INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

AIR DRUMS PLAYING DRUMS USING COMPUTER VISION

[1]E. SHIRISHA, [2]RAGIDDI POOJITHA, [3]KANUKUNTLA SREEJA, [4]JELLA ANILKUMAR

[1]Assistant Professor, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad,

sirishaedikoju@gmail.com

[2,3,4]BTech Student, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad

poojitharagiddi10@gmail.com , kanukuntlasreeja040@gmail.com ,anilkumarjella25@gmail.com

ABSTRACT: *The cost of a drum set is an investment that most aspiring drummers would eventually need to shoulder in order to continue their craft. What this research aims to do is to hasten the introduction of drummers to the drumming experience without the costs, and also to allow for drummers to be able to practice, at least casually, without a full drum set. This thus allows the experience of drumming to a wider audience. The solution we explore is the development of a prototype virtual drum set that would only require users to have a laptop with a camera, along with easily accessible markers representing the tips of drum sticks and knee movement, such as colored papers. OpenCV based on Python was used to implement this, and used the concept of color-biased blob detection for detecting the markers.*

I. INTRODUCTION

The drums is the most popular percussive instrument in the music industry today. Many beginners who aspire to be a percussive musician start out with learning how to play the drums. However, a typical drum set is usually expensive, takes a lot of space, and not easily transportable unlike other instruments such as the guitar or keyboard. Figure 1 shows the different components that consist of a standard drum set. The goal of this study is to build a system that will enable aspiring drummers to be able to play and practice the drums on thin air, with the use of computer vision. The idea is to translate a video being captured of a user playing the virtual drums, considering realistic movements with an actual drum set, to an audio synthesis of appropriate drum samples in real-time. Making use of a laptop's built-in web camera, the project aspires to create an implementation that would make it easier for people lacking the funds or equipment, to practice and learn the actual drums.

II. LITERATURE SURVEY

Implements the virtual drums using a computer vision approach. The advantage of these systems is that a camera is very accessible. However, the algorithms may be complex and difficult to implement in real-time. The work done by Bering and Famador locates and tracks the position of the hands in the video without the need for drum sticks. However, their work does not involve the movement of the feet which is important in recognizing the bass drum and the hi-hat control. Another work implemented by Brown et. al. uses orange markers attached to the end of the drum sticks to be able to locate and track the movement of the drum sticks. Again, their work does not involve the movement of the feet. The work by Rojo also used colored markers and implemented a dynamics computation for the volume setting of the synthesized drum samples. The most popular and marketable implementation making use of computer vision makes use of a high speed camera and reflective balls attached to the end of drum sticks.

After requirement gathering, the team comes up with a rough plan of software process. At this step the team analyzes if a software can be made to fulfill all requirements of the user and if there is any possibility of software being no more useful. It is found out, if the project is financially, practically and technologically feasible for the organization to take up. There are many algorithms available, which help the developers to conclude the feasibility of a software project.

The feasibility study mainly concentrates on bellow five mentioned areas. Among this Economic Feasibility Study is most important part of the feasibility analysis and Legal Feasibility Study is less considered feasibility analysis.

III. PROPOSED SYSTEM

The goal of this study is to build a system that will enable aspiring drummers to be able to play and practice the drums on thin air, with the use of computer vision. The idea is to translate a video being captured of a user playing the virtual drums, considering

ISSN: 2057-5688

realistic movements with an actual drum set, to an audio synthesis of appropriate drum samples in real-time. Making use of a laptop's built-in web camera, the project aspires to create an implementation that would make it easier for people lacking the funds or equipment, to practice and learn the actual drums.

These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

SYSTEM ARCHITECTURE

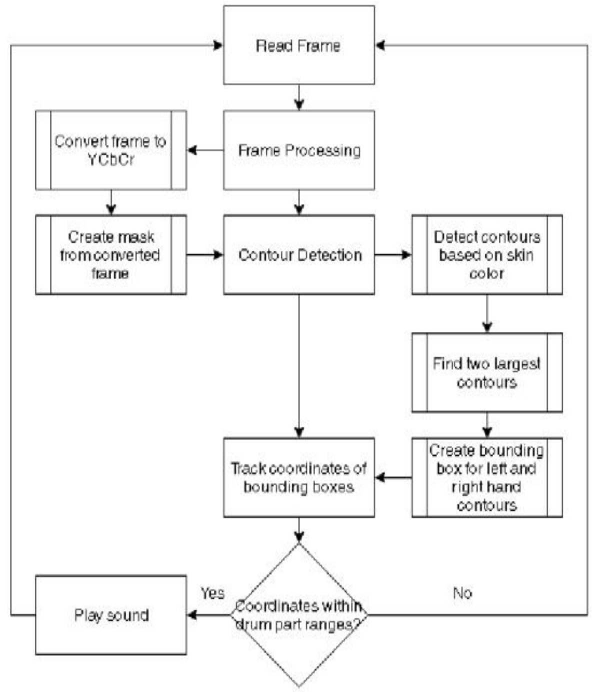


Fig.1 System architecture

Technical architecture

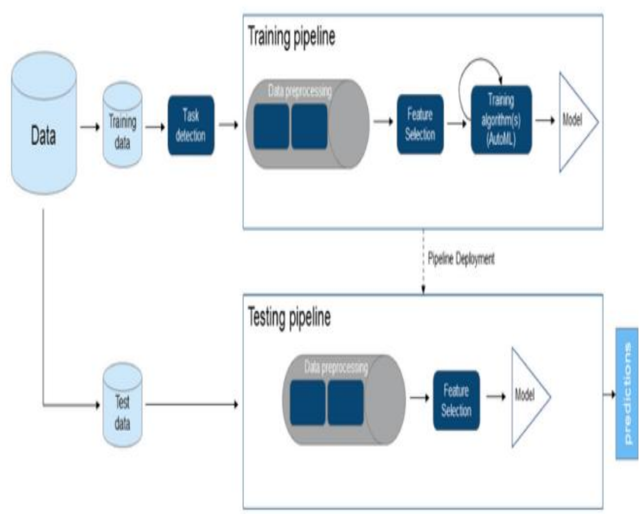


Fig.2 Technical architecture

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

METHODOLOGY

The prototype virtual drum system was developed using OpenCV 3.4.5 running on Python 3.6.8. The methodology is divided into three major steps: (1) Object Detection and Tracking, (2) Event Detection, and (3) Drum Sound Synthesis.

1) Object Detection and Tracking

For what can be said to be a complete camera-based drum system, four keypoints are needed to be detected and tracked: the two tips of the drum sticks, and the two feet for bass drum and hi-hat control. These four points should be used to control the sound that the drum system will produce. For this prototype, we limit the scope to tracking the two ends of the drum sticks, and the right knee of a user. The movement of the knee instead of the foot was chosen for detection in order to allow a user of the system to be near the camera of his/her laptop. The removal of the detection of the left foot was done since its movement is usually used for the open and close state of the hi-hat. The implementation of this would be better suited after the perfection of accuracies of knee or foot movement detection, as this would lead to more accurate states of the hi-hat. It should be noted that the hi-hat is usually closed for most drum sequences thus making this design choice acceptable for a prototype. Figure 3 shows an example setup and camera view given the desired key points

Fig.3 The setup of a user playing the virtual drum set

2) Blob Detection:

The prototyped system detects the three keypoints through blob detection based on color. As such, this method assumes that there are three different color ranges for the three keypoints, and that there are three differently colored markers attached to the ends of the two sticks and the right knee of a user. It should be emphasized that these colors should be different from each other,

ISSN: 2057-5688

and that the camera view should not contain items of the same color bigger than the sizes of the markers as seen by the camera. For each frame captured by a camera, thresholding is done for each keypoint to be identified. Dilation is done to each set of extracted pixels in order to make the extracted pixels be more blob-like. The largest blob for each threshold is determined to contain desired keypoint, and then the center of this blob is computed in order to get the position of the keypoint. Figure 3 shows an example frame where blob detection was implemented

Event Detection

Two methods were explored for detecting the event of striking a drum pad on the virtual drum set: (1) By Acceleration Computation, and (2) By Points Comparison. 1) By Acceleration Computation: With this explored method, we determine event of a drum pad being hit by calculating the dynamics of the points tracked by our system. Specifically, we calculate the position p(n), velocity v(n), and acceleration a(n) of the three keypoints. The calculation of the dynamics is given by the following equations. p(n) = x(n) y(n) (1) v(n) = kp(n) − p(n − 1)k ∆t (2) a(n) = v(n) − v(n − 1) ∆t (3) where ∆t is the time step corresponding to the number of frames per second (FPS) captured by the camera. For simplicity, we set this parameter to be a constant value since the time step between frames is varying due to the processing done in between frames captured. Since the points detected and tracked by the camera contain some measurement noise, we implement a Kalman filter for the calculation of the dynamics. Through the use of this filter, instead of obtaining the current position p(n) directly through computations made with the next frame captured by the camera, we make an estimated prediction of the the current position pˆ(n) using past data. The estimate of this position is given by the following equation: pˆ(n) = K(n) · p(n) + (1 − K(n)) · pˆ(n − 1) (4) where K(n) is the Kalman gain at time n. We use this estimated position instead of the measured position in calculating the dynamics of the points. Once the dynamics have been calculated, we detect that a drum pad trigger has occurred when a certain point has achieved the following criteria:

1) Computer Vision Algorithms Used:

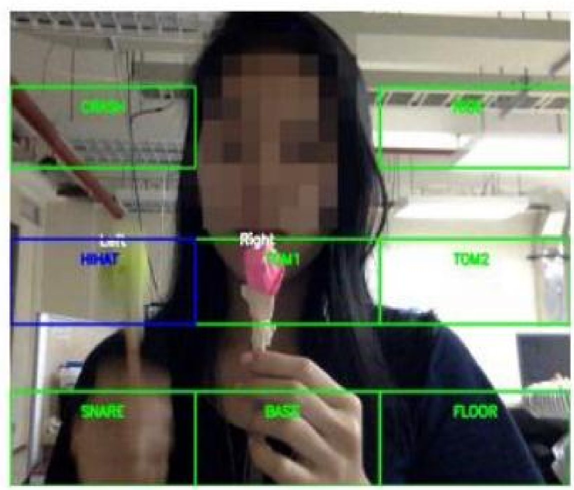
INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

For the algorithms used, the over-all system used blob detection by color for detecting the three desired keypoints since it is a simple and efficient technique useful for real-time applications. For event detection, we detect an event By Points Comparison for the two drum sticks, while for the right knee, we detect an event By Acceleration Computation. The reasoning behind this is that detecting By Points Comparison tends to be more robust since the requirement is just to detect where the keypoints for the sticks were in the previous frames, and then compare where they are in the current frame. And as for the right knee, we wanted a user to not be restricted to keeping his/her knee positioned at a certain location to comply with a bounding box. Through using By Acceleration Comparison, a user is then free to just move the knee up and downwards.

2) Modes of Play:

Two modes of play were included in our system. The user is asked to play the virtual drum set either by: (1) using two drum sticks, or (2) using two drum sticks with the right knee for bass drum control. For the first mode of play, the user can strike any of the drum pad in Figure 4 including the bass drum. For the second mode of play, the user can strike any of the drum pads with the sticks except for the bass drum, but the user can play the bass drum by moving the right knee similar to the movement for an actual drum set. Figure 4 shows the interface of a user playing the first mode of the system, while Figure 5 shows the second mode of the system. 3) Initial Calibration: Before the user can start playing the virtual drum set, he/she is asked to calibrate the system for the colored markers attached to the end of the drum



ISSN: 2057-5688

Fig.4 Example display seen by user playing the first mode of virtual drum

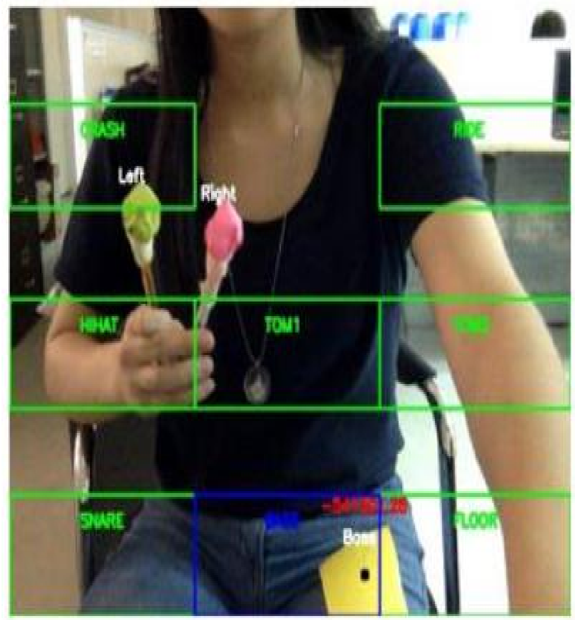
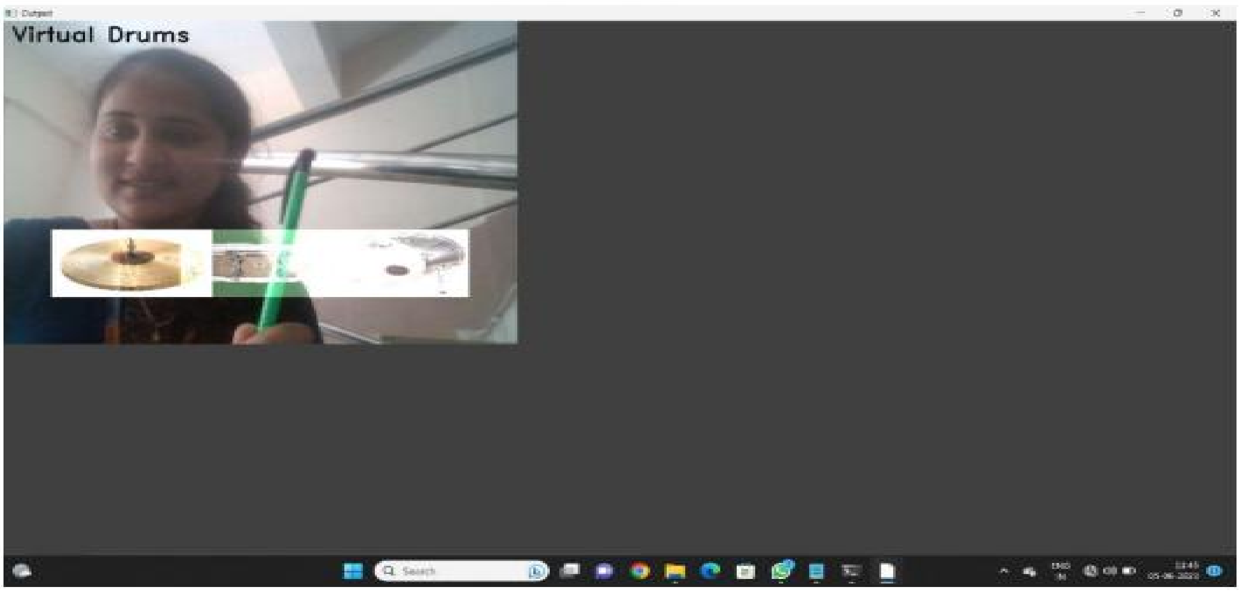


Fig.5 Example display seen by user playing the second mode of virtual drum

IV. RESULTS



V. CONCLUSION

We can say that we have achieved our goal of developing a prototype system for air drums, usable by beginner drummers and at a very low monetary cost. We have also built the first purely camera-based air drum system that includes the feet movement for bass drum control as far as we know. Moreover, the user interface of the system gives access to a more comprehensive set of drum components with the exclusion of the hi-hat due to its complicated dynamic properties. We have also built a more flexible system since it allows the usage of markers with different colors, shapes, and sizes. The tests for the detection-scheme also shows the feasibility of using color-based markers for real-time detection in everyday environments given controllable situations. The use of color based detection for real-time detection is simplistic, but due to its speed it is thus viable for the goal of gaining the fastest hits per minute possible in real time. We are able to achieve an estimated 513 hits per minute for the triggering of the drum pads. The conversion of the current code base to C++ is something to be explored in making the code run faster, and thus increase our hits per minute. Further refinement of the algorithms used for whether a drum pad was hit or not, along with the knee movement detection is something that is also needed to be done. As our tests show, we need to further refine them to achieve 100% usability even if it is just for casual drumming, as reliability of musical instruments is the most important factor for playing musical instruments. The achievement of this would also allow for the inclusion of hi-hat control which would then make the standard drum kit experience complete. Improvements to the user interface and user experience of the application is also something to be done for future work as the virtual drum system gets out of the prototype stage. Further development on using it as a USB controller and MIDI controller, would depend on how accurate the system as it is further developed could be. However, those purposes are indeed use cases that have potential for this system.

REFERENCES

[1] ”Janne”, “Standard 5-Piece Drum Kit Parts,” 2018. [Online]. Available: https://drumfortress.com/standard-5-piece-drum-kit-parts/

[2] “Freedrum: The Drumkit That Fits in Your Pocket,” 2017. [Online]. Available: https://www.kickstarter.com/projects/freedrum/freedrumthe-drumkit-that-fits-inyour-pocket

[3] S. R. F. Bering and S. M. W. Famador, “Virtual Drum Simulator Using Computer Vision,” pp. 370–375, 2017.

[4] B. Brown, D. Quenneville, and M. Shashoua, “CCS453 Computer Vision Spring 2016 Final Project: Air-drumming through Video Object-detection,” 2016. [Online]. Available: http://www.cs.middlebury.edu/{~}dquenneville/cs453/finalproject/

[5] F. Rojo, “Air Drums Using OpenCV and Python,” 2012. [Online]. Available: https://www.youtube.com/watch?v=MAnWwxTjL3k

[6] “Aerodrums,” 2019. [Online]. Available: https://aerodrums.com/ home/

[7]Prasadu Peddi (2017) “Design of Simulators for Job Group Resource Allocation Scheduling In Grid and Cloud Computing Environments”, ISSN: 2319-8753 volume 6 issue 8 pp: 17805-17811.