Music Manipulation through Gesticulation

Authors: Garrett Fosdick and Jair Robinson

Adviser: Jose R. Sanchez

Bradley University Department of Electrical and Computer Engineering

10/15/15

EXECUTIVE SUMMARY

There is a lack of dynamic and truly immersive opportunities available for people to experience their personal music libraries. Most digital and analog music players deliver a static play through of songs, disallowing the listener to manipulate the music to their liking in an enjoyable and intuitive fashion. Thus, the project proposed is an all-in-one system that accepts incoming audio and allows the user to apply a variety of effects to their liking with the use of multiple hand gestures that the system will recognize. The system for Musical Manipulation through Gesticulation is intended for personal and public entertainment. The system recognizes gestures from several feet away, and connects these gestures to audio effects that are applied to music being streamed through the system. This allows for two main features: the ability to alter music over a distance and an increased interactivity with the user.

Each audio effect will be tied to a recognized gesture. Through the usage of multiple gestures, the user will apply audio effects such as a low pass filter, high pass filter, chorus, distortion, and reverb to the input audio. The song will play from start to finish as normal, but at any point within the song's duration, the user can signal an audio effect to occur by the appropriate gesture. Even more, a motion to clear all effects, and a motion to control settings within an audio effect such as the cutoff frequency for the low-pass and high-pass filters, is at the users' control as well.

The implementation of the system will occur within the Bradley University computer labs for development, and outdoors and in various university rooms for testing. The experimentation and initial testing will be done using the program MATLAB. Full system implementation will occur on a Kinetis based Freescale Tower with a Leopard USB camera attachment. The full implementation will also use the Kinetis IDE for programming of the Kinetis board.

The expenses for the project come entirely from the hardware, even though it does contain software components. MATLAB is already owned by the university, and thus causes no expense to use, and the Kinetis IDE is free for download on their website. On the hardware end, there are two sources of expense, the Kinetis Freescale Tower, and the Leopard Imaging USB Camera. The Freescale Tower costs $294 and the USB Camera costs $92. In total, implementation of the project will cost $386.

Product performance will be judged using four categories. These four categories are latency, entertainment from use, ease of learning, and comfort from use. The latency of the product is the highest priority. The latency is how long it takes for the system to respond to user input. The goal for latency performance is to keep it as low as possible. The best possible result for latency is to have the latency be imperceptible to the user. Entertainment from use of the system is important, as it is designed to be an entertainment item. If it is not fun to use, the system will fail to solve one of the main problems it was addressing. Ease of learning's purpose is to make it easier for the user to learn how to use the system. The best-case scenario is to make it so the system is capable of being used immediately with no instruction. Finally, comfort of use means that the system is capable of being used without having to make any awkward feeling gestures. Design tradeoffs will be decided using these four performance factors.

ABSTRACT

Digital music is widely popular across the world due to its convenience and cost. This change of medium is flawed though in its loss of the dynamic and interactive nature present in music's analog form. This all in one system allows the user to manipulate audio from a distance by using hand gestures, providing an unparalleled interactive and dynamic personal music experience for the user. User interface is based around interactions with the system's camera. The images from the camera footage get compared to templates to recognize gestures. These gesture comparisons are made more dynamic and capable of handling distortions in the sequences time by using dynamic time warping. Without this it would be much harder to perform a gesture due to a strict definition. The templates for comparison are made from a training system to allow for more natural feeling gestures and a more dynamic system compared to the hard coded option. When a gesture is recognized, it's used to determine which effect to apply to the audio. All of these stated functions must be handled in real time for the system to be useful. The system hardware uses a mixture of digital signal processing integrated with controls.

TABLE OF CONTENTS

I. INTRODUCTION AND OVERVIEW

A. *Problem Background*:

The audio arts have always been important. Music in particular has been a powerful source of emotions, bonding, and beauty. The problem though, is that digitally stored music is static in the way that listeners experience it. One play through of a song will be exactly the same as the second third or fourth in the current listener experience. In addition, digitally stored music currently offers no interactivity with its listeners. At the moment, there are not any obvious mainstream or common methods to enjoy music exactly the way proposed in this document. Currently, for mobile devices there is an application called Brainwave that allows the user to control media players' playback functions (stop, play, and volume) with hand gestures [1]. There is an application named ControlAir for MacBooks that use the camera for gesture control of ITunes [2]. The most evolved is Geco, a package for MIDI expansion through hands, which requires a Leap Motion controller and MacBook and is tailored more towards music creation [3].

B. *Problem Statement*:

The goal of the system is to expand the possible ways in which music can be enjoyed. This goal is to be fulfilled mainly by expanding the listener's interactivity in a fun and enjoyable way, allowing users of this device to alter and affect their listening experience with simple gestures. These gestures will be designed to be fun, comfortable, and easy-to-use for the user. In addition, the system will include multiple gestures for interaction, as well as multiple effects linked to those gestures, usable in a variety of settings. The characteristics are listed in Table I listed below.

TABLE I. A SUMMARY OF SYSTEM ATTRIBUTES

Characteristics	F	O	M	C
Manipulates Audio	✔			
Recognizes Hand Gestures	✔			
Obtains Video Images	✔			
Manipulates Audio Selected By User	✔			
Functions Under Some Lighting Variance	✔			
Recognizes Multiple Gestures	✔			
Applies Multiple Audio Effects	✔			
Displays Audio Information	✔			
Low Latency		✔		
Fun to Use		✔		
Easy to Use		✔		
Comfortable to Use		✔		
Video Camera Takes Video Images			✔	
Image Processing Determines Hand Gestures			✔	
Gesture Recognition Determines Audio Manipulation Types			✔	
Stationary System				✔

(F)unctions, (O)bjectives, (M)eans, (C)onstraints

II. STATEMENT OF WORK

Different aspects of the system's operations will be discussed. In a high-level manner, both the system's potential fulfillment of each objective and the processes from input to output will be described.

A. System Diagrams:

The system is fairly simple from the input and output perspective due it only having three inputs and two outputs as shown in Fig. 1. The first of the three inputs is the electricity that powers the system. The system will receive its power from an outside source, an outlet, which was chosen over an integrated power source such as batteries due to the system's stationary nature. The next input into the system will be light. The light is the analog form of the images that the system's camera will convert to data for the system's use. The final input is the audio data. The audio data into the system will be in the form of an MP3, and be used as the base audio for manipulation.

As for system outputs, as previously stated, there are only two. The first is the audio output. This audio will be the analog version of the manipulated audio data that was input originally. The other one will be visuals. These visuals will communicate things such as current filters, and the current place in the song.

Fig. 1. Overall system block diagram

The glass box diagram contains four main components. The first component is a camera, which will take in the light input, and produce image data from it. The second component is the embedded device that will process the visual and audio data, and output that data once the embedded device finishes manipulating the data. The final two main components are the speakers and display screen. These both convert the data to something the user can hear or see, the speakers of course creating the auditory component, and the display screen providing the visual component. All of these components will be powered by the electric input. The glass box diagram represents subsystem one and is all displayed in Fig. 2.

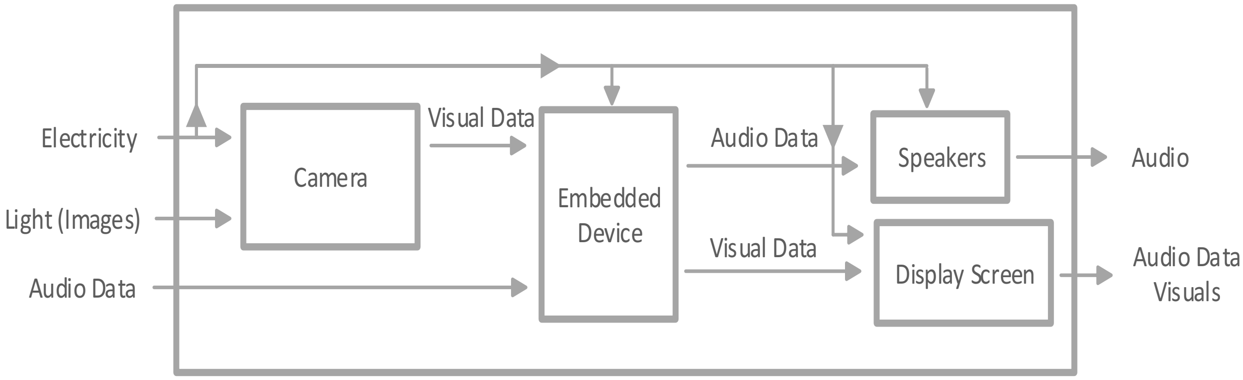


Fig. 2. System Glass Box Diagram

B. State Diagram:

The high-level system state diagram is shown in Fig. 3, which represents subsystem two and has five main states. The first state is the initialization of the system, preparing the system for use. After the initialization, the system becomes cyclical, performing processes in a loop. The system loop starts by obtaining the data to be processed by the system, followed by processing the image portion of the data, and then manipulating the audio data based on the previous step. The final step is to provide the data visually and audibly, before the loop begins again.

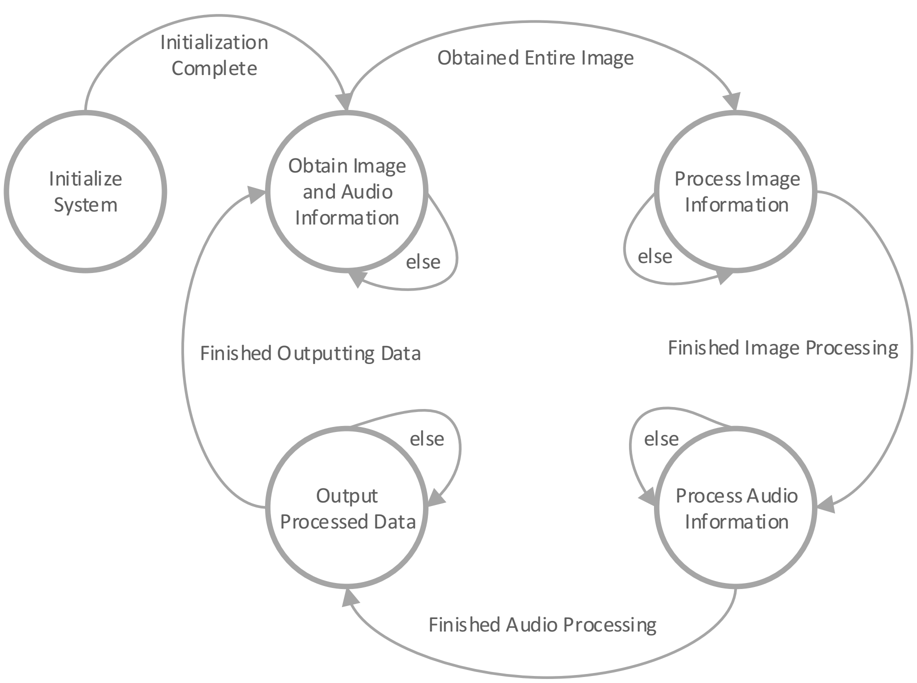


Fig. 3. State diagram of the system.

C. *Objectives:*

Measuring how well each objective of a system is completed is important to knowing the quality of the system that is being produced. To that point, Table V of Appendix A will be the measurement of with what quality the objectives are completed.

1) Latency:

The latency between user gestures and audio effects being applied to the audio can easily lead to the applied effects not working as the user intended. To that point the grading scale for latency is shown in Table VI of Appendix A.

2) Entertainment from Use:

The main goal of this system is to create a more enjoyable way to listen to music. Due to this, it is very important to measure how fun the system is to use. Table VII of Appendix A displays the metric in which fun will be measured. These measurements will be taken based off averages from user testing, and user feedback due to its subjective nature.

3) Ease of Use:

Due to the importance of the user being able to easily use the system, it must be considered under the objectives. Table VIII of Appendix A displays how the systems difficulty to use will be measured. The measure of difficulty to use is mainly based on the extent that a user will be able to immediately use the product without previous practice.

4) Comfort in Use:

The final objective has to do with the comfort with which the user can gesture. Awkward and unenjoyable gestures making the system unlikely for adoption when faced off against currently available tools. Table IX in Appendix A showing the metric of gesture comfort. Once more, this is highly subjective, and must be measured by user feedback.

D. *Functions:*

The system functions are the most important part of a system. These are what decide if the system actually works or not. Below are the functions of the system alongside their specifications that are required for a function to be considered working.

1) Function Specifications:

Manipulate Audio: Can apply the following effects to audio: low-pass filter, high-pass filter, chorus, reverb, and distortion.

Recognize Hand Gestures: Uses image processing to recognize hand motions with an at least 80% success rate, and cause the system to respond to them in less than a tenth of a second after the gesture. The system should check for movements 30 times a second, and use data from the most recent and past images to do so.

Obtain Video Images: Periodically retrieves colored images from the camera so that the system can process them. This should be performed at a frame rate of 30 times a second.

Manipulate Audio Selected by User: Allows the user to play and manipulate any MP3 file.

Function Under Some Lighting Variance: Capable of functioning under the effects of two different lighting types with an at least 80% success rate (Indoor and Outdoor lighting).

Recognize Multiple Gestures: Capable of recognizing five gestures: a clockwise circle, a counterclockwise circle, an "m" shape, an "8" shape, and a rocking "u" shape.

Connect Gestures to Audio: Capable of applying 5 different types of filters based on the gestures recognized by the system.

Display Audio Data: Displays current place in track, and currently added effects.

The functions are shown in Table II below with the corresponding system. The testing procedures are displaying in Appendix D.

TABLE II. REQUIREMENTS FOR FUNCTIONS AND SPECIFICATIONS

System Functions and Specifications		
System	**Function**	**Specification**
Overall	Obtain video images:	Retrieve at 30 frames per second
	Function under lighting variance:	Under two different lighting types with 80% success rate
	Display audio data:	Display current time in track and specific added effect
	Manipulate audio selected by user:	Allows user to play any MP3 File
Subsystem One	Connect gestures to audio:	Apply 5 different types of gestures to audio effects
	Manipulate Audio:	Apply five different effects to audio
Subsystem Two	Recognize hand gestures:	Check for movements 30 times a second with 80% success rate
	Recognize multiple gestures:	Recognize the five different programmed hand gestures

E. Design Approach and Method of Solution

The design of the system is based around research into related works, and experimentation in MATLAB.

1. Hand Tracking

The hand tracking method that will be used requires five stages to find the users hand. The first stage is Color Zeroing. In this stage, all colors that are similar to the users skin tone is reduced to zero. The further the color is from the skin tone, the higher it is. After this is done, the second stage is applied, which is thresholding of the color zeroed information. In this step, only colors close enough to zero are allowed to continue, making a filter for colors. Step three follows a similar process to step two, but instead of color, it uses motion. Anywhere there's enough motion, the image is permitted through. Step four then combines stages two and three, keeping the overlap between the two. In step four, the position of the hand is determined as the highest concentration of points permitted through. Finally, step five lowers down the area around the hand where future searches will occur to save on processing time.

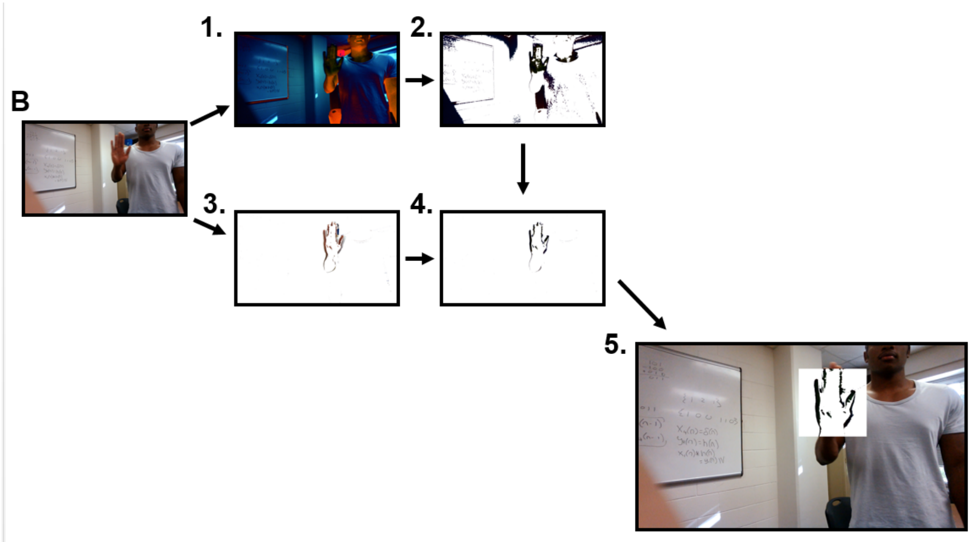


Fig 4. Hand detection method.

This method allows for easy hand detection, and has several advantages over other hand tracking methods. The advantages of this method is that it is capable of ignoring still hand colored objects such as heads, as well as moving objects that are not hand colored. Finally, it also saves considerably on processing power by only having to watch a small portion of the screen.

2. Gesture Recognition

Gestures occur at different rates and with some inaccuracy every time they occur. Due to this, gestures cannot just be simply compared to the template gesture. Instead, a method called Dynamic Time Warping is used. Dynamic Time Warping allows for time distortion to occur on both the gesture signal, and the template signal.

The Dynamic Time Warping system requires two different stages. The first stage is fairly simple. Two square matrixes are generated, one matrix is the gesture signal repeated horizontally, and the other is the template matrix repeated vertically. These matrixes are then subtracted from each other to create a matrix where every signal term is compared. An example of this can be found in Figure 5 below.

$$\underset{A}{\begin{bmatrix} 3 & 3 & 3 \\ 5 & 5 & 5 \\ 9 & 9 & 9 \end{bmatrix}} \quad \underset{B}{\begin{bmatrix} 4 & 2 & 3 \\ 4 & 2 & 3 \\ 4 & 2 & 3 \end{bmatrix}} \quad \underset{|A-B|}{\begin{bmatrix} 1 & 1 & 0 \\ 1 & 3 & 2 \\ 5 & 7 & 6 \end{bmatrix}}$$

Fig 5. Difference matrix example.

The second stage of Dynamic Time Warping is finding the cost of the bottom right square. The cost of the square is found by making a path from the top left to the bottom right of the matrix by expanding the currently cheapest route. The cheapest route always expands to any open squares below it, to the right of it, or diagonally of it. Once the bottom right corner is reach, the value in the bottom right corner is the cost of matching the two signals. If the cost is low enough, it's considered a match, and the effect tied to the gesture triggers. An example of this can be seen in Figure 6 below.

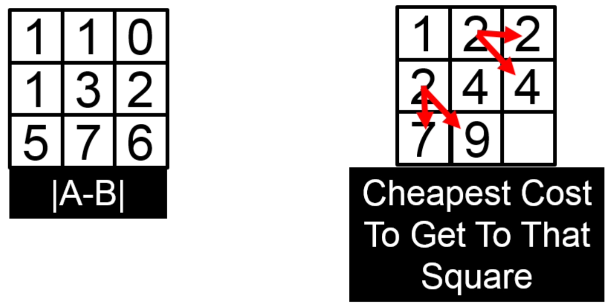


Fig 6. Cost matrix example.

This method allows for the system to handle gestures much more dynamically than the alternatives. Additional variation will also be added to increase processing efficiency. These variations will be costly route killing, where routes that exceed a certain cost are stopped, and a normal time bias. The normal time bias will make it so the signal is costlier when moving down or right instead of diagonally, thus biasing it towards the normal passage of time.

3. Audio Effects

Keeping in line with a foundation of simplicity, the audio effects that are implemented in the board will be simply coded and programmed into the system using C code. The audio effects include the low-pass filter, high-pass filter, chorus, reverb, and distortion, will be programmed and processed into the system, mainly on the audio. This allows for separate programming and testing from the visual aspect of the system when necessary.

Primary methods for testing will be subjective being that the techniques are primarily audio based. The user will rate the level of auditory satisfaction based on what they hear. Secondary testing results will be quantitative. These results will be the amount of times the appropriate audio effect is triggered from the applied gesture consistently and the processing time for the analog to digital conversion.

The testing of the audio and applied effects will be that if it sounds pleasing, distinct, and at a high enough volume, it will pass as acceptable. This is subjective, yet fits the overall tone that the project must provide a pleasant, enjoyable experience to users.

The testing of the analog to digital conversion will be that the lag needs to be 100 milliseconds or less. To calculate this, the clock cycles for each of the processes will be observed and recorded from the Kinetis IDE when programming the code onto the board. The clock cycles will be calculated to equivalent real world time in the measurement of seconds. Each of the effects and processes will be optimized and manipulated accordingly to allow for an analog to digital conversion time that meets the requirement of overall processing lag being less than 100 milliseconds and unperceivable to the user.

F. Economic Analysis

The development costs for the system low overall. The hardware, overall, comes to a cost of $386.19 dollars and the software does not cost. This includes the Kinetis Freescale Tower priced at $294.25, which includes the tower elevator for $39.00, an audio module for $106.25, and the control board for $149.00. The Leopard Imaging USB camera comes in at $91.94.

For software, the Kinetis IDE (integrated development environment) is within the software that is free to download and install. During the testing and research stages, MATLAB is used, which is provided by lab computers in the Electrical and Computer Engineering Department. There are not any expected substantial future expenses. This is all shown in the Table III below.

TABLE III. TOTAL COSTS OF PROJECT

Cost of Materials	
Description	Costs
Kinetis Freescale Tower	$39.00
Audio module	$106.25
Control board	$149.00
Leopard Imaging USB camera	$91.94
Kinetis IDE	$0.00
Final Costs	**$386.19**

G. Project Timeline

The timeline for the project is mostly organized in the Gantt chart displayed in Appendix B, showing the milestones that are necessary to overcome from the project's beginning to the finalized project. The scheduling reveals that the project is spaced appropriately and is very realistic to complete, especially in the sense that there is an adequate amount of time that is allotted for the testing and improvement of the system at the end of the design phase. The higher-level view of the most important milestones in the order that they must be completed is shown in Table IV below.

TABLE IV. OVERVIEW OF MILESTONES WITHIN SCHEDULE

Major Milestones to Complete	
Milestones	Date Due
Test Audio Processes in MATLAB	9/30/15
Test Video Processes in MATLAB	10/8/15
Implement Audio Effects on Hardware	11/12/15
Implement Visual Components on Hardware	12/4/15
Optimize Analog – Digital Conversion Speed	12/4/15
Sync Visual and Audio Components	2/15/16
Meet 80% Success Rate for Gesture Recognition	3/15/16
Have Project Finished	3/16/16

H. Division of Labor

The division of labor is between the two-team members, Garrett Fosdick and Jair Robinson. Mainly, the division of labor is divided between three different operations: work on the audio portion, work on the visual portion, and work on the system working as one as shown in Appendix C. Fosdick has control over the visual portion, whereas Robinson oversees the audio functionality. At designated moments within then the creation of the project, both will work on certain components together to combine the audio and visual functionality.

I. Societal and Environmental Impacts

Anyone who highly enjoys their music and would have the desire to interact with their music in a unique way would potentially enjoy the product. People who are able to make simple hand gestures, most likely from the ages 4 and up, could operate the product appropriately.

There are also some legal concerns that the product needs to not violate. Being that the product allows the user to manipulate an audio source, or song library of the user's choosing, there will be times that the product will manipulate copyright material. The product does not have the implementation of saving the output data directly, eliminating the possibility on the design's fault of replicating the music. This would be a problem if a user replicated the music and sold it without giving proper credit and rights to the creators of the original music.

There is a chance that the user could get hurt during the operation of this product. The gestures do not call for any extravagant movements, but to prevent the user from accidently harm, a disclaimer before use will help protect the user. The disclaimer will aim to advise the user to use caution and check the surroundings.

Also there is a possible liability to damage of the surrounding area around the user during use of this project. Naturally, the gestures and all in one system shouldn't pose a threat, but a disclaimer for the user to be aware of the environment, surroundings, and other objects will be applied,

The components in the all in one system, including the boards, camera, and tower are all RoHS compliant (Lead-Free, meeting Restriction of Hazardous Substances), meaning the system is environmentally safe and the materials are not violating restricted limits.

III. CONCLUSION

In summary, it is shown that a new way for music listeners to interact and enjoy their music is presented in this proposal. The system will allow a user to manipulate their song of choice in interesting and entertaining ways by use of hand gestures and apply an entertaining mix of audio effects to ultimately provide an innovative and dynamic manner to experience music. The main focus on design should be to develop a device that is used to create a fun experience unhindered by confusing controls, uncomfortable gestures, or latency in gestures and audio playback. The project is feasible to complete in the time allotted and is socially, economically, and environmentally friendly.

IV. REFERENCES

[1] Yee, Jojo. *Tech Support Alert*. Gizmo, 21 Nov. 2014. Web. Oct. 2015. <http://www.techsupportalert.com/content/control-your-music-players-hand-gestures.htm>.

[2] *MacWorld*. Ed. Lou Hattersley. IDG, 20 Feb. 2015. Web. Oct. 2015. <http://www.macworld.co.uk/how-to/mac-software/control-you-mac-with-hand-gestures-3598714/>.

[3] *GECO*. UWYN, n.d. Web. Oct. 2015. <http://uwyn.com/geco/docs/>.

V. APPENDIX A: METRICS FOR OBJECTIVES

TABLE V. METRIC FOR OBJECTIVE QUALITY

Metric	
Ideal	10
Good	7.5
Okay	5
Barely Tolerable	2.5
Useless	0

TABLE VI. METRIC FOR LATENCY QUALITY

Latency Metric	
Unnoticeable to User	10
Barely Noticeable to User	7.5
Somewhat Noticeable to User	5
Extremely Noticeable to User	2.5
Unusable due to Latency	0

TABLE VII. METRIC TO MEASURE FUN

Entertainment from Use (Based Off User Feedback)	
Extremely Enjoyable	10
Enjoyable	7.5
Somewhat Enjoyable	5
Barely Enjoyable	2.5
Frustrating/Not Enjoyable	0

TABLE VIII. METRIC FOR MEASURING DIFFICULTY OF SYSTEM TO USE

Ease of Use	
User Requires No Prep	10
User Requires Minimal Prep	7.5
User Requires Considerable Prep	5
User Requires Some Assistance	2.5
User Requires Considerable Assistance	0

.

TABLE IX. METRIC TO MEASURE GESTURE COMFORT

Comfort from Use (Based Off User Feedback)	
Gestures Feel Perfectly Natural	10
Gestures Feel Mostly Natural	7.5
Gestures Feel Somewhat Natural	5
Gestures Feel Somewhat Uncomfortable	2.5
Gestures Feel Extremely Uncomfortable	0

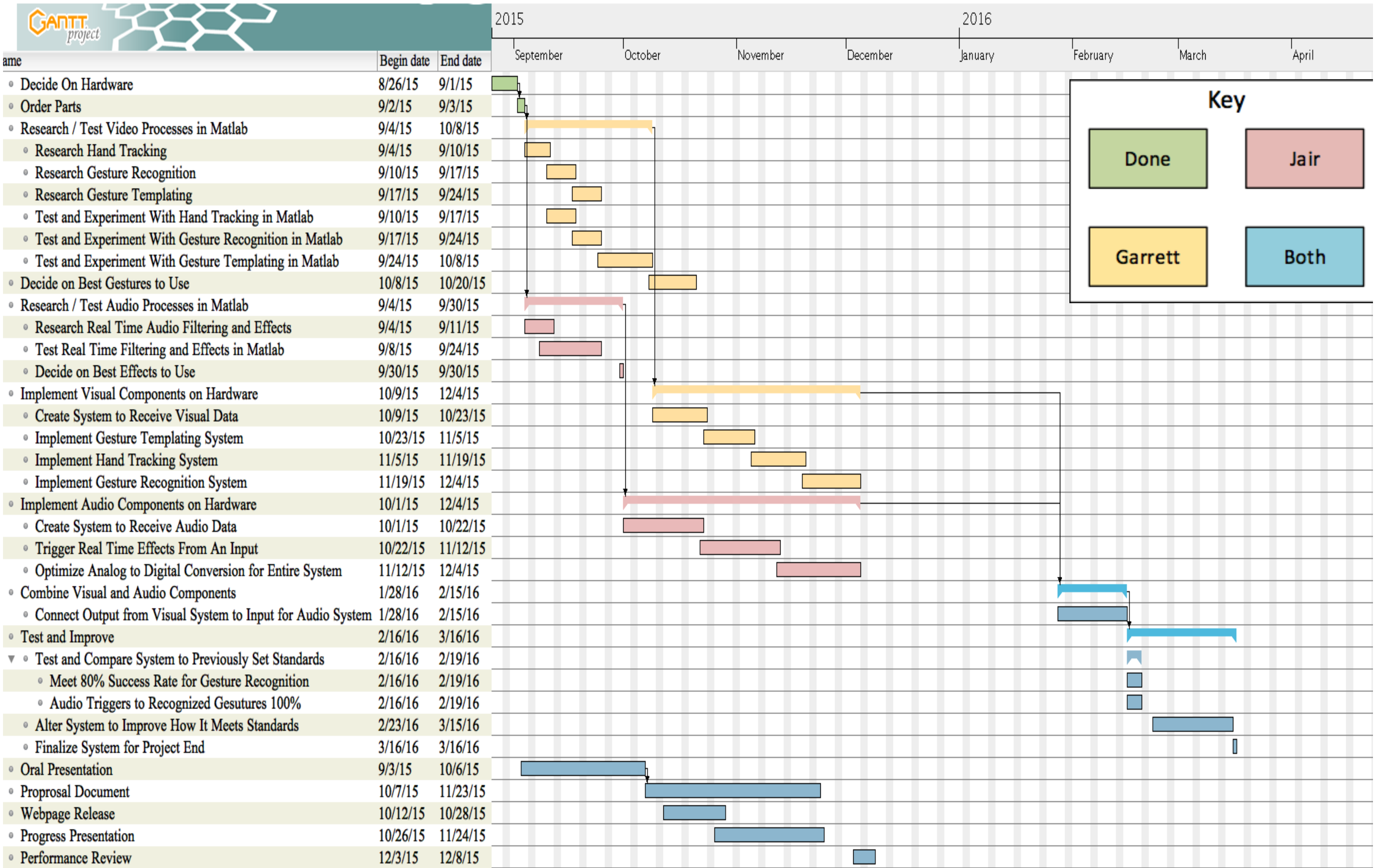
Pairwise Comparison Chart:

When comparing all the objectives to each other, two are found to be far more important than the others. The first and most important is the latency of the product. This is where many similar products fail, and often render the product useless. Latency has a very strong effect on the other three objectives. As for the second most important objective, that would be how fun the overall device is to use. As the product is made to enhance an entertainment experience, it is very important the user enjoys using it. For a more in depth comparison, Table X contains all comparisons of the objectives, and their total score.

TABLE X. A PAIRWISE COMPARISON CHART FOR THE PROPOSED SYSTEM

PCC	Low Latency	Fun to Use	Easy to Learn	Comfortable to Use	Total
Low Latency		1	1	1	3
Fun to Use	0		1	1	2
Easy to Learn	0	0		0	0
Comfortable to Use	0	0	1		1

VI. APPENDIX B: GANTT CHART SCHEDULE



ame	Begin date	End date
Decide On Hardware	8/26/15	9/1/15
Order Parts	9/2/15	9/3/15
Research / Test Video Processes in Matlab	9/4/15	10/8/15
Research Hand Tracking	9/4/15	9/10/15
Research Gesture Recognition	9/10/15	9/17/15
Research Gesture Templating	9/17/15	9/24/15
Test and Experiment With Hand Tracking in Matlab	9/10/15	9/17/15
Test and Experiment With Gesture Recognition in Matlab	9/17/15	9/24/15
Test and Experiment With Gesture Templating in Matlab	9/24/15	10/8/15
Decide on Best Gestures to Use	10/8/15	10/20/15
Research / Test Audio Processes in Matlab	9/4/15	9/30/15
Research Real Time Audio Filtering and Effects	9/4/15	9/11/15
Test Real Time Filtering and Effects in Matlab	9/8/15	9/24/15
Decide on Best Effects to Use	9/30/15	9/30/15
Implement Visual Components on Hardware	10/9/15	12/4/15
Create System to Receive Visual Data	10/9/15	10/23/15
Implement Gesture Templating System	10/23/15	11/5/15
Implement Hand Tracking System	11/5/15	11/19/15
Implement Gesture Recognition System	11/19/15	12/4/15
Implement Audio Components on Hardware	10/1/15	12/4/15
Create System to Receive Audio Data	10/1/15	10/22/15
Trigger Real Time Effects From An Input	10/22/15	11/12/15
Optimize Analog to Digital Conversion for Entire System	11/12/15	12/4/15
Combine Visual and Audio Components	1/28/16	2/15/16
Connect Output from Visual System to Input for Audio System	1/28/16	2/15/16
Test and Improve	2/16/16	3/16/16
Test and Compare System to Previously Set Standards	2/16/16	2/19/16
Meet 80% Success Rate for Gesture Recognition	2/16/16	2/19/16
Audio Triggers to Recognized Gesutures 100%	2/16/16	2/19/16
Alter System to Improve How It Meets Standards	2/23/16	3/15/16
Finalize System for Project End	3/16/16	3/16/16
Oral Presentation	9/3/15	10/6/15
Proprosal Document	10/7/15	11/23/15
Webpage Release	10/12/15	10/28/15
Progress Presentation	10/26/15	11/24/15
Performance Review	12/3/15	12/8/15

Fig. 7. Schedule of project tasks in Gantt chart.

VII. APPENDIX C: DIVISION OF LABOR

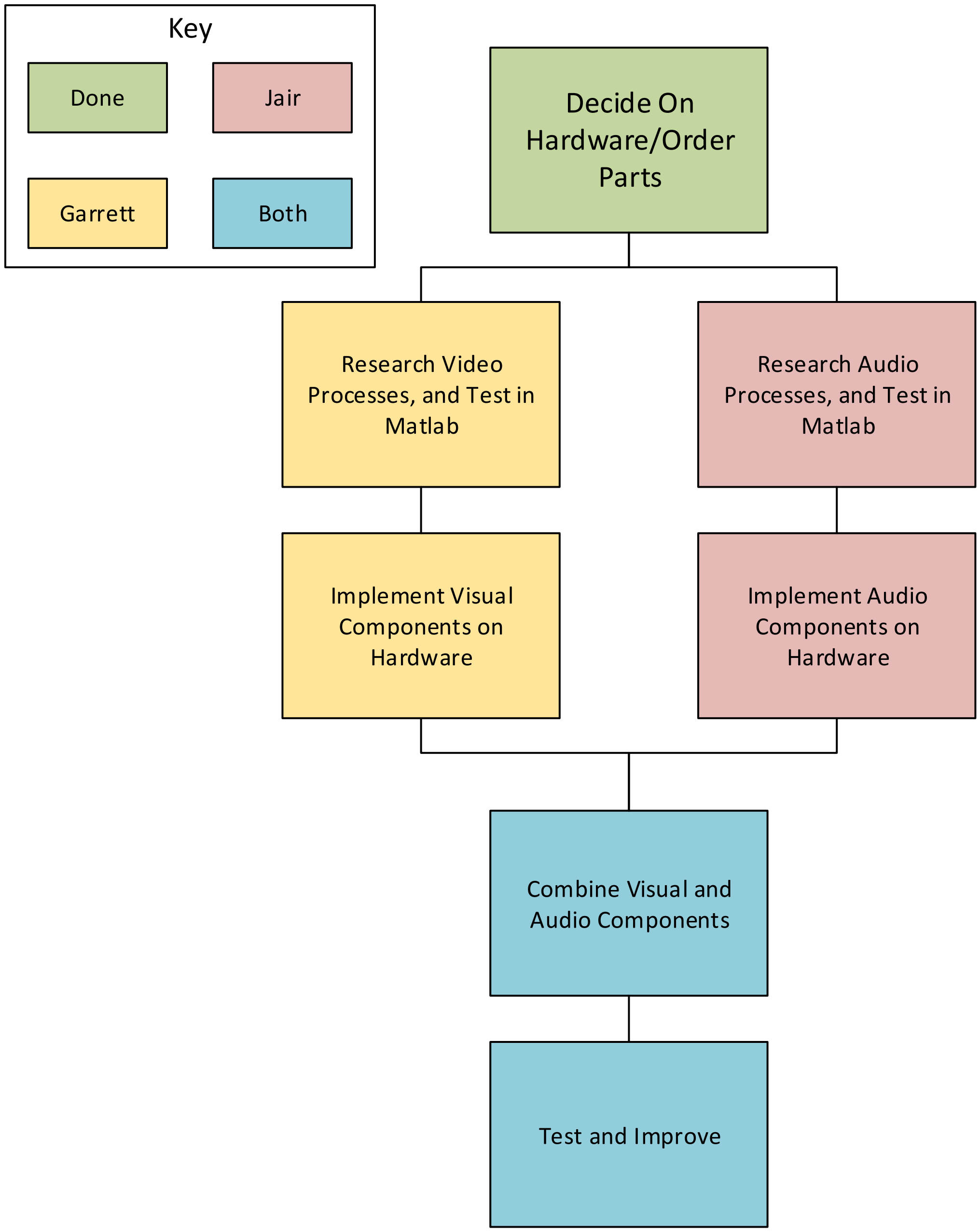


Fig. 8. Overall view of division of labor.

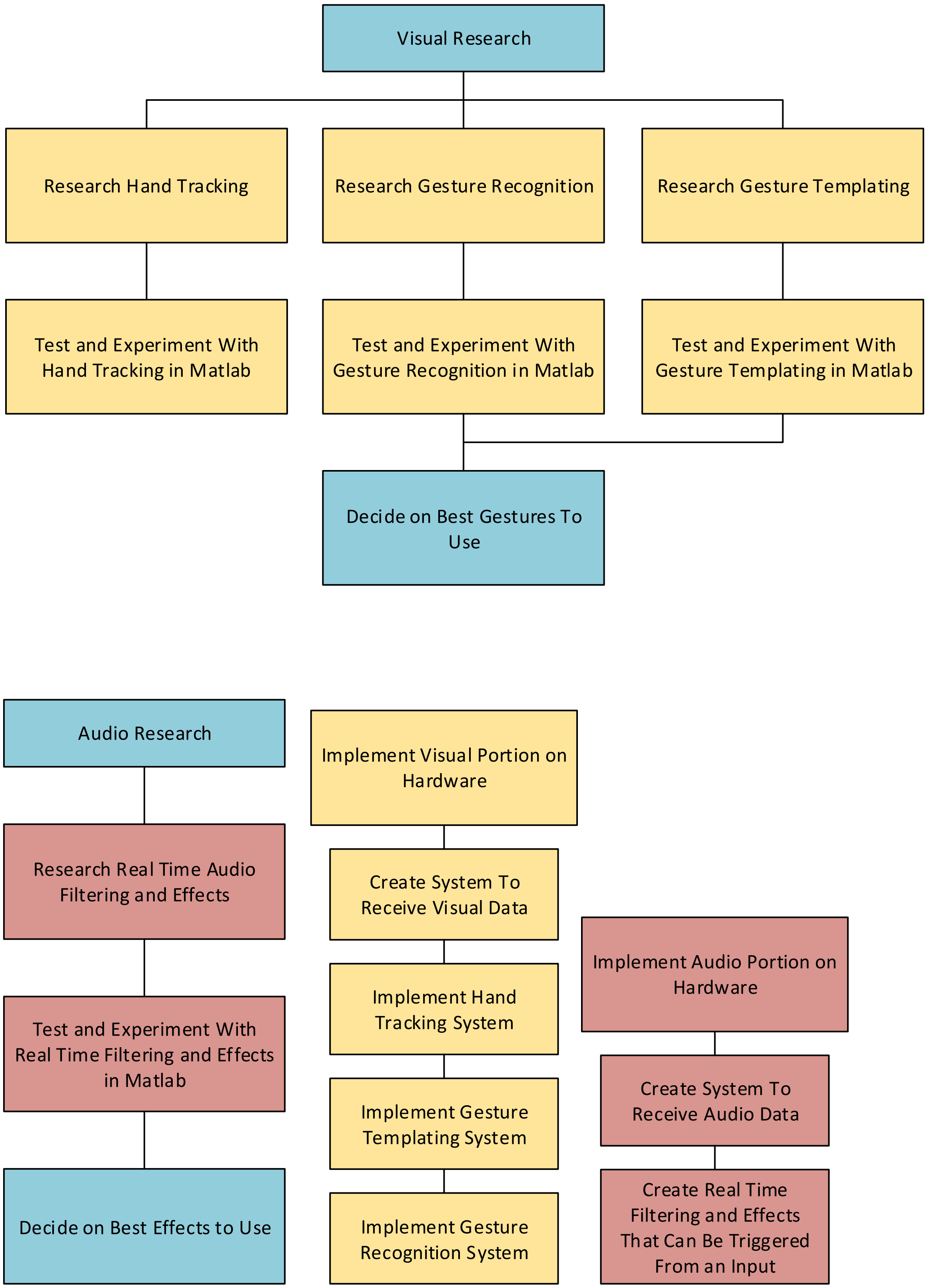


Fig. 9. Visual and audio research and implementation on hardware.

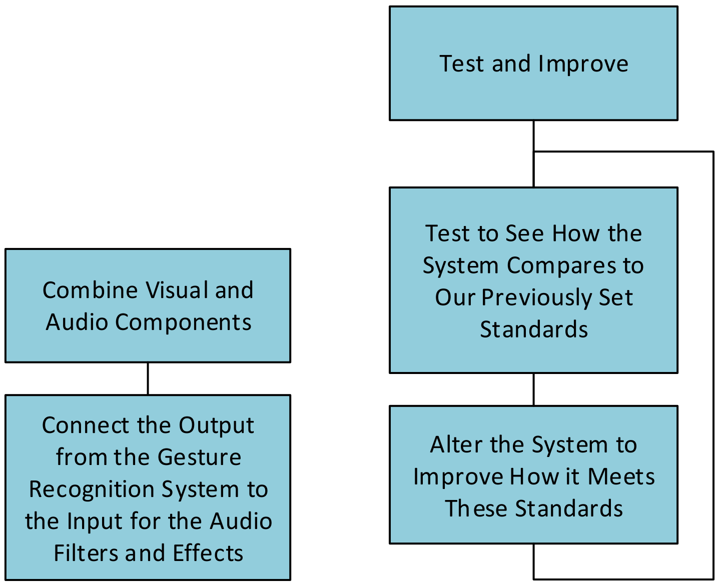


Fig 10. Testing and improvement of stage tasks.

VIII. APPENDIX D: TESTING PROCEDURES

A. Audio System:

Audio Effect Timing Testing

1. Turn on system as if using for regular functionality

2. Apply hand gesture for first audio effect

3. Record if there is perceivable lag between end of gesture and the audio effect being applied

4. Repeat steps 2 and 3 for four more iterations

5. Assess whether or not lag of effect is sensorially perceivable and obstructive to the user experience based on personal subjectiveness

6. Repeat steps 2 through 5 for each remaining audio effect

Satisfactory Auditory Testing

1. Turn on system as if using for regular functionality

2. Play song fully throughout without applying gestures

3. Note any crackling or pops in the sound if necessary

3. Assess whether or not song is clearly played through output device based on personal subjectiveness

4. Replay song

5. Apply hand gesture for audio effect one

6. Record whether the effect is audible based on personal subjectiveness

7. Repeat steps 5 and 6 for the other four audio effects

B. Video System:

Hand Tracking Testing

1. Turn on system as if using for regular functionality
2. Perform singular hand gesture
3. Detect if output audio effect is the correctly tied effect to gesture
4. Perform steps 2 and 3 for four more iterations
5. Record Results
6. Repeat Steps 2 to 5 for each hand gesture
7. Evaluate if 80% accuracy of detection of hand gestures is met

Tracking with Various Hands Testing

1. Turn on system as if using for regular functionality
2. Perform singular hand gesture
3. Detect if output audio effect is the correctly tied effect to gesture
4. Perform steps 2 and 3 for four more iterations
5. Record results
6. Repeat steps 2 to 5 for each hand gesture
7. Have separate users repeat steps 1 to 6 for two more iterations. Have a hand of different pigmentation for each iteration and record if each effect reaches an 80% success rate on various hands.

Tracking Under Various Lighting Testing

1. Turn on system as if using for regular functionality
2. Have an extra external light pointed at front of user
3. Perform singular hand gesture
4. Detect if output audio effect is the correctly tied effect to gesture
5. Perform steps 3 and 4 for four more iterations
6. Record results
7. Repeat steps 3 to 6 for each hand gesture
8. Repeat steps 1 through 7 again removing the external light in room
9. Record results and assess if results meet 80% success rate for each gesture