

# 움직임 인식응용을 위한 커널 밀도 추정 기반 학습용 데이터 증폭 기법<sup>+</sup>

(Data Augmentation using a Kernel Density Estimation for  
Motion Recognition Applications)

정우순<sup>1)</sup>, 이형규<sup>2)\*</sup>  
(Woosoon Jung and Hyung Gyu Lee)

**요약** 머신러닝(ML, Machine Learning)기반 응용에서의 인식성능은 적용된 모델의 종류와 크기, 학습환경 및 학습에 사용되는 데이터 등 다양한 요인에 따라 결정된다. 특히 학습에 사용되는 데이터가 충분치 않을 경우 인식성능이 저하되거나 과적합(Overfitting)등의 문제가 발생하기도 한다. 이미지 인식을 주요 대상으로 하는 기존 연구들은 학습을 위한 데이터셋이 풍부하고 검증된 데이터셋을 사용하여 학습 및 인식성능을 평가할 수 있다. 하지만 사용된 센서, 인식의 대상, 인식 상황이 다른 특정 응용들의 경우 데이터셋을 직접 구축해야 한다. 이런 경우, ML모델의 성능은 데이터의 양과 품질에 따라 달라진다. 본 논문에서는 이용 가능한 학습용 데이터가 충분치 않은 움직임 인식응용에 효율적으로 사용될 수 있는 비모수 추정 방식의 일종인 커널 밀도 추정 알고리즘을 사용하여 학습용 데이터를 증폭한 후, 사용된 커널의 종류에 따라, 원본 데이터의 수 및 증폭 비율에 따라 증폭된 데이터가 원본 데이터의 특징을 잘 반영하는지 인식 정확도 변화를 토대로 비교 분석한다. 실험결과, 본 연구에서 사용한 움직임 인식응용에서는 좁은 대역폭을 가진 Tophat 커널로 증폭된 데이터셋에서 최대 14.31%의 인식 정확도 향상을 확인하였다.

**핵심주제어:** 기계학습, 커널 밀도 추정, 데이터 증폭

**Abstract** In general, the performance of ML(Machine Learning) application is determined by various factors such as the type of ML model, the size of model (number of parameters), hyperparameters setting during the training, and training data. In particular, the recognition accuracy of ML may be deteriorated or experienced overfitting problem if the amount of data used for training is insufficient. Existing studies focusing on image recognition have widely used open datasets for training and evaluating the proposed ML models. However, for specific applications where the sensor used, the target of recognition, and the recognition situation are different, it is necessary to build the dataset manually. In this case, the performance of ML largely depends on the quantity and quality of the data. In this paper, training data used for motion recognition application is augmented using the kernel density estimation algorithm which is a type of non-parametric estimation method. We then compare and analyze the recognition accuracy of a ML application by varying the number of original data, kernel types and augmentation rate used for data augmentation. Finally experimental results show that the recognition accuracy is improved by up to 14.31% when using the narrow bandwidth Tophat kernel.

**Keywords:** Machine Learning, Kernel Density Estimation, Data Augmentation

\* Corresponding Author: hglee@duksung.ac.kr

+ 본 연구는 방위사업청과 국방과학연구소가 지원하는 미래전투체계 네트워크 기술 특화연구센터 사업의 일환으로 수행되었습니다.(UD190033ED)

Manuscript received August 10, 2022 / revised August 29, 2022 / accepted August 29, 2022

1) 대구대학교 정보통신공학과, 제1저자  
2) 덕성여자대학교 소프트웨어학과, 교신저자

## 1. 서론

머신러닝(ML, Machine Learning)기반 응용의 인식성능은 적용된 모델의 종류 및 크기, 학습 방법, 학습 데이터의 품질과 양에 의해 결정된다. 기존 ML 관련 연구의 대부분이 이미지 인식 관련 모델의 개발에 초점이 맞춰져 있으며, 정량적인 성능 비교를 위해 ImageNet(Deng, J. et al., 2009), CIFAR-10 (Krizhevsky, A. & Hinton, G., 2009), MNIST (Deng, L., 2012) 등과 같이 잘 알려진 오픈 데이터셋(Open data set)을 사용하여 모델을 학습하고 성능을 평가한다. 대표적으로 알려진 ImageNet 같은 경우, 백만 개 이상의 학습용 데이터가 잘 정리되어 있으므로 학습과 평가 과정에 필요한 충분한 정보가 반영된다고 볼 수 있다. 다만, 이러한 데이터셋들은 대부분의 컴퓨터 비전에 초점이 맞춰져 있을 뿐, 다양한 목적으로, 다양한 센서들이 사용되는 인식응용에서 학습 과정 및 객관적인 성능 비교를 위한 충분한 데이터셋이 부족한 것이 현실이다. 실제로 컴퓨터 비전을 제외한 대부분의 응용에서는 데이터셋을 직접 구축해야 할 필요가 있다. 특히 지도학습 분야에서 데이터셋 구축은 가장 큰 비용과 시간이 발생하는 단계로, 개인이나 작은 기관의 경우 연구/개발에 필요한 충분한 데이터셋을 구축하는 데에는 많은 한계가 존재한다.

ML 관련 대부분의 연구가 ML 기법의 인식 정확도 평가에 치중되어있으나, 높은 정확도를 달성하기 위해서는 충분한 데이터셋이 확보되어야 한다. 특히 컴퓨터 비전 분야에서는 데이터셋의 크기에 따라 달성 가능한 정확도의 한계가 확실히 드러난다. 사용한 데이터셋의 수가 5천만개 이하인 경우에는 92% 정도의 정확도를 달성하였지만, 데이터셋의 수가 1억 5천만개로 증가할 경우 98% 이상의 높은 정확도를 보이고 있다 (Wang, H et al., 2018).

본 논문에서는 데이터 부족으로 인해 충분한 인식성능을 발휘하지 못하는 환경에서 커널 밀도 추정 방식을 통해 데이터셋의 수를 증폭(Augmentation, 데이터 수를 늘리기)하고 증폭된 데이터가 원본 데이터의 특징을 얼마나 잘

반영하는지, 어느 정도의 인식성능 향상을 도출하는지에 관한 실험과 이에 따른 비교 분석을 한다.

데이터 증폭을 통한 학습의 경우, 제일 먼저 수집된 적은 양의 데이터로부터 새로운 데이터셋을 증폭하고, 증폭된 데이터를 기반으로 학습 과정을 반복하면서 증폭된 데이터의 유의미성을 평가한다.

데이터를 증폭하는 방법으로 데이터의 고유한 패턴특성을 잃지 않는 범위 내에서 다양한 방법을 활용할 수 있다. 적대적 생성 신경망(GAN, Generative Adversarial Networks), 오토인코더(Autoencoder), 가우시안 믹스처(GMM, Gaussian Mixture Model), 커널 밀도 추정(KDE, Kernel Density Estimation) 등의 기법이 적용될 수 있다.

GAN을 사용한 방법은 생성자(Generator)와 판별자(Discriminator)라는 네트워크를 각각 생성하여 학습하는 방식으로 지도학습과 비지도학습의 혼합형태이다 (Goodfellow, Ian J. et al., 2014). 이를 통해 생성된 데이터의 경우 원본 데이터의 특성을 유지하면서도 사람도 구분하기 힘들 정도의 고화질 이미지를 생성하는 등 가장 뛰어난 성능을 보이지만, 모델 자체의 크기가 매우 크며 학습에 상당한 시간이 소요되기 때문에 사용되는 응용에 한계가 존재한다.

오토인코더는 비지도학습의 일종으로, 입력을 출력으로 복사하는 신경망이며 주로 이상 탐지 응용에 사용된다 (Bank, D., et al., 2003; Kim, K. et al., 2020). GMM과 KDE는 위 2가지 기법과는 달리 신경망(NN, Neural Network) 기반이 아니며, 알고리즘의 적용이 간결하나, 거리 기반 알고리즘이므로 증폭되는 데이터셋의 수가 많아질 경우, 데이터 고유의 패턴특성을 잃게 되어 학습의 효과가 떨어질 수 있다. GMM의 경우, 데이터가 여러 개의 가우시안(Gaussian) 분포를 따른다는 가정을 가진 모수적(Parametric) 추정 방식이므로 군집화(Clustering)에 많이 적용된다. KDE는 관측된 데이터만으로 확률 밀도를 찾는 비모수적(Non-parametric) 추정 방식이다 (Izenman, A. J., 1991).

본 연구에 적용된 데이터셋 증폭 방법은 움직임 인식응용에 효율적으로 사용할 수 있는 KDE 기법

으로, 움직임 인식 센서에서 실제 수집된 데이터를 토대로 확률 밀도를 추정하므로 본 연구에서 사용된 비선형적인 데이터셋을 분석하기에 알맞은 방법이다. 원본 데이터셋은 직접 수집한 데이터로 구성되어 있으며, 움직임 인식에 사용되는 플렉스(Flex) 센서로부터 수집된 시간에 따른 움직임을 각도 정보로 표현하는 출력값들의 집합이다. 해당 데이터셋은 기존 연구(Jung, W. and Lee, H.G., 2022)에서 2축 플렉스 센서를 통해 손가락 동작을 인식하여 17가지의 손동작을 분류하는 인터페이스를 구현하기 위해 사용되었다. 감지에 부착된 플렉스 센서의 출력은 손가락 관절의 x축, y축에 해당하는 각도이다.

플렉스 센서로부터 수집된 원본 데이터셋은 KDE 기법을 사용하여 증폭하며, 증폭에 사용된 커널의 종류에 따른 인식의 정확도를 비교 분석하여 움직임 인식응용에 가장 적합한 커널이 무엇인지 탐색한다. 또한, 사용 가능한 데이터셋의 수에 따른 데이터 증폭의 필요성 및 기여도 평가를 위해, 수집된 원본 데이터셋 중 학습에 사용되는 데이터셋의 수와 증폭 비율을 변경하였을 때 움직임 인식 정확도를 비교/분석한다.

인식성능 평가를 위한 ML 모델로 비교적 구조가 단순한 MLP(Multi-Layer Perceptron)가 적용되었으며, 수집된 원본 데이터셋만으로 학습을 진행했을 때의 성능(정확도)과 KDE 방법을 통해 증폭된 데이터로 학습을 진행하였을 때의 인식 정확도를 비교하여 증폭된 데이터의 신뢰도를 평가한다. 평가의 공정성을 위해 모든 평가에는 증폭된 데이터가 아닌 수집된 원본 데이터셋만을 이용하였다. 실험결과, 사용 가능한 원본 데이터셋이 적은 경우 증폭비율에 따라 가우시안 커널의 경우 최대 11.13%의 정확도 향상을, Tophat 커널의 경우 최대 14.31%의 인식 정확도 향상을 보였다.

## 2. 데이터 증폭 기법

### 2.1 커널 밀도 추정(KDE) 기법

많이 쓰이는 2가지 밀도 추정 기법은 가우시안 믹스처(Gaussian Mixture)와 커널 밀도 추정

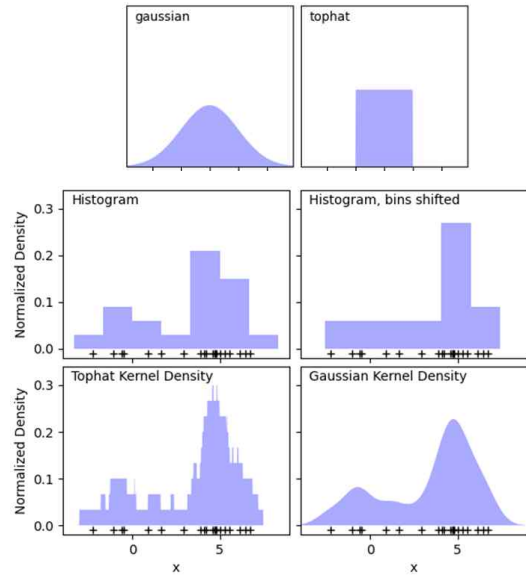


Fig. 1 Examples by kernel type in KDE (scikit-learn.org)

기법이다. 일반적으로 가우시안 믹스처 모델은 클러스터링의 맥락으로 구분되므로, 본 연구에서는 커널 밀도 추정 기법을 적용하였다. 밀도 추정의 가장 대표적인 예는 히스토그램이며, bin(Bin)과 각 bin 내의 데이터 수를 시각화한 형태이다.

커널 밀도 추정 기법은 관측된 데이터의 분포로부터 원래 변수의 분포 특성(밀도)을 추정하므로, 원본 데이터의 특성을 잘 보존한다는 특징이 있다. 예를 들어, “키-몸무게는 가우시안 분포를 따른다”고 가정하지 않고 실제 집계된 데이터를 통해 키-몸무게 분포를 추정한다.

KDE에서 사용되는 커널 함수의 정의는 적분값은 1인 함수이면서(식 1), 원점을 중심으로 대칭이며 non-negative인 함수이다(식 2). 식으로 표현하면 다음과 같다.

$$\int_{-\infty}^{\infty} K(u)du = 1 \tag{1}$$

$$K(u) = K(-u), K(u) \geq 0, \forall u \tag{2}$$

식에서 K는 커널(음이 아닌 함수)이며, 원도우 함수로도 불린다. 아래 식(식 3)은 KDE에서 랜덤 변수 x에 대한 확률밀도함수를 추정한다.

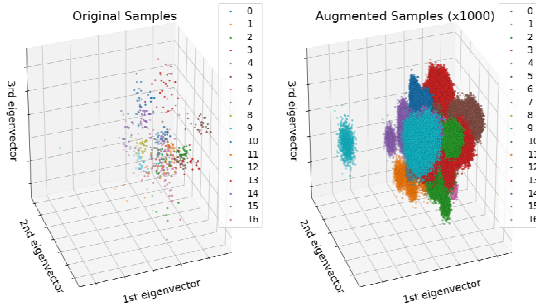


Fig. 2 Example of generating too many sets from insufficient data

$x_i$ 는 관측된 데이터 포인트 하나를 의미하며,  $h$ 는 대역폭(bandwidth)이라 불리는 파라미터로, 커널의 모양을 완만하거나 뾰족하게 만드는 역할을 한다.

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (3)$$

그림 1은 커널의 종류에 따른 모양(그림 윗부분) 및 확률 밀도 추정에 적용된 예시(그림 아랫부분)이며, 히스토그램, Tophat, 가우시안 커널 순서로 대역폭( $h$ )이 커지며 커널의 모양이 완만해지는 것을 알 수 있다. Tophat 커널(식 4)과 가우스 커널(식 5)에 대한 정의는 다음과 같다.

$$K(x, h) \propto 1, \text{ if } x < h \quad (4)$$

$$K(x, h) \propto \exp\left(-\frac{x^2}{2h^2}\right) \quad (5)$$

## 2.2 증폭할 데이터셋의 수 선택

본 연구의 목적은 데이터셋을 수집하기 힘든 환경에서 인식 정확도를 높일 수 있도록 학습에 사용되는 데이터셋의 수를 증폭하는 것이다. 일반적으로 직접 수집한 데이터셋의 수가 많으면 신뢰성이 높아진다. 하지만 데이터셋을 증폭하여 학습에 사용하는 경우, 너무 많은 데이터를 증폭하는 것은 오히려 원본 데이터의 성질을 잃거나, 편향이 발생할 수 있다. Zhu, X. et al. (2015)에 따르면, 고전적인 ML 기법을 사용하

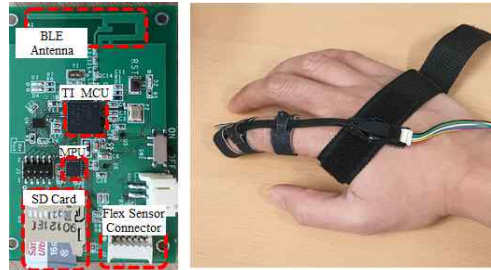


Fig. 3 The prototype (left) & the flex sensor attached on a hand(right)

는 경우, 일정 시점에서 인식 정확도가 포화 되며, 비교적 최신 기법인 깊은 신경망(DNN, Deep Neural Network) 기반 ML의 경우, 정확도가 포화 되는 시점이 상당히 미뤄지며 데이터셋의 수가 클수록 높은 성능을 보이나, 선형비례적인 효율성은 가지지 못한다. 결과적으로, 원본 데이터와 증폭할 데이터 간 최적의 비율을 탐색하는 과정이 필요하다. 그림 2는 매우 적은 수의 데이터셋에서 너무 많은 배수로 데이터셋을 증폭한 예를 나타내었다. 왼쪽 그래프는 총 17개의 클래스를 갖는 데이터셋에서 클래스 당 30개의 데이터셋을 도식화한 것이며, 오른쪽 그래프는 가우시안 커널을 적용하여 데이터셋을 1,000배 증폭시킨 예이다. 커널의 대역폭이 넓고, 과도한 배수로 원본 데이터를 증폭시키는 경우, 클래스 간 분포의 경계가 모호해지는 모습을 보여주게 되고, 이를 학습에 이용할 경우 인식 정확도가 저하되는 결과를 가져올 수 있다. 이는 해당 데이터셋의 특성과 원본 데이터셋의 수에 따른 최적의 데이터 증폭 비율을 탐색하는 과정이 필요함을 의미한다. 본 연구에서는 실험을 통해 학습에 사용되는 원본 데이터셋의 수와 증폭 비율 간 상관관계를 탐색하였으며, 그 결과는 3.3절에서 상세히 설명한다.

## 3. 실험결과

### 3.1 실험환경 구축

실험을 위해 수집된 원본 데이터셋은 손가락

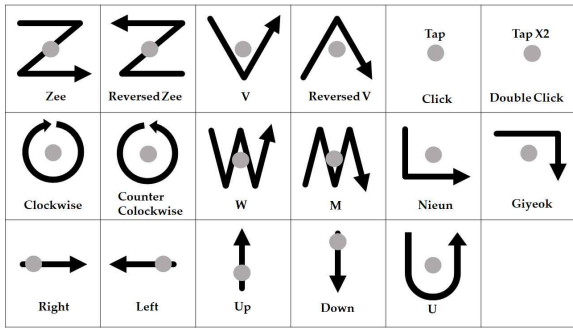


Fig. 4 Defined gestures in the dataset

관절의 움직임 각도로써, 2축 (x와 y축) 플렉스 센서의 출력값인  $-180^{\circ} \sim 180^{\circ}$ 의 32비트 ( x축 16비트, y축 16비트)로 구성된 실수 자료형이다. 해당 센서는 그림 3의 오른쪽 부분과 같이 웨어러블 형태로 손가락에 부착되어 시간에 따른 손가락의 움직임을 측정할 수 있다. 그림 3의 왼쪽 부분은 데이터 수집에 사용된 프로토타입으로 손가락 움직임 인식을 위한 선행연구(Jung, W. and Lee, H.G., 2022)에서 개발되었으며, 본 연구에서는 동일 프로토타입을 통해 실험에 사용된 원본 데이터를 수집하였다.

그림 4는 정의된 17가지의 손가락 동작(제스처)을 나타낸다. 해당 그림 중앙의 회색 원은 동작의 시작/종료 지점을 의미하고, 하나의 동작은 화살표 방향으로 진행되며, 모든 동작의 길이는 0.8~1.2초이다.

정의된 17가지 동작당 300개의 데이터셋이 수집되어 총 5,100개의 데이터셋을 센서로부터 직접 수집하였으며, 이 중 50%인 2,550개의 데이터셋은 평가용으로만 사용되며, 학습에 사용된 원본 데이터는 850~2,550개로 변화시키며 커널의 성능을 탐색한다.

**3.2 분류기 및 KDE 설정**

본 연구에서 적용한 분류기는 가장 간단한 ML 모델 중 하나인 MLP(Multi-Layer Perceptron)이며, 플렉스 센서의 출력은 하나의 인식 동작 시간 동안 x축 및 y축에서 각각 14개의 출력값이 수집되므로, 하나의 동작당 총 28개의 값(feature)으로 구성되어 있다 (그림 5 윗부분).

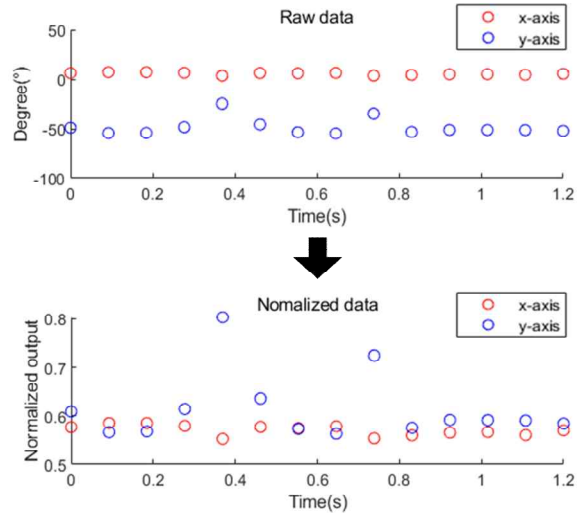


Fig. 5 Raw data (upper) & normalized data (lower) from the flex sensor

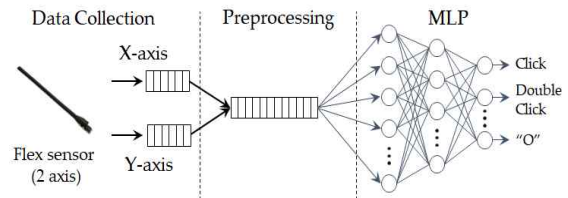


Fig. 6 Overview of the system

입력된 하나의 데이터셋은 각 동작별, 사용자별 편차를 최소화하기 위한 최대-최소(Min-Max) 정규화 과정을 (그림 5 아랫부분) 거쳐 MLP의 입력 계층으로 전달된다. 위 과정을 모두 도식화하면 그림 6과 같다. MLP는 Scikit-learn의 API를 사용하였으며, 계층별 노드 수는 입력 계층에서는 하나의 데이터셋을 구성하고 있는 값의 수와 같은 28개, 단일 은닉 계층에서는 150개, 출력 계층에서는 동작의 수와 같은 17개로 정의되었다. 활성화 함수로는 relu가 적용되었으며, 하이퍼파라미터로 최대 반복수(epoch)는 500, 학습률(learning rate)은 0.0075로 설정하였다. 배치 사이즈(batch size)는 Scikit-learn의 MLP Classifier 함수로 자동으로 설정된다. 실험에서 사용한 MLP 계층 구조 및 하이퍼파라미터들은 원본 데이터만을 사용한 설계공간탐색을 통해 최대의 학습성능을 보이는 설정들로,

Table 1 Results according to kernel type, number of collected samples and augmented rate.  
 (# of dataset used for testing = 2,550, # of recognized gestures (classes) = 17)

| # of original trainset per a gesture | Amp. factor | # of generated trainset per a gesture | # of total trainset | Acc_A(%) (Original) | Acc_B(%) (Gaussian) | Acc_C(%) (Tophat) | $\Delta$ (B-A) | $\Delta$ (C-A) |
|--------------------------------------|-------------|---------------------------------------|---------------------|---------------------|---------------------|-------------------|----------------|----------------|
| 50<br>(total 850 samples)            | 10          | 500                                   | 9,350               | 78.51               | 86.78               | 87.88             | 8.27           | 9.37           |
|                                      | 20          | 1,000                                 | 17,850              | 78.51               | 86.78               | 78.51             | 8.27           | 0.00           |
|                                      | 30          | 1,500                                 | 26,350              | 78.51               | 84.20               | 90.08             | 5.69           | 11.57          |
|                                      | 40          | 2,000                                 | 34,850              | 78.51               | 81.18               | 90.63             | 2.67           | 12.12          |
|                                      | 50          | 2,500                                 | 43,350              | 78.51               | 85.29               | 91.14             | 6.78           | 12.63          |
| 75<br>(total 1,275 samples)          | 100         | 5,000                                 | 85,850              | 78.51               | 77.22               | 89.53             | -1.29          | 11.02          |
|                                      | 10          | 750                                   | 14,025              | 77.69               | 88.82               | 90.04             | 11.13          | 12.35          |
|                                      | 20          | 1,500                                 | 26,775              | 77.69               | 77.69               | 91.57             | 0.00           | 13.88          |
|                                      | 30          | 2,250                                 | 39,525              | 77.69               | 87.53               | 90.75             | 9.84           | 13.06          |
|                                      | 40          | 3,000                                 | 52,275              | 77.69               | 86.94               | 91.76             | 9.25           | 14.07          |
| 150<br>(Total 2,550 samples)         | 50          | 3,750                                 | 65,025              | 77.69               | 81.65               | 91.88             | 3.96           | 14.19          |
|                                      | 100         | 7,500                                 | 128,775             | 77.69               | 77.69               | 92.00             | 0.00           | 14.31          |
|                                      | 10          | 1,500                                 | 28,050              | 92.24               | 91.29               | 94.71             | -0.95          | 2.47           |
|                                      | 20          | 3,000                                 | 53,550              | 92.24               | 91.22               | 94.04             | -1.02          | 1.80           |
|                                      | 30          | 4,500                                 | 79,050              | 92.24               | 86.82               | 92.20             | -5.42          | -0.04          |
|                                      | 40          | 6,000                                 | 104,550             | 92.24               | 90.20               | 95.14             | -2.04          | 2.90           |
|                                      | 50          | 7,500                                 | 130,050             | 92.24               | 79.76               | 94.55             | -12.48         | 2.31           |
|                                      | 100         | 15,000                                | 257,550             | 92.24               | 75.29               | 95.53             | -16.95         | 3.29           |

해당 파라미터들을 변화시키면서 유의미한 인식 성능 향상이 발견되지 않는 임계점을 중심으로 구성하였다.

앞 2.2절에서 기술된 바와 같이, KDE 기법의 주 파라미터는 대역폭이다. 대역폭에 따라 커널의 모양이 변경되며, 뽀족하거나 넓어지는 형태를 나타낸다. 본 연구에서는 최적의 대역폭을 탐색하기 위해, Scikit-learn의 API 중 GridSearchCV를 사용하여 최적의 대역폭을 탐색하였다. GridSearchCV의 기능은 해당 추정기(KDE)에 대한 지정된 파라미터(대역폭)를 그리드 탐색으로 최적화하는 것이다.

### 3.3 실험결과

먼저 증폭된 데이터셋이 원본 데이터셋의 특

성을 잘 유지하고 있는지 검증하기 위해 같은 수의 원본 데이터셋과 증폭된 데이터셋을 가지고 학습하였을 때의 인식 정확도를 비교하였다. 실제 수집된 원본 데이터셋의 최대수는 동작당 150개(전체 2,550개)이므로 같은 수의 증폭 데이터를 생성하기 위해 동작당 50개의 데이터셋을 3배 증폭하여 150개의 증폭된 데이터셋을 만든 후 증폭된 데이터셋만을 이용하여 동일한 모델을 학습한 후 인식 정확도를 비교하였다. 원본 데이터만을 사용한 경우 92.24%의 인식 정확도를 보인 반면, 같은 수의 증폭된 데이터셋을 사용한 경우 가우시안 커널에서 77.37%, Tophat 커널에서 84.43%의 인식 정확도가 관찰되었다. 이는 증폭된 데이터가 어느 정도 원본 데이터의 특성을 반영한 것으로 분석된다. 하지만 감소한 인식 정확도를 볼 때 원본 데이터의 특성이 그

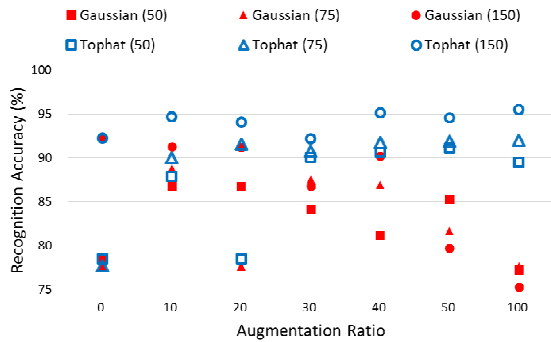


Fig. 7 Recognition accuracy variation with the number of augmented samples and kernel types.

대로 전달되지 못하고 일부 손실되었다는 것을 알 수 있다.

표 1은 좀 더 다양한 조건을 가지고 실험한 결과이며, 하나의 인식 동작에 대한 원본 데이터셋의 수, 증폭 비율과 증폭을 통해 생성된 새로운 데이터, 학습의 위해 사용된 총 데이터셋의 수와, 그에 따른 3가지 설정들의 인식 정확도를 나타낸 것이다. 테스트를 위한 데이터셋의 수는 2,550개로 고정되어있으며, 표의 “Acc\_A”에 해당하는 항목은 사용 가능한 원본 데이터셋만으로 학습된 MLP에 2,550개의 데이터셋으로 평가된 결과를 나타낸다. “Acc\_B”는 수집된 원본데이터에 추가하여 가우시안 커널이 적용된 KDE로 증폭된 데이터셋을 추가하여 학습을 진행하고, 테스트셋으로 평가된 결과, “Acc\_C”는 Tophat 커널의 결과이다. 모든 설정에 대해, 테스트셋의 2,550개의 데이터셋은 학습 과정에 노출되지 않는다. 수집된 원본 데이터셋의 수는 동작 당 50, 75, 150개(17 동작 전체로 계산하면 850, 1,275, 2,550개)로 변화하면서 인식 정확도의 변화를 분석하였다.

먼저, 가우시안 커널을 적용했을 때 발생하는 인식 정확도 변화를 보면, 학습에 사용되는 수집된 원본 데이터가 많을 경우(동작 당 50개의 원본 데이터셋만 사용) 데이터 증폭을 통해 인식 정확도가 최대 8.27% 향상되었다. 증폭 비율에 따른 결과를 분석해 보면 10배 이상의 데이터 증폭은 인식 정확도 향상률을 감소시키며,

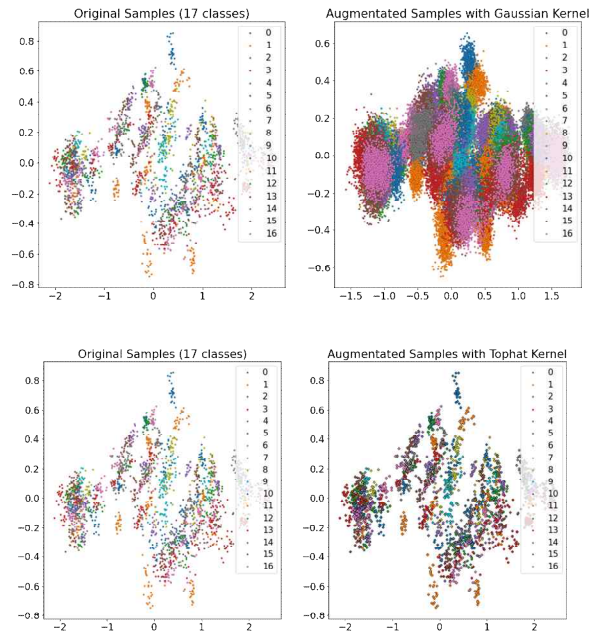


Fig. 8 Result of sample generation according to kernel type (upper: Gaussian, lower: Tophat)

100배 이상의 경우에는 증폭 전보다 오히려 정확도가 감소하는 현상을 보였다. Tophat 커널을 사용한 경우 최대 12.63%의 인식 정확도 향상으로 가우시안 커널보다 높은 인식 정확도 향상을 보였으며, 증폭비를 높임에 따라 선형 비례 관계는 아니더라도 인식 정확도가 조금씩 높아짐을 보였다.

학습에 사용된 원본 데이터셋을 증가시켰을 경우(동작 당 75개의 원본 데이터셋만 사용) 동작당 50개의 원본 데이터만을 사용한 경우보다 인식 정확도가 오히려 약간 감소하는 현상을 보인 반면, 증폭된 데이터를 통해 학습한 경우 가우시안 커널의 경우 최대 11.13%, Tophat 커널의 경우 최대 14.13%의 인식 정확도 향상이 관찰되었으며, 증폭비율 증가에 따른 인식 정확도 향상은 이전과 비슷한 양상을 보였다.

학습에 사용된 원본 데이터셋의 수를 최대로 사용한 경우 (동작 당 150개의 원본 데이터셋을 사용), 원본 데이터셋만을 사용해서도 인식 정확도가 92.24%로 향상되었다. 이에 따라 데이터 증폭을 같이 사용한 결과에 대한 인식 정확도의

향상도는 Tophat 커널의 경우 최대 3.29% 증가하여 95.53%의 향상된 정확도를 보인 반면, 가우시안 커널의 경우 인식 정확도 향상이 거의 없거나 오히려 감소하는 경향을 보였다.

그림 7은 커널의 종류, 사용된 원본 데이터의 수, 증폭 비율에 따른 인식 정확도 변화추세를 그래프를 통해 표현하고 있다. Tophat 커널로 증폭된 데이터는 사용된 원본 데이터셋의 수에 따라 약간의 편차를 보이지만, 전반적으로 증폭된 데이터셋의 수가 증가할수록 인식 정확도가 증가하는 추세를 보인다. 반면 가우시안 커널을 통한 증폭의 경우, 증폭 비율이 10배 이상일 경우 사용되는 원본 데이터셋의 개수와 상관없이 모두 인식 정확도가 감소하는 모습을 보인다.

그림 8은 수집된 원본 데이터셋과 KDE를 통해 원본의 100배만큼 증폭된 데이터를 주성분 분석(PCA, Principal Component Analysis)을 통해 2차원으로 차원 감소시켜 도식화한 그림이다. 가우시안 커널의 경우(위), 부드럽고 넓은 커널의 형태에 따라 데이터의 분포가 넓어짐에 따라 증폭되는 데이터의 수가 많아져 각 동작의 특성이 사라지지만, Tophat 커널의 경우(아래), 뾰족한 커널의 형태처럼 데이터의 분포가 넓게 퍼져지는 않아 증폭되는 데이터 수가 많더라도 어느 정도까지는 그 특성을 유지할 수 있는 것으로 분석되며, 본 연구의 대상인 움직임 인식 응용 데이터셋에는 뾰족한 형태의 커널을 적용하는 것이 더 유리하다고 볼 수 있다.

실험결과를 정리하면, 학습에 사용할 수 있는 원본 데이터셋의 수가 충분치 않을 때 데이터 증폭은 인식 정확도 향상에 많은 도움을 줄 수 있으며, 움직임 인식의 경우 데이터 증폭 비율이 높아지더라도 그 특성을 더욱 오래 유지할 수 있는 Tophat 커널을 사용한 KDE 기법이 가우시안 커널을 사용한 경우보다 더 적합하다고 볼 수 있다.

#### 4. 결론

본 연구에서는 충분한 학습용 데이터셋을 수집하기 어려운 움직임 인식응용을 대상으로, 신

뢰성 있는 데이터셋을 증폭하는 방법을 제안하고 분석하였다. 학습용 데이터 증폭 방법으로 비교적 사용이 간단한 커널밀도 추정 기법을 사용하였으며, 증폭을 위해 가우시안과 Tophat 커널을 사용하였다. 제안된 데이터 증폭기법의 효율성 검증을 위해 플렉스 센서로부터 수집한 움직임 데이터를 기반으로 사용된 원본 데이터의 양과 증폭 비율을 다양하게 설정한 후 MLP를 학습하였으며, 실험결과, 가우시안 커널이 최대 11.13%, Tophat 커널이 최대 14.31%의 인식 정확도 향상을 보였다. 두 커널 모두 사용 가능한 데이터가 적은 경우 상대적으로 높은 인식 정확도의 향상을 보였다. 다만 가우시안 커널의 경우 원본 데이터의 10배 정도로 데이터를 증폭할 경우 정확도의 향상이 확인되었지만, 그 이상 증폭할 경우 인식 정확도가 감소하였다. 반면 Tophat 커널로 증폭된 데이터셋들은 사용 가능한 원본 데이터셋이 충분치 않은 경우 100배의 증폭에서도 정확도 향상이 꾸준히 확인되었다. 이는 움직임 인식응용에는 Tophat 커널을 사용하는 것이 적절하다는 것을 나타낸다.

#### References

- Bank, D., et al., (2003). Autoencoders, <https://doi.org/10.48550/arxiv.2003.05991>
- Deng, J. et al. (2009). ImageNet: A Large-Scale Hierarchical Image Database, <https://doi.org/10.1109/CVPR.2009.5206848>.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6), 141 - 142.
- Goodfellow, Ian J. et al. (2014). Generative Adversarial Networks, <https://doi.org/10.48550/arxiv.1406.2661>
- Izenman, A. J. (1991). Review Papers: Recent Developments in Nonparametric Density Estimation, <https://doi.org/10.1080/01621459.1991.10475021>
- Jung, W. and Lee, H.G. (2022) Energy -



Accuracy Aware Finger Gesture Recognition for Wearable IoT Devices. Sensors, <https://doi.org/10.3390/s22134801>

Kim, K. et al. (2020). Fault Detection Using Signal Reconstruction Model Based on Autoencoder in Thermal Power Plant, *KIEE*, 69(6), 800 - 807.

Krizhevsky, A. & Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical Report, University of Toronto, Toronto, Ontario

Scikit-learn, [https://scikit-learn.org/stable/auto\\_examples/neighbors/plot\\_kde\\_1d.html#sphx-gl-auto-examples-neighbors-plot-kde-1d-py](https://scikit-learn.org/stable/auto_examples/neighbors/plot_kde_1d.html#sphx-gl-auto-examples-neighbors-plot-kde-1d-py)

Wang, H. et al. (2018). A Bayesian Scene-Prior-Based Deep Network Model for Face Verification. Sensors. <https://doi.org/10.3390/s18061906>.

Zhu, X., Vondrick, C., Fowlkes, C. and Ramanan, D. (2015). Do We Need More Training Data?, <https://doi.org/10.48550/arXiv.1503.01508>.



**정 우 순 (Woosoon Jung)**

- 정회원
- 대구대학교 임베디드시스템공학과 학사
- 대구대학교 정보통신공학과 석사
- (현재) 대구대학교 정보통신공학과 박사과정
- 관심분야 : 저전력 시스템 설계, 패턴인식, AI



**이 형 규 (Hyung Gyu Lee)**

- 정회원
- 서울대학교 전기컴퓨터공학부 석사
- 서울대학교 전기컴퓨터공학부 박사
- 대구대학교 정보통신대학 임베디드시스템공학부 부교수
- (현재) 덕성여자대학교 소프트웨어공학과 부교수
- 관심분야 : 저전력 시스템 설계, 에너지 하베스팅, 저전력 메모리