
FLAPPY BRID USING FACE RECOGNITIONAND ADAPTIVE AI**S.Abishaken^{*1}, S.K.Vedhanth^{*2}, Dr.P.Sujatha^{*3}**

^{*1,2}Student, Department Of Computer Applications, Vels Institute Of Science Technology And
Advanced Studies, Chennai - 117,Tamil Nadu, India

^{*3}Professor And Head, Department Of Computer Applications, Vels Institute Of Science Technology
And Advanced Studies, Chennai - 117,Tamil Nadu, India

DOI: <https://www.doi.org/10.56726/IRJMETS95184>

ABSTRACT

In this project, we created a modified version of the Flappy Bird game by adding some AI features to make it more interesting. The main idea is to use face recognition, where the system captures the player's face using a webcam and detects some basic features. Then, with the help of AI, the face is changed into a cartoon bird, so each player can play the game with their own customized character. Also, the game difficulty increases slowly as the score gets higher, so it doesn't feel too easy after some time. Apart from that, we added different backgrounds like city, village and night to make the game look better. There is also a multiplayer option where players can compete with each other, which makes it more fun. The game shows both current score and high score, so players can try to beat their previous record. We used Python, OpenCV and Pygame for developing the game, along with TensorFlow or PyTorch for AI parts. Overall, this project shows a simple way of using AI in games to make them more interactive and enjoyable.

KEYWORDS: Flappy Bird, Artificial Intelligence (AI), Face Recognition, Adaptive Difficulty, Game Development, Computer Vision, Personalized Gaming.

I. INTRODUCTION

Flappy Bird is a well-known arcade game and at first glance it looks very simple. The player just has to control a bird and avoid hitting pipes. But in reality, the game is not that easy and many users find it quite difficult after a few tries. This difficulty is actually one of the reasons why the game became so popular and addictive. However, the original version of Flappy Bird is quite basic. It does not include any personalization or smart features. Because of that, after some time, the game can feel repetitive. In this project, an attempt is made to improve the traditional Flappy Bird game by adding face recognition and artificial intelligence. The idea is not just to make the game harder or easier, but to make it feel a bit more personal and interactive.

At the beginning, the system uses a webcam to capture the player's face. One thing noticed during testing is that lighting conditions can sometimes affect detection, but in normal cases it works fine. After capturing the image, the face is processed and converted into a cartoon-style bird. This bird is then used in the game instead of the default one. So basically, each player gets their own version of the bird.

Another feature included is adaptive difficulty. Instead of keeping the same level throughout, the game changes depending on how the player performs. If the score keeps increasing, the game slowly becomes more challenging. But if the player is struggling, the difficulty does not increase too fast. This balance was adjusted after trying different settings.

Apart from that, different background themes like city, village, and night mode are added. This is a small feature, but it makes the game feel less boring over time. There is also a multiplayer option, which allows players to compete with others. This part was interesting to test. The system also keeps track of scores, including the highest score. This encourages players to try again and improve. The project is mainly developed using Python, along with OpenCV for face detection and Pygame for game development. Overall, the aim was to take a simple game and add a few meaningful improvements.

II. RELATED WORK

In recent years, artificial intelligence has been used in many games, especially to improve user experience. Most traditional games, including Flappy Bird, do not change based on the player. Because of this, the experience remains the same for everyone. One commonly used idea is adaptive difficulty. In this approach, the game adjusts itself based on player performance. For example, if a player is doing well, the game becomes harder. If

not, it stays easier. This helps in maintaining balance. Face recognition is another area that has improved a lot. It is mostly used in security systems, but recently it is also being used in games and applications. In gaming, it can be used to create avatars that look similar to the player. Some previous works have focused only on AI-based gameplay, while others focused on personalization. But combining both is not very common. This project tries to combine these two ideas in a simple game setup. It is also observed from different studies that personalized systems usually keep users more engaged. So adding such features can improve the overall experience.

III. PROPOSED APPROACH

The system designed in this project mainly focuses on improving gameplay by adding a few extra features. There are three main parts in the system. The first one is the face recognition module. It captures the player's face and converts it into a simple cartoon bird. The output is not very complex, but it works well for this purpose. The second part is the game engine. This handles everything related to the game, like movement, pipes, scoring, and screens such as start menu and game over. The third part is the adaptive AI system. This checks how the player is performing and adjusts the difficulty. During testing, it was seen that small gradual changes work better than sudden difficulty spikes. All these components are connected together to form the final system.

IV. METHODOLOGY

The system is divided into different modules for easier implementation. The face recognition module captures the player's face using a webcam. The image is processed and converted into a basic avatar. Sometimes detection may vary slightly, but overall it gives a usable output. The game engine controls the main logic. It includes bird movement, pipe generation, collision detection, and score calculation. The adaptive AI module monitors the player's score and survival time. Based on this, it adjusts parameters like speed and gap between pipes. The workflow is simple. First, the user opens the game and starts it. Then the face is captured and converted. After that, the gameplay begins. During the game, difficulty keeps adjusting. Finally, the score is displayed.

V. IMPLEMENTATION

The implementation is done using Python. OpenCV is used for capturing and detecting the face. In some cases, detection speed depends on system performance, but it works fine on most systems. Pygame is used to develop the game. It handles graphics, movement, and collision detection. Basic physics like gravity and jump movement are implemented. Pipes are generated at intervals, and collision is checked continuously. The adaptive system increases speed and reduces gaps as the score increases. This was tuned by testing different values. All modules are combined and tested together to form the final application.

VI. COMPARISON

EXISTING SYSTEM:

Features	Existing System
Personalization	No personalization
Difficulty Level	Fixed Difficulty
User Engagement	Limited
Gameplay Adaptation	Not Available
Game Modes	Single-Player Only
User Interaction	Basic Controls
Learning Capability	None

PROPOSEDSYSTEM:**Proposed System**

Face-Based avatar

Generation Adaptive

difficulty using AI

High due to personalization

Real-time adjustment based on

performance Single-player + Multiplayer

Interactive and dynamic

experience AI learns from player

behavior

VII. RESULT

The system was tested under different conditions. It was observed that players found the personalized avatar feature interesting. Seeing their own face in the game made it feel different from the normal version. The adaptive difficulty also worked as expected in most cases. The game did not feel too easy or too hard. Multiplayer mode and score tracking added some extra value to the system. However, in some cases, face detection was slightly affected by lighting, which can be improved in the future.

VIII. CONCLUSION

In this project, an attempt was made to improve the Flappy Bird game by adding face recognition and adaptive AI. The system was able to personalize the game and adjust difficulty based on performance. These changes made the game play more engaging compared to the original version.

IX. FUTURE SCOPE

There are several areas where this project can be improved. Face recognition can be made more accurate by using advanced deep learning models. Also, performance can be improved for low-end systems. The game can be extended to mobile platforms, which would make it more accessible. Additional features like gesture control or better graphics can also be added in future versions.

X. REFERENCE

- [1] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.
- [2] PygameCommunity, "Pygame Documentation", [Online].
- [3] Available: <https://www.pygame.org/docs/>
- [4] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010.
- [5] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed., Prentice Hall, 2010.
- [6] I. Goodfellow, Y. Bengio, and Courville, Deep Learning, MIT Press, 2016.
- [7] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1991.
- [8] Y. LeCun, Y. Bengio, and G. Hinton, "DeepLearning," Nature, vol. 521, pp. 436–444, 2015.