AIR-PIANO – Python Based Virtual Piano

Prof. Kapil Mundada[1], Apoorva Kunte[2], Nitya Munshi[3], and Sahil Mangle[4]
[1,2,3,4] *Instrumentation and Control Dept., Vishwakarma Institute of Technology, Pune, India*

***Abstract*—The fusion of computer vision and gesture recognition has enabled novel forms of human-computer interaction, especially in creative domains like music. Air Piano is a virtual instrument that lets users simulate piano playing through mid-air hand gestures, using only a webcam. MediaPipe handles real-time hand tracking[1], while OpenCV processes video input[2] and Pygame plays mapped audio notes for each finger. Gesture-based controls enable volume adjustment and musical scale switching, with voice feedback provided via pyttsx3. This system offers an engaging, touchless way to explore musical expression, showcasing the potential of computer vision in virtual instrument design.**

***Index Terms*—Virtual musical instrument, gesture recognition, hand tracking, computer vision, MediaPipe, OpenCV, Pygame, touchless interaction, human-computer interaction, music technology.**

I. INTRODUCTION

Recent progress in computer vision and gesture recognition has vastly improved contactless human-computer interaction, in general, and performance in digital music in particular. Unlike traditional instruments, touchless musical systems avoid physical sensors or MIDI, offering both greater accessibility and expressive potential. Real-time tracking of the hand brings new possibilities for interaction with virtual instruments. [1], [7]

This paper presents Air Piano, a virtual piano played entirely with the hands. It uses a webcam to capture hand movements, applies MediaPipe for hand landmark detection, and uses OpenCV for video processing. The system maps each finger to a specific note triggered by vertical motion of the finger.

To enhance interactivity, Air Piano includes gesture-controlled volume adjustment (thumbs-up/down), musical scale switching, and voice feedback via pyttsx3 for user guidance. Designed for ease of use and responsiveness, it demonstrates how widely available technologies can be used to build immersive, low-cost digital instruments.

This work contributes to the growing area of virtual musical interfaces and opens several perspectives for new educational, recreational, and therapeutic applications based on gesture-controlled music systems.[2],[7]

II. LITERATURE REVIEW

Zhang et al. [1] introduced MediaPipe Hands, a lightweight real-time hand landmark detection model capable of tracking 21 3D hand keypoints using only an RGB camera. This framework enables efficient hand gesture recognition without requiring depth sensors or specialized hardware, making it widely used in interactive computer-vision applications.

Bradski [2] introduced OpenCV, which is an open-source library of computer vision that can be used in handling real-time images, hand segmentation, gesture tracking, and feature extraction. It has had great flexibility and performance; hence, it acts as a standard tool in the development of gesture-based interfaces.
Amprimo et al. [3] conducted an accuracy and clinical applicability assessment of the MediaPipe hand-tracking framework, presenting high reliability related to fine-grained finger landmark detection and, therefore, supporting its use in precise motion-controlled interfaces, including virtual musical instruments.

Sung et al. [4] proposed an on-device real-time gesture recognition system optimized for low-latency execution, demonstrating efficient deployment of gesture-driven interaction on resource-constrained devices. Verma et al. [5] further combined MediaPipe with convolutional neural networks to improve hand-

shape recognition performance for sign-language interpretation, emphasizing the potential of computer-vision-based gesture systems for accessibility applications.

In the field of virtual musical interaction, Valbom and Marcos [6] designed a prototype of an immersive musical interface that explored new ways of expressive musical performance independent of the use of classic physical instruments. Mäki-Patola et al. [7] conducted experiments using virtual instruments and discussed interaction challenges regarding latency and controlling gestures that are still applicable to the modern touchless music systems.

The reacTable, developed by Jordà et al. [8], demonstrated tangible and interactive tabletop musical control environments that enabled collaborative performance, showing how non-traditional input devices can enhance creativity. LaViola et al. [9] explored hands-free navigation techniques in virtual environments that relied on gesture-based control, illustrating alternate forms of natural interaction. Harrison et al. [10] proposed Skinput, a system turning the human body surface into an input interface, demonstrating innovative alternatives for gesture input beyond classic hardware.

Overall, previous studies reveal a strong trend toward vision-based, touch-free accessible interaction interfaces. These collectively contribute to the development of systems like Air Piano, which merges MediaPipe-based hand tracking with OpenCV gesture recognition and real-time audio feedback for allowing non-contact, musical performance applicable to both education and assistance.

III. METHODOLOGY

The Air Piano system is developed for an emulated piano experience with hand gesture recognition, audio feedback in real-time, and low hardware dependency. The approach consists of four main phases: input acquisition, hand gesture recognition, mapping of hand-gesture-based interaction, and audio-visual feedback. A Python implementation makes use of various open-source libraries like MediaPipe, OpenCV, Pygame, and pyttsx3 for speech synthesis.

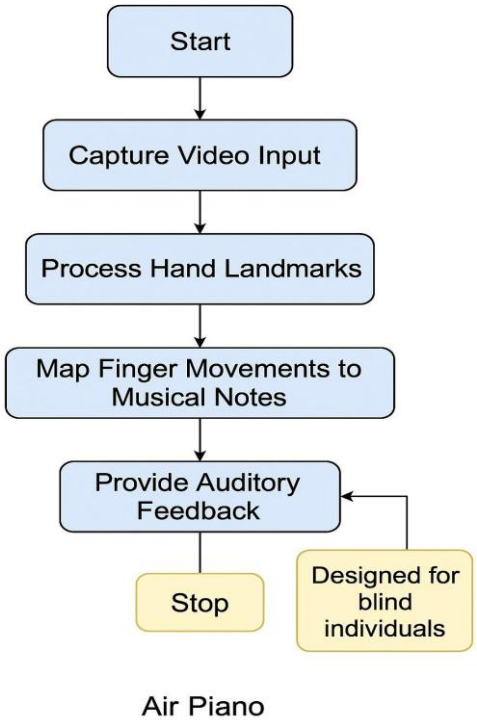

Fig 1. Flowchart

Figure 1 describes the flow of the proposed system.

A. Input Acquisition

A typical webcam is used as the main input device. It records real-time video frames that are processed via the OpenCV library. The video stream is flipped for easy interaction and resized for uniform processing.

B. Hand Tracking and Landmark Detection

Google's MediaPipe Hands solution is used to detect and track hand landmarks. The model detects 21 key points per hand, such as fingertips and joints, and can track up to two hands at a time. The system utilizes these landmarks to calculate the position and movement of fingers. Certain indices (e.g., 4, 8, 12, 16, 20) refer to the tips of the thumb, index, middle, ring, and pinky fingers respectively.

C. Gesture Interpretation and Mapping

Every identified gesture maps to a musical note, volume control, or guidance directive. The interpretation mechanism includes the following:

Note Playback:

Downward Y-direction finger tip movement past a threshold ($\Delta y > 0.02$) initiates a note.

A cooldown mechanism (200 ms) avoids repeated triggering.

Notes are assigned by reference to an off-the-shelf scale dictionary (default and minor scales), which maps every finger to a musical note (e.g., index → D4).

Volume Control: Volume is controlled via explicit gestures:
Thumbs Up: All fingers closed except thumb, with thumb above its base joint.
Thumbs Down: Same hand with thumb below its base joint.

Pressing the 'M' key alternates between default and minor music scales.

On-screen display and text-to-speech announcement (pyttsx3) are made for the current scale.

Hand Position Guidance:
When the wrist landmark crosses screen edges (x or y < 0.2 or > 0.8), audio feedback is issued to correct the hand position.

If a hand is not detected for more than 10 seconds, the user is reminded to put his/her hand in front of the camera.

D. Audio and Visual Feedback
Pygame mixer module is utilized to preload .wav files representing musical notes. When the correct gesture is detected, the respective note gets played. The interface gives real-time visual feedback by superimposing hand landmarks on the video stream and showing current scale and volume levels. The pyttsx3 engine speaks up key interactions like volume and guidance to improve accessibility and user experience.

V. RESULTS AND DISCUSSION

The Air Piano system was successfully implemented using Python with OpenCV, MediaPipe, and Pyttsx3 for voice feedback. During testing, the system accurately detected hand gestures and mapped each finger movement to corresponding musical notes in real-time. Both major and minor scales were supported, and users could switch between them using a simple keyboard toggle.
The system also integrated voice guidance for volume control and hand positioning. Volume adjustments by using thumbs-up/down gestures worked reliably, using a delay buffer to prevent them from false triggering. In practical tests, users were able to play melodies at different speeds, while visually impaired users received timely audio cues that improved orientation. The frame rate remained above 20 FPS on a mid-range laptop, which provided smooth visual feedback.

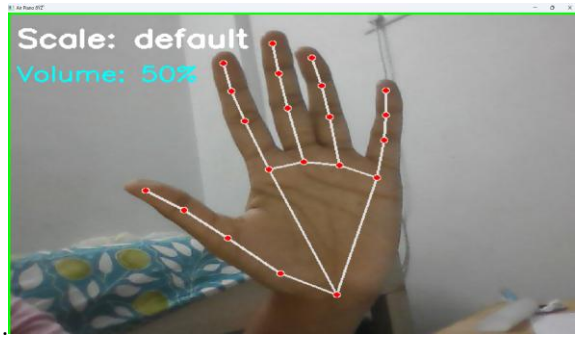

Fig 1 - This figure illustrates the hand gesture along with annotated landmark points.

VI. CONCLUSION

This project showcases an innovative, non-contact virtual piano interface that empowers visually impaired persons to play music with only hand gestures and voice feedback. By fusing real-time hand tracking with intuitive audio guidance, this system removes the dependency on visual interfaces or physical keys. Application proves that this implementation is effective for enabling basic musical interaction and the playing of notes in real-time, with proof of concept for inclusive music education and recreation.

VII. FUTURE SCOPE

- Multi-note and Chord Detection: Extend the system to support detection of simultaneous multi-finger input for playing chords.
- Recording and Playback of Sessions: Record the sessions for users to play back later for personal practice or sharing.
- Octave and Instrument Switching: Incorporate gesture-based control for shifting octaves and

changing sound banks-for example, from piano to guitar.

- Mobile or AR/VR Deployment: The system shall be ported to both smartphones and AR glasses for portable usage.
- Multi-language Voice Support: Providing voice guidance in various languages can enhance accessibility. A gamified learning mode that features beginner-friendly tutorials and challenges for learning music fundamentals.
- Haptic Feedback: Integrating with wearables provides a physical response that helps reinforce correct placement of fingers and confirmation of notes.

REFERENCES

[1] F. Zhang et al., "MediaPipe Hands: On-device real-time hand tracking," arXiv preprint arXiv:2006.10214, 2020.

[2] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.

[3] G. Amprimo et al., "Hand tracking for clinical applications: Validation of the Google MediaPipe Hand (GMH) and the depth-enhanced GMH-D frameworks," arXiv preprint arXiv:2308.01088, 2023.

[4] G. Sung et al., "On-device real-time hand gesture recognition," arXiv preprint arXiv:2111.00038, 2021.

[5] R. Verma et al., "Enhancing sign language detection through MediaPipe and convolutional neural networks (CNN)," arXiv preprint arXiv:2406.03729, 2024.

[6] L. Valbom and A. Marcos, "An immersive musical instrument prototype," IEEE Computer Graphics and Applications, vol. 27, no. 2, pp. 76–79, 2007.

[7] T. Mäki-Patola et al., "Experiments with virtual instruments," in Proc. Int. Conf. New Interfaces for Musical Expression (NIME), 2005, pp. 11–16.

[8] S. Jordà et al., "The reacTable: Exploring the synergy between live music performance and tabletop tangible interfaces," in Proc. 1st Int. Conf. Tangible and Embedded Interaction, 2007, pp. 139–146.

[9] J. J. LaViola Jr. et al., "Hands-free multi-scale navigation in virtual environments," in Proc. Symp. Interactive 3D Graphics, 2001, pp. 9–15.

[10] Harrison et al., "Skinput: Appropriating the body as an input surface," in Proc. SIGCHI Conf. Human Factors in Computing Systems, 2010, pp. 453–462.