(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0136597
(43) 공개일자 2021년11월17일

(51) 국제특허분류(Int. Cl.)
G06N 3/08 (2006.01) *G06K 9/62* (2006.01)
G06T 5/40 (2006.01)
(52) CPC특허분류
G06N 3/08 (2013.01)
G06K 9/627 (2013.01)
(21) 출원번호 10-2020-0055055
(22) 출원일자 2020년05월08일
심사청구일자 2020년05월08일

(71) 출원인
한국기술교육대학교 산학협력단
충청남도 천안시 동남구 병천면 충절로 1600 (한국기술교육대학교내)
(72) 발명자
김병철
세종특별자치시 달빛로 211(범지기마을10단지, 아름동), 1012동 101호
(74) 대리인
이철희

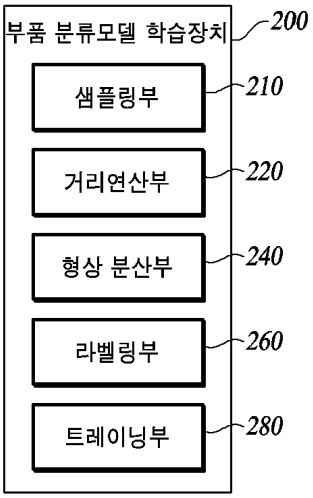
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 **부품의 3차원 모델 분류 방법, 이를 위한 컴퓨터 프로그램 및 장치**

(57) 요 약

본 발명은 부품의 3차원 데이터를 자동으로 분류하기 위해, 부품의 3차원 데이터를 점으로 샘플링(sampling)하고, 샘플링된 점의 거리를 연산하여 히스토그램으로 표현함으로써 1차원 데이터인 형상 분산(shape distribution)화 한 이후 형상 분사화된 데이터에 대응하는 부품의 종류로 라벨링(labeling)함으로써 트레이닝 셋을 추출하고, 신경망(neural network) 기반의 부품 분류모델을 학습시키는 학습방법, 컴퓨터 프로그램 및 그 장치와 학습시킴으로써 부품의 3차원 데이터를 자동으로 분류하게 하는 분류방법 및 그 컴퓨터 프로그램을 제공한다.

대 표 도 - 도2



(52) CPC특허분류
G06T 5/40 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1345317035
과제번호 NRF-2017R1D1A1B03028274
부처명 교육부
과제관리(전문)기관명 한국연구재단
연구사업명 개인기초연구
연구과제명 딥러닝과 형상분산을 이용한 3D 스캐닝 점군 데이터의 플랜트 기자재 인식
기 여 율 1/1
과제수행기관명 한국기술교육대학교 산학협력단
연구기간 2017.06.01 ~ 2020.05.31

명 세 서

청구범위

청구항 1

복수의 부품의 3차원 데이터를 일정 개수의 점(point)으로 각각 샘플링(sampling)하여 샘플링 데이터 셋(sampling data set)을 얻는 샘플링 과정;

상기 샘플링 데이터 셋의 샘플링 데이터 내 모든 점들 간 거리(distance)를 각각 연산하여 거리값 데이터 셋(distance data set)을 얻는 거리연산 과정;

상기 거리값 데이터 셋을 히스토그램(histogram)으로 각각 표현하여 형상분산 데이터 셋(shape distribution data set)을 얻는 형상 분산화 과정;

상기 형상 분산 데이터 셋을 부품의 종류로서 각각 라벨링(labeling)하여 트레이닝 셋(training set)을 추출하는 라벨링과정; 및

상기 트레이닝 셋을 이용하여 신경망(neural network) 기반의 부품 분류모델(part clustering model)을 트레이닝(training)하는 트레이닝 과정

을 포함하는 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 2

제 1항에 있어서,

상기 샘플링 과정은,

상기 3차원 모델 데이터를 삼각화(triangulation)하여 삼각형의 면적에 비례하게 샘플링하는 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 3

제 1항에 있어서,

상기 샘플링 과정은,

상기 샘플링 후 재표본(resampling) 방법을 수행하는 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 4

제 1항에 있어서,

상기 형상 분산화 과정은,

상기 거리값 데이터 셋으로부터 최소 거리값 및 최대 거리값을 산출하고, 상기 최소 거리값 및 최대 거리값의 차(difference)를 고정된 개수의 빈(bin)으로 나누어 히스토그램화하는 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 5

제 1항에 있어서,

상기 부품 분류모델은 합성곱 신경망(CNN; Convolution Neural Network) 기반의 부품 분류모델인 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 6

제 5항에 있어서,

상기 합성곱 신경망 기반의 부품 분류모델은 1차원 데이터에 적합하게 변경한 VGG(Visual Geometry Group) 신경

망 기반의 모델인 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 7

제 6항에 있어서,

상기 VGG 모델은,

크기가 25인 1차원 필터(1-D filter)가 적용되는 합성곱 레이어(convolutional layer)로 구성된 것을 특징으로 하는 부품의 분류모델 학습방법.

청구항 8

제 1항 내지 제 7항 중 어느 한 항에 따른 분류모델 학습방법에 의해 기 학습된 부품의 분류모델을 이용하여 부품의 3차원 데이터를 분류하는 것을 특징으로 하는 부품 분류방법.

청구항 9

제 1항 내지 제 7항 중 어느 한 항에 따른 부품의 분류모델 학습방법이 포함하는 각 과정을 실행시키기 위하여 컴퓨터로 읽을 수 있는 기록매체에 저장된 컴퓨터 프로그램.

청구항 10

제 8항에 따른 분류방법이 포함하는 각 과정을 실행시키기 위하여 컴퓨터로 읽을 수 있는 기록매체에 저장된 컴퓨터 프로그램.

청구항 11

복수의 부품의 3차원 데이터를 일정 개수의 점(point)으로 각각 샘플링(sampling)하여 샘플링 데이터 셋(sampling data set)을 얻는 샘플링부(sampling unit);

상기 샘플링 데이터 셋의 샘플링 데이터 내 모든 점들 간 거리(distance)를 각각 연산하여 거리값 데이터 셋(distance data set)을 얻는 거리연산부(distance calculator);

상기 거리값 데이터 셋을 히스토그램(histogram)으로 각각 표현하여 형상 분산 데이터 셋(shape distribution data set)을 얻는 형상 분산부(shape distribution unit);

상기 형상 분산 데이터 셋을 부품의 종류로서 각각 라벨링(labeling)하여 트레이닝 셋(training set)을 추출하는 라벨링부(labeling unit); 및

상기 트레이닝 셋을 이용하여 신경망(Neural Network) 기반의 부품 분류모델(part clustering model)을 트레이닝(training)하는 트레이닝부(training unit)

를 포함하는 것을 특징으로 하는 부품의 분류모델 학습장치.

발명의 설명

기 술 분 야

[0001] 본 실시예는 부품의 3차원 모델 분류 방법, 컴퓨터 프로그램 및 장치에 관한 것이다. 더욱 상세하게는 부품의 3차원 모델을 점으로 샘플링하고, 히스토그램으로 표현한 것을 학습모델에 학습하여 부품을 분류하는 부품의 3차원 모델의 분류 방법, 컴퓨터 프로그램 및 장치에 관한 것이다.

배 경 기 술

[0002] 이 부분에 기술된 내용은 단순히 본 실시예에 대한 배경 정보를 제공할 뿐 종래기술을 구성하는 것은 아니다.

[0003] 플랜트(plant)는 장기간의 운영을 통해 지속적인 유지 보수 및 설계 변경이 일어나 부품의 교체가 빈번하다. 지속적으로 변화하는 플랜트 설계 정보를 관리하기 위해, 변경된 플랜트의 형상을 3차원 CAD 모델로 재구성하여 저장 및 관리하려는 요구가 증가하고 있다. 이를 해결하기 위한 방법으로, 라이다(lidar) 와 같은 3차원 스캐너

를 이용해 플랜트를 스캔한 후 역공학(reverse engineering) 기술을 활용해 3차원 CAD 모델로 재구성하는 방법(비특허문헌 1, 2 참조)이 시도되고 있다. 그러나 이와 같은 방법으로 재구성한 CAD 모델에는 대상 부품의 형상만으로 어떤 부품인지는 알 수 없는 문제점이 있었다. 따라서 작업자가 형상을 보고 일일이 수작업으로 부품의 종류를 식별해야 한다.

[0004] 이러한 작업을 자동으로 하기 위해, 형상 기술자(shape descriptor)를 이용해 대상물을 식별하기 위한 다양한 연구(비특허문헌 3 내지 6 참조)들이 제안되어 왔으나, 인식률이 높지 않았다. 최근에는 합성곱 신경망(CNN; Convolutional Neural Network)을 이용한 방법(비특허문헌 7 내지 15 참조)들이 제안되었으며, 기존의 형상 기술자를 이용한 방법보다 더 좋은 인식률을 보여주고 있다.

[0005] 이에 따라, 합성곱 신경망에 이용하기 위한 3차원 형상을 표현하는 다양한 방법들이 제안되고 있다. 3차원 형상의 경우에는 2차원 이미지와 달리 다양한 표현 방법이 존재한다. 예컨대, 3차원 형상은 3차원의 볼륨 모델(volume model), 2차원의 다중 뷰 이미지(multi-view image), 점군(point cloud) 등을 이용하여 표현할 수 있다. 도 1은 본 실시예에 따른 신경망 모델에 이용하기 위한 3차원 데이터의 표현방법의 예시도이다.

[0006] 3차원 볼륨 모델을 이용하는 방법은 3차원 형상을 복셀(voxel) 형태로 표현하는 것이다. 도 1의 (a)와 같이, 복셀은 작은 정육면체 또는 부피요소로 나눠질 수 있는 오브젝트를 이용하여 2차원 픽셀을 3차원 형태로 구현하는 것이다. 복셀은 2차원 이미지를 확장한 개념이기 때문에 2차원 이미지에 성공적으로 적용되었던 신경망 구조를 확장하여 쉽게 적용할 수 있다. 대표적인 방법으로는 3D ShapeNets(비특허문헌 7 참조)와 VoxNet(비특허문헌 8 참조)이 있다. 또한 플랜트 부품의 점군을 분류하는 방법을 제안한 Kim 의 연구(비특허문헌 9 참조)에서도 2차원 이미지 분류에 사용하는 VGG(비특허문헌 16 참조) 구조를 3차원 형상에 적용하였다. 그러나, 복셀은 이미지의 해상도가 높아질수록 연산 복잡도가 부피에 비례하여 증가한다는 문제점이 있다.

[0007] 2차원의 다중 뷰 이미지 형태로 표현하는 방법은, 도 1의 (b)와 같이 3차원 형상을 다양한 방향에서 바라본 복수의 이미지로 표현하는 것이다. 도 1의 (b)는 비행기를 네 방향에서 본 이미지를 2차원으로 표현한 것이다. 복셀을 이용한 방법이 형상의 내부까지 표현한 데 반해, 2차원 다중 뷰 이미지는 형상의 외부를 다양한 각도에서 표현하는 데 초점을 맞춘다. 2차원 다중 뷰 이미지를 기반으로 한 신경망 기반 분류모델의 정확도는 복셀을 기반으로 한 경우보다 더 높다고 알려져 있다. 대표적인 방법에는 MVCNN(비특허문헌 10, 11 참조), PANORAMA 기반 방법(비특허문헌 12 참조), RotationNet(비특허문헌 13 참조)이 있다.

[0008] 점군을 이용하는 방법은 도 1의 (c)와 같이, 3차원 형상을 균일하게 분포한 3차원 점으로 표현하는 것이다. 도 1의 (c)에는 플랜트 부품을 점군으로 표현한 하나의 예시가 나타나 있다. 이 방법은 물체의 형상을 점들로 표현한다는 점에서 기존의 표현 방법들과 차이가 있고, 스캔 데이터(scan data)를 즉시 이용할 수 있다는 장점이 있다. 대표적인 방법에는 PointNet(비특허문헌 14 참조)과 PointNet++(비특허문헌 15 참조)가 있다.

[0009] 그러나 CNN 이 주로 사용되는 2차원 이미지 분류에 비해, 3차원 형상을 분류하는 경우에는 데이터의 차원이 증가하여 다루어야 하는 전체 데이터 및 신경망(neural network)의 크기가 커지고, 학습 시간 및 메모리가 더 많이 소요되는 문제점이 있었다.

[0010] 이에 따라, 3차원 형상을 1차원의 형상 분산(shape distribution)(비특허문헌 3 참조)으로 표현하고, 이를 이용하여 합성곱 신경망 기반의 분류모델을 학습시키는 방법이 필요하다.

선행기술문헌

비특허문헌

[0011] (비특허문헌 0001) (1) Kawashima, K., Kanai, S. and Date, H., 2014, “As-Built Modeling of Piping System from Terrestrial Laser-Scanned Point Clouds Using Normal-Based Region Growing,” Journal of Computational Design and Engineering, Vol. 1, No. 1, pp. 13~26.

(비특허문헌 0002) (2) Patil, A. K., Holi, P., Lee, S. K. and Chai, Y. H., 2017, “An Adaptive Approach for the Reconstruction and Modeling of As-Built 3D Pipelines from Point Clouds,” Automation in Construction, Vol. 75, pp. 65~78.

(비특허문헌 0003) (3) Osada, R., Funkhouser, T., Chazelle, B. and Dobkin, D., 2002, “Shape Distributions,” ACM Transactions on Graphics (TOG), Vol. 21, No. 4, pp. 807~832.

(비특허문헌 0004) (4) Kazhdan, M., Funkhouser, T. and Rusinkiewicz, S., 2003, “Rotation Invariant

Spherical Harmonic Representation of 3D Shape Descriptors,", Symposium on Geometry Processing, Vol. 6, pp. 156~164.

(비특허문헌 0005) (5) Bronstein, A. M., Bronstein, M. M., Guibas, L. J. and Ovsjanikov, M., 2011, "Shape Google: Geometric Words and Expressions for Invariant Shape Retrieval,", ACM Transactions on Graphics (TOG), Vol. 30, No. 1, pp. 1~20.

(비특허문헌 0006) (6) Knopp, J., Prasad, M., Willems, G., Timofte, R. and Van Gool, L., 2010, "Hough Transform and 3D SURF for Robust Three Dimensional Classification," European Conference Computer Vision, Berlin, Heidelberg, Germany, pp.5(on)89~602.

(비특허문헌 0007) (7) Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang,

(비특허문헌 0008) X. and Xiao, J., 2015, "3D ShapeNets: A Deep Representation for Volumetric Shapes," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1912~1920.

(비특허문헌 0009) (8) Maturana, D. and Scherer, S., 2015, "VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition," Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 922~928.

(비특허문헌 0010) B. C., 2018, "Shape Recognition of Plant Equipment from 3-D Scanned Point Cloud Data Using a Convolutional Neural Network,", Trans. Korean Soc. Mech. Eng. A, Vol. 42, No. 9, pp. 863~869.

(비특허문헌 0011) (10) Su, H., Maji, S., Kalogerakis, E. and Learned-Miller, E., 2015, "Multi-view Convolutional Neural Networks for 3D Shape Recognition," Proceedings of the IEEE International Conference on Computer Vision, pp. 945~953.

(비특허문헌 0012) (11) Qi, C. R., Su, H., Nieβner, M., Dai, A., Yan, M. and Guibas, L. J., 2016, "Volumetric and Multi-view CNNs for Object Classification on 3D Data,", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5648~5656.

(비특허문헌 0013) (12) Sfikas, K., Pratikakis, I. and Theoharis, T., 2018, "Ensemble of Panorama-based Convolutional Neural Networks for 3D Model Classification and Retrieval," Computers & Graphics, Vol. 71, pp. 208~218.

(비특허문헌 0014) (13) Kanezaki, A., Matsushita, Y. and Nishida, Y., 2018, "RotationNet: Joint Object Categorization and Pose Estimation Using Multiviews from Unsupervised Viewpoints," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5010~5019.

(비특허문헌 0015) (14) Garcia-Garcia, A., Gomez-Donoso, F., Garcia-Rodriguez, J., Orts-Escolano, S., Cazorla, M. and Azorin-Lopez, J., 2016, "PointNet: A 3D Convolutional Neural Network for Real-Time Object Class Recognition," 2016 International Joint Conference on Neural Networks (IJCNN), IEEE, pp. 1578~1584.

(비특허문헌 0016) (15) Qi, C. R., Yi, L., Su, H. and Guibas, L. J., 2017, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," Proceedings of the International Conference on Neural Information Processing Systems, pp. 5105~5114.

(비특허문헌 0017) (16) Simonyan, K. and Zisserman, A., 2015, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada.

(비특허문헌 0018) (17) Mikołajczyk, A. and Grochowski, M., 2018, "Data Augmentation for Improving Deep Learning in Image Classification Problem,", 2018 International Interdisciplinary PhD Workshop (IIPhDW), IEEE, pp. 117~122.

(비특허문헌 0019) (18) Chen, Z., Zhang, T., Cao, J., Zhang, Y. J. and Wang, C., 2018, "Point Cloud Resampling Using Centroidal Voronoi Tessellation Methods," Computer-Aided Design, Vol. 102, pp.

12~21.

(비특허문헌 0020) (19) Cheng, X., Zeng, M., Lin, J., Wu, Z. and Liu, X., 2019, "Efficient L0 Resampling of Point Sets," Computer-Aided Geometric Design, Vol. 75, Paper No. 101790.

(비특허문헌 0021) (20) Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, Vol. 15, No. 1, pp. 1929~1958.

(비특허문헌 0022) (21) Kingma, D. P. and Ba, J., 2015, "A Method for Stochastic Optimization," Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada.

(비특허문헌 0023) (22) Glorot, X and Bengio, Y, 2010, "Understanding

(비특허문헌 0024) the Difficulty of Training Deep Feedforward Neural Networks," Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp 249~256.

발명의 내용

해결하려는 과제

[0012] 본 실시예의 일 측면에 의하면, 부품의 3차원 데이터를 자동으로 분류하기 위해, 부품의 3차원 데이터를 점(point)으로 샘플링(sampling)하고, 샘플링된 점의 거리를 연산하여 히스토그램(histogram)으로 표현함으로써 1차원 데이터인 형상 분산(shape distribution)화 한 이후 형상 분사화된 데이터에 대응하는 부품의 종류로 라벨링(labeling)함으로써 트레이닝 셋(training set)을 추출하고, 신경망(neural network) 기반의 부품 분류모델(part clustering model)을 학습시킴으로써 부품의 3차원 데이터를 자동으로 분류하게 하는 데 주된 목적이 있다.

[0013] 나아가, 본 실시예가 제공하는 분류모델 학습방법은 신경망 기반의 부품 분류모델을 1차원 데이터에 적합하게 변경한 VGG(Visual Geometry Group) 모델인 것을 특징으로 하여, 3차원 데이터를 학습에 이용하는 것 대비 다루어야 하는 전체 데이터 및 신경망의 크기가 커지는 문제점을 해결하여 학습 시간 및 소요되는 메모리의 크기를 감소시키는 데 다른 목적이 있다.

과제의 해결 수단

[0014] 본 실시예의 일 측면에 의하면, 복수의 부품의 3차원 데이터를 일정 개수의 점으로 각각 샘플링(sampling)하여 샘플링 데이터 셋(sampling data set)을 얻는 샘플링 과정; 상기 샘플링 데이터 셋의 샘플링 데이터 내 모든 점들 간 거리(distance)를 각각 연산하여 거리값 데이터 셋(distance data set)을 얻는 거리연산 과정; 상기 거리값 데이터 셋을 히스토그램(histogram)으로 각각 표현하여 형상분산 데이터 셋(shape distribution data set)을 얻는 형상 분산화 과정; 상기 형상 분산 데이터 셋을 부품의 종류로서 각각 라벨링(labeling)하여 트레이닝 셋(training set)을 추출하는 라벨링과정; 및 상기 트레이닝 셋을 이용하여 신경망(neural network) 기반의 부품 분류모델(part clustering model)을 트레이닝(training)하는 트레이닝 과정 을 포함하는 것을 특징으로 하는 부품의 분류모델 학습방법을 제공한다.

[0015] 본 실시예의 다른 측면에 의하면, 본 실시예에 따른 부품의 분류모델 학습방법으로 기 학습된 분류모델을 이용하여 부품의 3차원 데이터를 분류하는 것을 특징으로 하는 부품 분류방법을 제공한다.

[0016] 본 실시예의 다른 측면에 의하면, 본 실시예에 따른 부품 분류모델 학습방법 또는 분류방법이 포함하는 각 과정을 실행시키기 위하여 컴퓨터로 읽을 수 있는 기록매체에 저장된 컴퓨터 프로그램을 제공한다.

[0017] 본 실시예의 다른 측면에 의하면, 복수의 부품의 3차원 데이터를 일정 개수의 점으로 각각 샘플링(sampling)하여 샘플링 데이터 셋(sampling data set)을 얻는 샘플링부(sampling unit); 상기 샘플링 데이터 셋의 샘플링 데이터 내 모든 점들 간 거리(distance)를 각각 연산하여 거리값 데이터 셋을 얻는 거리연산부(distance calculator); 상기 거리값 데이터 셋을 히스토그램(histogram)으로 각각 표현하여 형상 분산 데이터 셋(shape distribution data set)을 얻는 형상 분산부(shape distribution unit); 상기 형상 분산 데이터 셋을 부품의 종류로서 각각 라벨링(labeling)하여 트레이닝 셋(training set)을 추출하는 라벨링부(labeling unit); 및

[0018] 상기 트레이닝 셋을 이용하여 신경망(Neural Network) 기반의 부품 분류모델(part clustering model)을 트레이닝(training)하는 트레이닝부(training unit)를 포함하는 것을 특징으로 하는 부품의 분류모델 학습장치를 제공한다.

발명의 효과

[0019] 이상에서 설명한 바와 같이 본 실시예에 의하면, 신경망 기반의 분류모델을 학습시켜 부품의 3차원 데이터를 자동으로 분류하게 하는 효과가 있다.

[0020] 뿐만 아니라, 3차원 데이터를 형상 분산이라는 1차원 데이터로 차원 축소시킨 후 신경망 기반의 분류모델을 학습시킴으로써 분류모델의 학습에 소요되는 메모리 및 시간을 종래 대비 감축시키는 효과가 있다.

[0021] 이러한 형상 분산 기반의 학습은 3차원 모델의 크기, 위치 및 자세에 불변하므로 데이터 강화(data augmentation)가 별도로 필요하지 않다는 장점이 있다.

도면의 간단한 설명

[0022] 도 1은 본 실시예에 따른 신경망 모델에 이용하기 위한 3차원 데이터의 표현방법의 예시도이다.

도 2는 본 실시예에 따른 부품의 분류모델 학습장치를 나타내는 블록구성도이다.

도 3은 본 실시예에 따른 부품의 3차원 데이터를 점군(point cloud)으로 표현하는 예시도이다.

도 4는 본 실시예에 따른 점군으로 표현된 3차원 데이터를 히스토그램화하여 형상 분산 데이터를 얻는 예시도이다.

도 5는 본 실시예에 따른 신경망 기반의 부품 분류모델의 예시도이다.

도 6은 본 실시예에 따라 학습된 부품 분류모델의 검증 결과를 나타내는 그래프이다.

도 7은 본 실시예의 학습방법에 따라 기 학습된 부품의 분류모델을 이용하여 3차원 데이터를 분류하는 방법을 나타내는 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0023] 이하, 본 발명의 일부 실시예들을 예시적인 도면을 통해 상세하게 설명한다. 각 도면의 구성요소들에 참조부호를 부가함에 있어서, 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가지도록 하고 있음에 유의해야 한다. 또한, 본 발명을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.

[0024] 또한, 본 발명의 구성 요소를 설명하는 데 있어서, 제 1, 제 2 등의 용어를 사용할 수 있다. 이러한 용어는 그 구성 요소를 다른 구성 요소와 구별하기 위한 것일 뿐, 그 용어에 의해 해당 구성 요소의 본질이나 차례 또는 순서 등이 한정되지 않는다. 명세서 전체에서, 어떤 부분이 어떤 구성요소를 '포함', '구비'한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 '…부', '모듈' 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0025] 도 2는 본 실시예에 따른 부품의 분류모델 학습장치(200)를 나타내는 블록구성도이다.

[0026] 본 실시예에 따른 부품의 분류모델 학습장치(200)는 샘플링부(sampling unit, 210), 거리연산부(distance calculation unit, 220), 형상 분산부(shape distribution unit, 240), 라벨링부(labeling unit, 260), 및 트레이닝부(training unit, 280)을 전부 또는 일부 포함한다. 도 2에 도시된 부품의 분류모델 학습장치(200)는 일 실시예에 따른 것으로서, 도 2에 도시된 모든 블록이 필수 구성요소는 아니며, 다른 실시예에서 부품의 분류모델 학습장치(200)에 포함된 일부 블록이 추가, 변경 또는 삭제될 수 있다. 예컨대, 부품의 분류모델 학습장치(200)의 트레이닝된 결과를 검증하는 검증부(미도시)가 등이 추가될 수 있다.

[0027] 부품의 분류모델 학습장치(200)의 전처리(pre-processing)로는, 부품의 3차원 데이터를 점으로 샘플링(sampling)하고 샘플링된 점들 간 거리값을 연산하여 연산된 거리값을 기초로 히스토그램으로 나타내는 형상 분산(shape distribution) 데이터를 생성한다. 이렇게 생성된 형상 분산 데이터를 부품의 종류로 라벨링(labeling)하여 신경망(neural network) 기반의 부품 분류모델을 학습시킨다. 이하에서는 부품의 분류모델 학습

장치(200)에 포함된 각각의 구성요소에 대해 설명하도록 한다.

[0028] 샘플링부(210)는 부품의 3차원 데이터를 각각 일정 개수의 점(point)으로 샘플링한다. 점의 개수는 부품 분류모델의 성능에 따라 달라질 수 있다. 예컨대 점의 개수는 100,000(비특허문헌 17 참조)개일 수 있으나, 형태가 복잡하고 정교한 부품의 데이터를 분류하기 위해서는 샘플링되는 점의 개수가 이보다 많을 수 있다. 3차원 데이터를 보다 정교하게 샘플링하기 위해, 3차원 모델을 삼각화(triangulation)시키고 삼각형의 면적에 비례하여 샘플링할 수 있다. 또한, 점들이 균일하게 샘플링되도록 하기 위해, 샘플링 후 재표본(resampling) 방법(비특허문헌 18, 19 참조)을 수행할 수 있다. 예컨대, CVT(Centroidal Voronoi Tessellation) 기반의 재표본을 수행할 수 있다. CVT는 보로노이 셀(Voronoi cell)의 중심에 점을 위치시키는 것으로, 보로노이 셀은 특정 점까지의 거리가 가장 가까운 점의 집합으로 평면을 분할한 셀이다. 부품의 3차원 데이터를 점으로 샘플링한 예시는 도 3에 나타나 있다.

[0029] 샘플링부(220)는 샘플링 데이터 셋의 샘플링된 모든 점들 간 거리(distance)를 각각 연산하여 샘플링 데이터마다의 거리값 데이터를 얻는다. 이렇게 얻은 거리값 데이터를 바탕으로, 형상 분산(shape distribution)의 생성에 이용되는 거리값 데이터 셋을 얻는다.

[0030] 형상 샘플링부(240)는 샘플링부(220)에서 얻은 거리값 데이터 셋을 히스토그램으로 표현하는 형상 분산 데이터 셋을 만든다. 히스토그램(histogram)을 만들기 위하여, 거리값 데이터 셋의 각 거리값 데이터들간 최소 거리값과 최대 거리값의 차(difference)를 B개의 고정된 개수의 빈(bin)으로 나누어, 각 빈에 해당하는 거리값의 개수를 세어 빈을 채우는 방법을 이용할 수 있다.

[0031] 히스토그램 빈의 개수에 따라 히스토그램의 가로축 밀도가 달라지므로, 빈의 개수가 클수록 더욱 정밀한 형상 분산 표현이 가능하다. 거리값 데이터 셋을 형상 분산 데이터로 표현하는 예시는 도 4에 나타나 있다.

[0032] 3차원 모델을 이러한 형상 분산 데이터로 표현하면 히스토그램은 3차원 모델의 크기(size), 위치(position) 및 자세(posture)에 대해 불변하므로, 데이터 강화(data augmentation)(비특허문헌 17 참조)가 별도로 필요하지 않는 장점이 있다.

[0033] 라벨링부(260)는 형상 분산 데이터 셋을 대응하는 부품의 종류를 기초로 라벨링(labeling)하여 부품 분류 모델을 트레이닝시킬 트레이닝 셋(training set)을 추출한다. 라벨링이란 트레이닝 셋의 트레이닝 데이터(training data)에 정답값(target value)을 부여하는 것이다. 라벨링은 반드시 라벨링부(260)에서 이루어지는 것은 아니고, 샘플링부(210), 샘플링부(220), 형상 샘플링부(240)에서 각각 이루어질 수도 있다. 또한 샘플링 전의 3차원 데이터 또는 원시 데이터(raw data)에 이미 부품의 종류에 대한 라벨(label)이 주어진 것일 수 있고, 이 경우 이미 라벨링이 수행된 것으로 보아야 한다.

[0034] 트레이닝부(26)는 라벨링부(24)에서 추출한 트레이닝 셋을 이용하여 부품의 분류모델(part clustering model)을 학습시킨다. 부품의 분류모델의 성능을 향상시키기 위해서, 트레이닝부(26)는 홀드아웃(holdout) 방법을 사용하여 라벨링된 데이터 셋을 트레이닝 셋과 테스트 셋(test set)으로 나누어, 트레이닝 셋을 이용하여 학습시킨 분류모델의 성능을 테스트 셋을 이용하여 평가할 수 있다. 모델의 예측값과 실제값 사이의 차이인 손실함수(loss function)의 값을 최소화하기 위하여 예측값과 실제값 차이 연산에 기울기를 이용하는 그레디언트 하강(gradient descent)을 이용할 수 있다. 또한, 모델의 성능을 높이고자 트레이닝 셋을 계층 샘플링(stratified sampling)을 수행하여 트레이닝 셋 내 라벨링이 동일한 데이터 셋들의 집합인 각 클래스(class)를 균등하게 분포시킬 수 있고, 과적합(overfitting)을 방지하기 위하여 랜덤 서브샘플링(random subsampling)을 수행할 수 있으며, k-겹 교차 검증(k-fold cross validation)을 이용하여 학습시킬 수 있다.

[0035] 본 실시예에 따른 부품 분류모델은 신경망을 기반으로 하고 있다. 구체적으로 이미지 또는 형상 처리에 주로 사용되는 합성곱 신경망(CNN; Convolution Neural Network) 기반의 분류모델일 수 있다. 이러한 합성곱 신경망 기반의 분류모델은 VGG(Visual Geometry Group)-16 또는 VGG-19를(비특허문헌 16 참조) 1차원 데이터에 적합하게 변형한 것일 수 있다. 변형된 VGG-16 또는 VGG-19 모델은 도 5에 나타나 있다.

[0036] 본 실시예에 따른 부품 분류모델의 입력 노드에는 형상 분산 데이터가 제공되므로, 입력 노드의 크기는 히스토그램 빈의 개수와 동일하게 할 수 있다. 또한 추가적인 정보를 이용하지 않으므로 한 개의 채널로 구성할 수 있다. 본 실시예에 따른 부품 분류모델의 출력 노드는 부품 종류의 개수와 동일한 개수의 노드로 구성할 수 있다.

[0037] 부품의 분류모델 학습장치(200)를 이용하여 학습시킨 부품 분류 모델의 검증은 도 6에서 후술한다.

[0038] 단, 본 실시예는 독립된 부품에 대한 점군 데이터를 대상으로 하였으나, 반드시 이에 한정되는 것은 아니고 장

치(예: 플랜트) 전체의 점군 데이터부터 각 부품을 검출하는 것도 가능하다.

[0039] 도 3은 본 실시예에 따른 부품의 3차원 데이터를 점군(point cloud)으로 표현하는 예시도이다.

[0040] 도 3의 (a)는 45도 엘보우(45-degree elbow) 부품의 3차원 데이터를 점으로 샘플링한 예시도이다. 각 점은 3차원 형상의 표면뿐 아니라 내측까지 표현하도록 고루 분포되어 있다. 도 3의 (b)는 체크 벨브(check valve) 부품의 3차원 데이터를 점으로 샘플링한 예시도이다. 도 3의 (c)는 티(tee) 부품의 3차원 데이터를 점으로 샘플링한 예시도이다.

[0041] 도 4는 본 실시예에 따른 점군으로 표현된 3차원 데이터를 히스토그램화하여 얻은 형상 분산 데이터를 나타내는 예시도이다.

[0042] 도 4a의 (a)는 45도 엘보우의 형상 분산을, (b)는 체크 벨브의 형상 분산을 나타내고 있다. 이때 히스토그램의 빈의 수가 각각 512, 1,024개로 달라져도 전체적인 형상은 크게 달라지지 않음을 알 수 있다. 그러나 빈의 개수가 많아질수록 히스토그램의 가로축 간격이 줄어들어 세로축 값 또한 줄어들게 되어 보다 정교한 형상 분산을 얻을 수 있다.

[0043] 도 4b는 형태가 유사한 부품들의 형상 분산을 나타내고 있다. 도 4b의 (a)는 45도 엘보우의 형상 분산을, (b)는 90도 엘보우의 형상 분산을 나타내며 양 부품의 형상이 유사한 만큼 형상 분산의 전체적인 형상 또한 유사함을 알 수 있다. 부품 분류모델의 정확도를 높이기 위해 유사한 형태의 부품들의 형상 분산간 차이가 발생하도록 적절한 수의 점으로 샘플링하고 적절한 수의 빈으로 히스토그램화해야 한다.

[0044] 도 5는 본 실시예에 따른 신경망 기반의 부품 분류모델의 예시도이다.

[0045] 도 5의 (a)는 본 실시예에 따른 VGG-16 기반의 부품 분류모델, (b)는 본 실시예에 따른 VGG-19 기반의 부품 분류모델이다. VGG-16과 VGG-19의 차이는 모델에 적용되는 레이어(layer)의 수로, VGG-16에는 16개의 레이어가 적용되고 VGG-19에는 19개의 레이어가 적용된다. VGG-19에는 VGG-16보다 합성곱 레이어가 세 개 더 포함된 것이다. 도 5의 (b)에는 5의 (a)와 비교하여 ConvID(256)+ReLU 및 ConvID(512)+ReLU 두 개가 더 포함되어 있음을 알 수 있다.

[0046] 도 5에서 ConvID로 표현되는 합성곱 레이어(convolutional layer)는 크기가 25인 1차원 필터를 가지며, 스트라이드 (stride)는 1, 패딩(padding)은 입력과 출력의 크기가 동일하도록 설정되었다. ConvID 괄호에 기재된 숫자는 레이어에 적용되는 채널(channel)의 수이다. 본 합성곱 레이어에서는 1차원 필터를 사용하므로 필터는 별도로 표시하지 않았다. 합성곱 레이어의 채널 수는 64 개부터 시작하여 최대값 풀링 레이어를 지나갈 때마다 두 배로 크게 증가시켰다.

[0047] 또한, 각 합성곱 레이어 다음에 ReLU 활성화 함수(activation function)를 적용하였다. ReLU는 도 5에 ReLU로 표현되어 있다.

[0048] 도 5에서 MaxPoolingID로 표현되는 최대값 풀링 레이어(max pooling layer)는 크기가 2인 1차원 필터를 가지며, 스트라이드는 2로 설정된 것이다. 최대 풀링 레이어 다음에는 드롭아웃(dropout)(비특허문헌 20 참조)을 적용하여 과적합(overfitting)을 방지하였다. 구체적으로 드롭아웃 비율은 25 %일 수 있다(비특허문헌 20 참조). 드롭아웃은 도 5에 Dropout으로 표현되어 있다.

[0049] 도 5에서 Fn으로 표현되는 완전 연결 레이어(fully connected layer)는 512 개 또는 28 개의 노드로 각각 구성되며, 512 개 노드로 구성된 두개의 완전 연결 레이어에는 ReLU 활성화 함수와 50 % 의 드롭아웃이 적용되었다.

[0050] 도 5의 (a), (b) 모델의 가장 하단에는 소프트맥스 레이어(softmax layer)가 적용됨을 알 수 있다. 소프트맥스 레이어의 출력값은 형상 분산이 각 부품 종류일 확률을 나타내므로, 소프트맥스 레이어가 확률이 가장 큰 값을 가지는 출력노드를 인식시키도록 하였다.

[0051] 단, 도 5에 도시된 모델은 하나의 예시에 불과한 것으로 반드시 전술한 모델에 한정되지 않고, 각 레이어 또는 함수의 적용여부는 필요에 따라 생략, 변경 또는 추가될 수 있다. 예컨대, 모델의 가중치(weight)를 조정하기 위하여 정규화 레이어(normalization layer) 등이 추가될 수 있다.

[0052] 도 6은 본 실시예에 따라 학습된 부품 분류모델의 검증 결과를 나타내는 그래프이다.

[0053] 도 6의 검증과정은, 구글의 텐서플로우(TensorFlow)를 백엔드(backend)로 이용한 케라스(Keras)를 이용한 파이선(python) 환경에서 도 5에 도시된 부품의 분류모델을 구현하고 동일한 환경에서 부품의 분류모델 학습 및 실

험을 수행함으로써 얻은 것이다. 학습에 사용된 컴퓨터는 인텔 코어 i9-7900X CPU, 64 GB RAM 과 네 개의 NVidia Titan Xp 그래픽 카드를 장착하였다.

[0054] 본 검증은 28 개의 부품 종류로 라벨링된 1046 개의 데이터 셋을 기반으로 이루어졌다. 1,046 개의 데이터 셋을 704 개(70 %)의 트레이닝 셋과 128(12 %) 개의 검증 데이터 셋(validation set) 및 192 개(18 %) 의 테스트 데이터 셋으로 랜덤하게 분리하였고, 이중 트레이닝 셋을 이용하여 부품 분류모델을 트레이닝시켰다. 트레이닝 후 검증 데이터 셋을 이용해 하이퍼파라미터(hyperparameter)를 변경해가며 수십 번의 실험을 진행하였고, 가장 좋은 정확도가 나온 하이퍼파라미터를 선정하였다. 부품 분류모델 학습에 사용된 방법 및 하이퍼파라미터들은 다음과 같다. SGD(Stochastic Gradient Descent)를 사용해 32 개의 미니배치(mini-batch) 단위로 학습시키고, 1,000 에폭(epoch)을 수행하였다. 학습률(learning rate)은 0.001로 초기화하였으며, 아담(adam) 최적화(비특허문헌 21 참조)(모멘텀 파라미터로 β_1 = 0.9, β_2 = 0.999, ϵ = 10^{-8}를 적용)를 수행하였다. 신경망 가중치는 Xavier 초기화 방법(비특허문헌 22 참조)을 적용하였고, 바이어스(bias)들은 0으로 초기화하였다.

[0055] 도 6의 (a)는 트레이닝 및 검증 결과를 보여준다. 연산 시간은 입력 크기가 가장 큰 경우인 2,048 개에 대해서 약 6,941 초(1.9 시간)가 소요되었다. 비교를 위해 Kim의 방법(비특허문헌 9 참조)을 이용하여 복셀(voxel)을 이용한 트레이닝을 하였으며, 이 경우 약 88 시간이 소요되었다. 따라서, 형상 분산을 이용할 경우 학습 속도가 비약적으로 향상되는 것을 알 수 있다. 그러나, 형상 분산을 이용할 경우, 도 6의 (a)에서 볼 수 있듯이 학습 정확도는 96~98 % 수준이며, 검증 정확도는 87 % ~ 91 % 수준이다. 반면에, 복셀을 이용한 경우에는 학습 정확도는 형상 분산을 이용하는 경우와 유사하나, 검증 정확도가 약 97.16 % 였다.

[0056] 도 6의 (b)는 트레이닝 셋과 검증 데이터 셋으로 부품 분류모델을 학습시킨 후, 192 개의 테스트 셋을 이용하여 성능을 평가한 그래프이다. 도 6의 (b)에서 볼 수 있듯이, VGG-19 모델의 입력 크기가 2,048인 경우의 정확도가 94.79 % 로 가장 높음을 알 수 있다. 정확도가 가장 높은 VGG-19, 빈 크기 2048인 경우에 대해 오차 행렬(confusion matrix) 계산시, 본 실시예에 따른 부품 분류모델이 부품 형상을 잘 구분 하고 있음을 알 수 있었다. 잘못 분류하는 경우는 부품의 형상이 서로 유사한 경우였다. 다만, 복셀을 이용한 경우의 정확도는 97.29 % 였다.

[0057] 따라서, 도 6의 검증 결과를 통하여, 형상 분산을 이용하는 경우가 복셀을 이용하는 경우보다 정확도가 다소 떨어지나, 이는 부품 분류를 수행하기에는 나쁘지 않은 결과이며, 본 실시예에 따른 학습방법을 이용하여 학습 시 학습 시간의 향상이 현저함을 확인할 수 있다.

[0058] 도 7은 본 실시예의 학습방법에 따라 기 학습된 부품의 분류모델을 이용하여 3차원 데이터를 분류하는 방법을 나타내는 순서도이다.

[0059] 샘플링부(210)는 복수의 부품의 3차원 데이터 각각을 N개(N은 자연수)의 점으로 샘플링한다(S700). 보다 균일하게 샘플링하기 위해 샘플링 후 재표본 방법을 수행 할 수 있다.

[0060] 샘플링부(220)는 샘플링된 데이터마다 샘플링된 모든 점들 간 거리를 각각 연산하여 거리값 데이터 셋을 얻는다(S710).

[0061] 형상 샘플링부(240)는 거리값 데이터 셋을 히스토그램으로 각각 표현하여 형상 분산 데이터 셋을 얻는다(S720). 히스토그램으로 표현하는 경우 히스토그램의 빈의 개수는 분류 대상에 따라 달라질 수 있다.

[0062] 라벨링부(260)는 형상 분산 데이터 셋을 대응하는 부품의 종류로 각각 라벨링하고, 라벨링된 형상 분산 데이터 셋으로부터 트레이닝 셋을 추출한다(S740). 단, 라벨링은 S740 이전 단계(S720 내지 S700) 중 어느 한 단계에서 수행되거나 부품의 3차원 모델에 부여된 정보로서 이미 수행된 것일 수 있다.

[0063] 트레이닝부(280)는 추출된 트레이닝 셋을 이용하여 부품의 분류모델을 학습시킨다(S760).

[0064] 부품의 분류장치(미도시)는 학습된 부품 분류모델을 기반으로 부품의 분류를 수행한다(S780).

[0065] 도 7에서는 과정 각 과정들을 순차적으로 실행하는 것으로 기재하고 있으나, 이는 본 발명의 일 실시예의 기술사상을 예시적으로 설명한 것에 불과한 것이다. 다시 말해, 본 발명의 일 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 일 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 도 7에 기재된 순서를 변경하여 실행하거나 각 과정 중 하나 이상의 과정을 병렬적으로 실행하는 것으로 다양하게 수정 및 변형하여 적용 가능할 것이므로, 도 7의 시계열적인 순서로 한정되는 것은 아니다.

[0066] 한편, 전술한 바와 같이 도 2에 기재된 본 실시예에 따른 부품의 분류모델 학습장치(200)의 샘플링부(sampling

unit, 20), 거리연산부(distance calculation unit, 22), 형상 분산부(shape distribution unit, 24), 라벨링부(labeling unit), 및 트레이닝부(training unit, 28)의 동작은 프로그램으로 구현되고 컴퓨터로 읽을 수 있는 기록매체에 기록될 수 있다. 본 실시예에 따른 부품의 분류모델 학습장치(200)의 샘플링부(sampling unit, 20), 거리연산부(distance calculation unit, 22), 형상 분산부(shape distribution unit, 24), 라벨링부(labeling unit), 및 트레이닝부(training unit, 28)의 동작을 구현하기 위한 프로그램이 기록되고 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록장치를 포함한다. 이러한 컴퓨터가 읽을 수 있는 기록매체는 ROM, RAM, CD-ROM, 자기 테이프, 플로피디스크, 광 데이터 저장장치 등의 비 일시적인(non-transitory) 매체일 수 있으며, 또한 캐리어 웨이브(예를 들어, 인터넷을 통한 전송) 및 데이터 전송 매체(data transmission medium)와 같은 일시적인(transitory) 매체를 더 포함할 수도 있다. 또한, 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수도 있다.

[0067] 본 명세서에 설명되는 장치, 및 기법들의 다양한 구현예들은, 디지털 전자 회로, 집적 회로, FPGA(field programmable gate array), ASIC(application specific integrated circuit), 컴퓨터 하드웨어, 펌웨어, 소프트웨어, 및/또는 이들의 조합으로 실현될 수 있다. 이러한 다양한 구현예들은 프로그래밍가능 시스템 상에서 실행 가능한 하나 이상의 컴퓨터 프로그램들로 구현되는 것을 포함할 수 있다. 프로그래밍가능 시스템은, 저장 시스템, 적어도 하나의 입력 디바이스, 그리고 적어도 하나의 출력 디바이스로부터 데이터 및 명령들을 수신하고 이들에게 데이터 및 명령들을 전송하도록 결합되는 적어도 하나의 프로그래밍가능 프로세서(이것은 특수 목적 프로세서일 수 있거나 혹은 범용 프로세서일 수 있음)를 포함한다. 컴퓨터 프로그램들(이것은 또한 프로그램들, 소프트웨어, 소프트웨어 애플리케이션들 혹은 코드로서 알려져 있음)은 프로그래밍가능 프로세서에 대한 명령어들을 포함하며 "컴퓨터가 읽을 수 있는 기록매체"에 저장된다.

[0068] 컴퓨터가 읽을 수 있는 기록매체는, 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록장치를 포함한다. 이러한 컴퓨터가 읽을 수 있는 기록매체는 ROM, CD-ROM, 자기 테이프, 플로피디스크, 메모리 카드, 하드 디스크, 광자기 디스크, 스토리지 디바이스 등의 비휘발성(non-volatile) 또는 비 일시적인(non-transitory) 매체일 수 있으며, 또한 캐리어 웨이브(예를 들어, 인터넷을 통한 전송) 및 데이터 전송 매체(data transmission medium)와 같은 일시적인(transitory) 매체를 더 포함할 수도 있다. 또한, 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수도 있다.

[0069] 본 명세서에 설명되는 시스템들 및 기법들의 다양한 구현예들은, 프로그램가능 컴퓨터에 의하여 구현될 수 있다. 여기서, 컴퓨터는 프로그램가능 프로세서, 데이터 저장 시스템(휘발성 메모리, 비휘발성 메모리, 또는 다른 종류의 저장 시스템이거나 이들의 조합을 포함함) 및 적어도 한 개의 커뮤니케이션 인터페이스를 포함한다. 예컨대, 프로그램가능 컴퓨터는 서버, 네트워크 기기, 셋탑 박스, 내장형 장치, 컴퓨터 확장 모듈, 개인용 컴퓨터, 랩탑, PDA(Personal Data Assistant), 클라우드 컴퓨팅 시스템 또는 모바일 장치 중 하나일 수 있다.

[0070] 이상의 설명은 본 실시예의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 실시예들은 본 실시예의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 실시예의 기술 사상의 범위가 한정되는 것은 아니다. 본 실시예의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 실시예의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

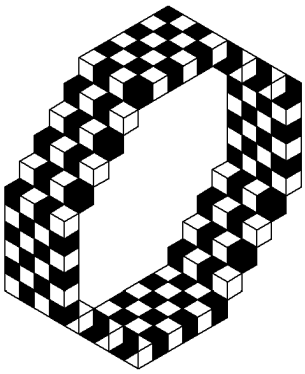
[0071] 200: 부품 분류모델 학습장치 210: 샘플링부

220: 거리연산부 240: 형상 분산부

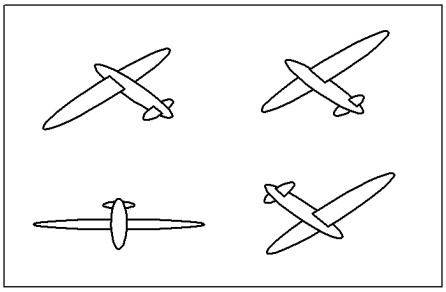
260: 라벨링부 280: 트레이닝부

도면

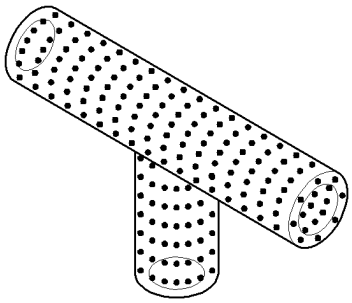
도면1



(a)

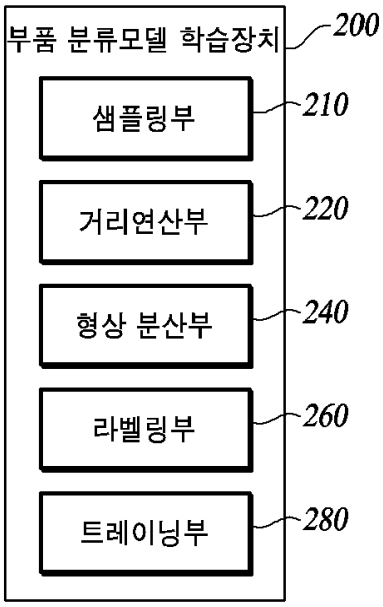


(b)

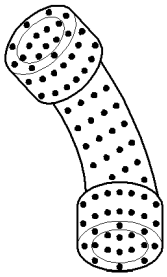


(c)

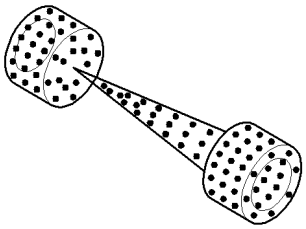
도면2


부품 분류모델 학습장치
200
샘플링부
210
거리연산부
220
형상 분산부
240
라벨링부
260
트레이닝부
280

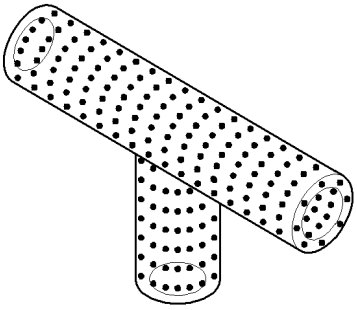
도면3



(a)

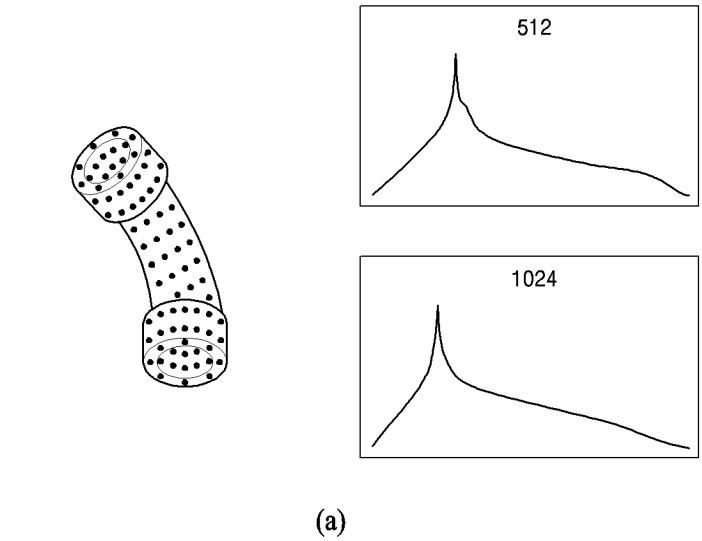


(b)

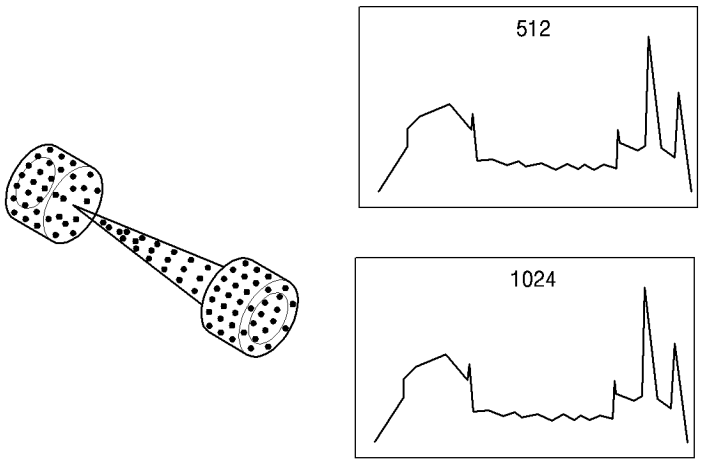


(c)

도면4a

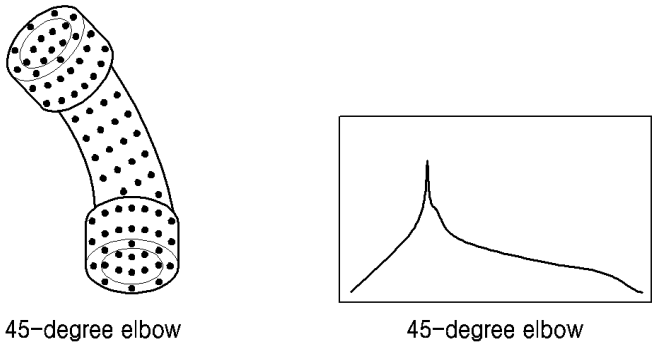


(a)

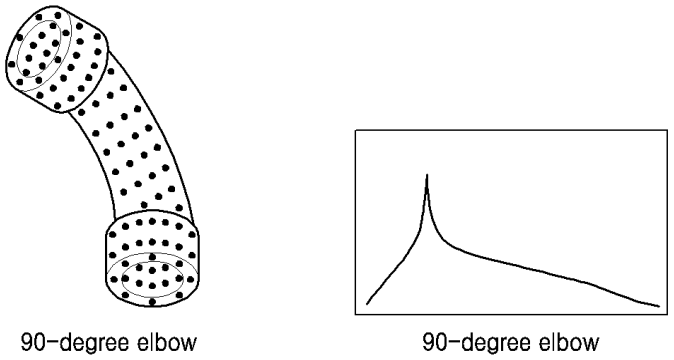


(b)

도면4b

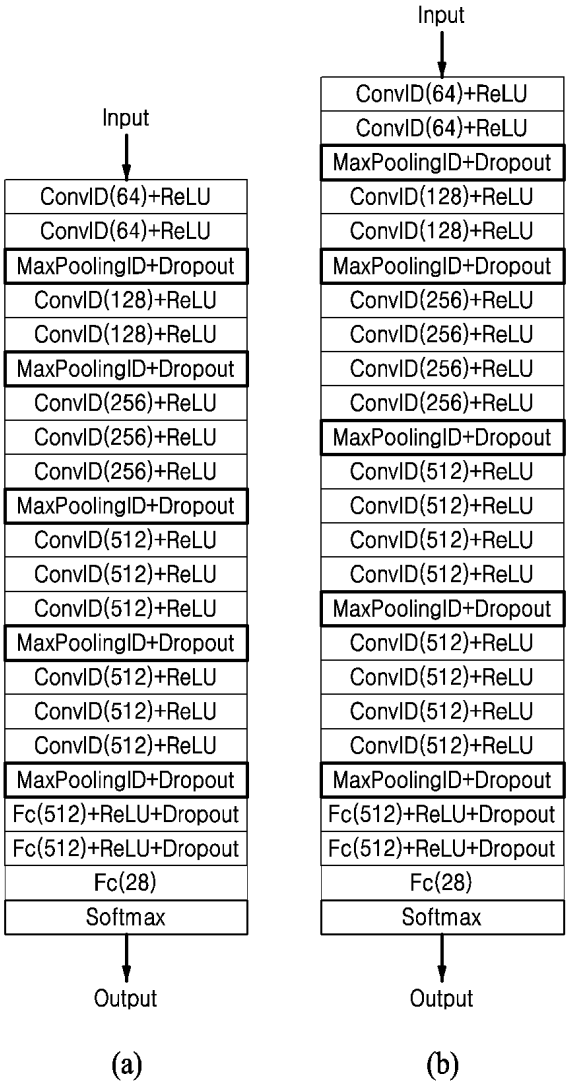

45-degree elbow
45-degree elbow

(a)

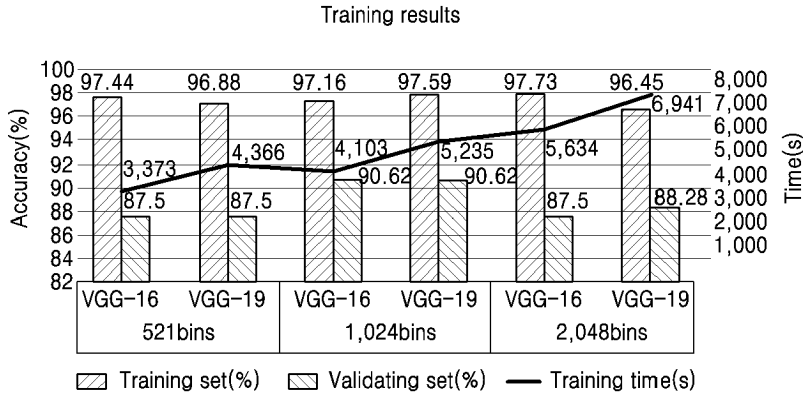

90-degree elbow
90-degree elbow

(b)

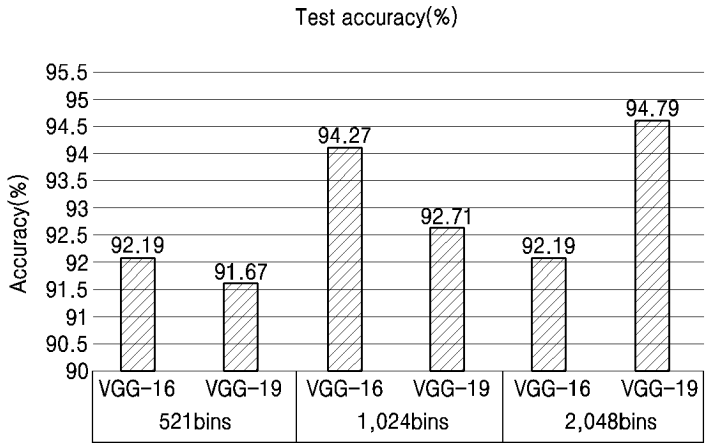
도면5


Input
ConvID(64)+ReLU
ConvID(64)+ReLU
MaxPoolingID+Dropout
ConvID(128)+ReLU
ConvID(128)+ReLU
MaxPoolingID+Dropout
ConvID(256)+ReLU
ConvID(256)+ReLU
ConvID(256)+ReLU
MaxPoolingID+Dropout
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
MaxPoolingID+Dropout
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
MaxPoolingID+Dropout
Fc(512)+ReLU+Dropout
Fc(512)+ReLU+Dropout
Fc(28)
Softmax
Output
(a)
Input
ConvID(64)+ReLU
ConvID(64)+ReLU
MaxPoolingID+Dropout
ConvID(128)+ReLU
ConvID(128)+ReLU
MaxPoolingID+Dropout
ConvID(256)+ReLU
ConvID(256)+ReLU
ConvID(256)+ReLU
ConvID(256)+ReLU
MaxPoolingID+Dropout
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
MaxPoolingID+Dropout
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
ConvID(512)+ReLU
MaxPoolingID+Dropout
Fc(512)+ReLU+Dropout
Fc(512)+ReLU+Dropout
Fc(28)
Softmax
Output
(b)

도면6

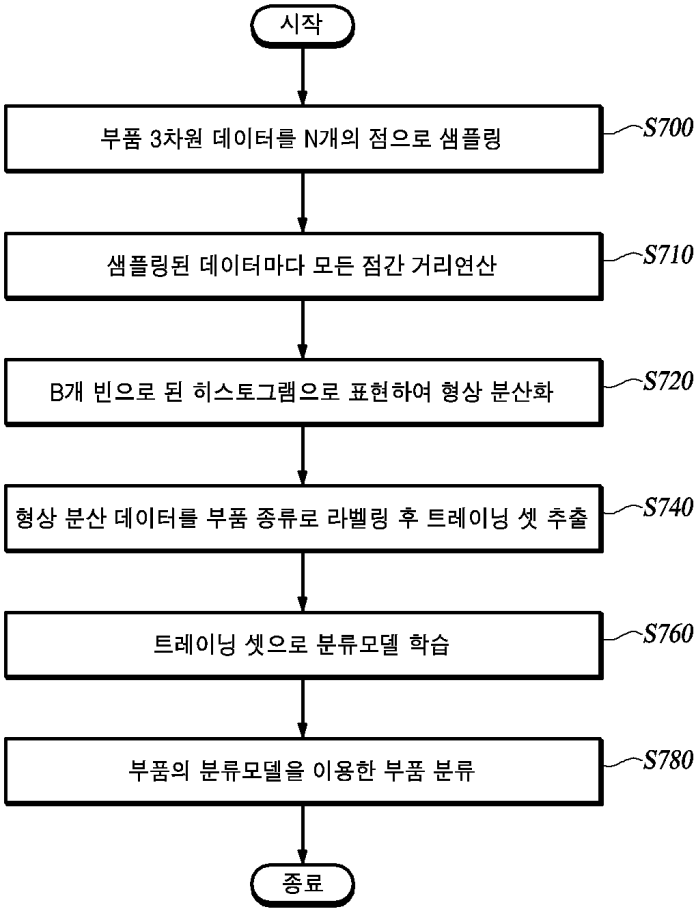

Training results
Accuracy(%)
Time(s)
97.44
96.88
97.16
97.59
97.73
96.45
3,373
4,366
4,103
5,235
5,634
6,941
87.5
87.5
90.62
90.62
87.5
88.28
VGG-16
VGG-19
521bins
1,024bins
2,048bins
Training set(%)
Validating set(%)
Training time(s)

(a)


Test accuracy(%)
Accuracy(%)
92.19
91.67
94.27
92.71
92.19
94.79
VGG-16
VGG-19
521bins
1,024bins
2,048bins

(b)

도면7


시작
부품 3차원 데이터를 N개의 점으로 샘플링
S700
샘플링된 데이터마다 모든 점간 거리연산
S710
B개 빈으로 된 히스토그램으로 표현하여 형상 분산화
S720
형상 분산 데이터를 부품 종류로 라벨링 후 트레이닝 셋 추출
S740
트레이닝 셋으로 분류모델 학습
S760
부품의 분류모델을 이용한 부품 분류
S780
종료