Convolutional Neural Networks-Based Anti-Weapon Detection for Safe 3D Printing

Giao N. Pham[1,*], Suk-Hwan Lee[2], and Ki-Ryong Kwon[3 Corresponding Author]
[*]Digital Innovation, Advance Analytics Center, FPT Software Company, Hanoi, Vietnam
[2]Dept. of Information Security, Tongmyong University, Pusan, South Korea 608-711
[1, 3]Dept. of IT Convergence & Application Engineering, Pukyong National University, Pusan, South Korea 48513

ABSTRACT

With the development of 3D printing technology anybody can print weapons with home 3D printer. In this paper, we would like to present an anti-weapon detection algorithm for safe 3D printing using the convolutional neural networks (CNNs) to prevent the printing of weapons in 3D printing industry. The proposed algorithm is based on training the D2 shape distribution of 3D weapon models by the improved CNNs. The D2 shape distribution of 3D weapon model is calculated from geometric features and points on the surface of 3D triangle mesh in order to construct a D2 vector. The D2 vector is then trained by the improved CNNs. The training and testing results show that the proposed algorithm is more accuracy than the conventional works and previous methods.

Keywords: 3D printing, 3D weapons, shape distribution, deep learning and convolutional neural networks.

1. INTRODUCTION

Three dimension (3D) printing revolution help users can realize their ideas from the digital models with 3D printers [1]. With the development of 3D printing technology, people can download 3D models on the internet and print out real objects with home 3D printers. Thus, anyone can search 3D weapons such as firearm, gun and knife to print out or share them unlimitedly. This has led to the fear of the increased violence and crime. Up to the present time, there is no solution to stop the print out of weapons in 3D printing. So, a solution to prevent the print out of weapons is necessary for 3D printing.

For meeting requirements above, we would like to propose an algorithm to detect anti-weapon for safe 3D printing. The main content of the proposed algorithm is to extract facets and vertices from 3D weapon model in order to compute the D2 shape distribution. The D2 shape distribution is used to construct a feature vector, and this data vector will be trained by a convolutional neural networks for detecting 3D weapon models. To clarify the proposed algorithm, we organized our paper as follow: in Sec. 2, we explain related works. In Sec. 3, we show the proposed algorithm in detail. Experimental results and the evaluation of the proposed algorithm will be shown in Sec. 4. Sec. 5 shows the conclusion.

2. RELATED WORKS

2.1 Shape Distribution

Shape distribution is used to represent the shape signature of a 3D model. It is a probability distribution sampled from a shape function measuring global geometric properties of the 3D model. The key idea of shape distribution is to transform an arbitrary 3D model into a parameterized function that can easily be compared with others (Fig. 1). R. Osada et al. [2] introduced shape distributions by shape functions: A3, D1, D2, D3 and D4 for 3D polygon mesh model. A3: measures the angle between three random points on the surface of a 3D model. D1: measures the distance between a fixed point and one random point on the surface. D2: measures the distance between two random points on the surface. D3: measures the square root of the area of the triangle between three random points on the surface. D4: measures the cube root of the volume of the tetrahedron between four random points on the surface.

[1]ngocgiaofet@gmail.com, [2]skylee@tu.ac.kr, [3]krkwon@pknu.ac.kr

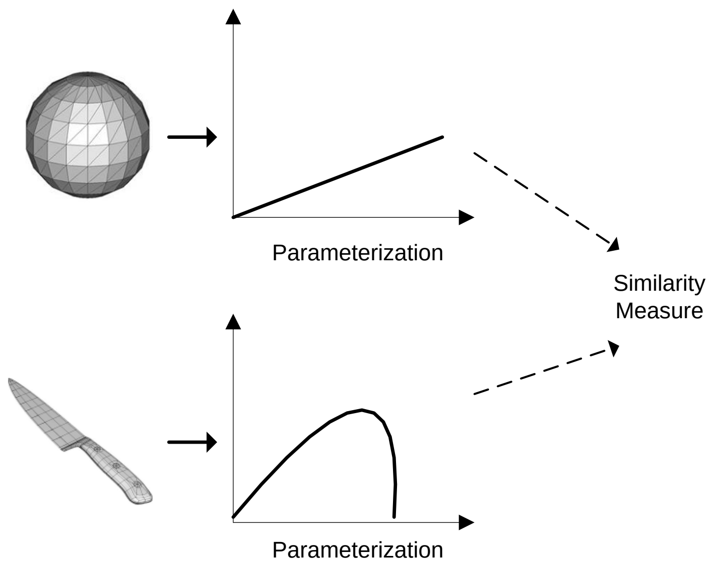

Figure 1. Shape Distribution Example.

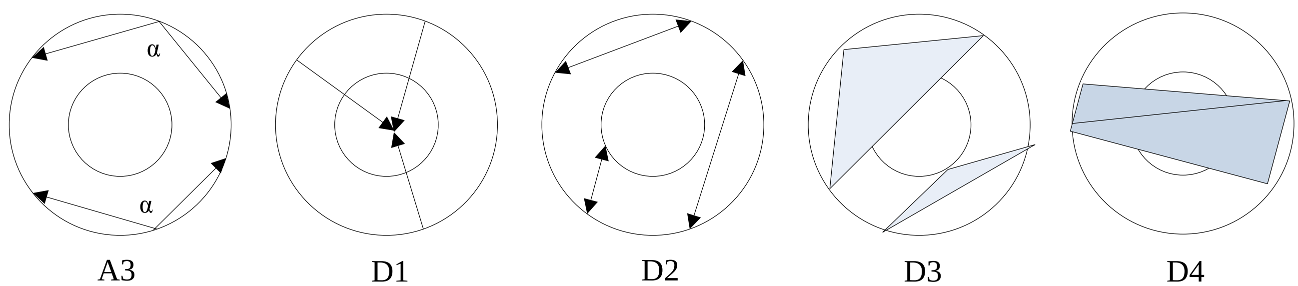

Figure 2. Shape Functions.

The experimental results [2, 3] showed that each the type of models is featured by the differential distributions and robustness with geometric attacks as rotation, scaling, translation and noise. Among the introduced shape distributions, the D2 shape distribution has the highest accuracy. Therefore, in the proposed algorithm D2 shape distribution is selected to extract and construct feature vectors from 3D triangle mesh for the training process by the convolutional neural networks.

2.2 3D Triangle Mesh

The input of 3D printing is 3D triangle mesh [4], which is designed by CAD software. A 3D triangle mesh contains a set of triangles. Each triangle includes three vertices. Each vertex is presented by three coordinates x, y and z. The purpose of the proposed algorithm is to detect anti-weapon for safe 3D printing from 3D triangle mesh. So, the target is to apply D2 shape function to 3D triangle mesh to calculate D2 shape distribution and construct D2 feature vector before training by the CNNs.

3. THE PROPOSED ALGORITHM

3.1 Overview

The proposed algorithm is shown detail in Fig. 3. Firstly, facets and vertices are extracted from 3D weapon triangle mesh in order to generate **N** pairs of two random points. The pairs of two points are random selected from points on the surface of 3D triangle mesh model including the vertices of 3D triangle mesh. The **N** pairs of two random points are then used to compute **N** distances from **N** pairs of two random points. The D2 shape distribution of 3D triangle mesh will be then computed from **N** distances in order to construct a D2 data vector for the training process by a CNNs. The dimension of

D2 vector is defined by users because the content of D2 shape distribution is the distribution of Euclidean distances between **N** pairs of randomly selected points on the surface of a 3D model. After data vector construction, D2 vector is trained by a CNNs. The structure of this CNNs consists of one convolution layer and neural networks. With the result of the training process, we can detect anti-weapon for safe 3D printing in order to anti the creation of weapons.

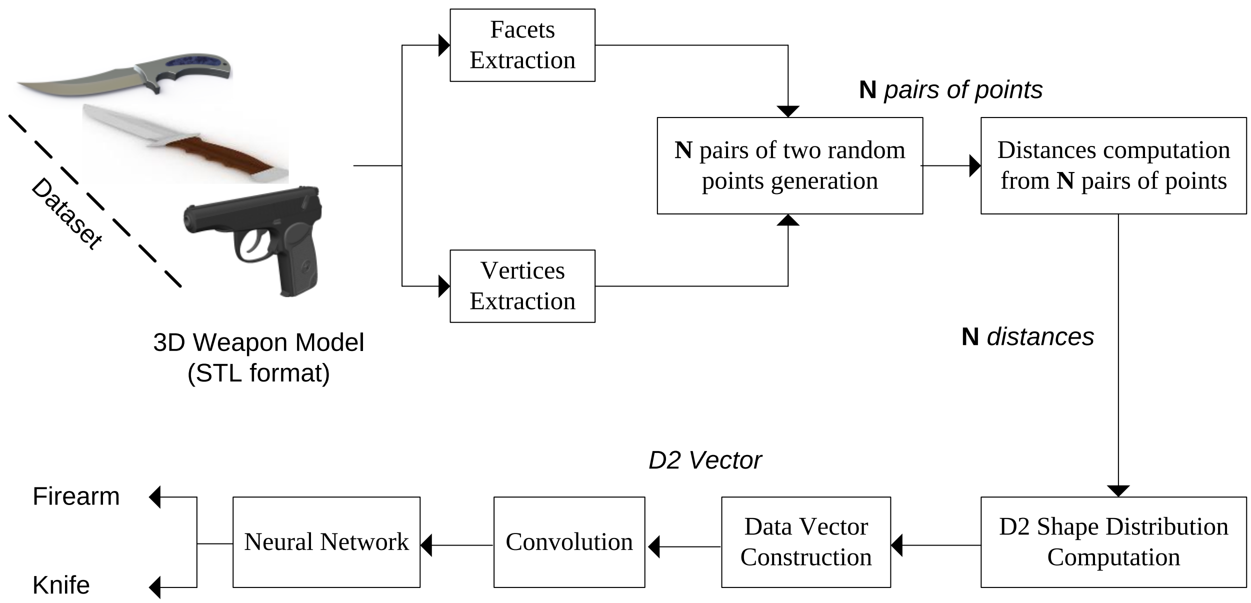

Figure 3. The proposed algorithm.

3.2 D2 Shape Distribution Computation and D2 Vector Construction

As the mention above, a 3D triangle mesh contains a set of triangles. Each triangle includes three vertices. The triangles are connected together to form a mesh of triangles. So, the number of vertices on the surface of 3D triangle mesh is always smaller than the number of facets. To brief, we consider a 3D triangle mesh $\mathbf{M} = \{\mathbf{V}, \mathbf{F}\}$ where **V** is a set of vertices; $\mathbf{V} = \{v_i \in \mathrm{R}^3 | i \in [1, N_V]\}$ and **F** is a set of facets; $\mathbf{F} = \{f_j | j \in [1, N_F]\}$ with f_j is formed from v_i. From a set of vertices and a set of facets, we obtain **N** pairs of two random points and calculate a set of distances from **N** pairs of two selected random points. Next, we find the maximum and minimum distances d_{min}, d_{max} from a set of distances above to determine the value range of distances. After finding d_{min}, d_{max}, we compute the D2 shape distribution of 3D triangle mesh by dividing the value range (d_{min}, d_{max}) into **B** bins and count the number of distances fall in each bin. Consequently, the D2 shape distribution of 3D triangle mesh is represented by a histogram of **B** bins and the value of each bin is the number of distances in that bin.

After the D2 shape distribution computation process, we construct a D2 vector from the D2 shape distribution. The number of bins is the number of dimensions of D2 vector. This mean the number of elements of D2 vector is equal the number of bins. Thus, D2 vector is a set of **B** elements.

3.3 D2 Shape Distribution Training by CNNs

The D2 vectors, which are computed from 3D weapon triangle mesh dataset, will be trained by a CNNs. The structure of this CNNs consists of a convolution layer and a neural networks. The structure of CNNs in the proposed algorithm is shown in Fig. 4. The input of convolution layer is D2 vectors. Each D2 vector is a set of **B** discrete elements, and the output of convolution layer is input neurons for neural networks.

For training process by the neural networks, we use the back propagation algorithm [4] to train the input neurons $\mathbf{N}_{input}$. The back propagation algorithm is separated into four distinct sections, the forward pass, the loss function, the backward pass, and the weight update. During the forward pass, $\mathbf{N}_{input}$ is passed through the whole network. The hyperbolic tangent function is applied as the active function to the hidden layer, and the soft-max function is applied to the output layer.

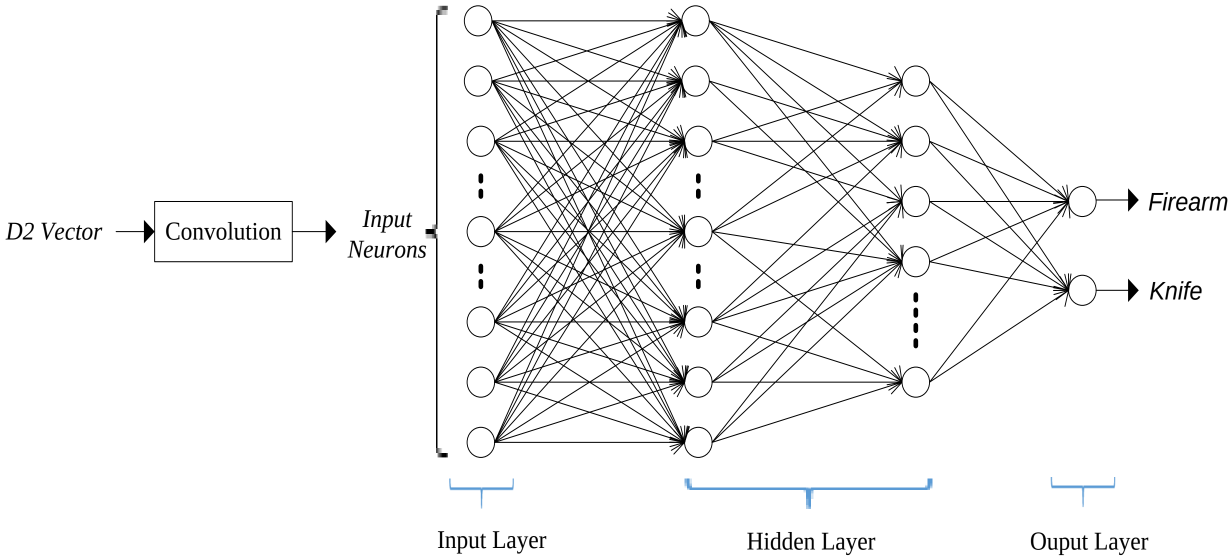

Figure 4. D2 shape distribution training by CNNs.

4. EXPERIMENTAL RESULTS & ANALYSIS

Table1. Experimental Results with CNNs.

No.	Number of models	Accuracy (%)		
		Training Result	Testing Result	Average
Dataset 1	100	55.00	30.00	42.50
Dataset 3	500	50.41	47.77	49.09
Dataset 3	600	62.50	62.50	62.50
Dataset 4	1000	76.75	80.50	78.62
Dataset 5	2000	94.94	99.64	97.29

Due to the fact that 3D printers use 3D triangle meshes as the input files, we collected 3D weapon models and other 3D models on internet for experiments. The format of 3D models is the STL format [5] which is the popular format of 3D triangle mesh. The mentioned weapon objects in this work are firearm and knife, thus we divide datasets into classes for the experimental process. The 80 percent of each dataset is used for the training process, and the 20 percent of each dataset is used for the testing process. The training and testing results with each dataset are shown detail in Tab. 1. The average accuracy is formed from 42.50 % to 97.29 % with datasets have the number of models from 100 to 2000. We can see that the accuracy of the proposed method is increased according to the number of models in dataset.

At this time, there is not any method for weapon anti in 3D printing. In order to evaluate the performance of the proposed method, we compare the accuracy of the proposed method with -the accuracy of the matching methods [2, 6, 7]. The average accuracy of the proposed method is 97.29 % after the training and testing process with dataset that has 2000 models. In Osada's method [2], author used five features for 3D model matching experiments and the highest accuracy of this method is 66.00 %. In Walter's method [6], he experimented with 6 classes of 3D hammer, mug, airplane, bottle, car and shoe models and get accuracies are 76 %, 86%, 85 %, 64%, 75 % and 68 % respectively. Thus, the average accuracy of Walter's method is 75.66 %. With Thomas' method [7], the highest accuracy of this method for 3D shape is 89 %. Fig. 5 show the performance comparison of the proposed method with the matching methods. Consequently, the proposed method has higher accuracy than previous methods.

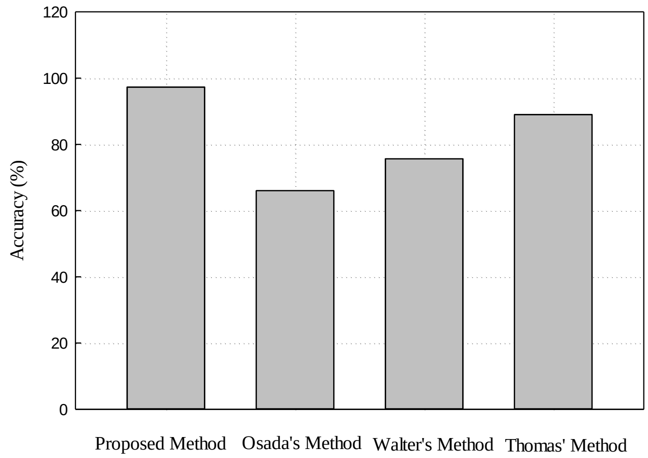

Fig. 5. Performance comparison the proposed method with matching method.

5. CONCLUSION

In this paper, we proposed a solution to restrict the printed out of weapons for safe 3D printing based on D2 shape distribution and CNNs. The D2 shape distribution is trained by CNNs in order to detect 3D weapon models as firearm and knife. It can prevent the printing of weapons as firearm, knife. The accuracy of the proposed algorithm is higher than conventional works. In future, we will collect more models to recognize many types of 3D weapon models.

ACKNOWLEDGEMENT

This work is supported by the by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2018-2016-0-00318) supervised by the IITP (Institute for Information & communications Technology Promotion) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (No. 2016R1D1A3B03931003, No. 2017R1A2B2012456).

REFERENCES

[1] United States Government Accountability Office, “3D Printing Opportunities, Challenges, and Policy Implications of Additive Manufacturing,” USA, June, 2015.
[2] R. Osada, F. Thomas, C. Bernard and D. David, “Shape Distributions”, ACM Transactions on Graphics, vol. 21, no. 4, pp. 807–832, Oct. 2002.
[3] C. Levi, R. Ruiz and Z. Huang, “A Shape Distribution for Comparing 3D Models”, Springer, LNCS 4351, Part I, pp. 54–63, 2007.
[4] R. Rojas, “Chap 7: The Backpropagation Algorithm,” Neural Networks, Springer-Verlag, Berlin, 1996.
[5] STL format in 3D printing, https://all3dp.com/what-is-stl-file-format-extension-3d-printing/, accessed in 2018.
[6] W. Walter and V. Markus, “Shape Based Depth Image to 3D Model Matching and Classification with Interview Similarity”, 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4865–4870, Sep. 25–30, 2011.
[7] F. Thomas, M. Patrick, K. Michael, J. Chen, H. Alex, D. David and J. David, “A Search Engine for 3D Models,” ACM Transactions on Graphics, vol. 5, no. 10, pp. 1-28, 2002.