력층의 크기보다 더 늘리거나 레이어의 개수를 늘리는 등의 방법을 적용한 많은 파생 Autoencoder 알고리즘이 존재한다.

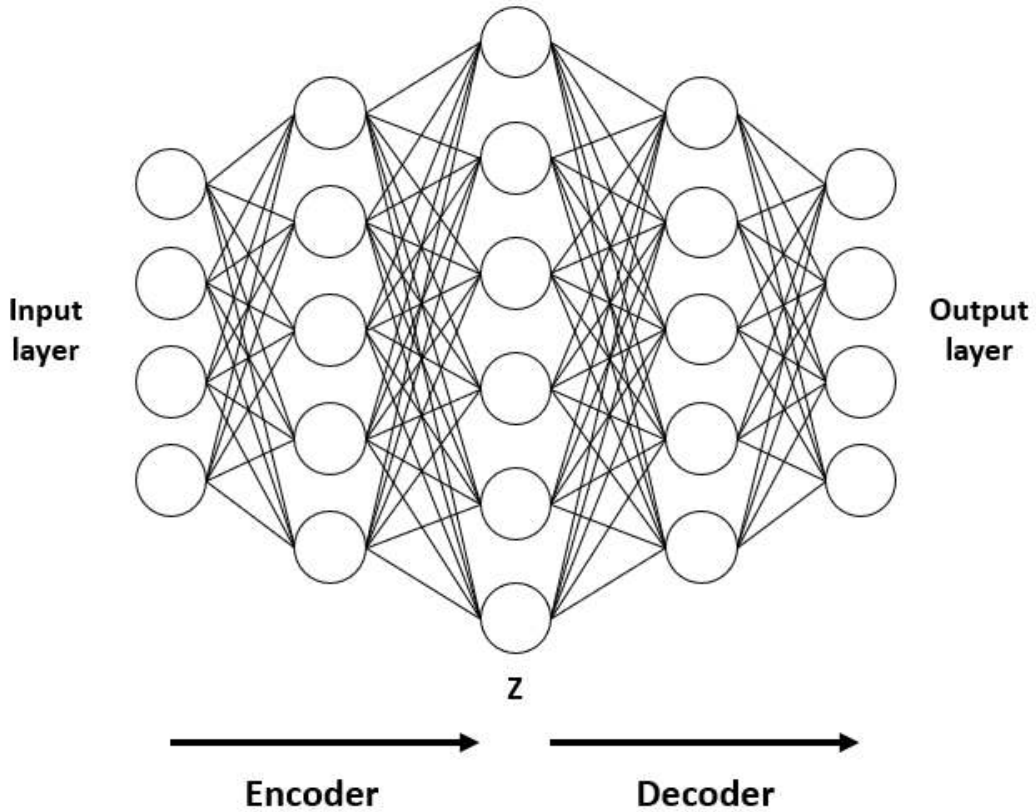


Fig. 2.23 Overcomplete Autoencoder

3.4 생성적 적대 신경망(Generative Adversarial Network, GAN)

앞서 데이터 부족과 데이터 불균형 문제의 해결방안으로 Data Augmentation, 오버 샘플링과 언더 샘플링, 오토 인코더의 방법에 대해 기술하였다. 그러나 이 중 몇몇 방법들은 데이터 전처리의 번거로움과 과적합 현상 또한 빈번하게 발생된다. 이에 반해 GAN은 비지도 학습을 이용한 알고리즘으로 데이터의 전처리가 거의 필요하지 않으며 학습되는 동안 자동으로 새로운 모조 데이터를 생성한다. 최근에는 훈련 데이터와 유사한 데이터 셋을 확보하는 것에 국한되지 않고 새로운 합성 데이터를 생성하는 연구 까지 진행되고 있다.

GAN은 생성자(Generator)와 판별자(Discriminator) 두 개의 모델이 공존하며 서로 적대적으로 학습하면서 성능을 개선하는 구조적 특징을 가지고 있다[1, 32]. 생성망 모델은 새로운 모조 데이터를 생성하는 기능을 하며 판별자 모델은 생성자 모델이 만든 모조 데이터가 진짜인지 가짜인지 판별하는 기능을 한다. 이러한 학습 과정이 반복 되면서 생성자 모델은 실제 데이터와 유사한 모조 데이터를 생성할 수 있게 되고 판별자 모델은 진짜인지 가짜인지 판단하는 진위 여부 능력이 개선된다.

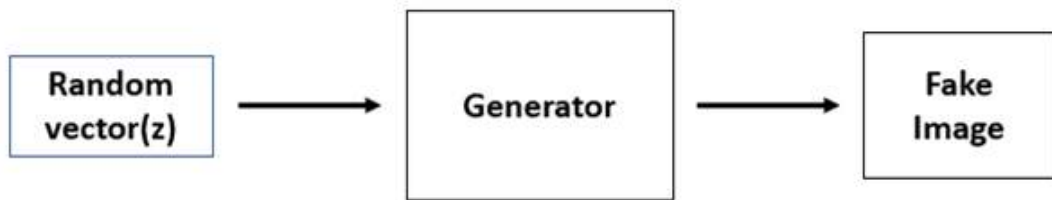


Fig. 2.24 Generator Training Process

먼저 생성자 모델은 z 를 입력으로 받아 모조 데이터를 생성한다. 입력 데이터인 z 는 모조 데이터를 생성하기 위해 잠재 공간(Latent Space)에서 추출한 데이터이다. 보통 z 의 분포는 정규 분포(Gaussian Distribution) 또는 균등 분포(Unifrom Distribution)를 사용하며 모조 데이터를 출력할 때마다 무작위로 추출되기 때문에 생성된 모조 데이터는 모두 다르다. 이후, 생성된 모조 데이터는 판별자 모델에게 전달되고 이러한 학습과정은 학습이 종료될 때까지 반복된다.

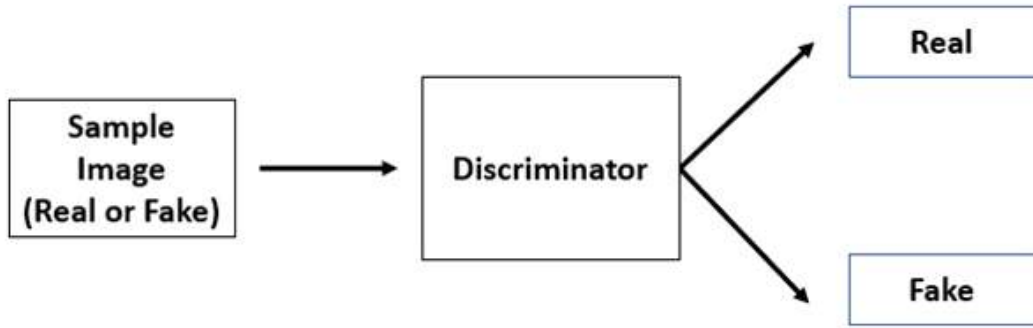


Fig. 2.25 Discriminator Training Process

판별자 모델은 실제 데이터 또는 생성자 모델이 만든 모조 데이터를 입력으로 받으며 이 두 가지의 데이터를 진짜와 가짜로 잘 구분하도록 학습된다. 이때, 출력은 진짜 데이터면 '1' 모조 데이터면 '0'의 결과를 출력한다. 출력한 결과는 다시 생성자 모델과 판별자 모델에게 전달되며 두 개의 신경망은 매개 변수를 업데이트하고 더 좋은 학습결과를 출력할 수 있도록 학습이 반복된다. 이처럼 생성자 모델과 판별자 모델은 학습이 종료될 때까지 반복적으로 이루어지며 학습과정에서 생성자 모델은 어느 시점부터 실제 데이터와 유사한 모조 데이터를 생성할 수 있게 된다.

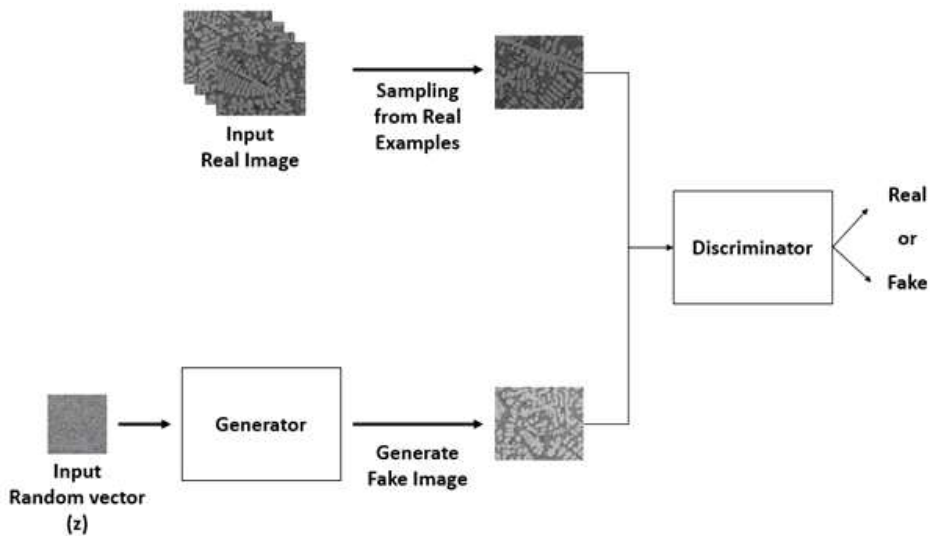


Fig. 2.26 GAN Training Process

이러한 GAN 알고리즘의 최종 목적은 생성자 모델이 실제 데이터가 가지는 분포와 근사한 모조 데이터를 출력하는 것에 있다. 대부분의 경우, 학습 초기에 생성자 모델이 출력한 모조 데이터는 거의 잡음 형태에 가까우며 이는 훈련 데이터의 분포를 학습하지 못했음을 의미한다. 그러나 학습시간이 경과됨에 따라 생성자 모델은 훈련 데이터와 비슷한 분포를 가진 모조 데이터를 생성할 수 있고 이는 GAN 모델의 학습 능력이 향상되어가고 있음을 알 수 있다.

그러나 GAN의 알고리즘으로 다양한 데이터를 적용하고 모조 데이터를 생성하기에는 여러 문제점들이 존재한다. 먼저 기울기 소실 문제로 앞서 말했듯 가중치가 업데이트 되지 않아 모델의 최적화가 어려워진다. 다음으로 모드 붕괴(Model Collapse) 현상이 있다. 모드 붕괴는 생성자 모델이 폭넓은 모조 데이터를 생성하지 못하고 하나의 이미지에 대해 비슷한 모조 데이터만 출력하는 경우를 말한다. 이는 판별자 모델의 진동이 심하거나 학습 데이터의 모드가 다양할 때 더욱 심하게 나타난다. 모드 붕괴 현상이 일어나 출력된 모조 데이터는 훈련 데이터와 유사할 수 있으나 하나의 이미지만을 계속 생성해 내기 때문에 다양한 학습 데이터의 출력이 불가능하고 가치가 떨어진다. 이외에도 구조 자체의 불안정(Unstable)으로 인해 학습이 어렵다는 단점이 있다. 이는 생성자 모델과 판별자 모델이 적대적인 구도로 학습하고 계층 간 연결구조가 완전 연결 구조로 되어있어 발생한다. 그로 인해 다양한 데이터의 도입이 어렵고 생성된 모조 데이터의 품질이 저조한 경우가 많다.

때문에 오늘날 이러한 문제점들을 보완한 여러 GAN 알고리즘들이 개발되었으며 현재도 연구 중에 있는 파생 GAN들이 존재한다. 기존 GAN은 비교적 작은 크기의 이미지를 이용해 모조 데이터를 생성하였으나 현재는 다양한 크기와 여러 종류의 데이터의 입력이 가능하며 영상, 이미지의 전환, 고해상도의 이미지 출력이 가능해졌다.

3.5 Conditional GAN(CGAN)

GAN은 생성자 모델에 입력되는 z 가 무작위로 추출되기 때문에 출력 데이터를 예측할 수 없다. 이에 반해 CGAN은 생성자 모델과 판별자 모델에 추가적인 정보(y)를 줌으로써 조건부 데이터의 확장이 가능하다[30]. 추가적인 정보는 데이터가 속해 있는 클래스의 레이블이나 다른 형태의 데이터가 될 수 있다. 이러한 학습 과정은 생성자 모델이 새로운 모조 데이터를 생성할 때 추가적인 정보에 맞게 z 의 추출이 가능해지며 판별자 모델 또한 입력 데이터와 추가적인 정보를 고려하여 데이터를 판별하는 구조를 가진다.

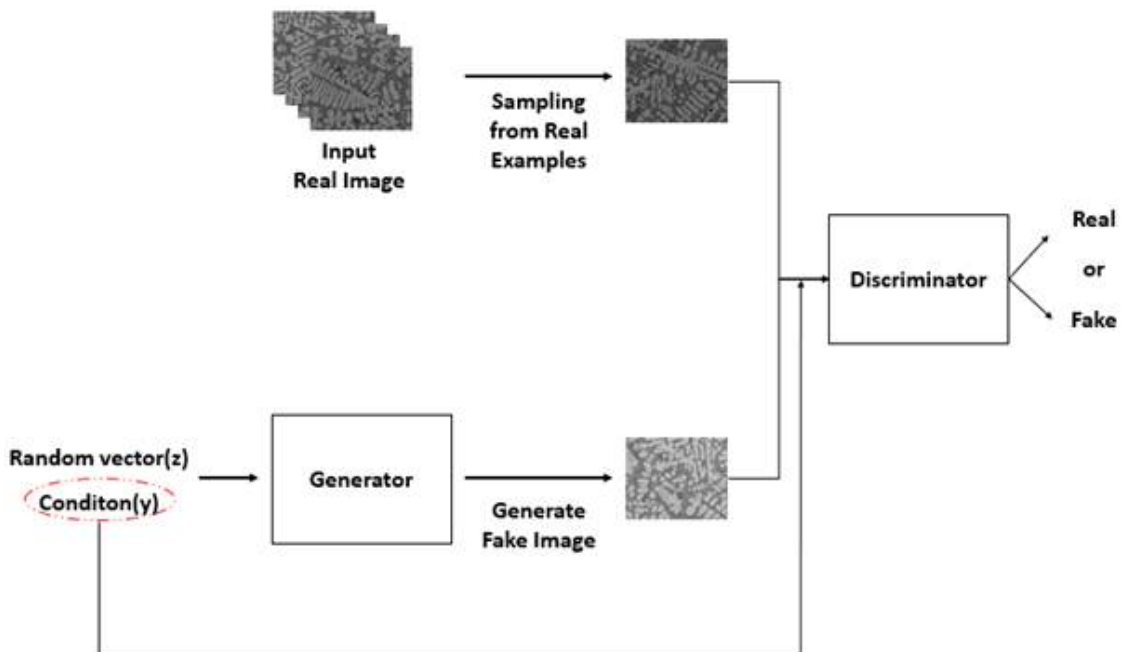


Fig. 2.27 CGAN Training Process

3.6 Cycle GAN

Cycle GAN은 데이터 클래스 간의 상호 변환의 가능한 알고리즘이다[33, 34, 47, 50]. 이 전에는 Pix2Pix를 이용하여 이와 같은 작업이 가능하였으나[16], 데이터 클래스 X에 쌍을 이루는 데이터 클래스 Y가 있어야 가능하며 이러한 데이터 셋은 확보하기 어려운 단점이 존재한다. 때문에 Cycle GAN은 서로 쌍을 이루는 데이터 클래스가 없어도 데이터 클래스 간의 변환이 가능하게 설계되었다.

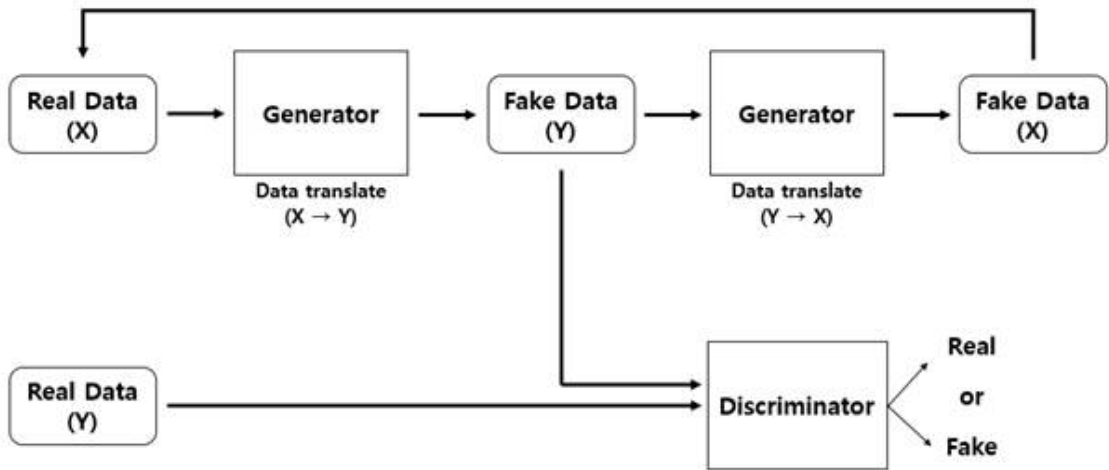


Fig. 2.28 Cycle GAN Training Process

Cycle GAN의 구조와 학습과정은 다음과 같다. 먼저 생성자 모델은 데이터 클래스 X와 대응하는 Y 클래스의 모조 데이터를 생성한다. GAN의 경우, 생성자 모델은 데이터 클래스 X와 유사한 모조 데이터의 생성이 가능하나 클래스 Y에 속하는 데이터는 생성하지 못한다. 때문에 생성된 모조 데이터를 다시 생성자 모델에 입력하여 처음에 입력된 데이터로 회귀하도록 학습하여 준다. 이 과정에서 생성자 모델은 데이터의 복원과 판별자가 출력한 결과값을 업데이트하여 데이터의 변환이 가능해지도록 학습된다. 입력된 데이터 클래스가 Y일 경우 또한 학습과정은 동일하다.

4. Al-Si alloy

Al은 내식성이 우수하고 전기 및 열전도성, 표면처리성 또한 뛰어나 주조용 합금으로 많이 쓰이고 있다. 주조용 Al 합금은 첨가 원소에 따라 분류되는데 그 중 Al-Si 합금은 유동성과 주조성, 가공성, 내식성 등이 좋아 사용빈도가 가장 높다.

Al-Si 합금의 2원계 상태도를 Fig. 4.1에 나타내었다. Si 공정점은 12.6wt%로 공정 온도인 577°C에서 Si는 Al에 최대 약 1.65%로 고용되는데 이후 온도가 낮아짐에 따라 고용도 또한 함께 낮아진다. 공정온도 577°C, Si 공정조성 12.6wt%를 중심으로 적은 Si를 함유할 경우 아공정 합금(Hypo-eutectic), 많은 Si를 함유할 경우 과공정 합금(Hyper-eutectic)이라 칭한다[8].

아공정 조성의 경우 액상선 온도에 도달하면 α -Al상이 먼저 생성된 후 온도가 감소함에 따라 양이 증대된다. 이후 공정온도인 577°C에 다다르면 고액공존영역(α -Al + Liquid)을 형성하며 공정온도에서는 잔류액상이 공정상인 α -Al+Si상으로 변태가 이뤄진다. 공정온도 이하로는 α -Al상과 α -Al+Si상의 공정상이 미세조직으로 형성된다.

공정 조성의 경우 공정온도 이상에서는 완전한 액체상태로 존재한다. 이후 공정온도에 도달하면 액상이 고체 상태의 α -Al+Si상의 공정상으로 형성된다. 공정온도 이하에서는 α -Al+Si의 공정상으로 구성된 미세조직으로 형성된다.

과공정 조성의 경우 액상선 온도에 도달하면 초정 Si상이 먼저 정출된다. 이후 온도 감소에 따라 Si상의 분율은 증가하게 되며 공정온도에 이르면 잔류액상은 α -Al+Si의 공정상으로 점차 응고된다.

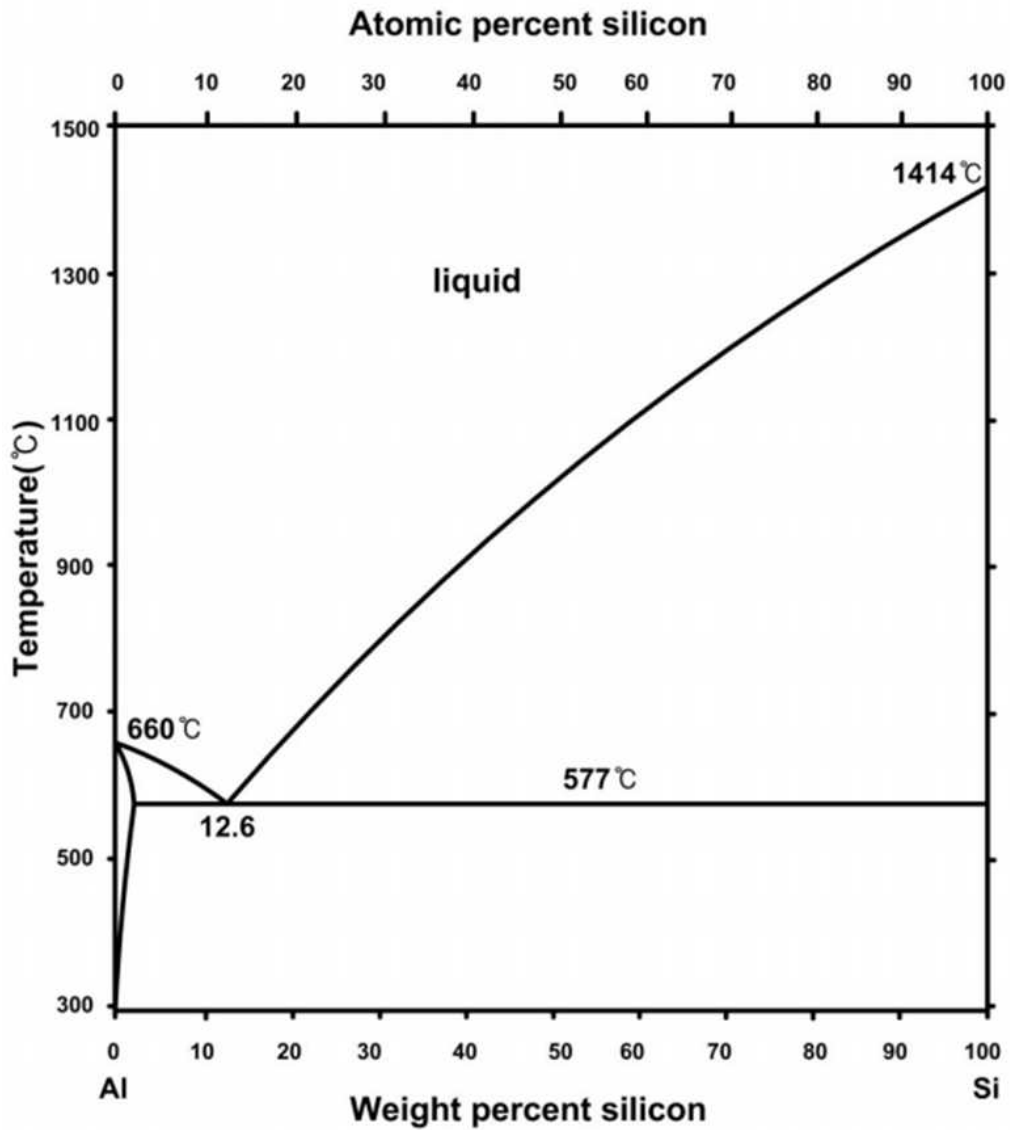


Fig. 2.29 Phase diagram of the Al-Si binary alloy system

제 3 장 모델링 설계 및 구현

3.1 DCGAN

데이터 생성을 위한 방법과 알고리즘들은 현재도 연구 중에 있다. 그중 DCGAN(Deep Convolutional Generative Adversarial Network)은 GAN의 구조적 변경을 통해 향상된 모델이다. GAN은 학습이 불안정하다는 것이 가장 큰 문제점으로 작용했다. 이에 DCGAN은 학습이 안정적으로 이뤄질 수 있게 개선되었으며 이로 인해 생성된 모조 데이터의 품질이 상당히 향상되었고 다양한 학습 데이터의 도입이 가능해졌다.

두 알고리즘의 학습과정은 거의 동일하나 가장 큰 차이점은 뉴런 간의 연결 구조에 있다. 기존 GAN의 완전 연결 계층 구조는 데이터의 공간적 정보가 소실되고 합성곱 계층보다 훨씬 많은 파라미터를 가지고 있어 학습시간이 오래 걸린다는 단점을 가진다. 때문에 DCGAN은 완전 연결 계층을 합성곱 신경망을 도입하여 이러한 문제점들을 보완하였다. 또한 신경망의 마지막 분류 단계의 레이어에서도 제거하고 Global Average Pooling으로 대체하였는데 이는 DCGAN 알고리즘의 연산량을 감소시키고 학습의 안정성에 기여한다.

3.2 알고리즘 설계

합성곱 신경망은 판별자 모델과 생성자 모델 둘 다 적용하였으나 연산 방식에 차이가 있다. 판별자 모델은 입력 데이터의 특징을 추출하고 분류하는 일반적인 Strided Convolution을 거친다. 그러나 생성자 모델의 경우 이와 반대의 연산을 거치는 Transposed Convolution을 사용한다. Transposed Convolution은 입력 데이터 크기의 데이터를 얻기 위한 연산 방식으로 특징 맵이 아닌 입력 데이터에 패딩을 적용하여 연산을 거칠수록 데이터의 크기를 키워 나간다.

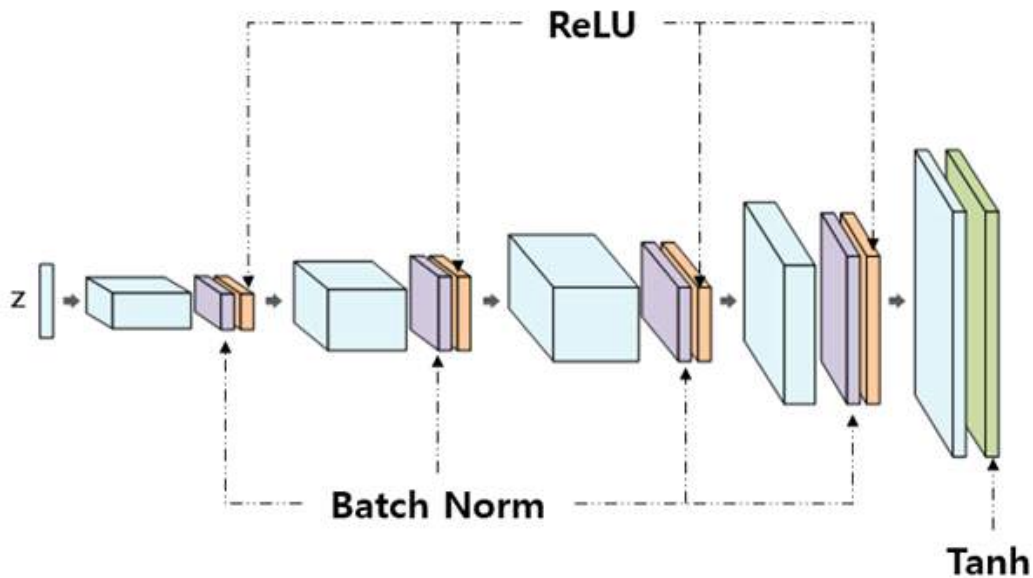


Fig. 3.1 Structure of DCGAN Generator

DCGAN 생성자 모델의 상세 구조를 Fig. 3.1에 나타내었다. 먼저 GAN과 마찬가지로 벡터화되어 있는 입력 z 를 입력으로 받아 앞서 언급한 Transposed Convolution 연산을 거치며 입력 데이터의 크기를 키워 나간다. 이후 4개의 Batch Normalization 레이어를 추가하였는데 이는 학습 속도를 상승시키고 데이터의 분포를 조절해 주는 역할을 한다. 이 외에 4개의 ReLU 활성화 함수층을 Batch Normalization 층 뒤에 배치해 두었으며 마지막 레이어에는 Tanh를 사용한다.

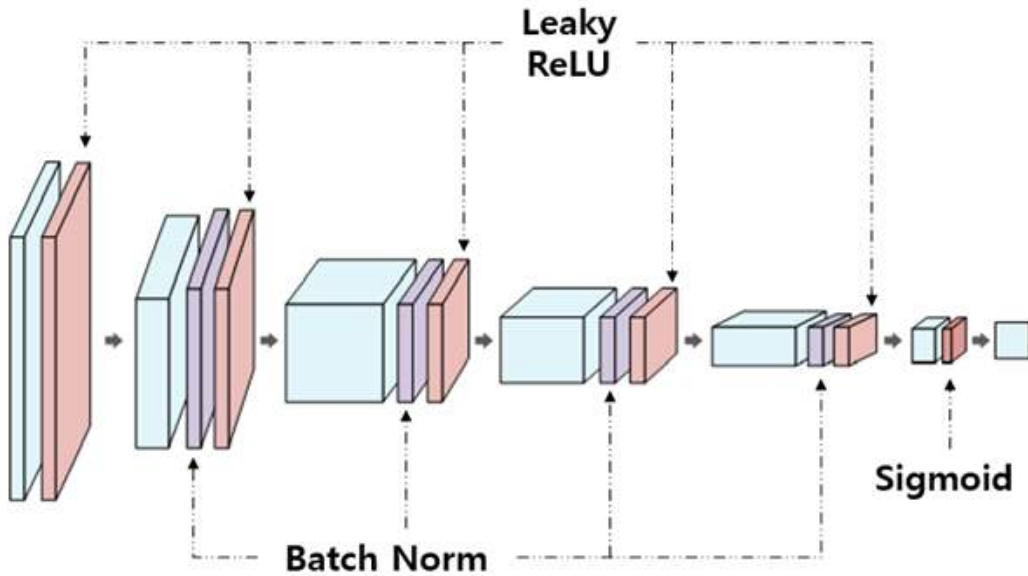


Fig. 3.2 Structure of DCGAN Discriminator

DCGAN 판별자 모델의 상세 구조는 Fig. 3.2에 나타내었다. 판별자 모델은 Strided Convolution 연산을 거치며 생성자 모델에게 전달받은 데이터 또는 실제 데이터의 크기를 점차 줄여 나간다. 이후 생성자 모델과 같이 4개의 Batch Normalization 레이어를 추가하였다. 생성자 모델과는 다르게 판별자 모델에는 5개의 Leaky ReLU 활성화 함수층을 적용하였으며 마지막 레이어에는 Sigmoid를 사용한다.

제 4 장 실험방법

본 논문에서는 DCGAN 알고리즘을 이용해 Al-Si 합금의 미세조직을 생성한다. 실험에 쓰이는 Al-Si 합금의 조성은 Al-Si6%, Al-Si9%, Al-Si12%, Al-Si15%로 각각 아공정, 공정, 과공정 조성에서의 미세조직을 이미지를 선택하였다. Al-Si 합금은 Si조성에 따라 각기 다른 미세조직 구조의 형태가 나타나며 대표적인 특징을 가지고 있다. 이에 한 가지 조성에 대한 미세조직 이미지뿐만 아니라 다른 구조적 특징을 가진 미세조직 이미지 또한 그려낼 수 있을지에 대해 알아보기 위해 각기 다른 4가지 조성의 Al-Si 합금에 대하여 재료를 선택하였다. 미세조직 데이터는 조연마와 미세연마를 거친 후 켈러시약(Keller's reagent : 95%, H₂O, 2.5%HNO₃, 1.5%HCl, 1.0%HF)을 이용해 에칭(Etching)하여 광학현미경으로 관찰 후 수집하였다.

4.1 실험환경

학습 과정의 실험환경은 다음과 같다. Windows 10, NVIDIA GEFROCE GTX 1060 Ti, Intel(R) Core(TM) i5-9600K CPU, 16.0GM RAM, Python 3.5의 환경에서 실험을 진행하였고 이 외에 CUDA 9.0, cuDNN은 이와 호환되는 7.6.4 버전을 설치하여 적용하였다.

CPU	Intel(R) Core(TM) i5-9600K CPU
RAM	16.0GB
GPU	NVIDIA GTX 1060 Ti
OS	Windows 10

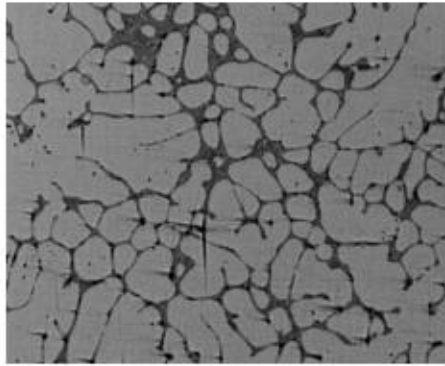
Table. 4.1 Computer Specifications

4.2 데이터 전처리 및 설계

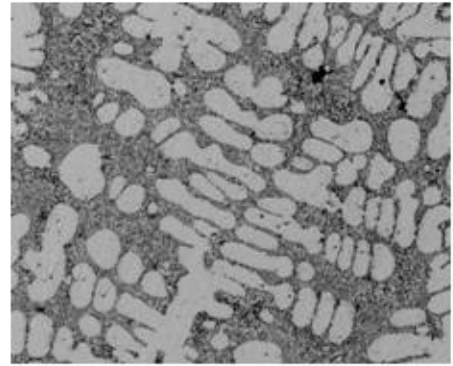
Al-Si6%, Al-Si9%, Al-Si12%, Al-Si15%의 미세조직 이미지는 모두 x100배의 배율과 너비(width)와 높이(height)의 크기를 645x530으로 고정하였고 하나의 조성당 최대 400장씩 수집하였다. 보통 미세조직 이미지의 크기는 더 큰 크기로 수집하는 것이 일반적이거나 학습 데이터의 크기가 클수록 학습 속도가 느려지고 생성자 모델이 학습 데이터의 분포를 파악하는 데 있어 매우 오래 걸린다는 단점이 있다. 때문에 이미지 데이터의 크기를 축소하여 수집하고 사용하였다. 수집된 미세조직 이미지는 모두 RGB(Red, Green, Blue) 채널을 갖는 3차원 이미지이며 명암에 대한 전처리는 따로 진행하지 않았다.

미세조직 이미지에서 축척표시(Scale Bar)는 표기해두지 않고 학습을 진행하였다. 앞서 말했듯, DCGAN에서 생성자 모델에 입력으로 사용되는 z 는 학습 데이터에서 무작위로 추출되어 진다. 이러한 학습 과정에 있어 축척표시가 완전히 추출되지 않고 일부만 추출될 가능성이 농후하다. 축척표시가 포함된 모조 데이터는 잡음으로 나타날 가능성이 있으며 다수의 표본이 추출되면 생성되는 모조 데이터의 품질은 매우 저조할 수 있다. 때문에 학습 데이터에는 축척표시를 제외한 나머지 미세조직 이미지만을 사용하였다.

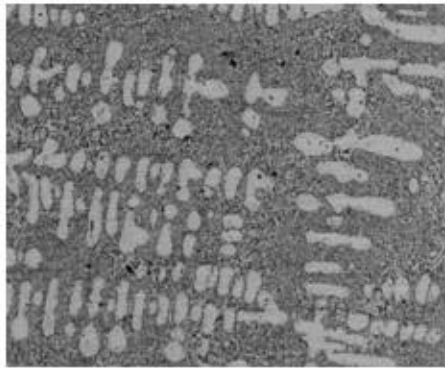
DCGAN과 같은 생성모델 알고리즘을 사용하는 주된 목적 중 하나는 제한된 데이터 셋을 확장하는 것에 있다. 이에 본 연구에서는 모조 데이터를 생성할 수 있는 최소한의 데이터를 이용해 실험을 진행하였다. 실험에 앞서, 100장을 기점으로 10장씩 데이터 수를 줄여가며 결과를 비교하였다. 그 결과 70장 미만의 데이터를 가지고 학습을 진행하였을 때 이미지의 특징을 학습하지 못하고 잡음 형태의 이미지만 출력하였다. 또한 학습횟수가 1000 Epoch를 초과했을 때도 같은 현상이 발생하였다. 따라서 실험에 사용된 데이터 수는 Al-Si 합금의 조성당 70장의 미세조직 이미지를 가지고 학습을 진행하였으며 학습횟수는 1000 Epoch로 고정하였다.



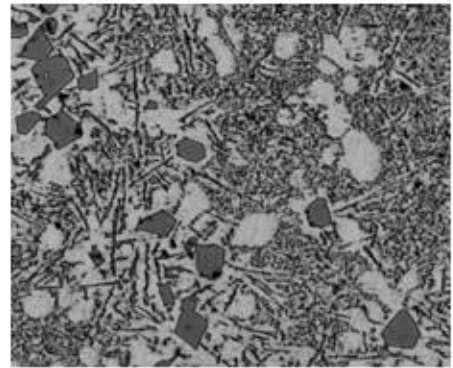
Al-6wt%Si



Al-9wt%Si



Al-12wt%Si



Al-15wt%Si

Fig. 4.1 Microstructure Image Preprocessing

4.3 데이터 검증 방법

생성된 모조 데이터는 분류기 모델인 Inception V3 알고리즘을 사용해 유사도를 측정한다. Inception V3 알고리즘은 1x1의 합성곱 연산을 사용하여 데이터의 차원축소와 연산량의 감소로 인해 빠른 결과의 출력이 가능하고 이 외에 3x3, 5x5 크기의 합성곱 연산 또한 사용함으로써 데이터의 특징을 효율적으로 추출할 수 있다. 또한 입력된 데이터의 분류에도 높은 정확도를 보여 이미지 분류에 많이 사용된다. Inception V3를 이용해 유사도를 측정하는 과정은 Fig. 4.2 에 나타내었다.

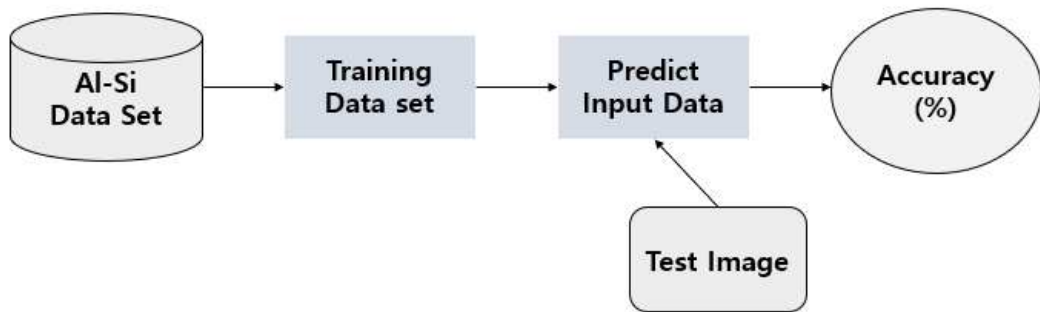


Fig. 4.2 Inception V3 Classification Process

앞서 말했듯, AI-Si 합금의 미세조직 데이터 70장을 이용해 AI-6wt%Si, AI-9wt%Si, AI-12wt%Si, AI-15wt%Si 의 모조데이터를 생성한다. 이후, 생성된 모조 데이터를 실제로 사용할 수 있는지 검증하기 위해 3가지 다른 조건으로 실험을 진행한다.

- 1) Real Image(70) – Fake Image(30), Non-DB Image(30)
- 2) Fake Image(70) – Real Image(30), Non-DB Image(30)
- 3) Real+Fake Image(70) – Real Image(30) Fake Image(30), Non-DB Image(30)

Fig. 4.3 Three Method of Verification

검증의 방법은 학습데이터와 모조데이터, 학습에 사용되지 않은 데이터로 구성되며 7 : 3 : 3 의 비율로 진행된다. 일반적으로는 4 : 3 : 3 로 검증 데이터의 비율을 구성하나 본 연구에서는 높은 신뢰성을 위해 검증 데이터의 비율을 소폭 높였다. 추가적으로 DCGAN 알고리즘을 이용해 새로운 합성 데이터를 생성할 수 있는지 알아보기 위해 Al-Si6%와 Al-Si12%, Al-Si9%와 Al-Si15% 의 데이터 셋을 혼합하여 Al-Si9%와 Al-Si12% 의 모조 데이터를 생성해 본다.

5. 실험결과

5.1 Al-Si 합금 미세조직 이미지 생성

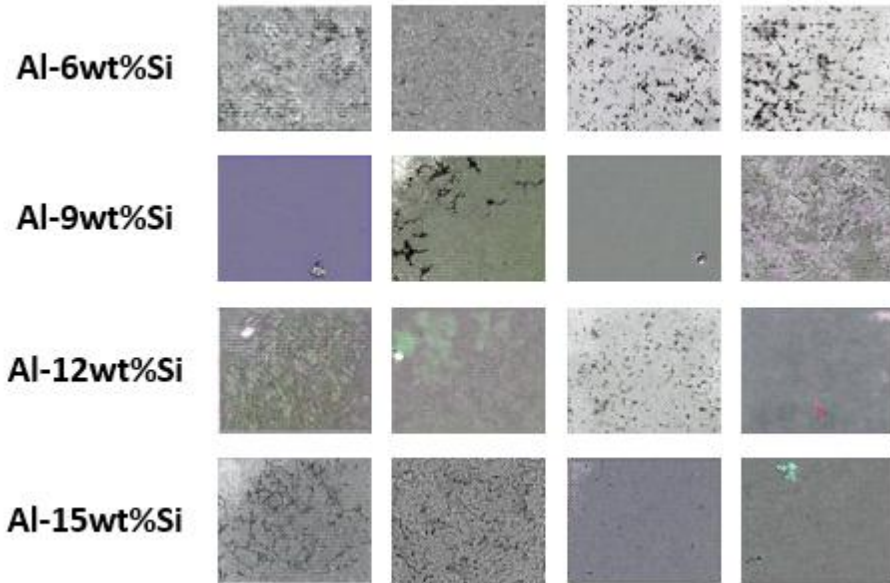


Fig. 5.1 Fake images generated early in the Epoch

Fig. 5.1은 학습 초기에 생성자 모델이 출력한 모조 데이터를 나타낸 것이다. 출력된 모조 데이터를 보면 모든 조성의 Al-Si 합금의 구조적 특징이 전혀 나타나지 않고, 잡음 형태의 이미지만 얻을 수 있었다. 이는 판별자 모델과 생성자 모델의 학습이 제대로 이루어지지 않았음을 의미한다. 그러나 학습이 진행됨에 따라 모조 데이터의 품질이 안정화 되어갔으며 Al-Si 합금 조성에 따른 구조적 특징 또한 잘 그려낸 이미지를 얻을 수 있었다.

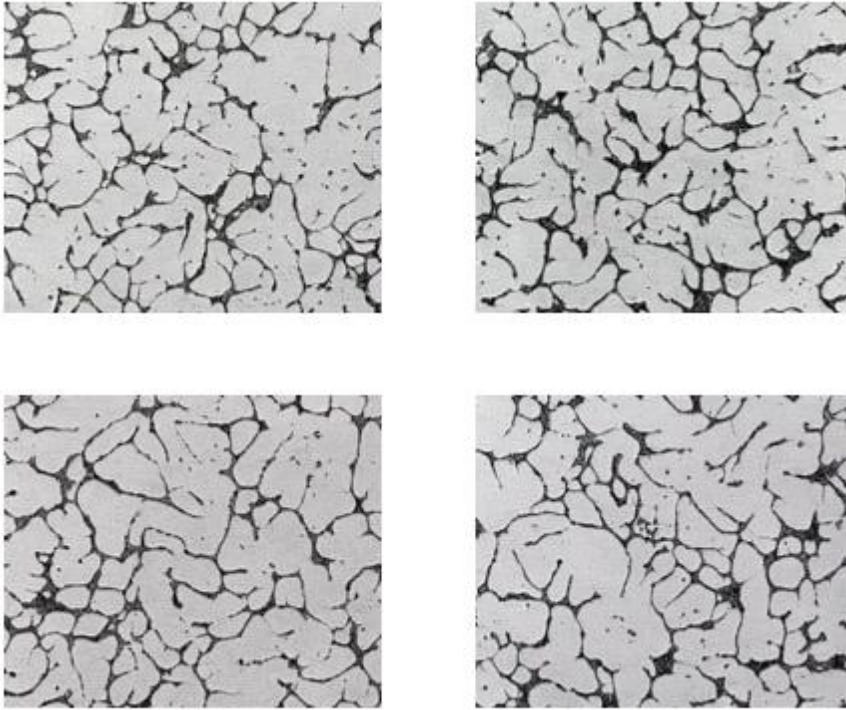


Fig. 5.2 Al-6wt%Si Fake Images

Al-6wt%Si 합금은 아공정 조성의 합금으로 미세조직 이미지는 초정 α -Al상이 주를 이루는 구조형상을 띄는 것이 특징이다. 위의 생성된 Al-6wt%Si 합금의 모조 데이터를 보면 대표적인 특징인 초정 α -Al상이 잘 생성된 것을 볼 수 있고 생성 데이터의 품질 또한 학습 초기에 비해 많이 향상된 것을 알 수 있다.

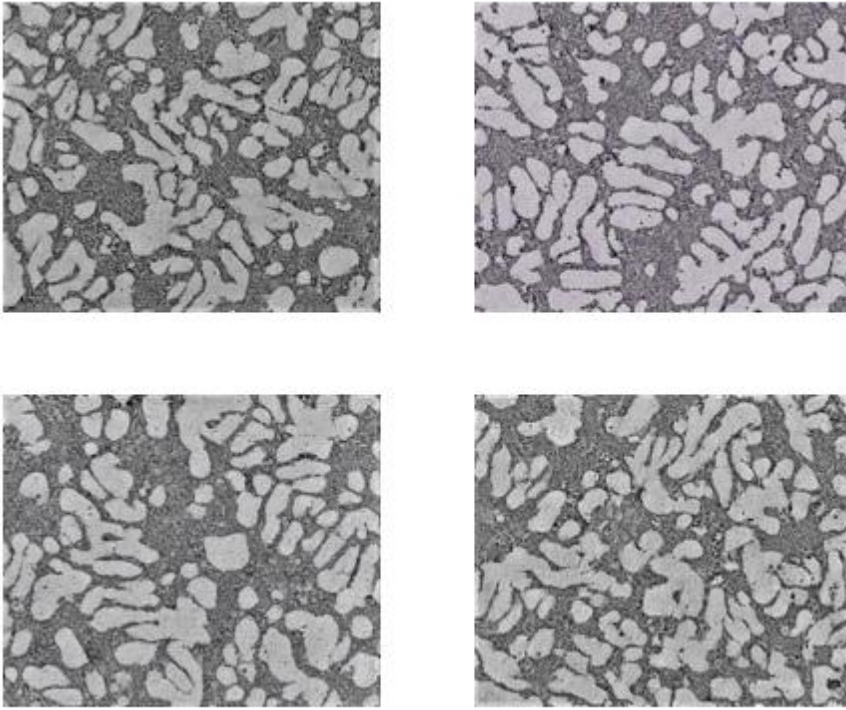


Fig. 5.3 Al-9wt%Si Fake Images

Al-9wt%Si 합금 미세조직 형상은 Al-6wt%Si 합금의 미세조직에 비해 응고가 진행됨에 따라 초정 α -Al상의 분율이 점차 줄어들고 α -Al+Si의 공정상이 같이 나타나는 것이 특징이며 α -Al상은 수지상의 형태로 나타난다. Fig. 5.3의 모조 데이터를 보면 실제 데이터와 비슷하게 초정 α -Al상이 Al-6wt%Si와 다르게 분율이 매우 줄어든 것을 확인할 수 있고 α -Al+Si의 공정상 또한 잘 나타난 것을 확인할 수 있다. 그러나 초정 α -Al상이 수지상으로 드러난 형태는 거의 드물게 나타났는데 이는 생성자 모델이 α -Al상의 크기가 분율이 줄어든 것에 대해서는 학습이 잘 된 것을 의미하나 수지상의 형태는 학습이 미흡하다는 것을 알 수 있다.

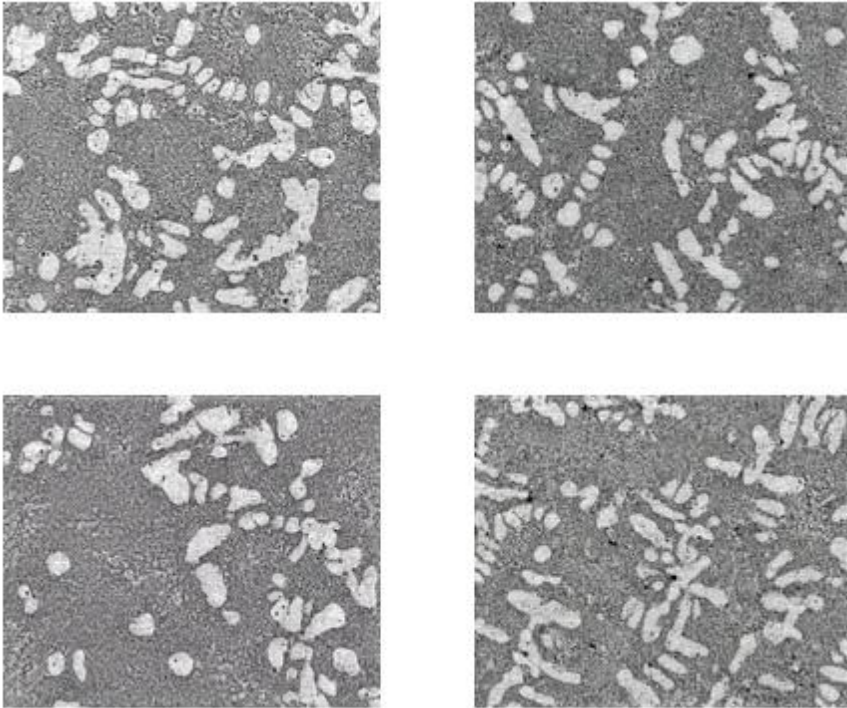


Fig. 5.4 Al-12wt%Si Fake Images

Al-12wt%Si 합금은 공정조성인 Al-12.6wt%Si 합금에 가장 가까운 아공정 합금으로 초정 α -Al상의 크기와 분율이 적게 나타난다. Fig. 5.4 의 모조 데이터들을 보았을 때 초정 α -Al상은 Al-9wt%Si에 비해 현저히 줄어든 것을 알 수 있고 α -Al+Si의 공정상은 반대로 크게 늘어난 것을 알 수 있다. 그러나 Al-12wt%Si 합금의 모조데이터 또한 수치상의 형태는 관찰하기 어려운데 이는 앞서 설명한 Al-9wt%Si 합금의 모조 데이터를 생성할 때와 비슷한 이유로 보인다.

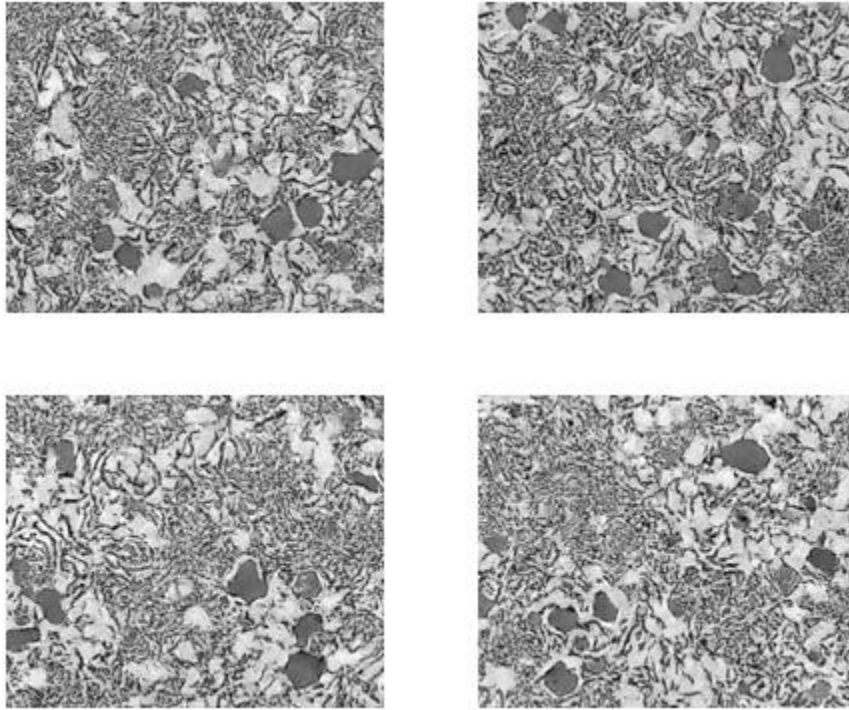


Fig. 5.5 Al-15wt%Si Fake Images

Al-15wt%Si 합금은 과공정 조성에 속하는 합금으로 조대한 Si상이 초정으로 정출되고 초정 α -Al상과 α -Al+Si의 공정상이 나타나며 Si침상 조직이 나타난다. Fig. 5.5에 생성된 모조 데이터를 보면 앞서 언급한 공정조직과 α -Al상 등의 특징들은 잘 생성되었음을 알 수 있으나, 초정 Si상과 Si침상 조직은 불온전한 형상을 띄고 있다. 실제 Al-15wt%Si 미세조직 이미지에서 초정 Si상의 경우 각진 형태의 형상을 띄고 Si침상 조직은 길고 조대한 형상으로 나타나나, 출력된 모조 데이터에서는 뭉그러진 형태로 나타났다. 이는 데이터의 전체적인 패턴은 학습이 잘 되었으나 데이터의 세부적인 구조는 학습이 미흡하다는 것을 알 수 있다.

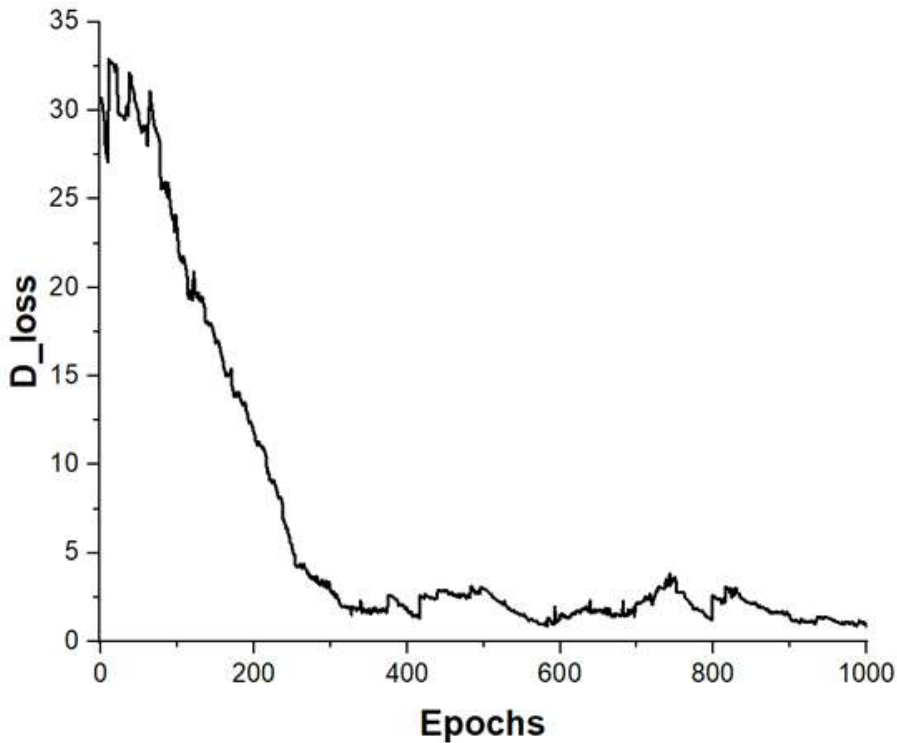


Fig. 5.6 The changes of discriminator loss function value in training process.

딥러닝 알고리즘은 학습 상태와 성능을 알아보기 위한 지표로 loss function을 사용한다. Fig 5.6은 DCGAN 알고리즘에서 판별자 모델의 loss를 나타낸 것으로 d_{fake_loss} 와 d_{real_loss} 를 합한 것으로 D_loss 로 표현한다. 앞서 말했듯, 판별자 모델은 진짜 이미지라고 판단하면 '1', 가짜 이미지라고 판단되면 '0'의 값을 출력한다. 따라서 판별자 모델이 진짜 이미지인지 가짜 이미지인지 판단하기 어려운 '0.5'의 값에 가깝게 출력되면 DCGAN 알고리즘의 학습이 이상적으로 진행된 것이라 판단할 수 있다. d_{fake_loss} 는 가짜 이미지라고 판단할 때의 확률이며 d_{real_loss} 는 진짜 이미지라고 판단할 때의 확률을 뜻한다.

D_loss 의 그래프의 변화 양상을 살펴보면 학습 초기에는 높은 손실 값을 출력하고 다소 많은 진동이 나타나나 300 Epoch 이후에는 0.5에 가까운 출력값을 내는 것으로 보아 학습이 안정적으로 이루어진 것을 확인할 수 있다.

5.2 Inception V3 알고리즘을 이용한 유사도 평가

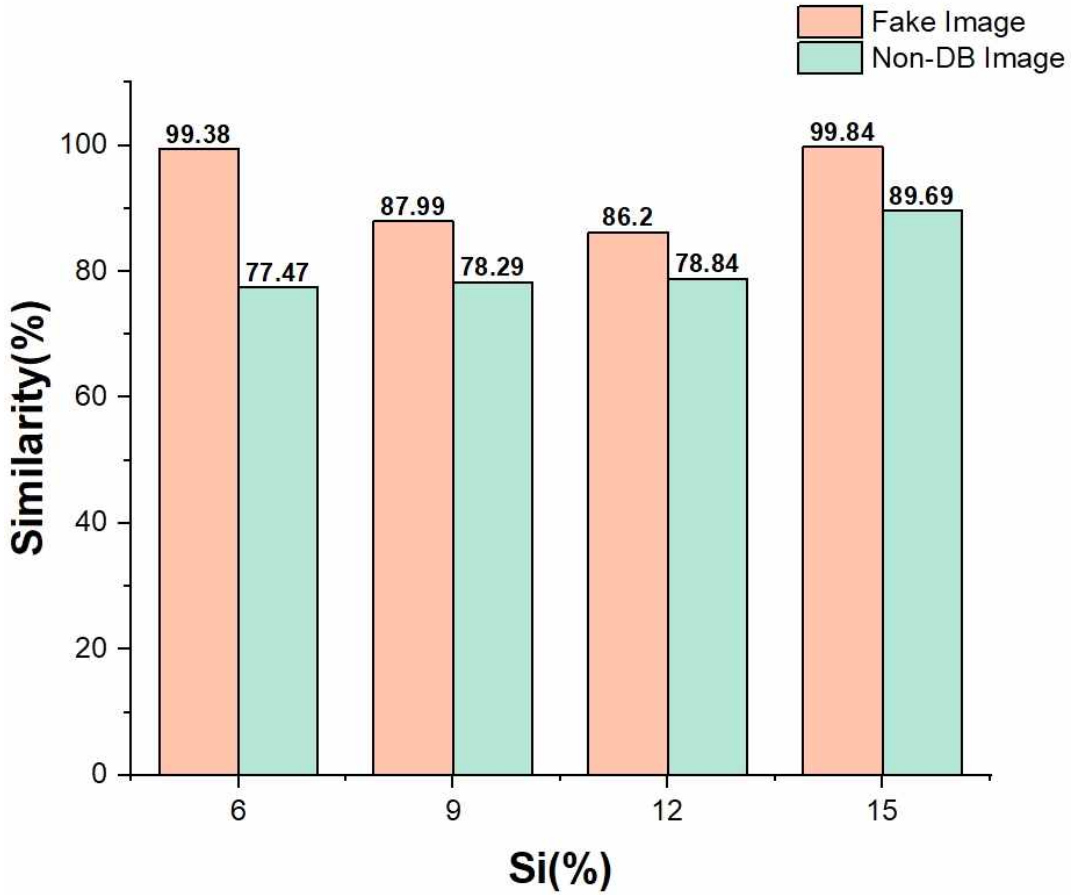


Fig. 5.7 Similarity bar graph when fake image and non-DB image are put in Inception V3 algorithm that trained real data

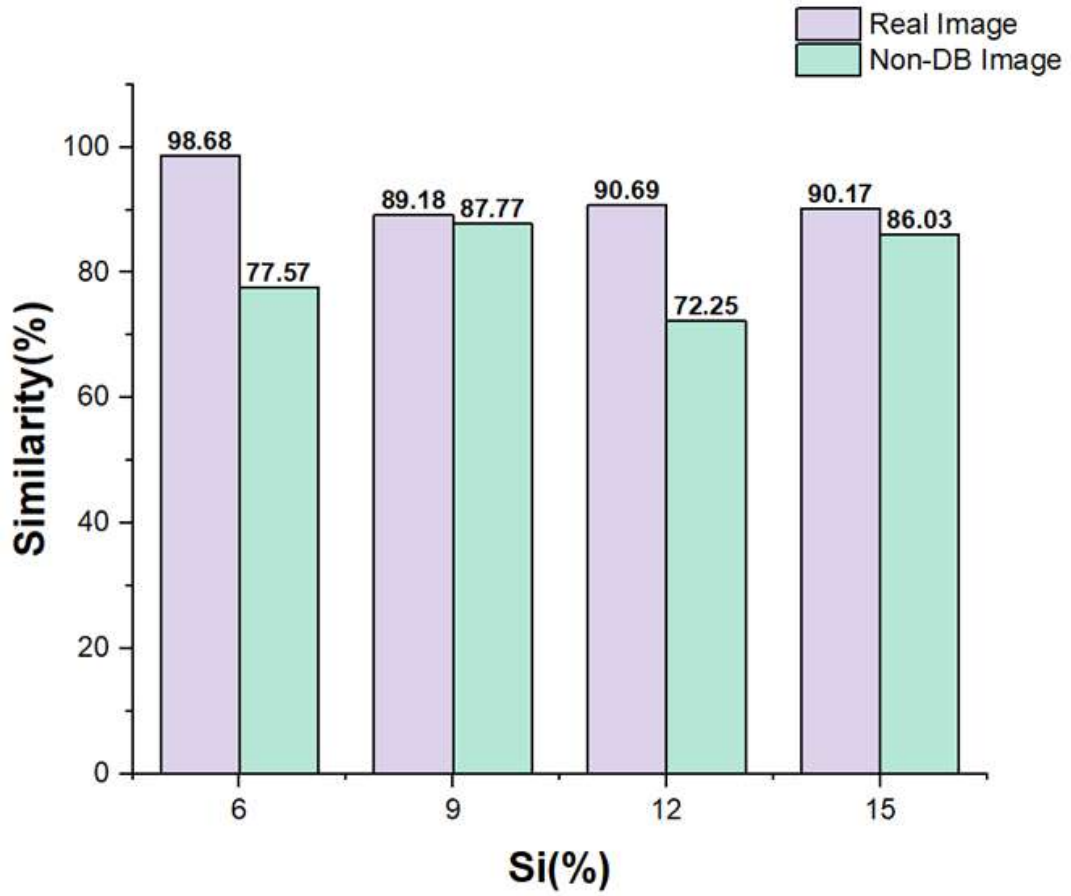


Fig. 5.8 Similarity bar graph when real image and Non-DB image are put in Inception V3 algorithm that trained fake data

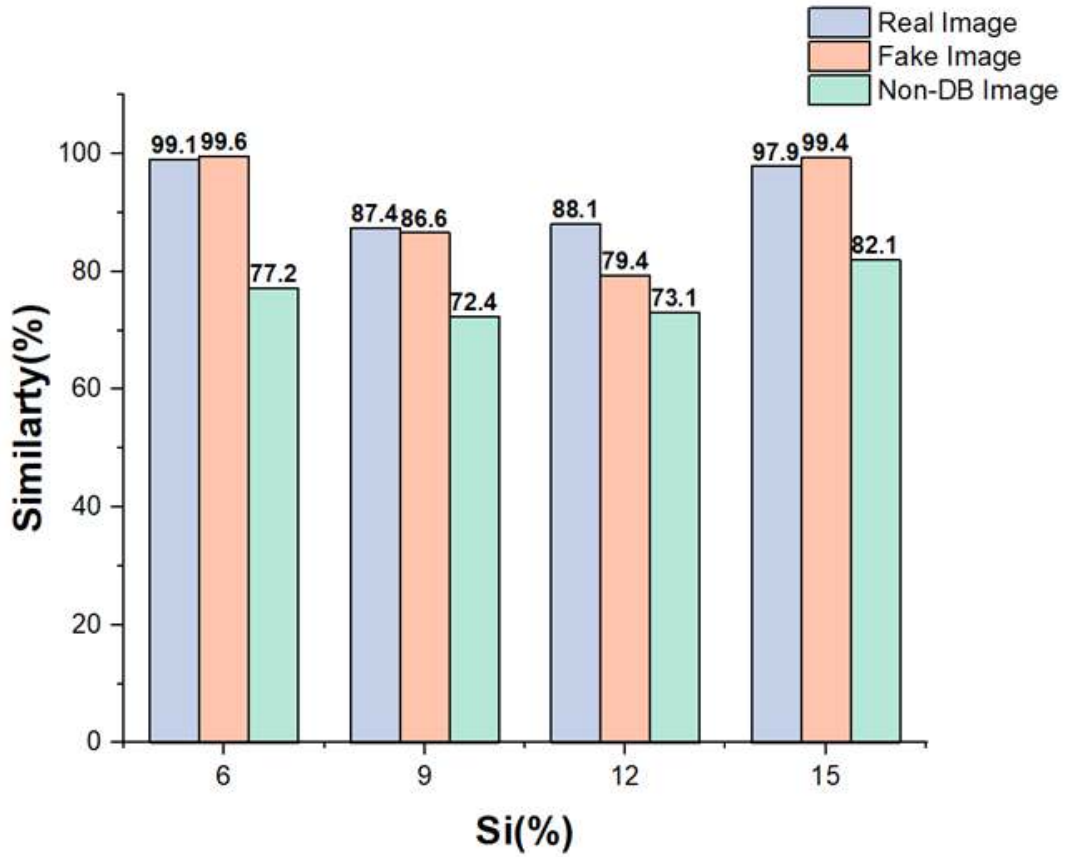


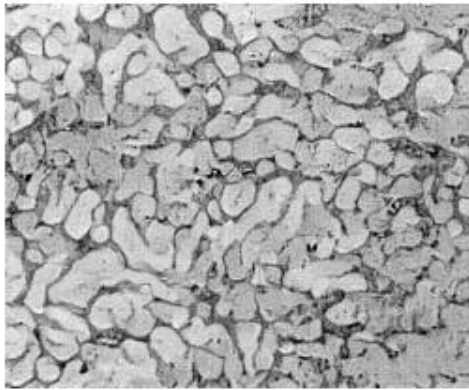
Fig. 5.9 Similarity bar graph when Real image, Fake Image and Non-DB image are put in Inception V3 algorithm that trained Fake data and Real data

앞서 말한 실험 조건에 따른 결과를 Fig 5.4 ~ Fig 5.6에 나타내었다. Fig 5.4의 경우 모조 데이터를 입력으로 넣었을 때 각 조성에서 99.38%, 87.99%, 86.2%, 99.84%의 평균 유사도를 나타내었고 학습에 사용되지 않은 데이터를 입력으로 넣었을 때 77.47%, 78.29%, 78.84%, 89.69%의 평균 유사도를 나타냈으며 이후 다른 실험 조건의 경우 또한 비슷한 결과가 도출되었다.

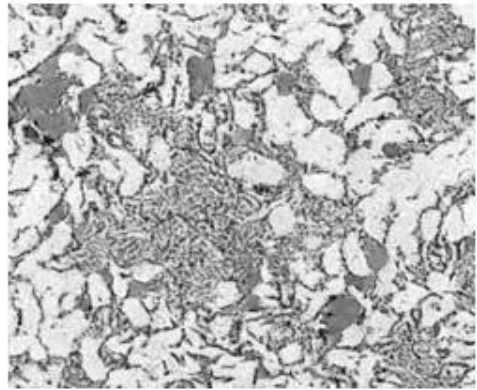
학습된 Inception V3 알고리즘에 실제 데이터와 학습 데이터를 테스트 이미지로 입력했을 경우 평균 85% 이상의 유사도를 나타내었으나 학습에 사용되지 않은 데이터를 입력했을 경우 평균 72% 이상의 유사도로 상대적으로 낮은 결과값이 도출되었다. 이러한 경향이 나타나는 것은 학습에 사용되지 않은 데이터들은 미세조직 이미지들의 배율과 크기가 각각 다른 점, 명암 등의 원인으로 보인다. 특히, 미세조직 이미지의 경우 촬영 배율에 따라 이미지의 차이가 나타나며 배율 차이가 클수록 차이는 극대화 된다. 그러나 학습에 사용되지 않은 데이터 또한 각 조성에 해당하는 유사도가 가장 높게 나온 것으로 보아 모조 데이터의 신뢰성 향상에 기여하는 것으로 판단된다.

5.3 합성 데이터 생성

이전 실험들은 각 Al-Si 합금의 조성에 해당하는 데이터 셋을 확보하기 위한 실험으로 새로운 합성 데이터를 그려내진 못했다. 이에 추가적인 실험으로 Al-6wt%Si, Al-12wt%Si 합금의 미세조직 이미지와 Al-9wt%Si, Al-15wt%Si 합금의 미세조직 이미지 데이터를 혼합하여 학습을 시행하였을 때 중간 조성인 Al-9wt%Si, Al-12wt%Si 합금의 미세조직 이미지를 그려낼 수 있는지 진행하였다.



**Al-6wt%Si + Al-12wt%Si
Microstructure Image**



**Al-9wt%Si + Al-15wt%Si
Microstructure Image**

Fig. 5.10 Composite image generated by mixing Al-6wt%Si and Al-12wt%Si, Al-9wt%Si and Al-15wt%Si datasets.

먼저 Al-6wt%Si와 Al-12wt%Si 미세조직 이미지를 혼합하여 생성된 모조 데이터는 Al-Si6wt%Si가 47.9% 유사도로 가장 높게 측정되었으며 Al-9wt%Si와 Al-15wt%Si 미세조직 이미지를 혼합하여 생성한 모조 데이터의 경우는 Al-15wt%Si가 71.32% 유사도로 가장 높게 측정되었다. 이는 전자의 경우, 공정상 조직의 분율이 적게 나타났기 때문인 것으로 보이며 후자의 경우 Si초정상의 모양이 다른 조직에 비해 특징이 뚜렷하기 때문에 나타난 결과로 판단된다.

제 6 장 결론 및 향후 연구

본 연구에서는 DCGAN 알고리즘을 이용해 제한된 Al-Si 주조 합금의 미세조직 이미지를 확장하는 실험을 진행하였다. 실험 이미지는 Al-6wt%Si, Al-9wt%Si, Al-12wt%Si, Al-15wt%Si 합금의 미세조직 이미지를 사용했으며 그 결과 다음과 같은 결론을 얻을 수 있었다.

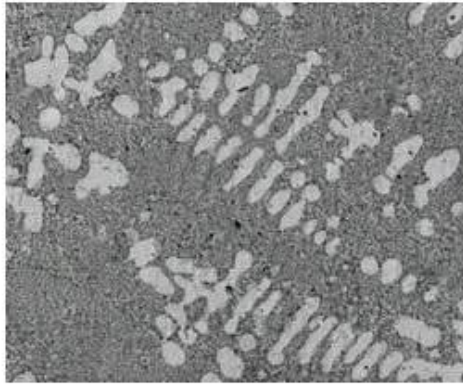
a. 생성된 모조 데이터들은 Al-Si 주조 합금의 각 조성에 해당하는 미세조직 이미지의 대략적인 구조적 특징은 잘 나타났으나 Al-15wt%Si 합금의 경우 Si 초정상과 침상 조직이 뭉그러진 형태로 나타났으며 아공정 합금의 경우 수지상 같은 연속된 패턴은 그려내진 못했다.

b. 생성된 모조 데이터를 Inception V3 알고리즘을 이용해 유사도를 측정해본 결과 해당 조성의 합금의 피크가 가장 높게 나왔으나 학습 데이터에 포함되지 않는 테스트 이미지의 경우 학습 데이터와 모조 데이터의 유사도에 비해 비교적 낮은 유사도를 나타내었다. 그러나 이는 학습 데이터의 포함되지 않는 이미지는 촬영 배율과 명암, 냉각속도의 조건 등이 다르기 때문에 나타난 결과로 보이며 평균 유사도 또한 평균 72% 이상이 나온 것으로 보아 결과의 신뢰성 향상에 기여하는 것으로 판단된다.

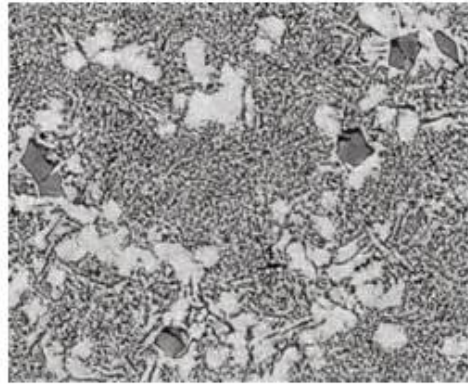
c. 학습상태를 나타내는 지표인 D_{loss} 그래프를 보았을 때 학습이 안정적으로 이루어진 것을 확인할 수 있으며 출력값이 낮아질수록 모조 데이터의 품질 또한 향상되었다.

d. 조성이 다른 Al-Si 합금 미세조직 이미지를 이용해 새로운 합성 데이터를 생성해 보았으나 상의 혼합 비율이 매우 불균일하고 낮은 유사도를 보였다. 따라서 새로운 합성 데이터를 생성하기 위해서는 다른 접근 방식이 필요한 것으로 판단된다.

이후, 결론 a의 문제점을 해결하기 위해 400장의 학습 데이터를 사용해 모조 데이터를 생성해 보았으며 그 결과를 Fig 6.1에 나타내었다.



Al-12wt%Si



Al-15wt%Si

Fig. 6.1 Generate Al-12wt%Si, Al-15wt%Si fake microstructure image using 400 training data

학습 데이터를 늘려 실험을 진행해 본 결과 Al-12wt%Si 미세조직의 경우 수지상의 패턴을 그려낼 수 있게 되었으며 Al-15wt%Si 미세조직의 경우 각진 형태의 Si초정과 비교적 직선 형태의 침상형태를 그려내었다. 이러한 결과를 토대로 더 많은 학습 데이터를 확보할 수 있다면 더 사실적인 미세조직 이미지를 얻어낼 수 있음을 확인하였다.

생성모델을 사용하는 주된 목적 중 하나는 제한된 데이터 셋을 확장하여 딥러닝에 적용하기에 충분한 학습 데이터 크기를 만들기 위함이다. 본 연구에서는 생성모델인 DCGAN을 이용해 Al-Si 합금의 모조 데이터를 생성해 보고 Inception V3 알고리즘으로 유사성을 비교해본 결과 실제 데이터로 사용할 수 있음을 확인하였다. 그러나 조성이 다른 미세조직 이미지를 사용해 중간 조성의 Al-Si 합금 미세조직 이미지는 생성하지 못했다. 때문에 새로운 합성 데이터를 생성하기 위해서는 다른 접근 방식이 필요한 것으로 보여진다.

참고 문헌

- [1] Ba, H. Improving Detection of Credit Card Fraudulent Transactions using Generative Adversarial Networks. arXiv preprint arXiv:1907.03355. (2019)
- [2] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, Vol.20, No.3, pp.273-297, 1995
- [3] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. SMOTE: synthetic minority oversampling technique. Journal of artificial intelligence research. pp. 321-357. (2002)
- [4] C. Robert, Machine learning, a probabilistic perspective, Taylor & Francis, 2014.
- [5] Deng L, Yu D. Deep learning: methods and applications[J]. Foundations and Trends® in Signal Processing, 2014, 7(3-4): 197-387.
- [6] E. Kim and W. Kim, "Face anti-spoofing based on combination of luminance and chrominance with convolutional neural networks", Journal of Broadcast Engineering, Vol. 24, No. 6, pp. 1113-1121, Nov. 2018
- [7] E. Philippe and C. Agon, "Time series data mining," ACM Computing Surveys (CSUR), vol 45, no. 12, pp. 1-34, Nov. 2012.
- [8] Francois R. Mollard, Merton C. Flemings and Eisuke F. Nuyama : Journal of Metals, November, 1987;34
- [9] Fritz, H. (INTELLIGENT AUTONOMOUS VEHICLES, Vol.2 No.-, [1995]) Neural Speed Control for Autonomous Road Vehicles
- [10] F. Rosenblatt, The Perceptron - a perceiving and recognizing automaton. Report 85-460-1, Cornell Aeronautical Laboratory (1957), F. Rosenblatt, Psychological Review 65, 386 (1958).
- [11] Goodfellow, I., Bengio, Y. and Courville, A., 2017, Deep Learning, MIT Press, Cambridge, MA, pp. 233~234
- [12] H. Huang, J. Joseph, M. Huang, and T. Margolina, "Automated Detection and Identification of Blue and Fin Whale Foraging Calls by Combining Pattern Recognition and Machine Learning Techniques," Proceeding of OCEANS 2016 MTS/IEEE Monterey, pp. 1-7, 2016
- [13] HINTON, Geoffrey E. A practical guide to training restricted Boltzmann machines. In: Neural networks: Tricks of the trade. Springer, Berlin,

Heidelberg, 2012. p. 599-619.

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville and Y. Bengio, "Generative Adversarial Nets," in NIPS, 2014.

[15] IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.

[16] Isola, P., J. Zhu, T. Zhou, and A. A. Efros, 2017. Imagenet-Image Translation with Conditional Adversarial Networks, Proc. of 2017 International Conference on Computer Vision and Pattern Recognition, Honolulu, HI, Jul. 22-Jul. 25, vol. 1, pp. 5967-5976.

[17] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in Proceedings of the ACM International Conference on Machine Learning, New York, pp.935-942, 2007.

[18] J. Yuna, H. Nagaharab, and I. Park, "Classification and restoration of compositely degraded Images using deep learning", Journal of Broadcast Engineering, Vol. 24, No. 3, May 2019.

[19] K. Alex, S. Ilya, and H. Geoffrey, "ImageNet Classification with Deep Convolutional Neural Networks," Proceeding of Advances in Neural Information Processing System, pp. 1097-1105, 2012

[20] Karen Simonyan & Andrew Zisserman, VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION, 2014

[21] Kong, Xiangjie, et al. "Mobility dataset generation for vehicular social networks based on floating car data." IEEE Transactions on Vehicular Technology 67.5 (2018): 3874-3886

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[23] LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. nature, 2015, 521.7553: 436.

[24] LECUN, Yann, 1995, 3361.10: 1995/KARPATHY, Andrej, 2014. P.1725-1732

[25] Li, X.-r.; Zhao, G.-z.; Zhao, L.-y.; (CONTROL AND DECISION, Vol.23 No.7, [2008]) [SCOPUS] Design of decision-tree-based support vector machines multi-class classifier based on vector projection

[26] Mani, I., & Zhang, I. (2003). kNN approach to unbalanced data distributions:

a case study involving information extraction. In Proceedings of workshop on learning from imbalanced datasets (Vol. 126).

[27] Marin T. Hagan, Howard B. Demuth, Mark Beale, Neural Network Design, PWS Publishing, Boston, 1995

[28] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," European conference on computer vision, Springer, Cham, pp. 818–833, Sep. 2014.

[29] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," J. Elsevier, vol. 40, no. 1, pp.200–210, Jan. 2013.

[30] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014

[31] Moonasar, V.;Venayagamoorthy, G.; (INTELLIGENT ENGINEERING SYSTEMS THROUGH ARTIFICIAL, Vol.10 No.-, [2000]) Artificial Neural Network Based Automatic Speaker Recognition Using a Hybrid Technique for Feature Extraction

[32] Mullick, S. S., Datta, S., & Das, S. Generative adversarial minority oversampling. In Proceedings of the IEEE International Conference on Computer Vision. pp. 1695–1704. (2019)

[33] M.Y. Liu and O. Tuzel, "Coupled Generative Adversarial Networks," Proceeding of Conference on Neural Information Processing Systems, pp. 469–477, 2016

[34] M.Y. Liu, T. Breuel, and J. Kautz, "Unsupervised Image-to-Image Translation Networks," Proceeding of Conference on Neural Information Processing Systems, pp. 700–708, 2017

[35] N. V. Chawla, et al., "SMOTE: synthetic minority oversampling technique," Journal of Artificial Intelligence Research, Vol.16, pp.321–357, 2002

[36] O. Carugo and F. Eisenhaber, Data Mining Techniques for the Life Sciences, Humana Press, vol. 609, 2010

[37] Perez, Luis, and Jason Wang. "The effectiveness of data augmentation in image classification using deep learning." arXiv preprint arXiv:1712.04621 (2017)

[38] Pierre Baldi, "Autoencoders, Unsupervised Learning, and Deep Architectures," JMLR: Workshop and Conference Proceedings 27:37–50, 2012

[39] Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised

- representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434 (2015)
- [40] Salamon, Justin, and Juan Pablo Bello. "Deep convolutional neural networks and data augmentation for environmental sound classification." *IEEE Signal Processing Letters* 24.3 (2017): 279–283
- [41] Sandberg, A.; Lansner, A.; Petersson, K. M.; Ekebe; (Network, Vol.13 No.2, 2002) [SCI,SCIE,SCOPUS] A Bayesian attractor network with incremental learning
- [42] Shijie, J., Ping, W., Peiyi, J. and Siping, H., 2017, "Research on Data Augmentation for Image Classification based on Convolution Neural Networks," *Chinese Automation Congress*, Vol. 1, No. 10, pp. 4165~4170.
- [43] S. K. Pal, S. Mitra, "Multilayer perceptron, fuzzy sets, and classification", *IEEE Transaction on Neural Networks*, vol. 3, no. 5, pp. 683–697, 1992.
- [44] S. Lee, C. Choi, and M. Kim, "CNN-based people recognition for vision occupancy sensors", *Journal of Broadcast Engineering*, Vol. 23, No. 2, March 2018, pp. 274–282.
- [45] Srivastava, Chandan, Lap Lambert Academic Publishing, [2011] Support vector data description : means of detecting artificially-induced postural degradation, machine learning, biomedical signal processing
- [46] T. Kim, T. Adali Fully complex multilayer perceptron for nonlinear signal processing *J. VLSI Signal Process. Syst. Signal Image Video Technol. Special Issue: Neural Networks Signal Process.*, 32 (2002), pp. 29–43
- [47] T. Kim, M. Cha, H. Kim, J.K. Lee, and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks," *Proceeding of International Conference on Machine Learning*, pp. 1857–1865, 2017
- [48] Y. Chen et al., "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, 2016
- [49] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel, *Neural Comput.* 1, 541 (1989).
- [50] Zhu, J. Y., T. Park, P. Isola, and A. A. Efros, 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, *Proc. of 2017 International Conference on Computer Vision*, Venice, Italy, Oct. 22–Oct. 29, vol. 1, pp. 2242–2251.

[51] Zou, B.; Li, L.-q.; (JOURNAL- HUBEI UNIVERSITY NATURAL SCIENCE EDITION, Vol.26 No.1, [2004]) The relation of solution between support vector machine regression and nu-support vector machine classification

[52] Z. Zhang, M. Lyons, M. Schuster, S. Akamatsu, Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron, IEEE Proceedings of the Second International Conference on Automatic Face and Gesture Recognition (FG' 98), Nara, Japan, 1998, pp. 454-459.