NeuroTune | AI-Enhanced Aim Trainer

ONG You Yang (Hans), and **LOW** Wang Chun, Mark

1 Introduction

The rapid rise of competitive gaming and eSports has captivated millions of players around the world, driving a continuous quest for improved performance, especially in first-person shooter (FPS) games where precise aiming is crucial. Games like ‘Counter-Strike 1.6,’ ‘Valorant,’ and the ‘Call of Duty’ series highlight the need for advanced aiming skills, making aim trainers an essential tool for both casual and professional gamers.

Traditional aim trainers such as Aim Lab and Kovaak's FPS Aim Trainer have been instrumental in helping players enhance their aiming abilities. These tools offer a variety of scenarios and drills tailored to improve different aspects of aiming. However, they often fall short of providing a truly personalized and adaptive training experience. Their one-size-fits-all approach does not cater to the unique needs of individual players. Typically, the user chooses between preset drills which vary the spawn interval, lifetime, size, and distance between targets to build muscle memory. This approach lacks the real-time adaptability to adjust the training difficulty based on player performance during a session.

To address these limitations, our research introduces NeuroTune, an AI-enhanced aim trainer that leverages on Fitts' Law, a fundamental principle in human-computer interaction. Fitts' Law predicts the time required to move a cursor to a target based on the distance and size of the target, allowing us to create a dynamic training environment. By incorporating this model, our aim trainer adjusts the key factors: target size, spawn position, and lifetime in real-time, offering a continuously challenging and personalized training regimen.

This paper explores the integration of Fitts' Law into aim training, discusses the theoretical foundations, and details the methodology for implementing an AI-driven aim training system. We will also present the results from initial testing and discuss the potential implications for the gaming industry. Our goal is to enhance the adaptability and personalization of aim trainers, setting a new standard for training tools in competitive gaming and eSports.

2 Background and Market Research

In the competitive gaming scene, players try to squeeze every bit of performance out of their setups or even themselves to find an edge over their opponents. The rapid growth of the competitive gaming and eSports industry has underscored the need for tools that help players improve their skills, particularly in first-person shooter (FPS) games where precision aiming is critical. Over the years, several aim trainers have been developed to assist gamers in honing their aiming skills.

2.1 Evolution of Aim Trainers

Figure 1 Counter-Strike 1.6 (1999) | Valve Corporation

The concept of aim training can be traced back to early FPS games like 'Counter-Strike 1.6,' which implicitly encouraged players to practice their aim through gameplay. With the increasing popularity of competitive gaming, dedicated aim trainer software such as AimLabs and Kovaak's FPS Aim Trainer emerged.

Figure 2 Valorant (2020) | Riot Games

Their popularity surged in recent years, especially due to the release of 'Valorant' in 2020. Streamers and professional players showcased these tools on their streams during live streams of their training sessions, leading to increased interest and adoption by viewers who wanted to emulate their favorite players.

Figure 3 AimLabs (2018) | State Space Labs, Inc

AimLabs, released in 2018, provides a variety of scenarios and drills designed to enhance different aspects of aiming. It offers detailed feedback on player performance, helping gamers identify areas for improvement. Similarly, Kovaak's FPS Aim Trainer, also released in 2018, features a wide range of customizable training scenarios, allowing players to focus on specific skills. Both tools have become widely adopted in the gaming community for their effectiveness in improving aim.

2.2 Limitations of Current Aim Trainers

Despite their popularity and utility, existing aim trainers have several limitations. Firstly, they often adopt a one-size-fits-all approach, providing a generic training experience that may not address the unique needs of individual players. This lack of personalization means that every player undergoes the same set of exercises, regardless of their skill level or required improvement areas.

Secondly, these tools lack real-time adaptability. While Aim Lab and Kovaak's offer a range of scenarios, they do not dynamically adjust the difficulty or nature of the targets based on the player's performance during a session. Real-time feedback and adjustment are crucial for effective training, to not waste the player’s time. Yet they are missing from these traditional aim trainers. For instance, a player who consistently performs well should face increasing difficulty to continue pushing their limits, while a struggling and overwhelmed player should encounter easier challenges to keep them engaged.

2.3 Introduction to Fitts' Law

To overcome these limitations, this research leverages on Fitts' Law, a fundamental principle in human-computer interaction. Fitts' Law predicts the time required to move to a target based on two primary factors: the distance to the target and the size of the target. The formula is expressed as:

$$\boldsymbol{T} = (\mathbf{a} + \mathbf{b})\ \boldsymbol{log_2}\frac{\boldsymbol{2D}}{\boldsymbol{W}}$$

T: The time required to reach the target.
a: Constant representing start/stop time.
b: Constant scaling the logarithmic term.
D: Distance to the target.
W: Width of the target.
log: Logarithmic relationship between distance and width.

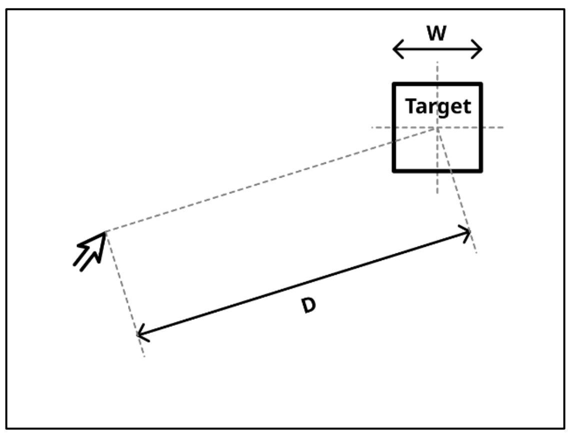

Figure 4 Illustration of Fitts' Law

The diagram in *Figure 4* represents Fitts' Law, which predicts the time required to move to a target based on the distance ***(D)*** to the target and the width ***(W)*** of the target. The dashed lines indicate the movement path, demonstrating how farther or smaller targets take longer to reach compared to closer or larger targets.

This model suggests that it will take a human longer to move a cursor onto targets that are farther away or smaller in size.

Fitts' Law has been applied extensively in various fields, including UI/UX design and games, to enhance user interaction and performance. By integrating this predictive model into an aim trainer, it is possible to create a dynamic training environment that adjusts key factors such as target size, spawn position, and lifetime in real-time based on the player's performance.

3 Methodology

3.1 Overview of the NeuroTune

NeuroTune is an AI-enhanced aim trainer is designed to provide a personalized and adaptive training experience. This section outlines the system architecture, the integration of Fitts' Law, and the real-time adaptability mechanism.

3.2 System Architecture

The aim trainer system is built using Unity, a widely-used game development platform. The system comprises several key components, including the target objects, the target spawner, and the performance tracker. The AI interacts which each component to create a seamless training experience.

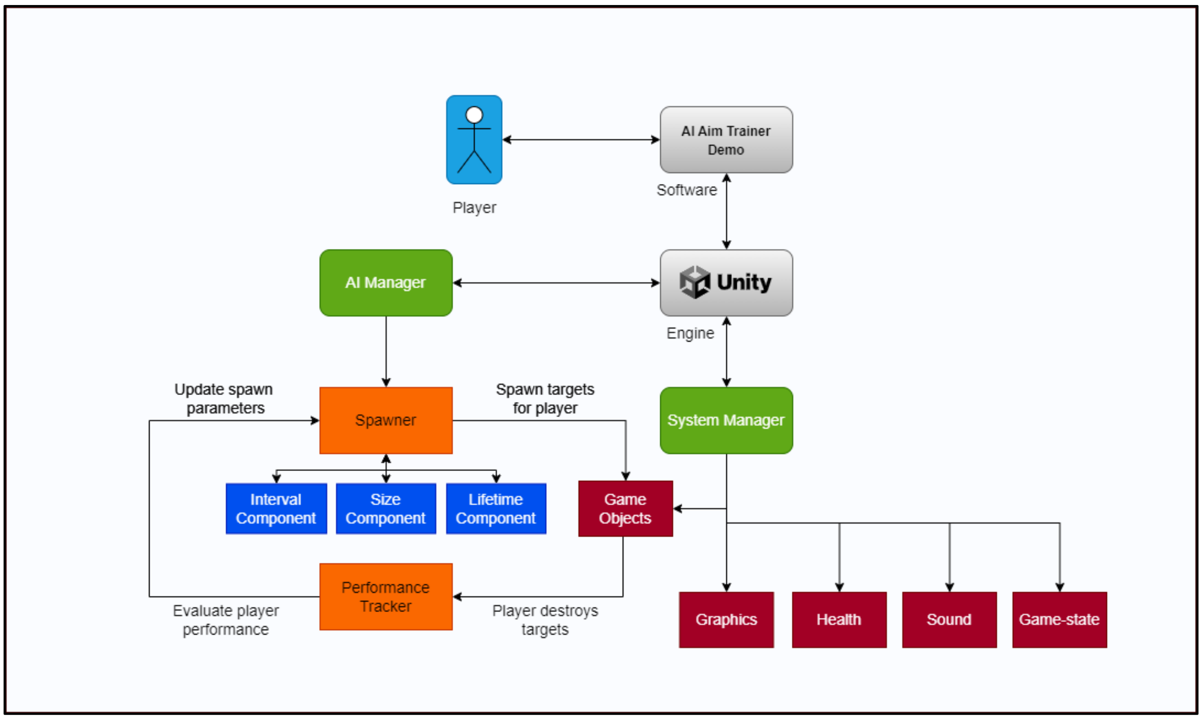

Figure 5: System Architecture of the NeuroTune | AI-Enhanced Aim Trainer

3.3 Integration of Fitts' Law in NeuroTune

In NeuroTune, Fitts' Law is applied to dynamically adjust the size of the targets based on the player's performance.

The CalculateTargetWidth function uses this principle to determine the appropriate target size based on the distance between the player and the target, along with the current difficulty level.

```
private float CalculateTargetWidth(float distance)
{
    // Fitts' Law to calculate target width and apply
addition factor
    float index = (currentDifficulty - a) / b;
    float targetWidth = sizeAdditionFactor + (2 * distance /
Mathf.Pow(2, index));

    // Ensure initial targets start at the default scale
    if (currentDifficulty == 1.0f)
    {
        targetWidth = defaultScale;
    }

    // Clamp target width to the defined range
    targetWidth = Mathf.Clamp(targetWidth, minScale,
maxScale);

    return targetWidth;
}
```

1. Firstly, the index is calculated using the formula $\frac{(\boldsymbol{CurrentDifficulty - a})}{\boldsymbol{b}}$,where $\boldsymbol{a}$ and $\boldsymbol{b}$ are constants from Fitts' Law.
2. Next, the target width is determined using the formula $sizeAdditionFactor + \frac{(\boldsymbol{2D})}{\boldsymbol{2^{index}}}$, which is derived from Fitts' Law. This ensures that the target size is appropriate for the given distance and difficulty level.
3. If the current difficulty level is at its initial value (1.0), the target width is set to the default scale.
4. Lastly, the calculated target width is clamped between the minimum and maximum scale values to ensure it stays within the defined limits.

3.4 Real-Time Adaptability

The AI system continuously monitors the player's performance, specifically tracking hits and misses. Based on the accuracy and response time, the system adjusts the difficulty level dynamically. The adaptation mechanism works by first determining accuracy. This is done by counting the number of successful hits within a spawn cycle. The percentage is then used like so:

1. If the player's accuracy is high, the system increases the difficulty by:
   - Decreasing the target size.
   - Increasing the spawn distance.
   - Reducing the target lifetime.

2. If the player's accuracy is low, the system decreases the difficulty by:
   - Increasing the target size.
   - Decreasing the spawn distance.
   - Increasing the target lifetime.

3.4.1 Adjusting Target Size

The target size is adjusted based on the player's performance. If the accuracy is high, the target size is decreased, and if the accuracy is low, the target size is increased. This is managed within the AdjustDifficulty method:

```
if (accuracy > 0.8f)
{
    defaultScale = Mathf.Max(minScale, defaultScale - 0.1f);
// Decrease scale
}
else if (accuracy < 0.5f)
{
    defaultScale = Mathf.Min(maxScale, defaultScale + 0.1f);
// Increase scale
}
```

This change in target size is then applied when the target is spawned:

```
GameObject target = Instantiate(targetPrefab, spawnPosition,
Quaternion.identity);
target.transform.localScale = new Vector3(targetWidth,
targetWidth, targetWidth);
```

3.4.2 Adjusting Spawn Distance

The spawn distance is modified by adjusting the spawn coordinates of the targets. The further the spawn position from the player, the higher the difficulty. This is handled in the SpawnTarget method:

```
Vector3 spawnPosition = new Vector3(
    Random.Range(minSpawnCoords.x, maxSpawnCoords.x),
    Random.Range(minSpawnCoords.y, maxSpawnCoords.y),
    Random.Range(minSpawnCoords.z, maxSpawnCoords.z)
);
```

3.4.3 Adjusting Target Lifetime

The lifetime of the target is adjusted to increase or decrease the urgency for the player to hit the target. A higher accuracy results in a shorter lifetime, and lower accuracy results in a longer lifetime:

```
if (accuracy > 0.8f)
{
    lifeTime = Mathf.Max(minLifeTime, lifeTime - 0.1f); //
Decrease lifetime
}
else if (accuracy < 0.5f)
{
    lifeTime = Mathf.Min(maxLifeTime, lifeTime + 0.1f); //
Increase lifetime
}
```

The adjusted lifetime is then applied to the target when it is spawned:

```
Target targetScript = target.AddComponent<Target>();
targetScript.lifetime = lifeTime;
```

4 Findings and Analysis

The testing phase provided valuable data on the effectiveness of NeuroTune compared to existing products in the market. A playtest was conducted to quantify and access its real-world impact on players. This section will detail the metrics and the testing process adopted along with the results.

4.1 Experiment methodology

4.1.1 Play-testers

In line with the goal of providing a market-leading training tool, 18 'Counter-Strike 2' players were sourced to participate in the 1-week experiment. To provide a comprehensive analysis, players of varying skill levels were used. The figure below shows the rank system from the game. For ease of understanding, the players are numbered according to their in-game ranks, for example, player 1's rank is 'Silver 1' (the lowest in the game) while player 16's rank is 'Global Elite' (the highest in the game). The image below serves as a reference to the rank system employed by 'Counter-Strike 2'.

Figure 6 Counter-Strike 2 Rank System

The participants were split into 3 groups with each group containing a fair mix of skill levels. Group A (odd-numbered players) utilized the existing aim trainer: Aimlabs, while

Group B (even-numbered players) trained on NeuroTune. Group C consisted of the last two players, Player 17 and Player 18, who were not allowed to use any aim trainer software during the experiment. This group served as a control group to provide a baseline for comparison.

4.1.2 Quantifying Performance

To numerically quantify the player's skill level, the popular steam workshop map for CS2 known as 'Aim Botz' created by the player: uLLeticaL was used. In the map, the player is tasked to kill 100 stationary enemy models in as short of a time as possible. Upon completion, a detailed statistics report will be generated. The key statistic being used to quantify their performance is their **Kills Per Minute (KPM)**.

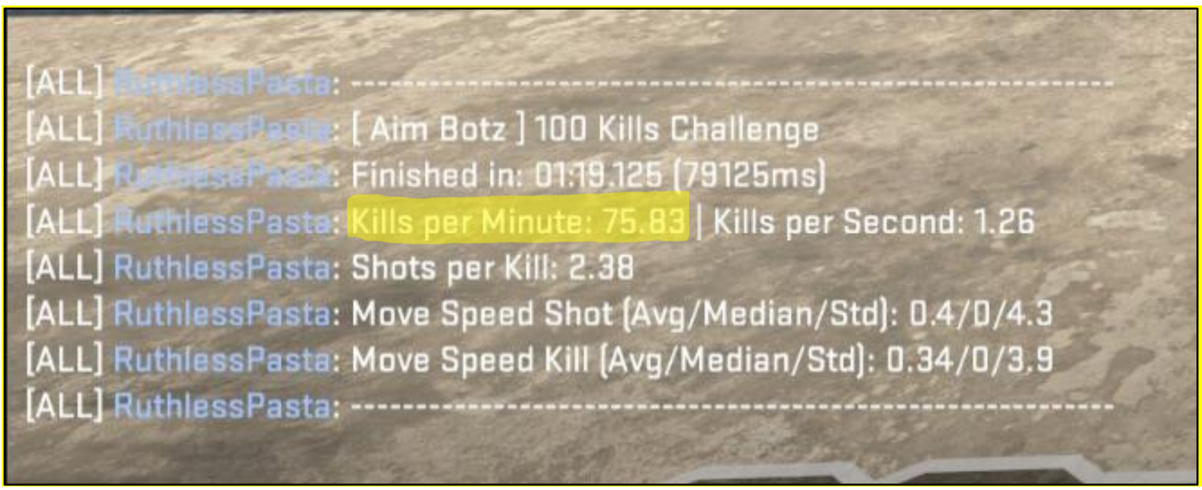

Figure 7 Post session statistics report from Aim Botz

4.1.3 Test Method

Over the course of a week, each day, every player was tasked to use their respective aim trainer for on average 10 to 30 minutes before recording their performance in 3 runs of the 'Aim Botz' test. Their average KPM over the 3 runs was logged each session for further analysis. Their rank after one week was also recorded although it is not used for comparison due to many other factors affecting rankings.

Figure 8 Shooting range in Aim Botz used to measure KPM

4.2 Playtest Data

The table presents the recorded KPM over the 7 days, the type of aim trainer used (if any), and their rank improvements, with the rank system referenced from *Figure 6*.

Table 1 Kills Per Minute (KPM) and Rank Improvements Over One Week.

Participant	Aim Trainer Type	Rank Before	Rank After	KPM Day 1	KPM Day 2	KPM Day 3	KPM Day 4	KPM Day 5	KPM Day 6	KPM Day 7
Player 1	Aim Labs	Silver I	Silver I (+0)	41.2	42.4	43.7	45.1	46.8	48.2	49.9
Player 2	NeuroTune	Silver II	Silver II (+1)	42.1	43.6	45.2	46.3	48.1	49.4	55.7
Player 3	Aim Labs	Silver III	Silver Elite Master (+3)	42.9	44.2	45.6	47.3	48.7	49.4	51.9
Player 4	NeuroTune	Silver Elite	Silver Elite (+0)	43.8	45.3	46.6	48.5	49.9	51.3	58.2
Player 5	Aim Labs	Silver Elite Master	Silver Elite (-1)	55.3	57.1	59.2	60.8	63.1	66.7	67.4
Player 6	NeuroTune	Gold Nova II	Gold Nova III (+1)	56.8	59.3	60.9	65.5	65.5	66.9	74.4
Player 7	Aim Labs	Gold Nova III	Gold Nova II (-1)	61.2	62.6	64.3	66.8	67.1	68.7	69.9
Player 8	NeuroTune	Gold Nova Master	Master Guardian II (+2)	61.9	63.5	64.8	68.4	68.4	70.8	75.8
Player 9	Aim Labs	Master Guardian I	Master Guardian II (+1)	70	71.2	73.2	74.6	76.8	77.6	79.3
Player 10	NeuroTune	Master Guardian II	Master Guardian I (-1)	71.5	72.8	74.3	75.7	77.2	78.9	85.1
Player 11	Aim Labs	Master Guardian Elite	Distinguished Master Guardian (+1)	74.2	75.7	77.4	78.6	80.3	81.7	83.1
Player 12	NeuroTune	Distinguished Master Guardian	Legendary Eagle (+1)	76.3	77.5	79.3	80.4	81.9	83.7	90.2
Player 13	Aim Labs	Legendary Eagle	Legendary Eagle (+0)	84.1	85.5	87.2	86.7	89.1	91.5	93.4
Player 14	NeuroTune	Legendary Eagle Master	Legendary Eagle Master (+0)	88.5	89.9	91.5	92.3	94.2	95.6	102.1
Player 15	Aim Labs	Supreme Master First Class	Global Elite (+1)	111.3	112.7	114.1	115.2	117.5	118.4	120.2
Player 16	NeuroTune	Global Elite	Global Elite (+0)	118.2	119.4	120.9	122.1	124.3	125.8	131.9
Player 17	NIL	Gold Nova I	Gold Nova II (+1)	55.9	61	58.3	55.1	56.2	53.8	56.4
Player 18	NIL	Legendary Eagle	Distinguished Master Guardian (-1)	83.7	81.2	78.4	85.2	86.1	84.8	85.9

4.4 Comparison of Baseline and Post-Training Metrics

The results from the playtesting phase provided valuable insights into the effectiveness of NeuroTune compared to AimLabs and a control group. *Figure 7 and Figure 8* collectively illustrate the performance improvements of individual players and the average improvements across different groups over a one-week period.

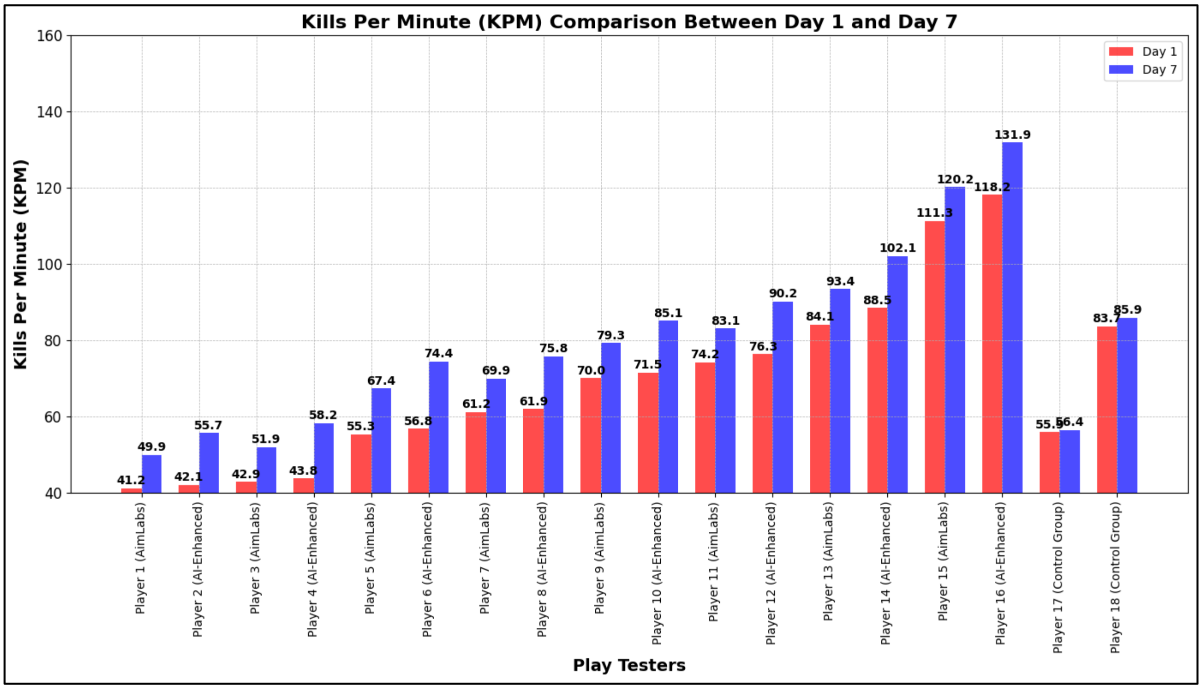

Figure 7 Improvement to Kills Per Minute (KPM) Between Day 1 to Day 7

4.4.1 Individual Performance Improvements

Figure 7 provides a detailed comparison of each player's Kills Per Minute (KPM) on the first day (Day 1) and the final day (Day 7). The players are divided into three categories: those using AimLabs (odd-numbered players such as Player 1, 3, 5, ..., 15), those using the NeuroTune (even-numbered players such as Player 2, 4, 6, ..., 16), and a control group with no aim trainer (including Player 17 and 18).

Across all groups, players demonstrated an increase in KPM by Day 7, indicating overall improvement in performance. Notably, the players using NeuroTune exhibited the most significant gains. For instance, Player 16 (NeuroTune) improved from a KPM of 118.2 on Day 1 to 131.9 on Day 7, while Player 14 (NeuroTune) showed an increase from 88.5 to 102.1 KPM. In contrast, players using AimLabs, such as Player 15, improved from 111.3 to 120.2 KPM, showing notable but comparatively smaller gains.
The control group, which did not utilize any aim trainer, displayed the least improvement. Player 17's KPM only slightly increased from 55.9 to 56.4, and Player 18's KPM rose from 83.7 to 85.9. This minimal improvement underscores the effectiveness of using specialized aim training tools for performance enhancement.

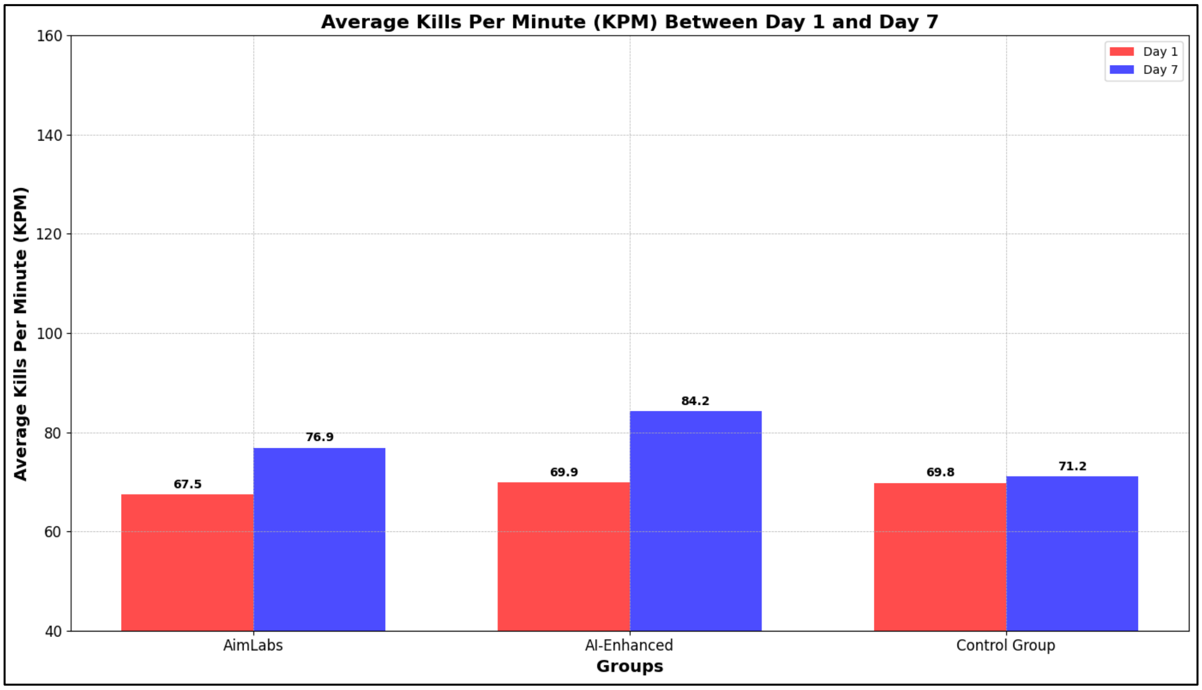

Figure 8 Improvement to Average Kills Per Minute (KPM) Between Day 1 and Day 7

4.4.2 Group Performance Improvements

Figure 8 provides a comparison of the average KPM between Day 1 and Day 7 for the three groups: AimLabs, NeuroTune, and the control group. The average KPM for NeuroTune group increased from 69.9 on Day 1 to 84.2 on Day 7, reflecting an average improvement of 14.3 KPM. This substantial increase highlights the NeuroTune's superior efficacy in boosting players' aiming skills.

In contrast, the AimLabs group showed an average KPM increase from 67.5 to 76.9, resulting in an average improvement of 9.4 KPM. While this group demonstrated significant progress, the improvement was less pronounced compared to the NeuroTune group.

The control group showed the least improvement, with the average KPM increasing from 69.8 to 71.2, yielding an average improvement of only 1.4 KPM. This marginal gain emphasizes the importance of dedicated aim training tools in enhancing player performance.

4.4.3 Comparison and Contrast

Combining the data from *Figures 7 and 8*, it is evident that NeuroTune group outperformed AimLabs and the control group in improving players' KPM. The NeuroTune group showed the highest individual and average improvements, indicating its effectiveness in providing a tailored and adaptive training experience. The AimLabs group also showed notable improvements, but to a lesser extent, while the control group exhibited minimal progress.

In conclusion, the results strongly suggest that NeuroTune is a more effective tool for improving player performance in first-person shooter games. The tailored training regimen, driven by AI and adaptive mechanisms, provides a significant advantage over traditional aim trainers and no training interventions. This finding underscores the potential of AI-driven solutions in enhancing competitive gaming performance.

5 Conclusion

Based on the analysis, we can conclude that aim trainers do provide players with a competitive advantage by honing their skills. However, by incorporating an adaptive AI into the existing concept, player performance can be further boosted. This can be attributed to constantly pushing players to their limits without unnecessarily fatiguing the player.

Although rank gain was not a metric of success during the research, play testers generally had improvements in rank across the board. As mentioned earlier, the wide gamut of factors affecting rank make for too many unknowns in quantifying if aim trainers provide any benefit.

Additionally, the play testers provided positive feedback. The consensus among play testers was that the adaptive approach ensured efficient training which motivated them to have longer training sessions compared to conventional options.

In the competitive scene where every bit counts, an AI-enhanced aim trainer would be an invaluable asset as an avenue for players to refine their skills and gain a competitive advantage over their opponents.

6 Calculations

6.1 AimLabs Group
Day 1 KPM Data:
- Player 1 (AimLabs): 41.2
- Player 3 (AimLabs): 42.9
- Player 5 (AimLabs): 55.3
- Player 7 (AimLabs): 61.2
- Player 9 (AimLabs): 70.0
- Player 11 (AimLabs): 74.2
- Player 13 (AimLabs): 84.1
- Player 15 (AimLabs): 111.3

Day 7 KPM Data:
- Player 1 (AimLabs): 49.9
- Player 3 (AimLabs): 51.9
- Player 5 (AimLabs): 67.4
- Player 7 (AimLabs): 69.9
- Player 9 (AimLabs): 79.3
- Player 11 (AimLabs): 83.1
- Player 13 (AimLabs): 93.4
- Player 15 (AimLabs): 120.2

Average Improvement for AimLabs:
- Player 1: 49.9 - 41.2 = 8.7
- Player 3: 51.9 - 42.9 = 9.0
- Player 5: 67.4 - 55.3 = 12.1
- Player 7: 69.9 - 61.2 = 8.7
- Player 9: 79.3 - 70.0 = 9.3
- Player 11: 83.1 - 74.2 = 8.9
- Player 13: 93.4 - 84.1 = 9.3
- Player 15: 120.2 - 111.3 = 8.9

Average improvement for AimLabs: (8.7 + 9.0 + 12.1 + 8.7 + 9.3 + 8.9 + 9.3 + 8.9) / 8 = 9.36

6.2 **NeuroTune |** ***AI-Enhanced Aim Trainer***
Day 1 KPM Data:
- Player 2 (NeuroTune): 42.1
- Player 4 (NeuroTune): 43.8
- Player 6 (NeuroTune): 56.8
- Player 8 (NeuroTune): 61.9
- Player 10 (NeuroTune): 71.5
- Player 12 (NeuroTune): 76.3
- Player 14 (NeuroTune): 88.5
- Player 16 (NeuroTune): 118.2

Day 7 KPM Data:
- Player 2 (NeuroTune): 55.7

- Player 4 (NeuroTune): 58.2
- Player 6 (NeuroTune): 74.4
- Player 8 (NeuroTune): 75.8
- Player 10 (NeuroTune): 85.1
- Player 12 (NeuroTune): 90.2
- Player 14 (NeuroTune): 102.1
- Player 16 (NeuroTune): 131.9

Average Improvement for NeuroTune:

- Player 2: 55.7 - 42.1 = 13.6
- Player 4: 58.2 - 43.8 = 14.4
- Player 6: 74.4 - 56.8 = 17.6
- Player 8: 75.8 - 61.9 = 13.9
- Player 10: 85.1 - 71.5 = 13.6
- Player 12: 90.2 - 76.3 = 13.9
- Player 14: 102.1 - 88.5 = 13.6
- Player 16: 131.9 - 118.2 = 13.7

Average improvement for NeuroTune: (13.6 + 14.4 + 17.6 + 13.9 + 13.6 + 13.9 + 13.6 + 13.7) / 8 = 14.3

6.3 Comparison of AimLabs Group and **NeuroTune** *Data*

- AimLabs players: Average improvement = 9.36
- NeuroTune players: Average improvement = 14.3
- Difference between AimLabs and NeuroTune: 14.3 - 9.36 = 4.94