Euphony Echo

Integrating Virtual Reality, Machine Learning, and Music Therapy to Enhance Emotional Expression and Anxiety Management for Individuals

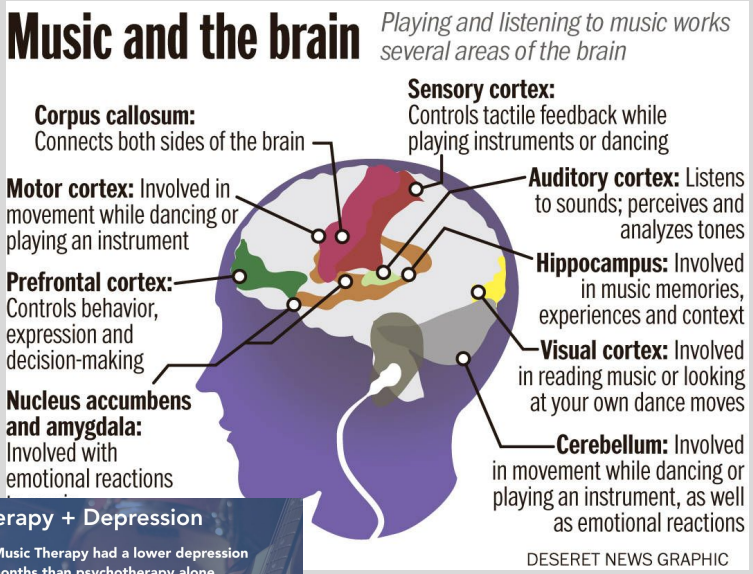
Alana Wu
Orange County School of the Arts

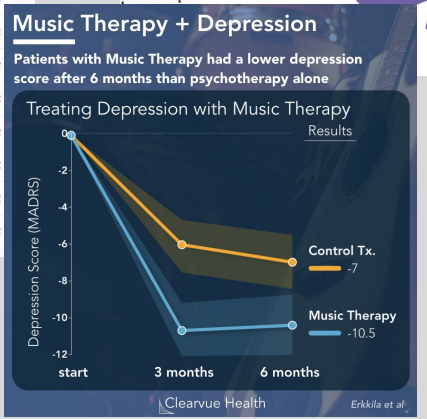
Motivation: Music and the Brain

Music Note To Frequency Chart

MixButton

NOTE	OCTAVE 0	OCTAVE 1	OCTAVE 2	OCTAVE 3	OCTAVE 4	OCTAVE 5	OCTAVE 6	OCTAVE 7	OCTAVE 8
C	16.35 Hz	32.70 Hz	65.41 Hz	130.81 Hz	261.63 Hz (A piano middle C)	523.25 Hz	1046.50 Hz	2093.00 Hz	4186.01 Hz (A piano's highest note)
C#/D♭	17.32 Hz	34.65 Hz	69.30 Hz	138.59 Hz	277.18 Hz	554.37 Hz	1108.73 Hz	2217.46 Hz	4434.92 Hz
D	18.35 Hz	36.71 Hz	73.42 Hz	146.83 Hz	293.66 Hz	587.33 Hz	1174.66 Hz	2349.32 Hz	4698.63 Hz
D#/E♭	19.45 Hz	38.89 Hz	77.78 Hz	155.56 Hz	311.13 Hz	622.25 Hz	1244.51 Hz	2489.02 Hz	4978.03 Hz
E	20.60 Hz	41.20 Hz (A bass's lowest note)	82.41 Hz (A guitar's lowest note)	164.81 Hz	329.63 Hz	659.25 Hz	1318.51 Hz	2637.02 Hz	5274.04 Hz
F	21.83 Hz	43.65 Hz	87.31 Hz	174.61 Hz	349.23 Hz	698.46 Hz	1396.91 Hz	2793.83 Hz	5587.65 Hz
F#/G♭	23.12 Hz	46.25 Hz	92.50 Hz	185.00 Hz	369.99 Hz	739.99 Hz	1479.98 Hz	2959.96 Hz	5919.91 Hz
G	24.50 Hz	49.00 Hz	98.00 Hz	196.00 Hz (A violin's lowest note)	392.00 Hz	783.99 Hz	1567.98 Hz	3135.96 Hz	6271.93 Hz
G#/A♭	25.96 Hz	51.91 Hz	103.83 Hz	207.65 Hz	415.30 Hz	830.61 Hz	1661.22 Hz	3322.44 Hz	6644.88 Hz
A	27.50 Hz (A piano's lowest note)	55.00 Hz	110.00 Hz	220.00 Hz	440.00 Hz	880.00 Hz	1760.00 Hz	3520.00 Hz	7040.00 Hz
A#/B♭	29.14 Hz	58.27 Hz	116.54 Hz	233.08 Hz	466.16 Hz	932.33 Hz	1864.66 Hz	3729.31 Hz	7458.62 Hz
B	30.87 Hz (A 5 string bass's lowest note)	61.74 Hz	123.47 Hz	246.94 Hz	493.88 Hz	987.77 Hz	1975.53 Hz	3951.07 Hz	7902.13 Hz





https://www.health.harvard.edu/newsletter_article/music-and-health
https://www.nccih.nih.gov/health/music-and-health-what-you-need-to-know

Existing Musical Therapy

- Thereoke
  - Geographical constraights and group setting
- SongBirds
  - Lack of home-based options
- Tang et al.’s Study
  - Lacks personal treatment & methodological inconsistencies

Problems of Current Musical Therapy

- Limited accessibility
- High cost
- Limited integration with technology
- Stigma and lack of awareness
- Adaptability for different needs

Integrating Technology with Music Therapy

AI & Machine Learning for Adaptive Therapy

Virtual Reality (VR) & Augmented Reality (AR) for Immersive Interaction

Biofeedback & Neurofeedback Integration

Expanding Accessibility via Mobile & Cloud-Based Platforms

Data Science for Therapeutic Insights

Solution: Euphony Echo

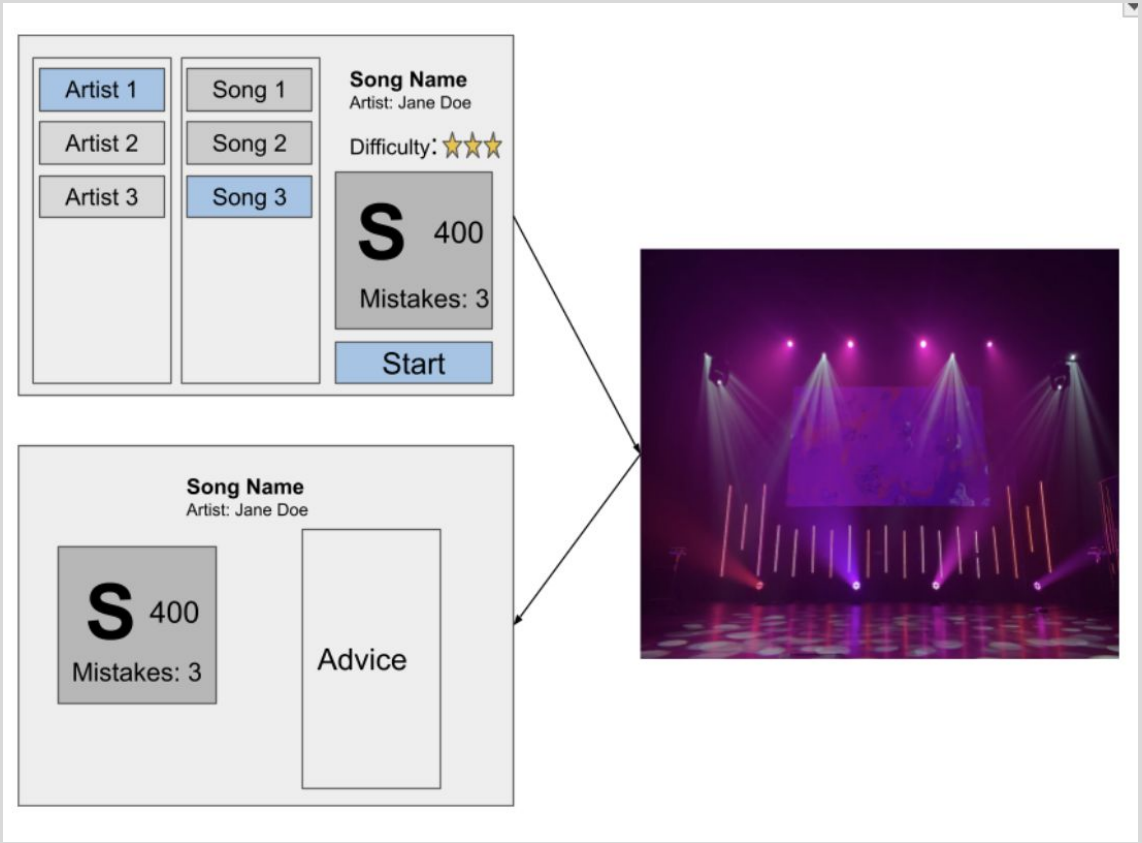
ABSTRACT

Performance anxiety, commonly known as stage fright, poses significant challenges for individuals aiming to enhance their singing abilities and stage presence [1]. To address this issue, we have developed a virtual reality (VR) simulation that replicates a concert environment, offering users a safe and controlled space to practice and improve their performance skills. The program integrates a Vocal Engine for real-time pitch analysis, an immersive VR environment that mirrors both on-stage and off-stage settings, and a diverse song library with synchronized lyrics [2]. Key challenges included ensuring accurate pitch detection, which we addressed by broadening the range of permissible pitch variations, and enhancing user immersion through detailed stage designs and dynamic movements. During experimentation, users engaged with the VR simulation across various scenarios, receiving immediate feedback to refine their vocal skills. Results indicated significant improvements in users' confidence and performance quality [3]. This innovative approach offers an accessible and effective solution for individuals seeking to overcome stage fright and develop their musical talents in a supportive virtual setting.

KEYWORDS

Singing, Unity, Virtual Reality, Scoring Algorithm

Solution: Euphony Echo



Built With:

Code Sample

```
0 references
void Start()
{
    // Initialize microphone input
    microphoneSource = gameObject.AddComponent<AudioSource>();
    microphoneSource.clip = Microphone.Start(null, true, 10, sampleRate);
    microphoneSource.loop = true;
    while (!(Microphone.GetPosition(null) > 0)) { }
    microphoneSource.Play();
    micSpectrum = new float[samples];

    // Initialize song source
    songSource = gameObject.AddComponent<AudioSource>();
    songSource.clip = songClip;
    songSource.Play();
    songSpectrum = new float[samples];
}

0 references
void Update()
{
    // Get spectrum data for both microphone and song
    microphoneSource.GetSpectrumData(micSpectrum, 0, FFTWindow.BlackmanHarris);
    songSource.GetSpectrumData(songSpectrum, 0, FFTWindow.BlackmanHarris);

    // Determine dominant frequency (pitch) for both
    float[] micFreq = DetermineDominantFrequencies(micSpectrum);
    float[] songFreqs = DetermineDominantFrequencies(songSpectrum);

    // Store pitch history
    micPitchHistory.Add(micFreq);
    songPitchHistory.Add(songFreqs);
    while (micPitchHistory.Count > AVERAGE_COUNT)
        micPitchHistory.RemoveAt(0);
    while (songPitchHistory.Count > AVERAGE_COUNT)
        songPitchHistory.RemoveAt(0);

    // Get averaged pitch values
    float[] avgMicFreqs = AveragePitches(micPitchHistory);
    float[] avgSongFreqs = AveragePitches(songPitchHistory);

    // Compare
    ComparePitches(avgMicFreqs, avgSongFreqs);
}
```

Code Sample

```
0 references
void Start()
{
    if (GameObject.Find("Selection") != null)
    {
        selector = GameObject.Find("Selection").GetComponent<SelectedSong>();

        if (selector.isInstrumental)
        {
            microphone.songClip = selector.selectedSong.instrumentalVersion;
        }
        else
        {
            microphone.songClip = selector.selectedSong.musicClip;
        }

        lyrics.text = selector.selectedSong.lyrics;

        endElements.GetChild(0).GetComponent<TextMeshProUGUI>().text = selector.selectedSong.songName;
        endElements.GetChild(1).GetComponent<TextMeshProUGUI>().text = selector.selectedSong.artist;
        if (selector.selectedSong.difficulty == song.Difficulty.Easy)
        {
            endElements.GetChild(3).GetChild(0).gameObject.SetActive(true);
            endElements.GetChild(3).GetChild(1).gameObject.SetActive(false);
            endElements.GetChild(3).GetChild(2).gameObject.SetActive(false);
        }
        else if (selector.selectedSong.difficulty == song.Difficulty.Normal)
        {
            endElements.GetChild(3).GetChild(0).gameObject.SetActive(true);
            endElements.GetChild(3).GetChild(1).gameObject.SetActive(true);
            endElements.GetChild(3).GetChild(2).gameObject.SetActive(false);
        }
        else if (selector.selectedSong.difficulty == song.Difficulty.Hard)
        {
            endElements.GetChild(3).GetChild(0).gameObject.SetActive(true);
            endElements.GetChild(3).GetChild(1).gameObject.SetActive(true);
            endElements.GetChild(3).GetChild(2).gameObject.SetActive(true);
        }
```

Code Sample

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[CreateAssetMenu(fileName="New Song", menuName="Song")]
4 references
public class song : ScriptableObject
{
    1 reference
    public string artist;
    1 reference
    public string songName;
    // public bool isInstrumental;
    1 reference
    public AudioClip instrumentalVersion;

    4 references | 1 reference | 1 reference | 1 reference
    public enum Difficulty { Easy, Normal, Hard }
    3 references
    public Difficulty difficulty;


    [TextArea(3, 10)]
    1 reference
    public string lyrics;

    1 reference
    public AudioClip musicClip;
}
```

artist.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[CreateAssetMenu(fileName="New Artist", menuName="Artist")]
public class artist : ScriptableObject
{
    public string artistName;
    public List<song> songs;
    public Color artistColor;
}
```

Demo

Experiment #1

Participant	Group	Pre-Test Stage Fright Score	Post-Test Stage Fright Score	Pre-Test Singing Accuracy (%)	Post-Test Singing Accuracy (%)	Engagement Score (1-10)
P1	Experimental	45	25	60	85	9
P2	Experimental	50	30	65	88	8
P3	Experimental	48	28	63	87	9
P4	Experimental	52	29	62	86	8
P5	Experimental	46	26	64	84	9
P6	Control	55	50	55	60	5
P7	Control	58	53	58	62	6
P8	Control	53	50	57	61	5
P9	Control	56	52	56	59	6
P10	Control	54	51	54	58	5

Experiment #1

Mean:
Pre-Test Stage Fright: 51.7, Post-Test: 39.4
Pre-Test Singing Accuracy: 59.4%, Post-Test: 73.0%
Engagement Score: 7.0/10

Median:
Pre-Test Stage Fright: 52.5, Post-Test: 40.0
Pre-Test Singing Accuracy: 59.0%, Post-Test: 73.0%
Engagement Score: 7.0/10

Lowest Values:
Stage Fright Post-Test: 25
Singing Accuracy Post-Test: 58%
Engagement Score: 5

Highest Values:
Stage Fright Pre-Test: 58
Singing Accuracy Post-Test: 88%
Engagement Score: 9

Experiment #2

Participant	Ease of Use (1-10)	Immersion (1-10)	Feedback Effectiveness (1-10)	Song Variety (1-10)	Overall Enjoyment (1-10)
P1	9	8	8	6	9
P2	7	8	8	8	9
P3	9	9	8	8	8
P4	9	8	8	8	9
P5	7	9	7	7	8
P6	7	9	7	9	9
P7	9	9	8	9	8
P8	8	8	8	9	9
P9	9	9	7	9	9
P10	9	8	7	8	8

Experiment #2

Mean Scores:
Ease of Use: 8.3
Immersion: 8.5
Feedback Effectiveness: 7.6
Song Variety: 8.1
Overall Enjoyment: 8.6

Median Scores:
Ease of Use: 9.0
Immersion: 8.5
Feedback Effectiveness: 8.0
Song Variety: 8.0
Overall Enjoyment: 9.0

Lowest Scores:
Ease of Use: 7
Immersion: 8
Feedback Effectiveness: 7
Song Variety: 6
Overall Enjoyment: 8

Highest Scores:
All dimensions :9

Future Development

- **Rhythm games**
- **Guided exercise (breathe control, warm ups)**
- **Expanded song catalog**
- **Larger range of environments**

Conclusion

- Immersive musical experience
- Facilitates emotional expression
- Anxiety management

References

[1] Hillecke, Thomas, Anne Nickel, and Hans Volker Bolay. "Scientific perspectives on music therapy." Annals of the New York Academy of Sciences 1060.1 (2005): 271-282.

[2] Wohlgenannt, Isabell, Alexander Simons, and Stefan Stieglitz. "Virtual reality." Business & Information Systems Engineering 62 (2020): 455-461.

[3] Anthes, Christoph, et al. "State of the art of virtual reality technology." 2016 IEEE aerospace conference. IEEE, 2016.

[4] Koelsch, Stefan. "A neuroscientific perspective on music therapy." Annals of the New York Academy of Sciences 1169.1 (2009): 374-384.

[5] Keltner, Dacher, et al. "Emotional expression: Advances in basic emotion theory." Journal of nonverbal behavior 43 (2019): 133-160.

[6] Kendall, Roger A., and Edward C. Carterette. "The communication of musical expression." Music perception .2 (1990): 129-163.

[7] Lena, Jennifer C., and Richard A. Peterson. "Classification as culture: Types and trajectories of music genres." American sociological review 73.5 (2008): 697-718.

[8] Vechtomova, Olga, Gaurav Sahu, and Dhruv Kumar. "Lyricjam: A system for generating lyrics for live instrumental music." arXiv preprint arXiv:2106.01960 (2021).

[9] Li, Xiaoye S. "An overview of SuperLU: Algorithms, implementation, and user interface." ACM Transactions on Mathematical Software (TOMS) 31.3 (2005): 302-325.

[10] Li, Erica XN, Dmitry Livdan, and Lu Zhang. "Anomalies." The Review of Financial Studies 22.11 (2009): 4301-4334.

[11] Leung, C. M., et al. "Karaoke therapy in the rehabilitation of mental patients." Singapore medical journal 39 (1998): 166-168.

[12] Grocke, Denise, Sidney Bloch, and David Castle. "The effect of group music therapy on quality of life for participants living with a severe and enduring mental illness." Journal of music therapy 46.2 (2009): 90-104.

[13] Tang, Qishou, et al. "Effects of music therapy on depression: A meta-analysis of randomized controlled trials." PloS one 15.11 (2020): e0240862.

[14] Seymour, Neal E. "VR to OR: a review of the evidence that virtual reality simulation improves operating room performance." World journal of surgery 32 (2008): 182-188.

[15] Gonçalves, Guilherme, et al. "Systematic review of comparative studies of the impact of realism in immersive virtual experiences." ACM Computing Surveys 55.6 (2022): 1-36.

Questions? Comments?
Thank You!

Email: alanawu87@gmail.com

EuphonyEcho code repo: https://github.com/alanawuu/VR-singing