

합성곱 신경망에서 이미지 분류를 위한 하이퍼파라미터 최적화

이재은¹, 김영봉¹, 김종남^{1*}

¹부경대학교 IT융합응용공학과

Hyperparameter Optimization for Image Classification in Convolutional Neural Network

Jae-Eun Lee¹, Young-Bong Kim¹, Jong-Nam Kim^{1*}

¹Dept. of Convergence & Applications Engineering, Pukyong National University

요약 합성곱 신경망 모형에서 높은 정확도를 얻기 위해서는 최적의 하이퍼파라미터를 설정하는 작업이 필요하다. 하지만 높은 성능을 낼 수 있는 하이퍼파라미터 값이 정확히 알려진 바가 없으며, 자료마다 최적의 하이퍼파라미터 값이 달라질 수 있기 때문에 매번 실험을 통해서 찾아야만 한다. 또한, 하이퍼파라미터 값들의 범위가 넓고 조합 수가 많기 때문에 시간과 계산량을 줄이기 위해서는 최적값을 찾기 위한 실험 계획을 먼저 한 후에 탐색을 하는 것이 필요하다. 그러나 아직까지 합성곱 신경망 모형에서 하이퍼파라미터 최적화를 위하여 실험계획법을 이용한 연구 결과가 보고되지 않았다. 본 논문에서는 이미지 분류 문제에서 통계방법 중 하나인 실험계획법의 요인배치법을 이용하여 실험 계획을 하고 합성곱 신경망 분석을 한 후에, 높은 성능을 갖는 값을 중심으로 그리드 탐색을 하여 최적의 하이퍼파라미터를 찾는 방법을 제안한다. 실험 계획을 통하여 각 하이퍼파라미터들의 탐색 범위를 줄인 후에 그리드 탐색을 함으로써 효율적으로 연산량을 줄이고 정확도를 높힐 수 있음을 보였다. 또한 실험 결과에서 모형 성능에 가장 큰 영향을 주는 하이퍼파라미터가 학습률이라는 것을 확인할 수 있었다.

• 주제어 : 그리드 탐색, 실험계획법, 하이퍼파라미터 최적화, 합성곱 신경망

Abstract In order to obtain high accuracy with an convolutional neural network(CNN), it is necessary to set the optimal hyperparameters. However, the exact value of the hyperparameter that can make high performance is not known, and the optimal hyperparameter value is different based on the type of the dataset, therefore, it is necessary to find it through various experiments. In addition, since the range of hyperparameter values is wide and the number of combinations is large, it is necessary to find the optimal values of the hyperparameters after the experimental design in order to save time and computational costs. In this paper, we suggest an algorithm that use the design of experiments and grid search algorithm to determine the optimal hyperparameters for a classification problem. This algorithm determines the optima values of the hyperparameters that yields high performance using the factorial design of experiments. It is shown that the amount of computational time can be efficiently reduced and the accuracy can be improved by performing a grid search after reducing the search range of each hyperparameter through the experimental design. Moreover, Based on the experimental results, it was shown that the learning rate is the only hyperparameter that has the greatest effect on the performance of the model.

• Key Words : Grid search, Design of experiment, Hyperparameter optimization, Convolutional neural network(CNN)

Received 12 August 2020, Revised 29 August 2020, Accepted 22 September 2020

* Corresponding Author Jong-Nam Kim, Dept. of IT Convergence & Applications Engineering, Pukyong National University, 45, Yongso-ro, Nam-gu, Busan, Korea. E-mail: jongnam@pknu.ac.kr

I. 서론

오늘날 자동차, 금융, 헬스케어 등 다양한 분야에서 기계학습(ML: machine learning) 알고리즘이 활용되고 있다. 이 중 합성곱 신경망(CNN: convolutional neural network) 알고리즘은 주로 이미지, 영상, 자연어처리 분야에서 뛰어난 성능을 보이고 있다. 합성곱 신경망을 포함한 기계학습에서는 높은 정확도를 얻기 위하여 최적의 하이퍼파라미터(hyperparameter) 값을 사용해야 한다. 하이퍼파라미터는 사람이 직접 입력을 해야 하는 값인데 자료마다 최적값이 다르고, 아직까지 어떤 하이퍼파라미터를 사용해야 하는지 이론적으로 밝혀진 바가 없다. 이러한 문제를 해결하기 위하여 다양한 연구가 진행되어왔다. 대표적인 하이퍼파라미터 최적화 방법으로는 매뉴얼 탐색(Manual Search), 그리드 탐색(Grid Search), 랜덤 탐색(Random Search), 베이지안 최적화(Bayesian Optimization), Tree Parzen Estimator (TPE)[1-3] 등이 있다. 그 외에 기존의 방법들을 조합하여 응용한 알고리즘들[4-5] 등이 있으며, 합성곱 신경망의 하이퍼파라미터를 최적화하기 위한 방법[6-9] 등이 연구되고 있다.

본 연구에서 제안하는 방법은 실험계획법 중 요인 배치법(factorial design)을 이용하여 선별한 각 하이퍼파라미터의 탐색 범위에 대하여 그리드 탐색을 통해 하이퍼파라미터 최적화를 수행한다. 이는 광범위한 탐색 범위를 선별함으로써 불필요한 계산량과 시간을 줄이며, 높은 성능을 내는 값들 주변을 탐색하여 정확도를 높이게 된다.

본 논문의 구성은 다음과 같다. 제II장에서 합성곱 신경망과 기존의 하이퍼파라미터 최적화 방법들을 소개한다. 제III장에서는 본 논문에서 제안하는 알고리즘에 대하여 기술하고, 제IV장에서는 제안한 알고리즘의 실험 결과를 확인하고 분석한다. 마지막으로 제V장에서는 본 논문에 대한 결론을 맺는다.

II. 관련 연구

합성곱 신경망 모형의 성능을 높이기 위해서는 하이퍼파라미터 최적화 작업이 필요하다. 먼저 학습 자료를 이용하여 오차 함수 $f(x)$ 가 최소가 되는 하이퍼파라미터들의 θ^* 값을 구한다. 그리고 이 값들을 이용하여 검증 자료에서 반복 작업을 통해 오차함수가 최

소가 되는 최적의 하이퍼파라미터 x^* 값들을 구한다. 하이퍼파라미터 최적화의 모형식은 다음과 같다.

$$\begin{aligned} x^* &= \min_{x \in \mathfrak{N}} f(x, \theta^*; S_{validation}), \\ \text{s.t. } \theta^* &= \arg \min_{\theta} f(x, \theta; S_{train}) \end{aligned} \quad (1)$$

여기서 $S_{validation}$ 과 S_{train} 은 각각 검증 자료와 학습 자료를 의미한다.

대표적인 하이퍼파라미터 최적화 방법으로는 이전 실험을 통해 얻은 경험과 지식을 활용하여 하이퍼파라미터 탐색 범위를 정한 후 매번 값을 변경하면서 최적 값을 찾는 매뉴얼 탐색(Manual Search), 경험에서 얻은 지식을 활용하여 하이퍼파라미터 범위를 설정한 후 일정한 간격으로 모든 조합을 실험하는 그리드 탐색(Grid Search), 범위 내 모든 조합 대신 난수 발생기를 이용하여 임의로 값을 선정하는 랜덤 탐색(Random Search)[1]이 있다. 하지만 매뉴얼 탐색과 그리드 탐색은 모든 공간을 탐색하기 때문에 많은 시간과 계산량이 요구되어 비효율적이고, 랜덤 탐색은 높은 정확도를 보장하지 못한다. 베이지안 최적화는 목적함수에 확률적 추정을 수행하는 Surrogate 모델과 최적값의 후보를 추천하는 Acquisition 함수로 구성된 방법이다[2]. 위의 방법들에 비해 베이지안 최적화는 탐색 속도가 향상되고 효율적으로 탐색을 하지만 탐색 과정이 복잡하여 이를 보완하기 위한 연구들이 계속 진행되고 있다[4-5]. 또한, 베이지안 최적화의 surrogate 모델 중 Tree Parzen Estimator(TPE)를 이용하여 연속형 뿐만 아니라 범주형 하이퍼파라미터도 사용할 수 있는 방법이 발표되었으며[3], 그 외 합성곱 신경망 하이퍼파라미터를 최적화하기 위한 다양한 방법들이 연구되고 있다[6-9]. 최근에는 랜덤포레스트(Random Forest) 모형의 하이퍼파라미터를 최적화하기 위하여 요인배치법과 반응표면분석을 조합한 방법이 제안되었다[10]. 탐색 범위를 선별 할 수 있는 장점이 있지만 반응표면분석을 이용한 탐색 과정이 쉽지 않기 때문에 이러한 어려움을 해결하기 위한 연구들이 진행되고 있다.

III. 실험방법

자료에 따라 높은 정확도를 나타낼 수 있는 하이퍼

파라미터의 값이 다르며, 각 하이퍼파라미터의 특성에 따라 탐색 범위가 넓은 경우도 있어서 적절한 하이퍼파라미터 값을 찾는 데 많은 시간과 계산량이 발생한다. 본 논문에서는 실험계획법을 이용하여 줄어든 하이퍼파라미터의 탐색 범위 안에서 반응표면분석 대신 그리드 탐색을 이용하여 조밀하게 최적값을 분석함으로써 효율적으로 계산량을 줄이고 정확도를 높이며 좀 더 편리하게 탐색할 수 있는 방법을 제안한다. 실험계획법은 제품의 특성에 영향을 미치는 인자들과 이들의 관계를 찾아보기 위한 실험을 실시하여 최적의 조건을 찾는 통계적 기법이다. 실험계획법들 중에서 요인배치법은 모든 인자들의 주효과와 교호작용을 검정할 수 있으며 실험에 영향을 크게 미치는 변동요인을 찾는 데 효과적인 방법으로 알려져 있다. 요인배치법은 인자의 수가 n 이고, 각 인자의 수준 수가 k 일 때 k^n 요인배치법이라 한다. 본 논문에서 제안한 알고리즘에서는 요인배치법 중 각 수준이 2개적인 2^n 요인배치법을 사용하였으며 여기서 각 수준은 하이퍼파라미터 탐색 범위의 하한과 상한, n 은 하이퍼파라미터 개수를 나타낸다. 하이퍼파라미터가 A, B, C 3개이고, 각 하이퍼파라미터의 수준이 2개이며 반복수가 r 일 때 2^3 요인배치법에 대한 모형식은 아래와 같다.

$$y_{ijkm} = \mu + a_i + b_j + c_k + (ab)_{ij} + (ac)_{ik} + (bc)_{jk} + (abc)_{ijk} + \epsilon_{ijkm},$$

where $i, j, k = 0, 1, m = 1, 2, \dots, r$

위 식에서 a_i 는 하이퍼파라미터 A 의 i 수준에서의 효과, $(ab)_{ij}$ 는 하이퍼파라미터 B, C 의 교호작용, $(abc)_{ijk}$ 는 하이퍼파라미터 A, B, C 의 교호작용을 나타낸다. 여기서 ϵ_{ijkm} 는 실험의 오차에 해당하는 확률 변수로서 평균 0, 분산 σ_E^2 인 정규분포 $N(0, \sigma_E^2)$ 를 따른다. 이 모형은 아래와 같이 회귀모형 식으로도 표현될 수 있다.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{12} x_1 x_2 + \beta_{13} x_1 x_3 + \beta_{23} x_2 x_3 + \beta_{123} x_1 x_2 x_3 + \epsilon$$

여기서 $\beta_0, \beta_1, \beta_2, \beta_3$ 은 각 하이퍼파라미터의 효과,

$\beta_{12}, \beta_{13}, \beta_{23}, \beta_{123}$ 은 하이퍼파라미터의 교호작용을 의미한다.

하이퍼파라미터를 최적화 할 때 가장 기본적으로 사용되는 방법 중 하나가 그리드 탐색이다. 그리드 탐색은 경험을 통하여 얻은 지식을 이용하여 하이퍼파라미터의 탐색 범위를 설정한 뒤, 일정한 간격으로 값을 변경하면서 모든 조합을 실험하고 최적의 하이퍼파라미터 값을 찾는 방법이다. 비록 탐색해야 할 범위가 넓거나 하이퍼파라미터의 수가 많을 경우 시간과 계산량이 기하급수적으로 증가하는 문제점이 발생하지만 모든 조합을 실험할 수 있다는 큰 장점 때문에 아직까지도 많은 연구자들이 사용하고 있다.

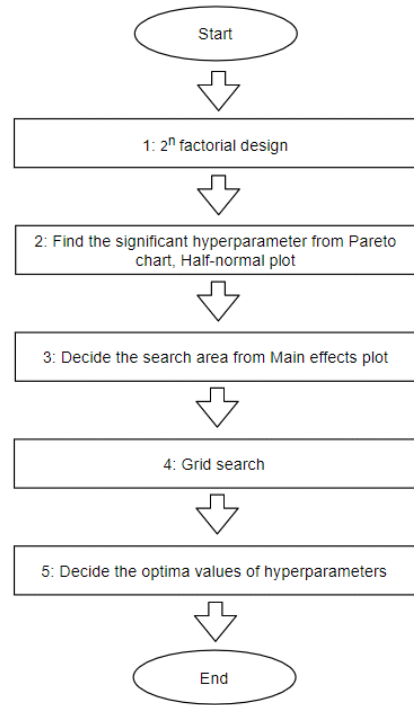


Fig. 1. Procedure of the proposed algorithm

본 논문에서는 위의 두 방법인 2^n 요인배치법과 그리드 탐색을 조합한 하이퍼파라미터 최적화 알고리즘을 제안한다. 먼저 각 하이퍼파라미터의 탐색 범위를 정한 후, n 개의 하이퍼파라미터와 2개의 수준수를 고려한 2^n 요인배치법을 이용하여 실험한다. 이때 각 하이퍼파라미터의 낮은 수준과 높은 수준은 탐색 범위의 하한값과 상한값으로 지정하여 실험한다. 실험 결과를

바탕으로 파레토 그림, 효과의 반정규 확률 그림을 평가하여 통계적으로 유의한 하이퍼파라미터가 있는지 확인 후, 주효과 그림에서는 두 수준 중에서 정확도가 높은 수준을 확인한다. 주효과 그림으로부터 높은 정확도를 갖는 하이퍼파라미터들의 수준 조합을 찾은 후, 선별된 수준 조합의 탐색 범위를 재지정하여 그리드 탐색을 시행한다. 그리드 탐색의 결과에서 가장 높은 정확도를 갖는 조합을 찾아 최적의 하이퍼파라미터 값으로 결정한다. 그림 1은 위에서 설명한 제안 알고리즘의 절차를 나타내고 있다.

IV. 실험 결과 및 고찰

본 연구에서 제안한 알고리즘의 성능을 평가하기 위하여 합성곱 신경망 모형의 최적화에 많이 고려되어 왔던 하이퍼파라미터들 중 합성곱 계층(convolutional layer)의 수, 완전연결 계층(fully-connected layer)의 수, 그리고 학습률(learning rate)을 이용하여 이미지 분류 실험을 수행하였다. Kaggle의 ‘Dogs vs Cats’ 자료를 사용하였으며, 이미지 중 개와 고양이가 같이 있는 사진, 개 또는 고양이의 크기가 매우 작은 사진 등을 제외한 총 10,000개의 이미지를 이용하여 분석하였다. 이때 입력 자료의 너비와 높이는 각각 256으로 적용하였다. 합성곱 신경망 모델에서 필터크기 (3,3), 필터 수 64, 활성화 함수 Relu, 배치크기 64, 반복횟수(epoch) 20으로 정의하였으며, 과적합(overfitting)을 방지하기 위하여 드롭아웃을 0.2로 정의하여 분석하였다.

먼저 세 개의 하이퍼파라미터에 대하여 2^3 요인배치법을 이용하여 표 1과 같이 실험 설계를 한 뒤 합성곱 신경망 분석을 하였다. 총 8가지 조합에 대하여 실험을 하였으며, 분석 결과 중 합성곱 계층 수 6, 완전연결 계층 수 3, 학습률 0.001일 때 정확도가 90.8%로 가장 높게 나왔고, 그 다음으로 합성곱 계층 수 6, 완전연결 계층 수 1, 학습률 0.001일 때 88.8%의 정확도를 보였다. 또한 회귀모형 설명력은 $R^2 = 0.992$ 이며 표 2에서와 같이 합성곱 계층 수의 효과는 0.0333, 완전연결 계층 수의 효과는 -0.0287, 학습률의 효과는 -0.1448로 나타났다. 이는 합성곱 계층 수는 많을수록, 완전연결 계층 수는 적을수록, 그리고 학습률은 낮을수록 모형성능을 향상시킨다고 해석할 수 있다. 합성곱 계층 수와 완전연결 계층 수의 p-값은 각각 0.246, 0.280 이므로 유의수준 5%에서 통계적으로 유의하지

않은 반면 학습률의 p-값은 0.059로서 5% 근방의 값이며, 유의수준 1%에서는 유의하다고 할 수 있기 때문에 모형 성능에 영향을 줄 수 있는 하이퍼파라미터라고 판단된다. 두 하이퍼파라미터가 함께 움직여서 성능에 변화가 나타날 경우 교호작용이 발생하게 되는데 본 연구의 실험에서는 교호작용이 발생하지 않았다. 그림 2와 그림 3은 각각 파레토 그림과 표준화된 효과의 반정규 확률 그림이며 위의 결과와 같이 학습률이 모형 성능에 가장 큰 영향을 주는 하이퍼파라미터라고 나타났다. 그림 4에서, 주효과도에서는 합성곱 계층 수가 높은 수준일 때, 완전연결 계층 수와 학습률은 낮은 수준일 때 높은 정확도를 나타내었다. 이 결과를 바탕으로 합성곱 계층 수 [6, 7], 완전연결 계층 [1, 2], 학습률 [0.0001, 0.001, 0.01]로 범위를 재설정 후 그리드 탐색을 하였다.

Table 1. Levels for the 2^3 full factorial design

하이퍼파라미터	낮은 수준 (-)	높은 수준 (+)
합성곱 계층 수	3	6
완전연결 계층 수	1	3
학습률	0.001	0.1

Table 2. Results for the 2^3 full factorial design (* $p < 0.1$)

하이퍼파라미터	계수	표준오차	t-값	$P(> t)$
(절편)	0.7063	0.014	52.122	0.012
합성곱 계층 수	0.0333	0.014	2.461	0.246
완전연결 계층 수	-0.0287	0.014	-2.122	0.280
학습률	-0.1448	0.014	-10.690	0.059 *
합성곱 계층 수 : 완전연결 계층 수	-0.0063	0.014	-0.461	0.725
합성곱 계층 수 : 학습률	-0.0136	0.014	-1.000	0.500
완전연결 계층 수 : 학습률	-0.0314	0.014	-2.321	0.259

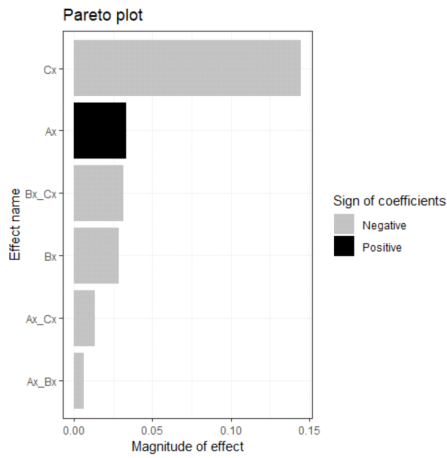


Fig. 2. Pareto plot of the hyperparameters

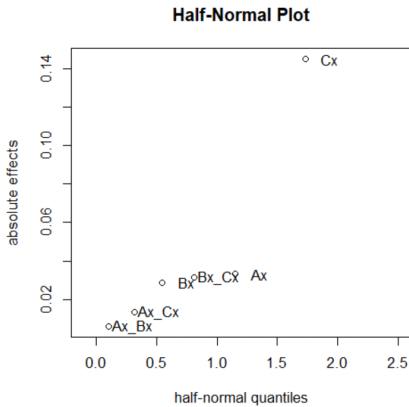


Fig. 3. Half-Normal plot of the hyperparameters

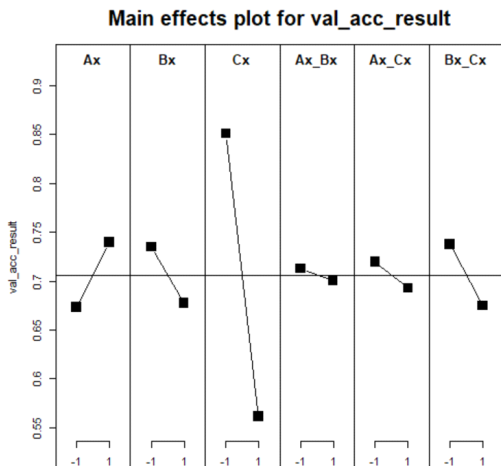


Fig. 4. Main effects plot of the hyperparameters

표 3은 그리드 탐색 결과를 나타낸 것이다. 분석 결과 합성곱 계층 수 7, 완전연결 계층 수 1, 학습률 0.001일 때 정확도가 91.42%로 가장 높게 나타났다. 특히 학습률 값에 따라 정확도의 차이가 크게 나타났는데, 학습률이 0.001일 때 정확도가 90% 이상, 0.01일 때 82~26%, 0.0001일 때 70.5% 이하의 정확도를 보였다. 이는 요인배치법에서 학습률이 모형 성능에 영향을 준다는 결과와 일치하는 것을 보여준다. 학습률은 가중치를 얼마만큼 업데이트 할 것인지 결정하는 값인데 이 때 값이 작을 경우 조밀하게 탐색을 하기 때문에 최적값을 찾아가는데 오랜 시간이 소요된다. 따라서 학습률은 작을수록 높은 정확도를 보이지만 너무 작을 경우 오히려 정확도를 떨어뜨릴 수 있다는 것을 그리드 탐색을 통해서 확인할 수 있다.

Table 3. Accuracy of grid search

하이퍼파라미터		학습률		
		0.0001	0.001	0.01
합성곱 계층 수-6	완전연결 계층 수-1	70.50	91.18	85.78
	완전연결 계층 수-2	64.25	90.37	82.4
합성곱 계층 수-7	완전연결 계층 수-1	61.98	91.42	86.05
	완전연결 계층 수-2	53.9	91.41	83.26

V. 결론

본 논문에서는 이미지 분류 문제에서 합성곱 신경망의 하이퍼파라미터 최적값을 찾기 위한 알고리즘을 제안하였다. 본 논문의 제안방법은 실험계획법을 이용하여 하이퍼파라미터 탐색 범위를 좁혀 불필요한 계산량과 시간을 줄였으며, 앞의 실험 결과를 바탕으로 각 하이퍼파라미터들의 높은 성능을 갖는 값들에 대하여 그리드 탐색을 함으로써 모형 성능의 정확도를 향상시켰다. 또한 2ⁿ 요인배치법을 통하여 모형 성능에 크게 영향을 미치는 하이퍼파라미터를 찾을 수 있었고, 교호작용이 있는지도 확인할 수 있었다.

ACKNOWLEDGMENTS

본 연구는 산업통상자원부와 한국산업기술진흥원의 “지역혁신 클러스터 육성사업(R&D P0004797)” 으로 수행된 연구결과임.

REFERENCES

[1] J. Bergstra, Y. Bengio, “Random search for hyper-parameter optimization,” The Journal of Machine Learning Research, vol. 13, no. 1, pp. 281-305, 2012.

[2] J. Snoek, H. Larochelle, R. P. Adams, “Practical bayesian optimization of machine learning algorithms,” Proceeding of Neural Information Processing Systems, pp. 2951-2959, 2012.

[3] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, “Algorithms for hyper-parameter optimization,” Proceeding of Advances in Neural Information Processing Systems, vol. 63, no. 3, pp. 2546-2554, 2011.

[4] A. Klein, S. Falkner, S. Bartels, P. Hennig, F. Hutter, “Fast bayesian optimization of machine learning hyperparameters on large datasets,” Proceeding of International Conference on Artificial Intelligence and Statistics, pp. 528-536, 2017.

[5] S. Falkner, A. Klein, F. Hutter, “BOHB: Robust and efficient hyperparameter optimization at scale,” Proceeding of International Conference on Machine Learning, pp. 1437-1446, 2018.

[6] T. Hinz, N. Navarro-Guerrero, S. Magg, S. Wermter, “Speeding up the hyperparameter optimization of deep convolutional neural networks,” International Journal of Computational Intelligence and Applications, vol. 17, no. 2, 2018.

[7] S. Albelwi, A. Mahmood, “A framework for designing the architectures of deep convolutional neural networks,” Entropy, vol. 19, no. 6, 2017.

[8] S. S. Talathi, “Hyper-parameter optimization of deep convolutional networks for object recognition,” Proceeding of IEEE International Conference on Image Processing, pp. 3982-3986, 2015.

[9] R. Andonie, A. C. Florea, “Weighted random search for CNN hyperparameter optimization,” International Journal

of Computers Communications & Control, vol. 15, no. 2, pp. 432-445, 2020.

[10] G. A. Lujan-Moreno, P. R. Howard, O. G. Rojas, D. C. Montgomery, “Design of experiments and response surface methodology to tune machine learning hyperparameters, with a random forest case-study,” Expert Systems with Applications, vol. 109, pp. 195-205, 2018.

저자 소개

이 재 은 (Jae-Eun Lee)



2013년 2월 : 부산대학교
통계학과(이학사)
2015년 2월 : 부산대학교
통계학과(이학석사)
2020년 2월 : 부경대학교
통계학과(이학박사)
2020년 7월~현재 : 부경대학교

인공지능연구소 연구원

관심 분야 : 머신러닝, 하이퍼파라미터 최적화

김 영 봉 (Young-Bong Kim)



1987년 2월 : 서울대학교
계산통계학과(이학사)
1989년 2월 : 한국과학기술원
전산학과(공학석사)
1994년 8월 : 한국과학기술원
전산학과(공학박사)
1994년~1995년 :

삼성전자정보기술연구소 선임연구원

관심 분야 : 컴퓨터 그래픽스, 3D 프린터, VR/AR

김 종 남 (Jong-Nam Kim)



1997년 2월 : 광주과학기술원
정보통신공학과(공학석사)
2001년 8월 : 광주과학기술원
기전공학과(공학박사)
2001년 8월~2004년 2월 : KBS
연구원
2004년 3월~현재 : 부경대학교

IT융합응용공학과 교수

관심 분야 : 비디오압축, 영상처리, 컴퓨터비전