

## Research Article

# Deep Learning of 3D High-Precision Model Digital Engraving of Next-Generation Games Based on Artificial Intelligence

**Yue Zhao** 

*School of Design Art, Wuxi Institute of Technology, Wuxi, Jiangsu 214121, China*

Correspondence should be addressed to Yue Zhao; 201410821005@stu.shmtu.edu.cn

Received 21 February 2022; Revised 3 April 2022; Accepted 18 April 2022; Published 9 May 2022

Academic Editor: Qiangyi Li

Copyright © 2022 Yue Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to improve the intelligence of the next-generation game model, a new game artificial intelligence came into being. Through the in-depth understanding and research of the next generation of game model production technology, the current industry general production process is summarized, and a set of effective production methods is expounded. In the context of the rapid development of digital technology, the game industry increasingly recognizes the importance of digital carving technology for the design and production of the next generation of games. The design and production of the next-generation game model can make up for the lack of refinement of the traditional next-generation game character model and the unreal mapping effect. This paper focuses on two ways: behavior tree and machine learning, designing game AI with more anthropomorphic perception and flexible behavior. Finally, the basketball player competition AI designed with a behavior tree and machine learning demonstrates its intelligent behavior.

## 1. Introduction

Digital engraving technology refers to the use of digital design software to create works in the virtual three-dimensional space created by computer. It is a new form of digital creation. Digital engraving technology is mainly used to make super-realistic three-dimensional models, which are made by digital engraving software such as ZBrush and Mudbox. Art lovers can create high-quality and high-precision digital engraving works with their own imagination and creativity. Compared with the traditional carving process, it subverts the relatively old and cumbersome carving mode, liberates sculptors from manual sculpture, provides them with broad development space and new technical means, and provides more possibilities for artistic innovation and diversification of artistic forms. It is a new trend in the development of the game industry to design and make the next-generation game role model through digital engraving technology [1]. The development of this technology will inject new vitality into the next-generation game industry. By analyzing the advantages and characteristics of digital engraving technology in the production of

next-generation games, this paper introduces and expounds on the concept of digital engraving technology and next-generation games, the design concept, and production process of next-generation game characters and finally demonstrates that digital engraving technology has great potential and strong development advantages in the design and production of next-generation game character models through examples. Exquisite game graphics and innovative game play can be loved by players, but in addition to these directions, what else can we work towards? In recent years, in order to meet the machine performance requirements of major game manufacturers, both computer consoles and mobile computers, their hardware technology has been greatly improved so that the artificial intelligence modules originally at the end of the game can get more computing power. This leads to the research direction of this paper, which is the game artificial intelligence. If you study in this direction, you not only can add fun to the game but also can have a good interaction with the game. An excellent game AI should also build an emotional bond with the player to become the player's company and give it an immersive experience.

Next-generation games refer to games made with advanced technology and high-tech engines that have not been widely used compared with similar games today, representing the popular trend of the next era [2]. At this stage, the next-generation technology is mainly embodied in high-definition and realistic games. It was originally synonymous with high-end games and only appeared on arcades and high-end TV game consoles. Games just opened on next-generation game consoles such as PS2, NGC, and Xbox are called next-generation games. Specifically, the game runs on a high-end game console; the engine technology is advanced; and the picture is more realistic than the traditional online game. The game that can create an immersive and happy game experience is the next-generation game. Xbox, PS3, and Wii are the next-generation games. However, with the rapid development of computer network technology, the concept category of next-generation games began to extend, and network next-generation games and mobile next-generation games began to be classified into next-generation games [3]. The etymology of the next generation is directly quoted from the "next generation" in Japanese, indicating the consciousness of the coming era. It is a general expression of the future world. From the existing literature on the next generation, there is no rigorous academic definition, but the concept of the next generation is generally used. The flow chart of the 3D model production is shown in Figure 1.

## 2. Literature Review

Radesky and others said that traditional art forms are divided into space art and time art, while time and space are relatively binary and have no compatible preconditions. Until the emergence of film, cross-category integration of art has been realized, and games have more comprehensive characteristics than films [4]. Balendran and others believe that the comprehensiveness of the game, on the one hand, is that the game realizes the combination of multiple art categories, pushes the audio-visual language to another peak on the basis of the film, and produces a special group between virtual and reality, that is, the player group [5]. Xiang and others said that players' instinctive pursuit based on game interactivity has promoted the industrial development of the game in a very short time since the emergence and development of the game, surpassed the film and other fields, and led the new direction of the development of the cultural industry. On the other hand, the comprehensiveness of the game is reflected in the characteristics of multidepartment division of labor and cooperation in the process of game production [6]. Chertow said that first of all, a game needs inspiration and creative ideas, which are expressed in words and written in the form of a text plan for communication and discussion. Then, the art team carries out art creation and original painting design according to the character settings in the game plan and further presents the text in the form of the picture image. Lay the foundation and provide a reference basis for the next technical

implementation and then the technical implementation link of the game. Build the character model through professional game production software and try to make corresponding maps according to the artistic style of character design. After making the model, equip music and sound effects according to the overall setting of the game to realize the coordination of audio-visual language [7]. Chiang and others proposed that when testing the game, the internal personnel of the general advanced company should conduct the test and then select some of the actual players for test observation, so as to correct the errors in time after finding the problems before the official release [8]. Wills and others said that the epochal nature of anything will be affected by the social production level in this period, so different cultural representative forms based on the representation of productivity level have emerged in different times [9]. Ferris and others said that in the oral age, music has become the main form of people's artistic appreciation. In the printing age, books are an important way for people to exchange and disseminate information. In the information age, the popularization and application of network and computer technology have led to the transfer of culture from the social elite to the public, and films, TV dramas, and games have become the main ways of mass consumption and entertainment [10]. Reddie and others believe that although people have different attitudes towards games at present, some people think that it plays with things and loses its will. Excessive addiction to games will cause people to lose their way in real life. Especially for young people, games not only affect physical health but also may form a distorted outlook on life due to violence and pornography in games because they are in a period of shaping life values and worldviews [11]. Rodríguez and others said that everything has two sides, both bad and good, and the main reason for the bad is due to people's improper application of technical means, not the bad impact of the form itself. The same is true of the game. As a cultural form with artistic attributes, it has internal factors to help people carry out aesthetic experiences, help people complete the virtual mapping of themselves in real life, and realize people's needs for high-level self-realization [12]. Bao and others said that the current game is following the tide of the era of virtual reality technology and is gradually realizing a more realistic reality experience so that players can completely break away from the barrier of space so that the body can also enter the game, form the simultaneous emptiness of body and perception, and really help people get rid of the shackles of reality and seek different feelings and breakthroughs in the game. Since the game is more and more experiential and experience means a perceptual realization process, it also shows that the artistry of the game has not been lost [13].

Based on this study, this paper proposes deep learning of digital carving of 3D based on artificial intelligence. Design a 3D shooter with playability. On the one hand, it is for the integrity of the paper results, and on the other hand, it is also for the order to carry the decision technology and perception system studied in this paper so that it can be effectively tested in a structured game running environment. On the contrary,

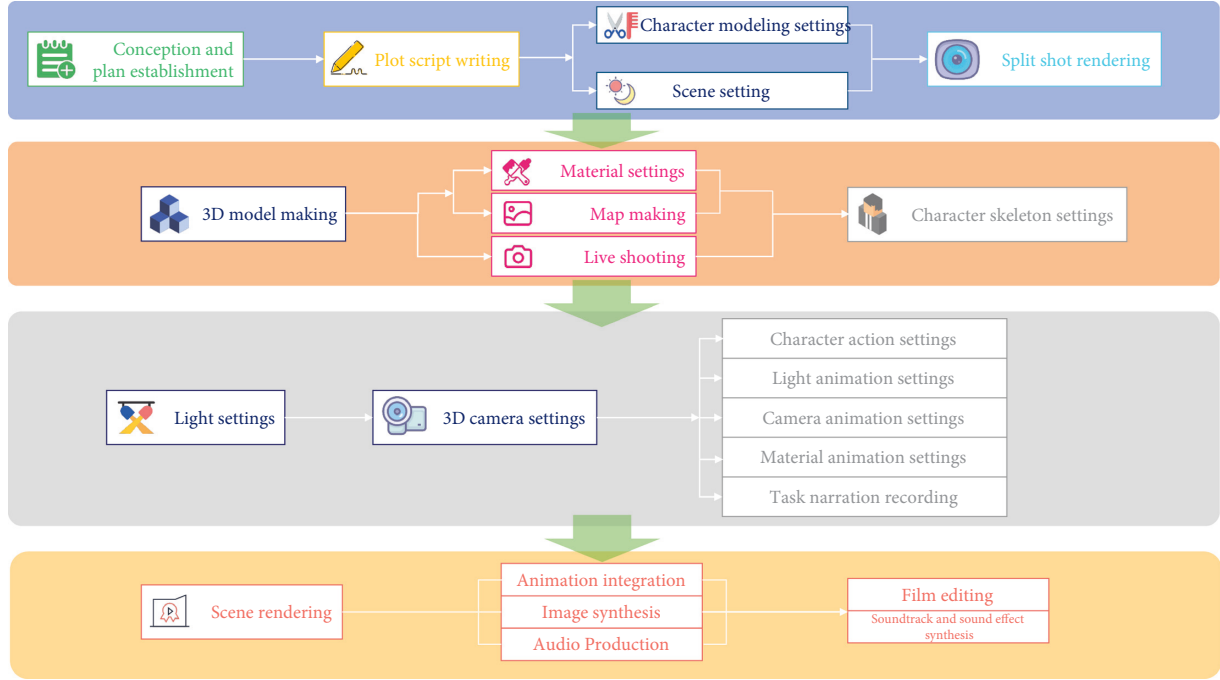


FIGURE 1: 3D model making flow chart.

by analyzing the data during the game operation, the feasibility and the stability of the relevant technology can be obtained.

### 3. Method

**3.1. Convolutional Neural Network Theory Research.** In recent years, the application of convolutional neural network (CNN) has made good progress in the problems of many computer fields. First, we introduce the common layer of convolutional neural network composition.

**3.1.1. Convolutional Layer.** The convolutional layer functions to extract features from the most original training data through different convolutional kernels and is a key part of the convolutional neural network model.

During the convolution operation, we suppose that the input image size is  $n \times n$ , after  $k$  convolution kernels, the stride size is  $s$ , and the size of the convolution layer is  $p$ ; the output matrix size of the convolution layer is shown in the following equation:

$$\lfloor \frac{n+2p-f}{s} + 1 \rfloor \times \lfloor \frac{n+2p-f}{s} + 1 \rfloor \times k. \quad (1)$$

Essentially, both 3D and 2D convolutions are weighted sums of the convolutional kernels and their locally connected regions. Differently, both the convolutional kernels used for 3D convolution and their locally connected regions are all 3D cubes. If the input size is  $l$ ,  $w$ , and  $h$ , the stride of the convolution kernel is  $s$ ; the padding size is  $p$ ; the dimension is  $ffc$ , which is shown as follows:

$$\lfloor \frac{l+2p-f}{s} + 1 \rfloor \lfloor \frac{w+2p-f}{s} + 1 \rfloor \times \lfloor \frac{h+2p-f}{s} + 1 \rfloor \times k. \quad (2)$$

**3.1.2. Pooling Layer.** The pooling layer is generally used to extract and compress the feature data, which can be considered a downsampling process, and generally appears after the convolutional layer, including the average pooling method and the maximum pooling method.

The average pooling formula is shown in the following equation:

$$f(x) = \frac{1}{T} \sum_{i=1}^T x_i, \quad (3)$$

where  $x_i$  is the input of the average pooling layer and  $T$  is the total input of this layer.

The maximum pooling formula is shown in the following equation:

$$f(x) \max_{1 \leq i \leq T} (x_i), \quad (4)$$

where  $x_i$  is the input for the maximum pooling layer.

**3.1.3. Fully Connected Layer.** The fully connected layer makes a nonlinear combination of the feature information extracted by the convolutional layer and the pooling layer to obtain the output results, and the dropout regularization method is usually introduced to prevent overfitting. Briefly, the dropout method enables better transformation invariance and good generalization performance by randomly “opening” and “closing” the neurons in the convolutional

neural networks and effectively improves the data fit ability of the networks. The formula is shown in the following equation:

$$f(x) = \begin{cases} \frac{x}{1-p}, & \text{if } u > p, \\ 0, & \text{else } u \leq p, \end{cases} \quad (5)$$

where  $x$  is the input and  $u$  and  $p$  are both random natural real numbers, whose relative size determines the “on” and “off” states of the neurons in the network.

**3.1.4. Activation Function.** The activation function often acts as a medium to transfer the upper layer of neuron nodes to the next layer of neurons of the convolutional neural network. Common nonlinear activation functions include Sigmoid, ReLU, and Tanh. The Sigmoid activation function formula is shown in the following equation:

$$f(z) = \frac{1}{1 + e^{-z}}. \quad (6)$$

However, the Sigmoid function also has defects: the existence of a derivative saturation region makes the network prone to gradient disappearance or gradient explosion; the weights are only updated one way in backpropagation, resulting in slow convergence; and the existence of the power function makes the training time long.

For the problem that the Sigmoid function is not centered around zero, the Tanh function is proposed. However, there are still some power operation and gradient vanishing problems. The formula is shown in the following equation:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \quad (7)$$

The ReLU function is a piecewise linear function that can achieve one-sided inhibition. The definition is shown in formula (8). The advantage is that there is no power function and the fast calculation speed and the convergence speed are also faster.

$$f(x) = \max(0, x). \quad (8)$$

Subsequently, the researchers proposed the Leaky ReLU function to solve the “DeadReLU” problem of the ReLU function. It is defined as shown in the following equation:

$$f(y) = \max(\varepsilon y, y), \quad (9)$$

where  $\varepsilon$  is the slope factor, which is a constant and is generally set to a smaller gradient value, such as 0.01.

**3.2. Common Game AI Design Methods.** Machine learning is a complex interdisciplinary subject, which covers many subjects such as algorithm, convex analysis, statistics, probability theory, and approximation theory. It is very difficult. It is at the forefront of current scientific and technological research. The goal of machine learning is to enable machines to imitate human behavior in some

aspects and finally form machines with independent behavior according to a given data set or other learning methods. There is no exact and unified definition of machine learning at present, but a statement highly praised in the academic circles is: “for a certain type of task  $T$  and performance measurement  $P$ , if a computer program’s performance measured by  $P$  on  $t$  improves with experience  $E$ , then we call the computer program learning from experience  $E$ ” [14]. This was put forward by Professor Tom Mitchell of Carnegie Mellon University in his book *Machine Learning*. Figure 2 depicts a machine learning definition for a machine graph.

Machine learning can be divided into supervised learning, unsupervised learning, and reinforcement learning. Supervised learning requires that the training data have labels. It is responsible for obtaining a function from the data set, and then using some new unlabeled data, through this function, the reasonable output can be obtained. Usually used in classification, it is a common technology in the decision tree and neural network training. The data set used in unsupervised learning has no label. The goal is to detect the similarity in the sample and then classify it. Generally speaking, it is to let the machine learn decision-making by itself, rather than human beings teach it. It is usually applied to clustering. Reinforcement learning also needs to learn how to do, let the machine constantly “try and error,” and tell the machine the direction of learning through reward and punishment in the process of trial and error. The common algorithms in machine learning are decision tree algorithm, naive Bayesian algorithm, artificial neural network algorithm, and so on. In specific reinforcement learning, the commonly used algorithms are Q-learning, proximal policy optimization (PPO), and so on. Reinforcement learning, also known as reinforcement learning, reinforcement learning, or evaluation learning, is an important part in the field of machine learning. According to the reward and punishment results after decision-making, the decision-making model is optimized in order to obtain the maximum benefits. Reinforcement learning is applied in many fields, such as game theory, genetic algorithm, and cybernetics. Reinforcement learning model is usually composed of four parts: state, action, reward, and policy [15–17] (see Figure 3).

State is a specific state obtained by the agent observing the environment. Through the state value function, we can get the expectation that the state  $s$  at time  $t$  can get returns in the future. Action is the output generated by the agent through observing the environment and then through a certain strategy, which will reffect the environment and generate rewards. Policy is the core of reinforcement learning [18]. The strategy is to process the state obtained from the observation environment and finally obtain the output action. Different strategies choose different actions in decision-making, which will affect the effect of the whole learning. The method used by the agent to select actions in the environment is called strategy  $\pi$ , and the strategy that can obtain the most reward among all optional strategies is called the optimal strategy. Reward is the characteristic of reinforcement learning. In the process of observing and

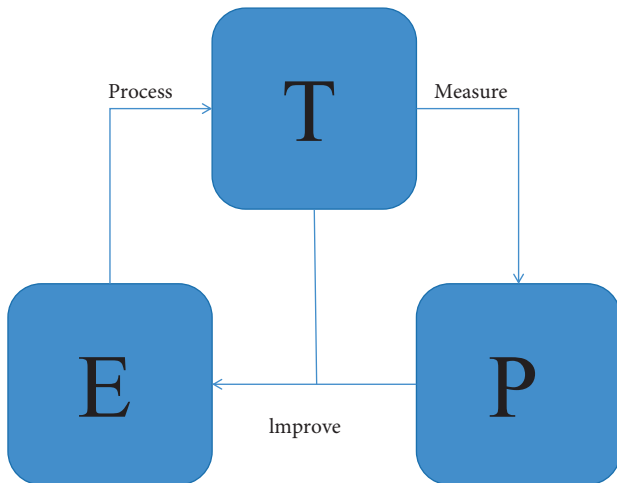


FIGURE 2: Description of machine learning definition.

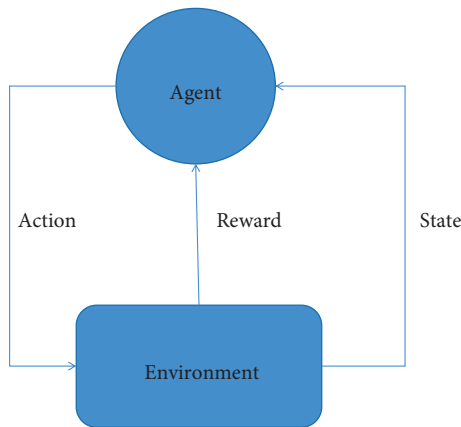


FIGURE 3: Reinforcement learning model.

affecting the environment, the agent obtains reward and punishment signals. According to this reward and punishment signal, the strategy in turn affects the next behavior decision [19].

ML agents is a plug-in based on reinforcement learning, so when using it, we must design several necessary elements in reinforcement learning. One is state, which is called observations here; the other is action, which is the policy output of the agent; and the last is reward, which is the reward method for the agent. Observations: the so-called observation refers to all the necessary vectors that the agent needs to observe in order to make the best decision [20]. The “necessary” here means that if we need the agent to observe the position of an enemy, we will directly tell the agent the position of the enemy as an observation vector. However, when the enemy enters the stealth state, its position should not be used as the observation vector. The way of observation can be digital or visual. However, digital observation can be divided into continuous and discrete. In most cases, we will choose to use the continuous method. In a relatively simple training environment, we can also choose to use the discrete method. In the ML agent plug-in, you can also set multiple

iterative observation vectors to achieve the memory effect of the last observation vector. For the digital observation vector, it should be normalized as much as possible or limited to the range of  $-1$  to  $1$  [21]. The following formula shows a general normalization method:

$$\text{NormalizedValue} = \frac{\text{CurrentValue} - \text{MinValue}}{\text{MaxValue} - \text{MinValue}}, \quad (10)$$

where NormalizedValue represents the normalized value, CurrentValue represents the current observed value, MinValue represents the minimum value of this observation vector, and MaxValue represents the maximum value of this observation vector.

**Actions:** the so-called actions are the instruction vectors output by the policy algorithm after obtaining the observation vector. Similar to the observation vector, the action vector can also be continuous or discrete. Usually, when designing an agent, we need to consider whether the vector controlling the agent is continuous or discrete [22]. For example, when we train a shooting robot, the action vector controlling the shooting force should be continuous because the change of force is a continuous form. When we train an agent to move back and forth, we only need about five types of action values. At this time, we should choose the discrete method. In particular, it should be noted that the value range of the output action vector of different strategies is different, which needs to be considered clearly in advance; otherwise, we will not get the expected training results. In our commonly used PPO algorithm, its output action vector value range is between  $-1$  and  $1$ . **Reward:** reward is an important part of reinforcement learning [23]. In the beginning, the agent does not know what to do but just gives some strategic actions randomly. The reward is used to guide the agent to learn in which direction. When the agent does the right thing, we should give a positive reward, but generally speaking, the reward is not easy to be too large. When the agent does something that should not be done, it should also be given a certain punishment, which is also not easy to be too large. The reasonable setting of reward methods can be of great help to the training of agents. In order to prevent the agent from finding the “loophole” of reward, it is not appropriate to set a very complex reward method. ML agents can combine unity running environment with machine learning, mainly relying on three components. They are learning environment, external communicator, and python API, respectively. The so-called learning environment is the game running environment created through unity, including all environment models, personas, and so on [24]. The agent will obtain the corresponding observation vector from this environment as the basis for training. External communicator is a module for unity to communicate with external Python API. It is located within the unity environment. Python API is a module completely independent of unity. It contains all algorithms used for machine learning training. Here, we mainly use algorithms related to reinforcement learning [25]. The ML-Agents structure diagram is shown in Figure 4.

The goal of the shooting robot is to imitate the process of human shooting action and smoothly throw the basketball



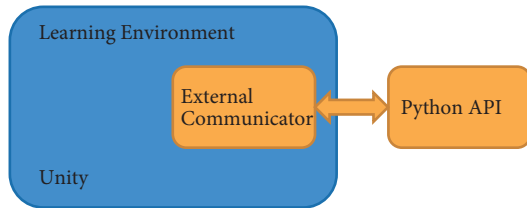


FIGURE 4: ML agents structure.

into the basket as shown in Figure 5. In real life, when shooting, human beings need to pay attention to the height and position information of the basket and adjust the strength and angle of the shot of the basketball according to their distance from the basket. Of course, they also need to aim at the basket. In order to simulate this process to the greatest extent and make some compromise and simplification, it is assumed that the robot can fully aim at the basket without shooting deviation. At the same time, it is guaranteed to use the best shooting angle every time. It only needs the robot to control its own shooting strength [26]. It is assumed that NPC faces baskets with international standard height. Therefore, we can remove the height of the basket from the observation vector. We only need to let NPC pay attention to the distance between the basket and itself. This distance refers to the distance between the projection of the basket on the horizontal plane and NPC. In order to facilitate training, the maximum distance and minimum distance must be defined in advance to facilitate the unit processing of real-time distance.

When we set the horizontal distance between player and basket as the observation vector, we can process the action vector action. The action vector is the policy output given by the training model according to the observed state vector. In fact, the model does not know what the agent will do with the action vector. It only focuses on whether the output given by itself can be rewarded [27], so as to decide how to optimize the next decision-making behavior. For basketball players, the basketball projection force is obviously a continuous vector space, so the action vector space should be continuous with a size of 1. If the PPO training model is used, the value range of the vector will be between  $-1$  and  $1$ . After scaling and translation, we map the value range of vector  $\alpha$  between  $0$  and  $1$ . Then, the linear interpolation function is used to interpolate in minforce and maxforce to obtain the actual strength of basketball projection. The reward setting of shooters is relatively simple, that is, when they use the appropriate strength to throw the ball into the basket or hit the target object, they will be given a larger positive reward. If the projected ball fails to hit the target, a smaller negative reward is given [27].

#### 4. Results and Analysis

Before training, it can be predicted that in the training of this shooting robot, observation and decision-making cannot be

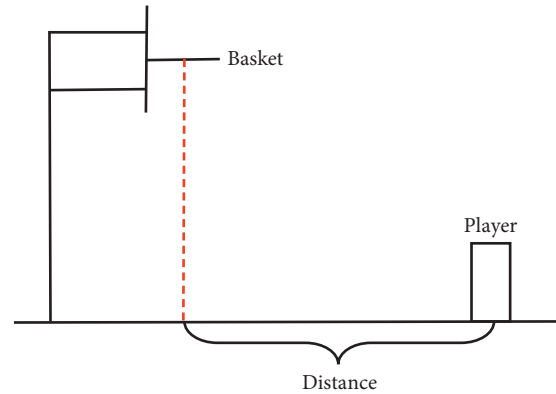


FIGURE 5: Schematic diagram of shooting.

carried out according to a fixed time because, after requesting a decision, the basketball is still in the flight process, and the next decision-making time cannot be predicted, which requires that the shooting robot agent must make decisions on demand, that is, the developer must control the timing of decision-making in the code. The proportion of baskets in the whole training area is very small, which leads to the sparse reward in the training environment. If the conventional reinforcement learning training method is used, it will undoubtedly take a long training time to achieve convergence. Therefore, it is considered to combine the idea of course learning, increase the size of the basket in the early stage of training so that it can easily obtain rewards, and then gradually reduce the size of the basket and increase the difficulty of training based on the increase of rewards or training process. The reinforcement learning training method used does not need a training set but only needs to set up appropriate reward methods to let the agent learn independently. Here, training methods such as course learning and curiosity are adopted. The hardware environment trained for this experiment is shown in Table 1.

The software environment trained for this experiment is shown in Table 2.

It is clear from the training results in Figure 6 that the light blue lines are the way of using ordinary reinforcement learning, and the learning rate is very slow. After training 100,000 steps, the cumulative reward is still negative. After about 700,000 steps, the convergence rate decreases very fast. According to this trend, it still needs a lot of training steps to get a good convergence. The red line is the training result after using reinforcement learning and curiosity. The initial reward is less, but the learning rate has been significantly improved, and the convergence is much faster than before. At about 1 million steps, the training result is very close to convergence. The blue line is the training result of using reinforcement learning, curiosity, and course learning. Because course learning reduces the learning difficulty, the speed is very fast in the early stage of learning. Even if the difficulty is increased later, the fluctuation of the curve is

TABLE 1: Experimental hardware environment.

Project	Configuration
CPU	Intel(R) Core™ i5-3230M @2.6 GHz
GPU	NVIDIA GeForce 710M
Memory	8G
Video memory	2G

TABLE 2: Experimental software environment.

Project	Configuration
Operating system	Windows 10
Python	3.6.1
ML-Agents	0.14.1
Jupyter	4.6.1
TensorFlow	1.7.1
Conda	9.0

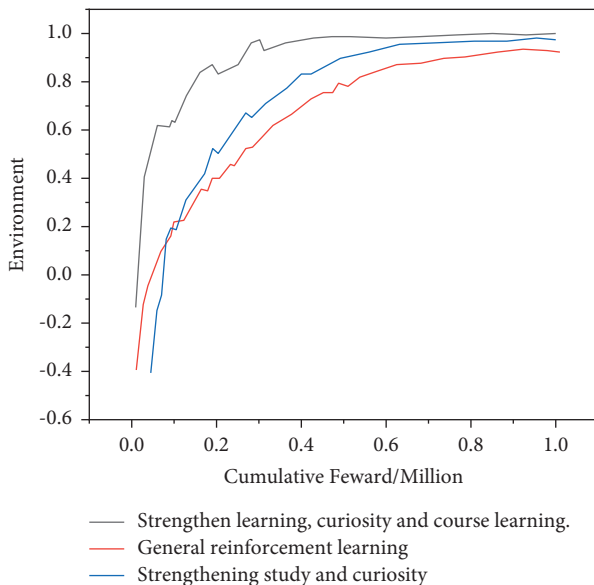


FIGURE 6: Training results of shooting robot.

very small, which significantly speeds up the speed of convergence. After about 700,000 steps, it has almost completely converged.

## 5. Conclusion

Through the analysis of several common game AI design methods, the design method of using behavior tree and machine learning is determined. Through practice, the design process of the two design methods is introduced in detail, and some simple demo game artificial intelligence are realized. This paper focuses on the design and implementation of multiple NPCs based on behavior trees and machine learning. In the design and production of a basketball player's game artificial intelligence, the combination method of behavior tree and machine learning is proposed. The strategy model obtained by machine learning is encapsulated into some nodes in the behavior tree, and the two are organically combined to learn from each other. Finally,

through the description of the overall operation effect of the game, the intelligent behavior of all game AI is displayed.

There is much more to improve in the machine-learning-based game AI design approach used in this paper. In the process of designing the reward methods and the training methods, we rely on the experience setting, so, in future work, we also need to strengthen the learning of our own machine learning knowledge. In addition, the training results of a complex model rely on the training mode, so, with the development of machine learning technology, more learning algorithms should also be provided to obtain the models they need faster. You can also continue to study deeply from the direction of hierarchical reinforcement learning so that developers can train highly complex game AI and make games more interesting.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The author declares that there are no conflicts of interest.

## References

- [1] S. Cho, S. Geun Choi, D. Kim, G. Lee, and C. BongSohn, "How to generate image dataset based on 3d model and deep learning method," *International Journal of Engineering & Technology*, vol. 7, no. 3.34, pp. 221–225, 2018.
- [2] M. M. Tareh, N. Zhu, T. A. A. Ali, A. S. Hameed, and M. L. Mutar, "Transfer learning to detect covid-19 automatically from x-ray images using convolutional neural networks," *International Journal of Biomedical Imaging*, vol. 2021, Article ID 8828404, 9 pages, 2021.
- [3] Y. Wu, G. M. White, T. Cornelius et al., "Deep learning li-rads grading system based on contrast enhanced multiphase mri for differentiation between lr-3 and lr-4/lr-5 liver tumors," *Annals of Translational Medicine*, vol. 8, no. 11, 701 pages, 2020.
- [4] J. S. Radesky, S. Eisenberg, C. J. Kistin et al., "Overstimulated consumers or next-generation learners? parent tensions about child mobile technology use," *The Annals of Family Medicine*, vol. 14, no. 6, pp. 503–508, 2016.
- [5] S. Balendran, S. Liebmann-Reindl, A. S. Berghoff et al., "Next-generation sequencing-based genomic profiling of brain metastases of primary ovarian cancer identifies high number of brca-mutations," *Journal of Neuro-Oncology*, vol. 133, no. 3, pp. 469–476, 2017.
- [6] L. Xiang, B. Poźniak, and T. Y. Cheng, "Bacteriological analysis of saliva from partially or fully engorged female adult rhipicephalus microplus by next-generation sequencing," *Antonie van Leeuwenhoek*, vol. 110, no. 1, pp. 105–113, 2017.
- [7] D. S. Chertow, "Next-generation diagnostics with crispr," *Science*, vol. 360, no. 6387, pp. 381–382, 2018.
- [8] M. Chiang, R. E. Azouzi, L. Gao, J. Huang, C. J. Wong, and S. Sen, "Guest editorial: smart data pricing for next-generation networks," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 4, pp. 641–644, 2020.
- [9] J. Wills, J. Herbohn, M. O. M. Moreno, M. S. Avela, J. Firn, and J. Firn, "Next-generation tropical forests: reforestation

- type affects recruitment of species and functional diversity in a human-dominated landscape,” *Journal of Applied Ecology*, vol. 54, no. 3, pp. 772–783, 2017.
- [10] R. Ferrus, V. Marojevic, and L. Goratti, “Next-generation softwarez wireless networks [from the guest editors],” *IEEE Vehicular Technology Magazine*, vol. 13, no. 4, 78 pages, 2018.
- [11] A. W. Reddie, B. L. Goldblum, K. Lakkaraju, J. Reinhardt, M. Nacht, and L. Epifanovskaya, “Next-generation war-games,” *Science*, vol. 362, no. 6421, pp. 1362–1364, 2018.
- [12] A. C. Rodríguez, C. Roda, F. Montero, P. González, and E. Navarro, “An interactive fuzzy inference system for teletherapy of older people,” *Cognitive Computation*, vol. 8, no. 2, pp. 318–335, 2016.
- [13] T. Bao, X. Zhang, T. Yu, X. Liu, and D. Wang, “A stackelberg game model of real-time supply-demand interaction and the solving method via reinforcement learning,” *Zhongguo Dianji Gongcheng Xuebao/Proceedings of the Chinese Society of Electrical Engineering*, vol. 38, no. 10, pp. 2947–2955, 2018.
- [14] G. D. Landro, O. Amoroso, T. A. Stabile, E. Matrullo, A. Lomax, and A. Zollo, “High precision differential earthquake location in 3d models: evidence for a rheological barrier controlling the micro-seismicity at the irpinia fault zone in southern apennines,” *Geophysical Journal International*, vol. 203, no. 3, pp. 1821–1831, 2018.
- [15] X. Yang, H. J. Cao, Y. P. Chen, L.-B. Zhu, and B. J. Li, “An analytical model of chip heat-carrying capacity for high-speed dry hobbing based on 3d chip geometry,” *International Journal of Precision Engineering and Manufacturing*, vol. 18, no. 2, pp. 245–256, 2017.
- [16] Y. Song and Y. S. Ho, “High-resolution depth map generator for 3d video applications using time-of-flight cameras,” *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 386–391, 2017.
- [17] M. S. Bhatia and A. Goyal, “Games people play: the next-generation addiction,” *Medical Journal Armed Forces India*, vol. 75, no. 2, pp. 119–120, 2019.
- [18] Z. Lina, “A next generation ocean robot explores the arctics,” *Mechanical Engineering*, vol. 138, no. 5, pp. 34–39, 2016.
- [19] J. Renz, R. Miikkulainen, N. R. Sturtevant, and M. H. M. Winands, “Guest editorial: physics-based simulation games,” *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 8, no. 2, pp. 101–103, 2016.
- [20] A. Marchand, “The power of an installed base to combat lifecycle decline: the case of video games,” *International Journal of Research in Marketing*, vol. 33, no. 1, pp. 140–154, 2016.
- [21] R. P. Barneva, K. Kanev, B. Kapralos, M. Jenkin, and B. Brimkov, “Integrating technology-enhanced collaborative surfaces and gamification for the next generation classroom,” *Journal of Educational Technology Systems*, vol. 45, no. 3, pp. 309–325, 2017.
- [22] Shaun and Campbell, “Top jobs filled at siemens gamesa,” *Windpower Monthly*, vol. 33, no. 5, p. 9, 2017.
- [23] S. Ranadheera, S. Maghsudi, and E. Hossain, “Minority games with applications to distributed decision making and control in wireless networks,” *IEEE Wireless Communications*, vol. 24, no. 5, pp. 184–192, 2017.
- [24] J. Freeman, “It’s a small (cell) world,” *Land Mobile Wireless Communications for Businesses*, vol. 23, no. 1, 34 pages, 2016.
- [25] K. Wakeford and C. J. Harrington, “Dynamic adjustment of difficulty in an online game based on hardware or network configuration,” *Korean Journal of Radiology*, vol. 16, no. 4, pp. 936–941, 2017.
- [26] P. Fanning, “Generation games,” *Eureka*, vol. 39, no. 8, 5 pages, 2019.
- [27] Plaisance and Heather, “Playing at the next level: a history of american sega games,” *Reference Reviews*, vol. 31, no. 6, pp. 39–40, 2017.