JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 27 No. 12, pp. 59-68, December 2022
https://doi.org/10.9708/jksci.2022.27.12.059

Deep Learning Similarity-based 1:1 Matching Method for Real Product Image and Drawing Image

Gi-Tae Han*

*Professor, Dept. of Computer Engineering, Gachon University, Seongnam, Korea

[Abstract]

This paper presents a method for 1:1 verification by comparing the similarity between the given real product image and the drawing image. The proposed method combines two existing CNN-based deep learning models to construct a Siamese Network. After extracting the feature vector of the image through the FC (Fully Connected) Layer of each network and comparing the similarity, if the real product image and the drawing image (front view, left and right side view, top view, etc) are the same product, the similarity is set to 1 for learning and, if it is a different product, the similarity is set to 0. The test (inference) model is a deep learning model that queries the real product image and the drawing image in pairs to determine whether the pair is the same product or not. In the proposed model, through a comparison of the similarity between the real product image and the drawing image, if the similarity is greater than or equal to a threshold value (Threshold: 0.5), it is determined that the product is the same, and if it is less than or equal to, it is determined that the product is a different product. The proposed model showed an accuracy of about 71.8% for a query to a product (positive: positive) with the same drawing as the real product, and an accuracy of about 83.1% for a query to a different product (positive: negative). In the future, we plan to conduct a study to improve the matching accuracy between the real product image and the drawing image by combining the parameter optimization study with the proposed model and adding processes such as data purification.

▸ **Key words**: Deep Learning, CNN(Convolutional Neural Network), Drawing Image, Real Product Image, Siamese Network, 1:1 Verification

[요 약]

본 논문은 주어진 현품 영상과 도면 영상의 유사도를 비교하여 1:1 검증을 위한 방법을 제시한 것으로, CNN(Convolutional Neural Network) 기반의 딥러닝 모델을 두 개로 결합하여 Siamese Net을 구성하고 현품 영상과 도면 영상(정면도, 좌우 측면도, 평면도 등)을 같은 제품이면 1로 다른 제품이면 0으로 학습하며, 추론은 현품 영상과 도면 영상을 쌍으로 질의하여 해당 쌍이 같은 제품인지 아닌지를 판별하는 딥러닝 모델을 제안한다. 현품 영상과 도면 영상과의 유사도가 문턱 값(Threshold: 0.5) 이상이면 동일한 제품이고, 문턱 값 미만이면 다른 제품이라고 판별한다. 본 연구에서는 질의 쌍으로 동일제품의 현품 영상과 도면 영상이 주어졌을 때(긍정 : 긍정) “동일제품”으로 판별할 정확도는 약 71.8%로 나타났고, 질의 쌍으로 다른 현품 영상과 도면 영상이 주어졌을 때(긍정: 부정) “다른제품”으로 판별할 정확도는 약 83..1%를 나타내었다. 향후 제안한 모델에 파라미터 최적화 연구를 접목하고 데이터 정제 등의 과정을 추가하여 현품 영상과 도면 영상의 매칭 정확도를 높이는 연구를 진행할 예정이다.

▸ **주제어**: 심층학습, 합성인공신경망, 도면 영상, 현품 영상, 샴신경망, 1:1 검증

• First Author: Gi-Tae Han, Corresponding Author: Gi-Tae Han
*Gi-Tae Han (gthan@gachon.ac.kr), Dept. of Computer Engineering, Gachon University
• Received: 2022. 11. 21, Revised: 2022. 12. 05, Accepted: 2022. 12. 12.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

두 영상 간 유사성을 비교하기 위하여는 각 영상에서의 특징을 추출하여 서로 간 비교하는 연구로 최근에는 딥러닝 기술을 이용하여 컴퓨터가 스스로 영상의 특징을 학습하고 학습된 특징을 바탕으로 영상의 분류 및 패턴을 인식하는 합성곱 신경망 연구가 진행되어왔으며, VGGNet, ResNe(X)t, DenseNet, MobileNet, EfficientNet 등이 대표적이다[1-6]. 그러나, 직관적 임에도 불구하고, 유사성 비교를 통한 영상 매칭은 실세계 영상에서 일반화하기가 쉽지 않으며, 그 성능은 키 포인트 탐지기(Key Point Detector)와 지역 특징 기술자(the Local Feature Descriptor)의 질에 의존한다. 최근 이미지 유사성 비교 알고리즘으로 Siamese Net이 다양한 형태로 활용되고 있다. Siamese Net은 정확하게 동일한 파라미터(Parameter)와 커널 값(Weight)을 갖는 동일한 두 개의 신경망으로 이루어진다[7]. 이 두 개의 동일한 신경망에 두 개의 다른 입력 영상을 각각 넣고 계산하여 출력되는 값을 비교하는 형태로 두 영상의 유사도를 계산하며, 종종 출력 벡터 중의 하나는 미리 계산하여 비교 대상인 다른 벡터에 대한 베이스라인(Baseline)을 형성해 놓을 수 있다.

본 논문에서는 도면 영상과 현품 영상을 입력으로 받아 입력된 현품 영상과 도면 영상이 동일제품의 것인지 아닌지를 판별하는 방법을 제안한다. 제품 영상과 도면 영상을 매칭하기 위한 방법의 준비단계로는 우선 제품 영상과 도면 영상에 대하여 동일한 제품으로의 그룹과 동일하지 않은 다른 제품의 그룹으로 그룹핑한 후 두 개의 영상을 랜덤하게 선택하되 동일한 그룹에서 선택된 경우는 유사도를 1로 설정하고 서로 다른 제품의 그룹에서 선택된 경우에는 유사도를 0으로 설정하여 학습한다. 학습을 위한 망의 구성은 2개의 CNN 모델을 쌍으로 구성한 Siamese Net으로부터 각 CNN 모델의 뒷 단 FC Layer의 Feature Vector에 대한 Similarity를 계산하여 같은 제품의 현품 영상과 도면 영상일 경우는 유사도를 1로 주고, 다른 제품의 현품 영상과 도면 영상일 경우에는 유사도를 0으로 주어 학습을 진행한다. 모델의 학습에 사용하는 유사도 함수에는 Cosine Similarity를 사용하고, Loss Function으로는 BCEWithLogitsLoss를 사용한다. 제안한 모델로 유사도를 계산한 결과 유사도가 0.5 이상인 경우를 같은 제품으로 간주하고 0.5 미만인 경우를 다른 제품으로 판단하도록 했을 때 동일한 현품 영상과 도면 영상을 동일제품으로 판별하는 정확도는 71.8 %이고, 다른 현품 영상과 도면 영상의 쌍을 다른 제품으로 판별하는 정확도는 83.09%로 나타났다. 현품 영상과 도면 영상의 매칭은 크로스 도메인에서의 매칭으로 단순 CNN 모델(WideResNet)을 이용하여 어떤 현품 영상이 동일제품의 도면 영상의 클래스로 분류하는 분류 문제로 접근한 경우는 Top 1 정확도가 38.0%, Top 3 정확도가 53.4% 정도로 아주 낮게 나오는 것에 비하여 유사도를 학습하는 문제로 접근하면 정확도를 현실 세계에서 사용 가능한 정도로 끌어 올릴 수 있다. 이 연구는 향후 좀 더 학습에 활용할 데이터를 정제하고 GA(Genetic Algorithm) 혹은 HSA(Harmony Search Algorithm) 등을 활용한 파라미터의 최적화 같은 연구[8]를 접목할 경우 특정 기관에서 원제품과 불법 제품과의 유사도 등을 비교하여 판별 기준으로 삼는 업무 등에 활용할 가치가 클 것으로 보인다. 이후 논문의 구성은 2장에서 관련 연구로 Siamese Network과 Similarity 및 다양한 Loss Function에 대하여 살펴보고, 3장에서는 도면영상과 현품영상을 매칭하기 위한 학습모델과 추론모델을 제안하며, 4장에서는 제안한 모델의 학습과 테스트 방법 및 추론한 결과를 제시하고, 마지막 5장에서는 결론으로 제안한 방법에 대한 타당성을 보인다.

II. Preliminaries

1. Deep Learning-based Image Classification Model

딥러닝 기반 영상분류는 각 영상에 대하여 레이블을 붙여 해당 레이블로 인식하도록 학습하는 지도형 기계학습 방법 중의 하나이다. 딥러닝 인공 신경망 중 CNN은 학습이 진행되면서 영상의 특징을 표현하는 Weight와 Bias가 Back Propagation에 의하여 갱신되면서 영상의 특징을 잘 추출하는 모델로 완성된다.

영상분류 모델의 기본을 이루는 CNN 모델은 Fig. 1과 같이 Convolutional Layer를 거치면서 영상의 특징들을 추출하고 Fully Connected Layer와 SoftMax를 통하여 영상을 분류한다.

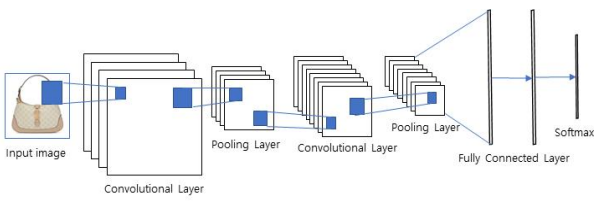


Fig. 1. CNN structure

영상은 직관적임에도 불구하고 두 영상 간 매칭에 사용하기 위하여 실 세계의 영상들을 일반화하는 것은 어려운 일이다. 그러나 이미지에서 특징을 추출하고 비교 대상 영상 간에 추출된 특징을 벡터화하여 두 벡터의 유사도를 구한다면 이미지를 매칭 하는 방법에 접근할 수 있다. 최근에는 두 개의 이미지로부터 특징 벡터를 추출할 수 있는 SiameseNet을 개선한 방법들이 등장하여 다양한 분야에 응용되고 있다.

2. Siamese Network Model

Siamese Net은 완전히 같은 파라미터와 웨이트를 갖는 동일한 CNN기반 망을 두 개로 구성한 신경망이다. Siamese Net은 분류해야 하는 영상의 클래스 종류가 매우 많고, 특정 클래스에 대한 충분한 데이터를 확보하기가 어렵거나 모델이 학습하지 못한 객체를 분류해야 하는 환경에서 사용하기 위하여 제안된 신경망이다[7]. 이러한 문제를 해결하기 위하여 One-shot Learning 혹은 Few-shot Learning이 등장했으며 이 방법은 다른 데이터로 학습된 모델을 이용하여 소규모나 한번 본 객체에 대해서도 그 객체가 어떤 클래스에 속하는지를 예측할 수 있도록 한다. 이것은 Fig. 2와 같이 두 개의 다른 입력 영상에 대하여 동일한 Weight를 적용하고 벡터를 계산하여 비교하게 되며, 입력으로 들어오는 두 영상이 같을 경우 유사도를 1로 부여하고 다를 경우 0을 부여하여 학습을 진행한다. 식(1)은 두 벡터 간 L1 벡터를 계산하는 식이다.

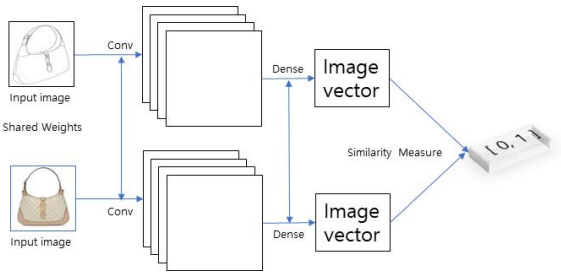
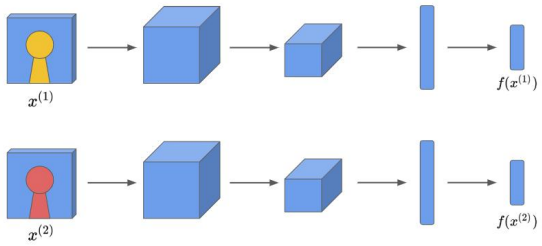


Fig. 2. Sample of Siamese Network

$$\sigma(\sum_j \alpha_j |\mathbf{h}_{1,L-1}^{(j)} - \mathbf{h}_{2,L-1}^{(j)}|) \quad (1)$$

이때 학습에 사용하는 Loss는 다음 식(2)과 같이 Cross-entropy를 사용한다.

$$\mathcal{L}(x_1^{(i)}, x_2^{(i)}) = \mathbf{y}(x_1^{(i)}, x_2^{(i)}) \log \mathbf{p}(x_1^{(i)}, x_2^{(i)}) + (1 - \mathbf{y}(x_1^{(i)}, x_2^{(i)})) \log (1 - \mathbf{p}(x_1^{(i)}, x_2^{(i)})) + \boldsymbol{\lambda}^T |\mathbf{w}|^2 \quad (2)$$



$$d(x^{(1)}, x^{(2)}) = ||f(x^{(1)}) - f(x^{(2)})||_2^2$$

이를 통해서 추출된 벡터 간의 거리는 서로 유사한 영상 간에는 높은 유사도를 가지고, 서로 다른 영상 간에는 낮은 유사도를 가지도록 학습이 진행되는데, 이와 같은 방법을 Similarity Learning 혹은 Metric Learning이라고 한다.

3. Cosine Similarity

비교를 위한 두 벡터를 A, B라 할 때 코사인을 사용한 유사도 함수는 아래 식(3)과 같다. 측정은 벡터 A와 B 사이의 코사인 각을 계산하여 값이 0에 가까우면 두 벡터가 서로 90도로 직교하는 것으로 유사하지 않은 것을 의미하며, 값이 1에 가까운 것은 매우 유사하다는 것을 의미한다[9-10].

Fig. 3은 유사도를 학습하는 모델을 나타내고 있다.

$$similarity = cos(\Theta) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} \quad (3)$$

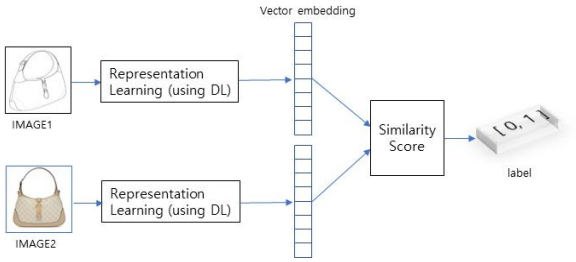


Fig. 3. Cosine Similarity Learning Model

4. Loss Function

(1) Binary Cross Entropy Loss

Fig. 3과 같은 모델의 프로세스에서 위 식(1)으로 두 인코딩 벡터의 L1 vector를 구하여 해당 Vector를 Hidden layer에 통과시킨 후, Output layer에서 Sigmoid(W*L1 vector + b)를 출력하게 하고, 두 영상이 다르면 0, 비슷하면 1이 나오도록 신경망을 학습하는데 위 식(2)의 Binary cross entropy의 Loss function을 사용한다.

(2) Contractive Loss

한 쌍의 데이터를 파라미터를 공유하는 두 개의 CNN에 전달하여 임베딩 $G(X_1)$, $G(X_2)$를 추출하고 식(4)와 같이 L2 Norm을 계산하여 임베딩 $G(X_1)$, $G(X_2)$ 간의 거리(Distance)를 구한다.

$$Dw(\overrightarrow{X_1}, \overrightarrow{X_2}) = \| G_W(\overrightarrow{X_1}) - G_W(\overrightarrow{X_2}) \|_2 \quad (4)$$

이것을 식(5)의 loss function에 활용한다.

$$L(W, (Y, \vec{X_1}, \vec{X_2})^i) = (1-Y)L_S(D_W^i) + YL_D(D_W^i) \quad (5)$$

식(5)에서 두 데이터가 가까운 거리에 있으면 Y=0이 되어 (1-Y)는 1이므로 전반부만 남고 후반부는 없어진다. 여기서 L_s는 동일 쌍 사이의 Loss, L_d는 다른 쌍 사이의 Loss를 나타내며 L_s와 L_d를 풀어쓰면 식(6)과 같다.

$$L(W, Y, \vec{X_1}, \vec{X_2}) = (1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m - D_W)\}^2 \quad (6)$$

식(6)에는 max()가 이용되어 m > Distance 경우와 m < Distance 경우로 두 가지로 나누어 접근할 수 있다[11]. 여기서 m은 margin을 의미하다.

첫 번째 경우는 Loss 값이 존재하여 Distance가 m만큼의 크기가 되도록 CNN의 파라미터 업데이트를 진행하고, 두 번째 경우는 max() 함수를 거치면 Loss가 0이므로 가중치의 업데이트가 없다. 여기서 m은 사람이 정해줘야 하는 값(하이퍼 파라미터)으로, 여러 번 시도 없이는 최적의 m을 정하기 어렵다. Distance가 m 값보다 작은 모든 경우에는 모든 임베딩 간 거리가 m으로 수렴하여 정보를 잃어버릴 수 있어 이를 보안한 것이 Triplet Loss이며, Fig. 4와 같다.

(3) Triplet Loss

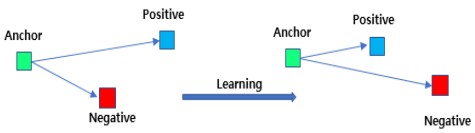


Fig. 4. Triplet Loss

학습을 통해서 두 개의 객체가 Positive(동일 혹은 유사) 하다면 객체 간의 거리는 줄이고, Negative(다른 객체) 하다면 객체 간 거리를 늘리는 방법인데, 기준이 되는 영상을 A(Anchor), 긍정 영상을 P(Positive), 부정 영상을 N(Negative)이라고 할 때

$$||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha \leq 0 \quad (7)$$

로 표현할 수 있고, 손실함수를 일반화하여 정리하면 아래 식 (8)로 나타낼 수 있다[12].

$$L(A, P, N) = max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha, 0) \quad (8)$$

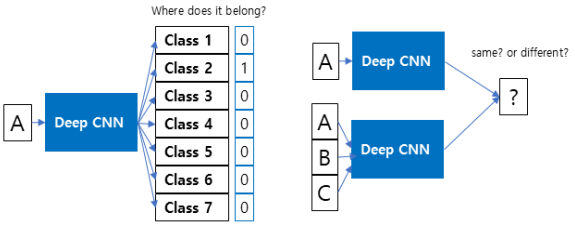
여기서 α는 A가 P와의 거리보다 N과의 거리가 다른 클래스라는 것을 보증할 만큼 충분하게 멀리 떨어졌다는 것을 나타내는 파라미터이다.

실제 배치 트레이닝 시, m을 배치 샘플의 수라고 할 때 Loss는 아래와 같이 계산된다.

$$J = \sum_{i=1}^{m} L(A^{(i)}, P^{(i)}, N^{(i)}) \quad (9)$$

Triplet Loss의 단점은 A, P, N에 사용되는 이미지를 random 하게 골랐을 때, Loss 가 너무 쉽게 0이 된다는 것입니다. 즉, d(A,N)은 d(A,P)보다 거의 항상 크기 때문에 d(A,P) - d(A,N) + a 은 항상 0보다 작아지게 되고 Loss는 0이 되게 되므로 훈련이 잘 안 된다. 그러므로 실제 훈련시 구분하기 어려운 세 개의 이미지가 비슷한, d(A,P) 와 d(A,N)의 차이가 크지 않은 이미지를 우선적으로 사용한다.

5. Difference between CNN and Siamese Net



(a) CNN (b) Siamese Net

Fig. 5. CNN and Siamese Net

CNN 모델은 CNN을 통과한 결과 어떤 클래스에 속하는가를 학습하는 모델이라면 Siamese Net은 두 객체가 같은가 다른가를 놓고 학습하는 모델이다.

III. The Proposed Method

본 논문에서는 주어진 현품 영상과 도면 영상을 1:1 매칭하여 현품 영상과 도면 영상의 유사도를 가지고 비교 쌍이 동일한 제품인지 아닌지를 판별하는 방법을 제안한다. 제안한 모델의 학습과 테스트에 사용할 데이터는 제품의 다양한 현품 영상과 제품에 해당하는 도면 영상(정면도, 좌우 측면도, 사시도)이 필요하며, 제품에 대한 현품 영상은 실제 촬영 영상이나 웹상에서 크롤링한 영상을 활용하고, 해당 도면 영상은 특허정보검색서비스(www.kipris.or.kr) 사이트에서 다운받은 도면 영상을 사용한다. 제안한 모델은 단순한 CNN 계열의 분류를 다루는 문제가 아니라 현품 영상과 도면 영상 간의 CROSS-DOMAIN에서의 영상 검색과 같은 문제로 볼 수 있다[13-15].

CROSS-DOMAIN에서 질의에 대한 타겟 결정 문제는 분류 문제가 아닌 해당 두 객체 영상에 대한 유사도 문제로 접근해야 하므로 제안한 방법에서는 모델을 CNN의 두 쌍으로 구성한 Siamese Net의 기본개념을 바탕으로 한쪽 CNN 망에는 현품 영상을 입력하고 다른 쪽 CNN 망에는 도면 영상을 입력하여 동일한 제품일 경우는 유사도를 1로 놓고 다른 제품일 경우는 유사도를 0으로 놓아 학습을 진행하며, 추론의 경우는 두 쌍의 도면 영상과 현품 영상이 제안한 모델을 통과한 후의 특성 Vector 간의 거리를 가지고 유사도를 계산하여 두 영상이 동일한 제품인지 다른 제품인지를 판별하는 시스템 개발을 위한 딥러닝 모델을 제안하다.

1. Proposed Learning Model

Fig. 6은 본 논문에서 제안한 방법의 학습모델을 도시한 것이다.

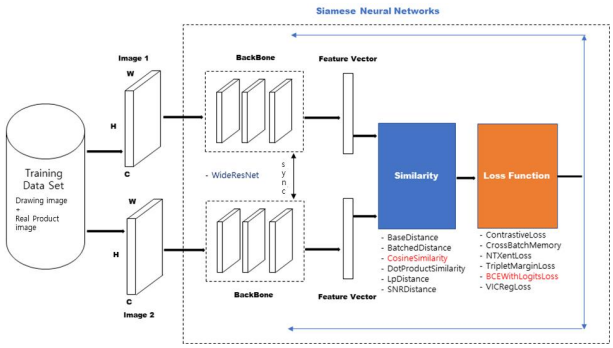


Fig. 6. The proposed learning model for detecting the similarity between the real product image and the drawing image

한 제품에 대한 현품 영상과 도면 영상을 쌍으로 받아서 유사도를 판별하기 위한 학습모델은 여러 층으로 구성된 기본 CNN을 한 쌍으로 구성하고, 각 CNN 망의 끝단의 FC(Fully Connected) Layer에서 특징 벡터를 추출한다. 각 CNN으로부터 추출된 특징 벡터들로부터 유사도를 계산하여 현품 영상과 도면 영상이 같은 제품이면 유사도를 1로 설정하고, 다른 제품이면 유사도를 0으로 설정하여 Cross Entropy Loss 함수를 통하여 학습을 진행한다.

2. Proposed Inference Model

Training Data Set으로 따로 분류된 현품 영상과 도면 영상을 사용하여 현품 영상과 도면 영상을 질의로 받아 두 개의 영상에 대한 유사도를 계산하고 유사도가 특정 문턱값(T: Threshold)을 기준으로 그 이상이면 동일제품으로 판별하고, 미만이면 다른 제품으로 판별하는 모델로 Fig. 7과 같이 구성한다.

추론 모델은 두 개의 질의 데이터를 각각 입력으로 받아 쌍으로 이루어진 CNN 모델과 CNN 모델의 끝단에서 각 모델의 특징 벡터를 추출하는 Layer 및 각각의 모델로부터 도출된 특징 벡터에 대한 Similarity를 계산하는 Layer까지는 제안한 학습 모델과 동일하게 구성하며, Similarity(S)와 문턱 값(T)의 비교를 통하여 동일제품인지 다른 제품 인지의 Result를 도출하는 단계만 다르게 구성한다.

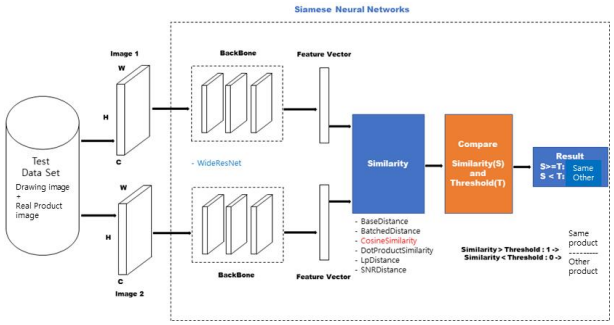


Fig. 7. Proposed similarity inference model between real product image and drawing image

3. Model Construction

제안한 모델은 제품의 현품 영상과 도면 영상을 1:1 매칭 하여 두 영상의 유사도를 가지고 현품 영상과 도면 영상이 동일한 제품인지를 판별하는 시스템에 사용하기 위한 인공지능 학습과 추론 모델이다. 제품에 대한 영상은 다양한 각 도로 촬영하여 획득하거나 획득한 영상을 Augmentation으로 다수의 영상을 확보할 수 있으나 도면 영상은 평면도, 정면도, 좌·우측면도, 사시도 등으로 제한이 되어 있어 학습에 참여시킬 데이터는 불과 몇 장(5장 내외)에 불과하다. 본 논문에서는 다양한 각 도나 회전 크

기 등에 관계없이 촬영된 현품 영상이나 웹 크롤링된 영상을 도면 영상과 유사도를 비교하여 매칭하는 모델로 일반 CNN의 모델의 Layer들로 구성된 모델을 쌍으로 붙이고 각 모델의 끝단 FC Layer를 특징 Latent Vector를 획득하는 Layer로 활용한다. 이후 학습모델에서는 두 모델의 끝단에서 획득한 특징 벡터의 유사도를 구하고 그 유사도 값을 Cross Entropy Function에 입력하여 Back Propagation을 통하여 학습을 진행한다. 제안한 학습모델과 추론모델은 Siamese Net 형식의 쌍으로 구성하기 때문에 두 모델의 형태가 동일하며, Weight의 갱신도 동기화되어 동일하게 이루어져야 한다. 세부적인 구성 요소들은 다음과 같다.

제안한 학습모델은 다음과 같이 4개의 컨볼루션 레이어는 동일하게 구성하고, FC(Fully Connected) Layer의 특징 벡터 간의 유사도를 본 논문에서는 Cosine Similarity 거리를 구하는 방법을 통하여 유사도를 도출하며, 학습모델에서는 Loss Function으로 BCE(Binary Cross Entropy) 방법을 사용하여 학습을 진행하고 추론 모델에서는 Cosine Similarity를 가지고 유사도를 판단하여 동일제품인지 다른 제품인지를 판별한다. 입력 영상으로부터 층의 구성은 다음 Table 1과 같다.

Table 1. Layer Configuration of Proposed Model

Input Image: 105*105	
Conv 64 10x10, Relu, Batch Norm	64×96×96
Max Pool	64×48×48
Conv 128 7x7, Relu, Batch Norm	128×42×42
Max Pool	128×21×21
Conv 128 4x4, Relu, Batch Norm	128×18×18
Max Pool	128×9×9
Conv 256 4x4, Relu, Batch Norm	256×6×6
Fully Connected	4096×1
Cosine Similarity	
1. Learn: Loss Function, Binary Cross Entropy 2. Inference: Compare Similarity and Threshold	

제안한 모델의 입력 영상은 도면 영상과 현품 연상 모두 105 × 105의 크기이며, 4개의 합성곱 Block의 각 Block에서는 Relu와 Batch Normalization을 적용한다.

FC에서 4,096의 특징 벡터를 추출하는 단계까지 두 개의 망이 한 쌍으로 구성되며, 양쪽 망의 각 출력에 대한 Similarity를 계산하여 Loss Function으로 학습을 진행한다. 추론은 도면 영상과 현품 영상을 쌍으로 구성한 두 개의 입력이 모델을 통과한 결과인 두 개의 특징 벡터로부터 도출한 Similarity와 Threshold(0.5 설정)를 비교하여 동일제품인지 다른 제품인지를 판별한다. Fig. 8은 제안한 방법의 Layer 구성도이다.

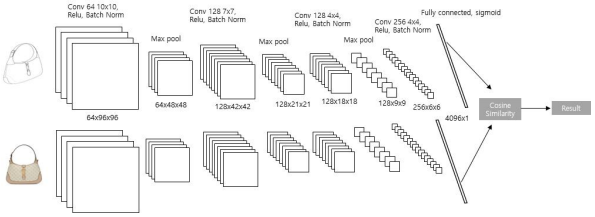


Fig. 8. Layer configuration diagram of the proposed model

IV. Experiment and Result

실험에 사용할 데이터로는 도면 영상은 KIPRIS(특허정보 서비스)에 게시된 도면 영상(정면도, 평면도, 좌우측면도, 사시도)을 다운받아 확보하였으며, 현품 영상은 도면 영상에 해당하는 제품의 사진을 직접 촬영 혹은 웹 스콜링하는 형태로 확보하였다. 도면 영상의 경우는 하나의 제품을 한 클래스로 보았을 때 한 클래스에 몇 장(4~5장)의 영상밖에 존재하지 않은 상황이다.

1. Preparation of Experimental Data

제안한 방법에서는 도면 영상의 제한사항을 고려하여 도면 영상은 각 제품별로 정면도, 평면도, 좌·우측면도를 각 1개씩 확보하고, 현품 영상은 각 제품의 영상에 대하여 좌우 이동, 상하 이동, 회전, 기울기, 확대 축소, 좌우 반전 등 Augmentation을 수행하여 20장씩을 확보하였다.

본 연구에서는 제품에 대한 현품을 구매하여 촬영해야 하는 어려움과 KIPRIS 시스템에 등록된 도면 영상과의 일치성 확인 등의 어려움을 감안하여 KIPRIS에 등록된 도면 영상 중에서 제품에 대한 현품 영상의 획득이 가능한 163개의 제품을 선정하여 학습을 진행하였다. 실험에 참여할 데이터의 구성은 전체 163개의 품목에 대하여 도면 영상은 각 제품당 4가지의 도면(정면도, 평면도, 좌·우측면도) 영상 652개와 현품 영상으로는 한 개의 제품당 20개씩 영상 3,260개를 확보하였다. Fig. 9는 현품 영상과 도면 영상의 샘플을 보이고 있다.

Fig. 9. Sample of the real product image and the drawing image

2. Organization of Training Data

각 현품 영상과 도면 영상으로부터 현품 영상을 기준으로 Training Data Set과 Test Data Set을 4 : 1의 비율로 나누어 실험을 진행한다.

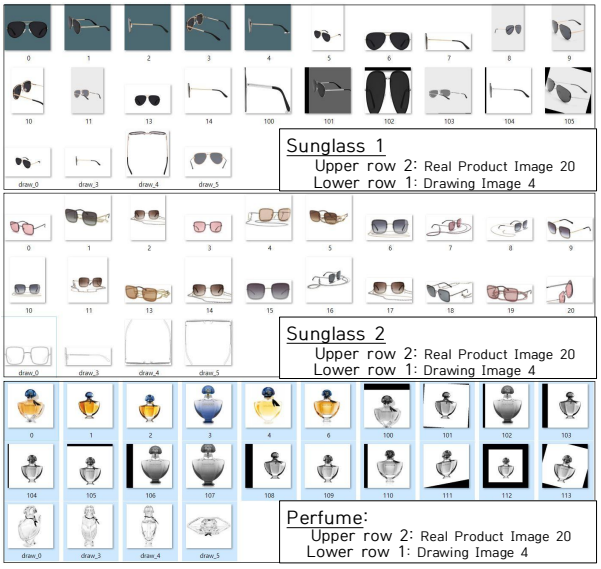


Fig. 10. Training DataSet by product (three samples)

Fig. 10은 학습을 위한 Training DataSet의 예를 보여주는 것이다. Fig. 10과 같은 형태의 163개의 Training DataSet에서 학습에 참여할 쌍을 임의(randomly)로 쌍을 뽑아 학습에 참여시킨다. 먼저 학습에 참여할 동일제품의 데이터 쌍의 경우의 수를 살펴보면 하나의 제품에 도면 영상 4개와 현품 영상인 20개인 24장으로 구성되어 있으므로 그중에서 2개를 선택하면 ${}_{24}C_2$로 되고 163개의 제품이므로 ${}_{24}C_2 \times 163$으로 44,988개이고, 다른 제품의 데이터 쌍의 경우의 수는 현재 기준으로 삼은 제품을 제외한 제품 수인 162개가 모두 24개의 영상을 가지므로 그중에서 2개를 뽑고 163개 제품이므로 ${}_{162\times24}C_2 \times 163$로 7,556,326개가 된다.

3. How to Learn

학습은 도면과 현품 영상을 선택하여 Fig. 10과 같이 쌍을 이루어 학습에 참여시키는데, 163개의 제품에 대하여 학습을 진행하되 현재 기준이 되는 제품의 도면 영상과 동일한 제품의 현품 영상을 뽑는 경우와 현재 기준이 되는 제품(도면 영상, 현품 영상)과 다른 제품(도면 영상, 현품 영상)의 영상을 뽑을 확률을 1:1 비율로 정하여 학습을 진행한다. 학습을 진행할 때에는 현품과 도면, 도면과 도면, 현품과 현품을 Fig. 10와 같은 Training Data Set으로부터 임의(randomly)로 Fig. 11과 같이 선택하여 동일한 제품일 때에는 유사도를 1로 다른 제품일 때에는 유사도를 0으로 놓고 학습을 진행한다.

(a) real:drawing (b) drawing:drawing (c) real:real

Fig. 11. Types of data participating in learning

학습에 사용한 GPU는 Geforce RTX 3090 24GB이며, 학습 Epochs는 200으로 설정하고 Batch size는 128로 설정하였다. Augmentation으로는 회전, 왜곡, 좌우 이동, 상하 이동, 확대, 축소, 좌우 플립을 사용하였다. 163개의 제품에 대하여 Positive 즉, 동일한 제품의 도면이나 현품 영상으로 쌍을 이루는 경우의 수와 Negative 즉, 다른 제품의 도면 영상이나 현품 영상으로 쌍을 이루는 경우의 수에서 학습 시 1 Epoch 마다 Positive와 Negative의 비율을 1:1로 설정하여 램덤으로 각 90,000쌍의 데이터를 뽑아 학습에 참여시켰고, 테스트 시에는 Positive와 Negative의 비율을 1:1로 설정하여 800쌍의 데이터를 뽑아 테스트하도록 하였다.

조기 수렴 및 오버피팅을 방지하기 위해 150 Epoch 이후부터 Test data set에 대해 Loss 값이 변하지 않으면 학습을 종료하도록 하였으며, 본 실험에서는 198 Epoch에서 제안한 모델이 수렴하였다.

4. Test Data Configuration

추론을 위한 데이터의 구성은 전체 데이터 중에서 현품 영상에 대하여 학습과 테스트의 비율을 4 : 1로 나누어 테스트에 참여할 데이터를 분리하고, 도면 영상은 Target이므로 학습과 테스트에 공동으로 이용하고 현품 영상은 학습에 참여하지 않은 데이터로만 구성한다. 테스트를 위한 질의 쌍의 구성 예는 Fig. 12와 같이 동일제품 안에서 도

면 영상과 현품 영상을 구성하여 동일제품인지를 판별하는 테스트 쌍의 구성과 Fig. 13과 같이 다른 제품의 도면 영상과 현품 영상의 쌍을 구성하여 동일제품이 아닌지를 판별하는 형태로 테스트 데이터를 구성한다.

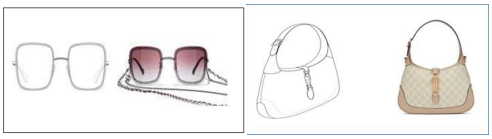


Fig. 12. Drawing image and real product image of the same product

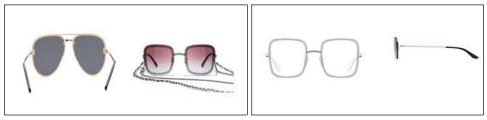


Fig. 13. Drawing image and real product image of other product

5. How to Test

현재 테스트의 기준이 되는 제품에서의 도면 영상을 A1, 현재 기준이 되는 제품과 동일한 제품에서의 현품 영상을 A2라 하면 동일제품의 판별을 위한 쌍은 A1 : A2로 구성할 수 있고, 현재 기준이 되는 제품에서의 도면 영상을 A라 하고 기준이 되는 제품과 다른 제품에서의 현품 영상을 B라 할 때,

(1) 동일제품 여부 판별(A1 : A2): Positive

163개 중 한 개의 동일제품 클래스로부터 뽑은 도면 영상과 현품 영상을 쌍으로 제안한 모델에 입력하여 Similarity를 계산하고 Similarity가 Threshold 값 이상이면 동일한 제품으로 판별하게 된다. 동일한 제품의 쌍을 동일한 제품으로 판별하면 TRUE이고, 다른 제품으로 판별하면 FALSE가 된다. 전체 추론 건수 중 TRUE가 되는 건수를 가지고 정확도를 계산한다.

(2) 다른 제품 여부 판별(A : B): Negative

163개의 제품 클래스로부터 각 도면 영상과 현품 영상을 각기 다른 클래스로부터 뽑아 쌍으로 구성하고, 제안한 모델에 입력하여 Similarity를 계산하고 Similarity가 Threshold 값 미만이면 다른 제품으로 판별하게 된다. 다른 제품의 쌍을 다른 제품으로 판별하면 TRUE이고, 동일한 제품으로 판별하면 FALSE가 된다. 전체 추론 건수 중 TRUE가 되는 건수를 가지고 정확도를 계산한다.

6. Results and Comparison of Approaches with Existing Models

본 논문에서 제안한 모델에 대하여 앞의 4.5절의 방법으로 테스트를 진행한 결과를 살펴보면, 제품이 도면과 매칭되는 정확도를 동일제품을 동일제품이라고 판별하는 정확도는 71.78%의 정확도를 나타내었고, 다른 제품을 다른 제품이라고 판별하는 정확도는 83.09%를 나타내었다. 표 2는 제안한 모델의 유사도 검출 정확도이다.

Table 2. Similarity accuracy of the proposed model (TV: Threshold Value)

Method \ TV	0.40	0.45	0.50	0.55	0.60	0.70
A1 : A2	76.69	75.46	71.78	69.35	66.87	53.99
A : B	77.57	80.47	83.09	85.58	87.82	92.31
Average	77.13	77.97	77.44	77.47	77.35	73.15

일반적으로 영상에서의 객체 분류는 지도 학습 모델인 CNN(Convolutional Neural Network) 기반의 ResNet50/152이나 ConvNext등의 다양한 모델들이 활용되고 있다. 본 논문에서는 제안한 모델에 학습과 테스트에 사용한 동일한 실험 데이터를 가지고 WideResNet과 ConvNext-tiny 모델을 사용한 객체 분류와 비교하여 제안한 방법이 유사도 검출에 의한 1:1 매칭 방법에 타당성이 있음을 확인한다. 주어진 환경에서 Top N 질의를 하려면 현품 영상을 질의 입력으로 주고 도면 영상이 출력으로 나와야 하는데, 이것은 마치 CROSS DOMAIN처럼 영역이 다른 데이터 간의 검색이 된다. 검증된 CNN 모델인 WideResNet을 통하여 동일한 실험 데이터를 분류했을 때 Top 1에서 38.04%, Top 3에서 53.37%, Top 5에서는 60.74%의 정확도가 관측되었고, 학습데이터를 상대적으로 많이 요구하는 최신 분류 모델인 ConNext-tiny에서는 학습 참여 데이터가 충분치 않아 학습이 제대로 이루어지지 않은 결과 Top 1이 3.68%, Top 3가 8.59%, Top 5는 12.89%의 낮은 정확도를 확인할 수 있었다.

Table 3. Classification accuracy of drawings and real product images by CNN

Method \ Top N	Top 1	Top 3	Top 5
Wide-ResNet	38.04	53.37	60.74
ConvNext-Tiny	3.68	8.59	12.89

V. Conclusions

본 본문에서는 도면 영상과 현품 영상의 1:1 매칭을 위한 딥러닝 학습모델을 제안한 것으로 Few-Shot Learning 형태로 몇 장의 영상으로도 학습을 진행하고 1:1 쌍으로 추론을 진행하여 질의로 들어온 영상이 도면 영상과 현품 영상이 동일제품 인지 다른 제품인지를 유사도 검출로 판별하는 모델을 제안하였다. 제안한 모델에 대하여 학습을 완료하고 테스트를 진행한 결과는 동일제품의 도면 영상과 현품 영상을 동일제품이라고 판별(A1:A2)하는 정확도는 71.78%, 다른 제품의 도면 영상과 현품 영상을 다른 제품이라고 판별(A:B)하는 정확도는 83.09%로 나타났다. A1:A2가 A:B보다 정확도가 낮게 나타난 것은 A1:A2의 경우에 학습에 참여하는 쌍의 수가 A:B의 경우 학습에 참여하는 쌍의 수보다 상대적으로 적게 존재하기 때문이다. 동일제품과 다른 제품의 비율을 1:1로 램덤으로 각각 90,000쌍의 데이터를 뽑아서 학습에 참여시키는데 제안한 실험 환경에서는 동일한 제품의 쌍이 44,988가지 경우만 존재한다. 그러므로 학습에 참여한 동일제품의 쌍은 2번 이상 중복을 허용한 상태이다. 제안한 방법에 대하여 학습에 활용할 데이터를 좀 더 정제하고 모델의 Layer 수, Kernel 크기, Kernel 개수 등 Hyper parameter의 최적화와 같은 연구를 진행한다면 불법 복제품의 판독과 같은 업무에 제안한 모델의 활용이 가능할 것이다.

ACKNOWLEDGEMENT

This work was carried out with the Research Funds Support of Gachon Univ. and Neowine Co., Ltd. in 2022.

REFERENCES

[1] Karen Simonyan, Andrew Zisserman, “Very Deep Convolutional Networks for large-scale image recognition'(VGGNet)”, Computer Vision and Pattern Recognition, cs.CV in Sep. 2014, ICLR 2015. arXiv:1409.1556

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Microsoft Research “Deep Residual Learning for Image Recognition,” Computer Vision and Pattern Recognition, Tech report pp770-778, 2015.

[3] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger, “Densely Connected Convolutional Networks”, Computer Vision and Pattern Recognition, cs.CV in Aug. 2016, CVPR 2017. arXiv:1608.06993

[4] Mingxing Tan 1 Quoc V. Le 1, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks” Machine Learning, cs.LG in May 2019, ICML 2019. arXiv:1905.11946

[5] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, Saining Xie, “A ConvNet for the 2020s,” Computer Vision and Pattern Recognition, cs.CV in Jan. 2022, CVPR 2022. arXiv:2201.03545

[6] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications” Computer Vision and Pattern Recognition, cs.CV in Apr. 2017, CVPR 2017. arXiv:1704.04861

[7] Gregory Koch, Richard Zemel, “Ruslan Salakhutdinov, ”Siamese Neural Networks for One-shot Image Recognition”, Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37

[8] Seong-Hoon Kim, Zong Woo Geem, Gi-Tae Han, “Hyperparameter optimization method based on harmony search algorithm to improve performance of 1D CNN human respiration pattern recognition system”, Sensors 2020, 20, 3697(Published: 1 July 2020). DOI:10.3390/s20133697

[9] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, Jianbin Jiao, “Image-Image Domain Adaptation with Preserved Self-Similarity and Domain-Dissimilarity for Person Re-identification”, Computer Vision and Pattern Recognition, cs.CV in Jul. 2017, CVPR 2018. arXiv:1711.07027

[10] Grigori Sidorov1, Alexander Gelbukh1, Helena Gomez-Adorno1, and David Pinto2, “Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model”, Computacionary Sistemas Vol. 18, No. 3, pp. 491-504, Sep. 2014.

[11] Raia Hadsell, Sumit Chopra, Yann LeCun, “Dimensionality Reduction by Learning an Invariant Mapping”, 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), Jun. 2006. DOI:10.1109/CVPR.2006.100

[12] Florian Schroff, Dmitry Kalenichenko, James Philbin, “FaceNet: A Unified Embedding for Face Recognition and Clustering”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 2015. DOI:10.1109/CVPR.2015.7298682

[13] Conghui Hu, Gim Hee Lee, “Feature Representation Learning for Unsupervised Cross-domain Image Retrieval”, Computer Vision and Pattern Recognition, cs.CV in Jul. 2022. ECCV 2022. arXiv:2207.09721

[14] Qingjie Meng, Daniel Rueckert, Bernhard Kainz “Unsupervised Cross-domain Image Classification by Distance Metric Guided

Feature Alignment", Machine Learning, cs.LG in Aug. 2020, arXiv:2008.08433

[15] Xiaoping Zhou, Xiangyu Han, Haoran Li, Jia Wang, Xun Liang, "Cross-domain image retrieval: methods and applications", International Journal of Multimedia Information Retrieval, Vol 11, pp 199-218, Jul. 2022.

Author

Gi-Tae Han received the B.S. in Computational Statistics from Chungnam University, Korea, in 1882 and received the M.S. and Ph.D. degrees in Computer Science and Electronic Engineering from Hanyang University, Korea, in 1995 and 2001, respectively. Dr. Han joined the faculty of the Department of Computer Science at Kyungwon University, Seongnam, Korea, in 1992. He is currently a Professor in the Department of Computer Engineering, Gachon University. He is interested in computer vision, artificial intelligence and deep learning, and image retrieval system.