

Reading Music From Images

Dept. of CIS - Senior Design 2012-2013 Final Report

Stefan Zhelyazkov
stz@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Eric O'Brien
ericob@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

David McDowell
dmcd@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Prof Boon Thau Loo
boonloo@gmail.com
Univ. of Pennsylvania
Philadelphia, PA

ABSTRACT

From a historical perspective, there have been many scientific attempts to find a relationship between images and music with respect to the emotions that both subjects invoke. In modern times, these scientific efforts have been facilitated by the use of computers aiming to discover evidence for objective correspondence between visual and audible information. Based on existing research, this research is targeted to analyze currently used algorithms for reading music from images by comparing their output based on feedback from a pool of testers. The second goal is to provide an improved algorithm for conversion of images into music that will extend on the algorithms analyzed in the experiments. This is a process that uses the results of two existing algorithms in the design of a third algorithm, focal analysis.

1. INTRODUCTION

The relationship between music and colors is a relatively old topic of interest. Some of the first accounts of analyzing this idea can be found in ancient Greek literature. A very serious attempt to unify the two concepts was made by the philosophers Aristotle and Pythagoras. In his work titled *De sensu et sensibilibus*, Aristotle states that colors are always in numerical proportion to one another and similarly, “we may regard all these colours [viz. all those based on numerical ratios] as analogous to the sounds that enter into music” [4].

Similar theory on the topic was held by Pythagoras, who presented the idea of universal harmony based on proportions. Pythagoras’ claim is that everything in the Universe uses the language of mathematics and can be described by numbers and ratios. According to his theory, rhythm, music, and mathematics are mutually related through proportions [14].

The main objective of this research is to further analyze the chromatic attributes of images, building upon the existing results of three papers [11] [8] [13]. The notions of *harmony* and audiovisual *sensation* are converted into a contemporary mathematical form through the use of basic building blocks called “chromatic indices” and “chromatic bricks” [13] and produce an algorithm that converts images to pleasant pieces of music. The subjective factor of being *pleasant* is scientifically undefined. Therefore, in order to justify final results, a comparison is made between the results of the direct mapping algorithm and our focal analysis algorithm. Amazon’s Mechanical Turk ([3]) is used to gener-

ate results (workers submit responses to a survey containing generated music from images). Further information on the performance evaluation can be found in Section 6.

This research aims to unveil the underlying relationship between graphical content and music. Its results are anticipated to serve as a source of inspiration for composers, musicians, and anyone who works with music. It may also serve as an inspiration for non-musicians to enter the field of music. Furthermore, entertainment remains one of the major justifications for our research. Many would love to try *hearing* the music of the images around them. In addition, the core algorithm could be implemented for mobile devices, a market in which we would expect much interest. Taking a photo of a beautiful landscape and listening to the sounds of nature are expected to be considered entertaining and attractive to large groups of people.

2. TERMINOLOGY

Our discussion of the following implementation will involve some basic knowledge of musical terms, as well as a few terms that we have defined for the purposes of our research.

- **Pitch/Tone** - A specific frequency of sound. All sounds have a pitch, though the term pitch is most often used in music for sustained sounds that come from instruments or voices.
- **Note** - A name given to a specific pitch for ease of reference. For instance, the note for the pitch representing 440Hz is called A.
- **Interval** - The distance between two pitches in terms of their relative frequencies. For example, the note A4 (440Hz) is said to be an octave (eight increments) below A5 (880Hz).
- **Timbre** - The quality of a sound that allows us to differentiate between different types of instruments. Even when playing the same notes, a piano sounds very different from a guitar, and this is because of the timbre of the sounds they produce. Timbre is generally described with subjective words like “warm”, “dark”, and “bright”, which suggests that images with similar subjective qualities could be assigned a timbre in their generated musical piece. Timbre is modified by changing the type of instrument when playing music.
- **Melody** - In music where only one note is playing at a time, the melody is simply the pattern of notes being

played. In music where multiple notes are played at the same time (most music), the melody is the most important set of notes in the piece. In many songs, the melody is the part which the lead singer sings.

- **Counterpoint** - The intentional combination of two or more melodies which are played simultaneously. Both melodies are equally important, and consequently their juxtaposition achieves a unique effect.
- **Key/Scale** - A set of notes that are related by their distance (in Hz) from one foundational note. These notes generally sound cohesive together when used exclusively in a piece of music. For instance, the key C major is comprised of the notes C D E F G A B, and the key E major is comprised of E F# G A B #C D.
- **Chromatic (music)** - Describes a sequence of two or more notes that are adjacent to one another. For example, the notes (C C#) are said to be chromatic because there are no notes between C and C#.
- **Chromaticism (music)** - The use of notes not belonging to the scale in the given key. (e.g., the use of the note D# in a piece written in the key C major). While this does not sound “cohesive” by the previous definition of a key, these notes can be used selectively by composers to create more interesting pieces of music.
- **Dissonance** - The combination of notes that sound harsh or unpleasant. This can arise from the use of chromaticism. While dissonance by definition sounds harsh and displeasing, in certain contexts it can be pleasing and powerfully emotional.
- **Chromatic Index** - This is a field of research and is fully described in [6]. In short, it is the musical tone that corresponds to a particular color after a key has been chosen.
- **Chromatic Brick** - A term introduced in the paper [13]. It is the pairing of a chromatic index and a duration (in seconds), both of which are decided upon by combining adjacent pixels of similar color into one component - the chromatic index is decided by the color of the block and the duration by the length of the block.
- **Musical Phrase** - A number of notes that are grouped together and form some cohesive pattern. Similar to a phrase in literature, phrases make up the whole of a musical piece.
- **Harmonic** - Relating to harmony, or groups of sounds that are pleasing.

3. RELATED WORK

A number of researchers and programmers have explored ways in which images can be related to music. Contemporary research in the area of music composition from visual information has been made primarily in two directions - direct mapping and chromatic analysis. The following are high-level explanations of both direct mapping and chromatic analysis. In Section 4, we go into detail about our implementation of these different approaches.

3.1 Direct-Mapping

The “Direct-Mapping” approach broadly describes a system in which each pixel of an image is assigned a note value based on some consistent rule. Applying this technique, visual information is directly converted to musical notes. This approach has one major known shortcoming - it only reads the numerical information from the images and completely ignores the subjective factor of feelings. However, this approach has been the most common proposal for the past 300 years [12]. There exist applications which directly convert images to music in this way, but their rules for generating music are completely arbitrary. For example, the program AudioPaint [9] uses the red and green components of a pixel to determine the amplitude of the left and right audio channels, and the blue component is ignored entirely.

3.2 Chromatic Analysis

The second approach is to perceive the information from the image and attempt to identify a sense of what the image represents. This direction of research is more interesting because it accounts for the subjective factor of images, namely the feelings they evoke in the observer. Understanding how these feelings occur and what they are based on are essential to relating their graphical information to music. Modern findings on the topic introduce the ideas of *chroma* (the Greek word for *color*), *chromatic index*, and *chromatic brick*. Those concepts are established fundamentals in some research directions [13] [8]. The relationship between colors, music, and chroma is well studied and will be a major premise of this paper. Their complete relationship is described thoroughly in [13] [6]. The specific colors and their associated chromatic indices are shown in Figure 1.

Color	X (chroma)
White	1.0
Turquoise	1.1
Green	1.2
Yellow/Gold	1.3
Orange	1.4
Red	1.5
Pink	1.6
Blue	1.7
Purple	1.8
Brown	1.9
Grey	2.0
Black	2.1

Figure 1: Chromatic Index Table

Many researchers have attempted to explore the connection between the visual and auditory systems [12] [6], and some have even devised algorithms that attempt to concretize the mapping of pixels to melodic phrases [13] [11] [8]. We are most interested in the research conducted by D. Margounakis and D. Politis [13]. In their papers, they describe an algorithm called “Chromatic Analysis” which transforms

common pixels of an image into musical phrases. A notable shortcoming of this approach, however, is the lack of user testing. Without the subjective approval of other people, it is impossible to determine whether or not the software is effective at generating music. Our research solves this issue by having user tests as an evaluation criteria.

3.3 Relating Music and Images

Research on the connection between vision and sound goes in both directions. In addition to turning images to music, there exists research about visualizing music as images [7]. This research is akin to designing compelling music visualizers found in popular audio playback software such as iTunes. If future research involves generating real-life imagery from music, we would have some very interesting complementary research.

A phenomenon known as synesthesia can help us learn more about how humans relate sounds to images. Derived from the Greek for “joined sensation,” synesthesia is a condition where people can have multiple disjoint sensations in response to one stimulus. This condition had not been widely researched until about 20 years ago, as patients who complained about these symptoms were not taken seriously or were diagnosed with an “overactive imagination” as children. However, interest has grown in recent years, possibly due to more information about how perception functions in the brain [5]. We are interested in looking at how our results integrate with the existing knowledge of the problem and if there is an inherent relationship between the medical findings and our findings. The subfield most applicable to this research is called chromesthesia, which is a condition where music and sounds invoke feelings of color.

4. SYSTEM MODEL

The entire system is comprised of a framework for receiving images and running an algorithm to generate musical notes, the two implemented algorithms which work within this framework, and a mobile application which allows the algorithm to be demonstrated to users. This section describes the system in greater detail.

4.1 Overview

At its core, the system loads an image, runs an algorithm over the image, and returns a set of musical notes. These notes can then be played directly by our system, or they can be saved in the form of sheet music to be played by a musician. The system is also tested as described in Section 6. Figure 2 shows a diagram of the system’s major components. A basic description of the process is as follows:

1. The two algorithms (direct mapping, focal analysis) receive an image as input.
2. The image is converted to music based on the two algorithms described later in this section.
3. The results of the algorithms are then posted on Amazon Mechanical Turk ([3]) so that the workers there can give feedback which one of the algorithms is doing better than the others. Based on their feedback, certain parts of the algorithm are modified in order to experiment and see which features of the algorithms provide optimal results. Testing is more thoroughly described in Section 6.

4.2 Direct-Mapping Algorithm

This algorithm is given the image as an object, giving it access to the individual pixel colors at every x, y coordinate. From this, more information can be extracted from the image, such as brightness, saturation, sharpness, but these factors have been discarded in favor of treating color as the main and sole component for note composition. The direct mapping algorithm computes a musical note for each pixel by adding the red, green, and blue color components, then maps that value to one of the 88 keys of a piano using the modulo function. For example, a solid blue pixel with RGB values of 0, 0, and 255, respectively, will be mapped to the $(255+0+0)\%88 = 79$ th key on a piano, which is a D#. This has some obvious issues - namely, a solid blue color will map to the exact same note as a solid red color, and other color to note mappings do not seem to have any logical basis. This issue is expected, as the direct mapping algorithm is merely a proof of concept to show that our framework for running the algorithm functions as correctly. As described in Section 3, other image to music mapping programs use completely arbitrary rules, and our goal was to see what could be done with note creation and image analysis.

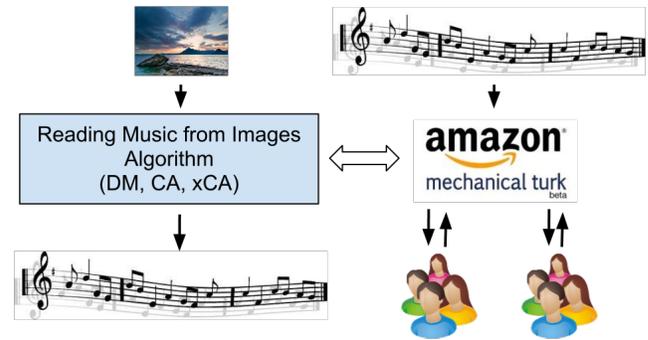


Figure 2: System Diagram

4.3 Extended Chromatic Analysis, xCA

4.3.1 General Outline

Extended Chromatic Analysis consists of five distinct steps:

1. Reading and resizing the image.
2. Performing “focal analysis” on the resized image to identify the focal objects.
3. Extracting the focal objects from the image and converting them to nodes in a graph.
4. Performing chromatic analysis on each node that produces the chord progressions for that node.
5. Connecting all chord progressions into a music melody.

4.3.2 Image Input and Resizing

When an image is provided to the program as input, it is scaled through an image resizing procedure. The maximum bounds for height and width were set to 400 pixels each. After the user selects an image, the ratio of width to height is computed and the longer side, if greater than the limit, is scaled by a factor to 400. Then the factor is applied over the entire image, resulting in a picture with each side being at maximum 400px.

4.3.3 Focal Analysis

Focal Analysis is an image processing procedure that extracts the objects which attract the viewers' attention first (schematically represented in Figure 3). When a viewer looks at an image, the eyes focus on the highly contrasting objects of the image, and next they perceive the rest of the picture. This natural behavior is the inspiration behind the "Focal Analysis" procedure - the objects captured by this image processing algorithm can then be converted to music first, after which the rest of the image is converted to music.

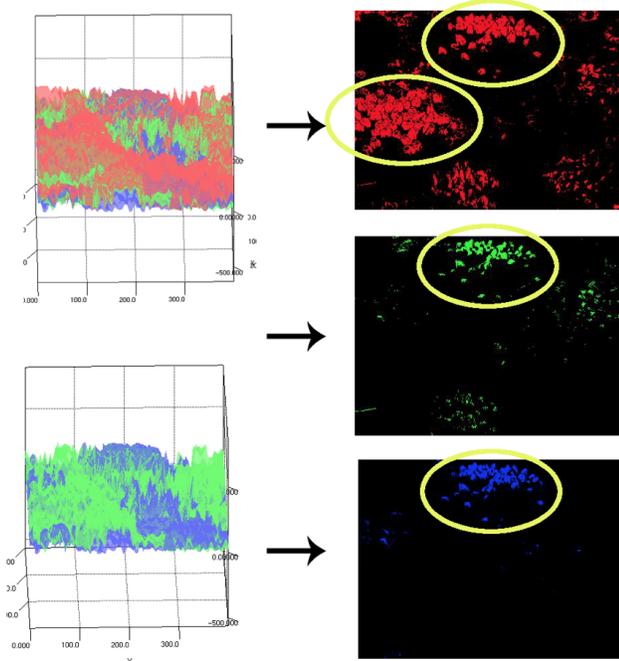


Figure 3: Focal Analysis Diagram

Focal Analysis was discovered after performing 3D histogram color analysis. Each image provided to the program is visualized in an xyz-plane where the xy-plane is the dimensions of the image and the z-plane contains the values of each color channel plotted. This could be similarly viewed as three independent plots of the the red, green, and blue channels over the image. Since all values for a channel are numbers in the closed range [0;255], the 3D plots are easily analyzable.

Each of the histograms is sliced into the average red, green, and blue value of the whole image. From the obtained results, slicing each histogram layer in the corresponding average color for the channel produces the best results. As an example, the red slice in the histogram is cut at 132 (average red), 95 (average green), and 63 (average blue), but the best

result is provided from slicing it at 132. The same pattern is observed for the rest of the slices, so the same procedure is applied for them as well.

Each of the slices from the previous step is passed into a color distribution procedure. Since 256 is 16^2 , the range for the distribution is set to 16 and then 16 different layers are extracted from the slice. For example, on the r-132 red slice, 16 layers are created, each layer representing the grouping of pixels from [0;15], [16;31], [32;47],... [240;255]. The layers that have 0 elements, i.e. no pixels within the range, are discarded. The rest of the layers are ordered by how many pixels are contained within them. Then the ordered distribution is passed to a selection function.

The selection function makes a grid on top of each layer where a cell from the grid is 20x20 pixels. The numbers are chosen from the bounds of the maximum dimensions: 400 is 20^2 . In this procedure, each grid cell delineates a portion of the layer. Then the amount of color in each cell is evaluated. If it is more than $\frac{1}{3}$, or 0.334, the cell is considered valuable - this means that there is an object in this location and, from the experimental procedure, also suggests that there would be other dense objects nearby. The value of $\frac{1}{3}$ was established experimentally. After performing the operation on a number of images it was found that majority of the layers that have cells of density below $\frac{1}{3}$ do not perform as well in capturing the focal objects of the image, while the majority of the grid cells of density equal or above $\frac{1}{3}$ contain focal objects. These observations led to selecting $\frac{1}{3}$ as the limiting value.

After all layers from all color channels have been analyzed by the procedure, those containing dense color cells are grouped into clusters and passed along to the graph construction algorithm.

4.3.4 Converting Layers to Graph Nodes

The layers are passed a linked structure to a Graph constructor (see Figure 4). As they arrive, each layer is either a red, green, or blue colored piece. However, the coordinates of the pixels of each R, G, B color are the same as the coordinates of the focal objects from the original image. Given this fact, the graph constructor scans each layer according to its x and y coordinates, while reading color data for the same pixels from the original image. Additionally, it also keeps track of the area of each focal object.

When a focal object is read, its average color is computed. This average color is then mapped to the closest of the chromatic indices given in Figure 1. This allows for the creation of a 'chromatic vertex', which is a simple data structure, representing a node in the graph - it contains the closest chroma based on the table and the area that the chroma occupies. This information is used when performing chromatic analysis on each node. The 'chromatic graph' is a chain of chromatic vertices.

In the end all chromatic vertices are ordered by the area that they occupy - this is due to the fact that each chromatic vertex represents a focal object of a highly contrasting color. Its area describes how likely it is for a viewer to look at it before looking at anything else - the larger it is, the more likely it is at the focus of the image.

After all the vertices have been ordered by area starting from the largest one to the smallest one, a procedure called "shrinking" is applied. Shrinking is connecting two or more neighboring vertices of the same color into one large chromatic

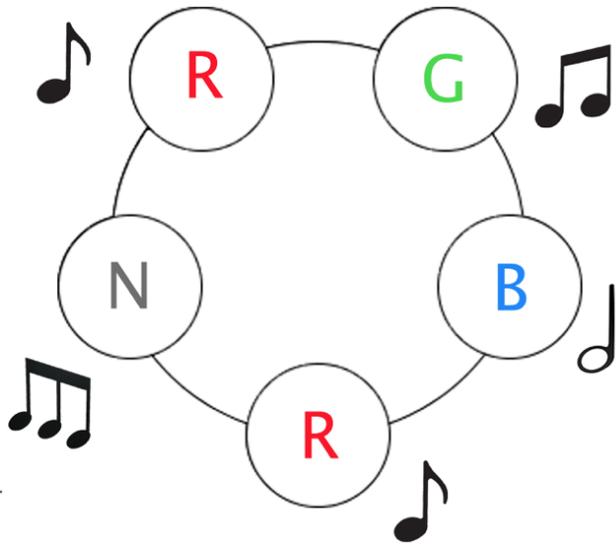


Figure 4: Graph Generation Diagram

matic vertex of the same color but with a combined area. Shrinking is performed until the graph cannot be shrunk any more.

4.3.5 Chord Progression Generation

The following table of indices in Western scale keys are used to create 'moods' with chord progressions.

Mood	Index in key	CI	Instrument
alternative	6, 4, 1, 5	≤ 1.0	piano
canon	1, 5, 6, 3, 4, 1, 4, 5	≤ 1.1	piano
cliche	1, 5, 6, 4	≤ 1.2	clarinet
cliche2	1, 6, 3, 7	≤ 1.3	xylophone
endless	1, 6, 2, 4	≤ 1.4	trumpet
energetic	1, 3, 4, 6	≤ 1.5	marimba
memories	1, 4, 1, 5	≤ 1.6	guitar
rebellious	4, 1, 4, 5	≤ 1.7	viola
twelvebarblues	1, 1, 1, 1, 4, 4, 1, 1, 5, 4, 1, 5	≤ 1.8	violin
wistful	1, 1, 4, 6	≤ 1.9	guitar
grungy	1, 4, 3, 6	≤ 2.0	guitar
sad	1, 4, 5, 5	≤ 2.1	accordion

Figure 5: Chord Progression Diagram

The data from this table is used to generate a general chord progression for the chromatic vertex. The chromatic index for each vertex is indexed into the table by the 'CI' key. Then the appropriate mood is selected and an algorithm for generating a chord progression within the mood, given C major as a starting key, is used to generate a chord progression for the given vertex. Given the chromatic index, the appropriate instrument is chosen. Based on the mood, the instruments were classified as cheerful or minor and were associated with suitable moods.

4.4 Music Generation

After chord progressions are generated for all chromatic vertices, the progressions are combined together in a song. With the help of jFugue [10], the song is recorded in MIDI format, which can later be played from any device with audio support.

4.5 Mobile Application

One compelling way to demonstrate our algorithm is with an interactive mobile application. With it, a user can take a picture with their mobile device, then play the generated music to "hear" how the image sounds. The structure of this application is illustrated in Figure 6 below:

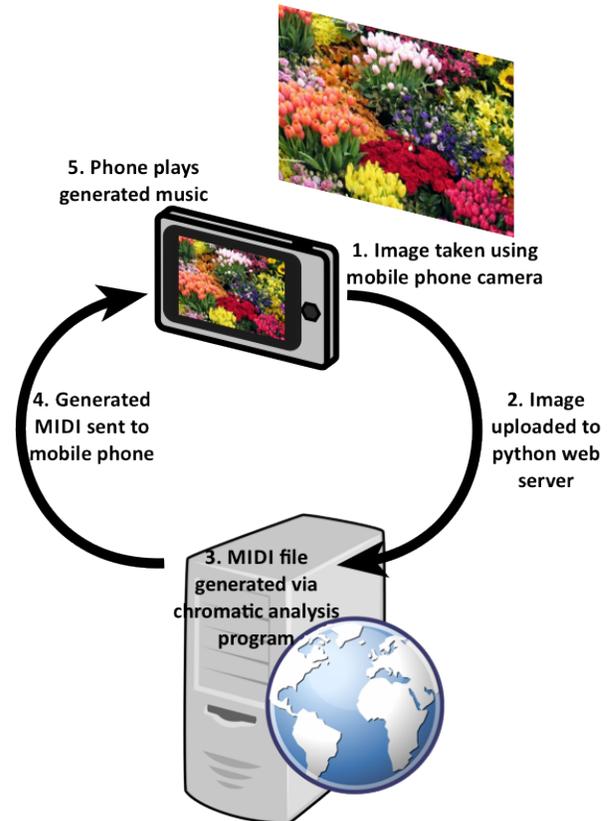


Figure 6: Android Application Diagram

The mobile application is implemented for the Android operating system. This allows the code to be consistent with the Java implementation of the core program. Additionally, Android is currently the most popular mobile operating system in terms of installed users, and a future commercial application release would reach the largest number of users compared with Apple's iOS or Windows Phone. A commercial release will be possible only after modification of the focal analysis algorithm to function entirely on the device.

The webserver runs using the Python BaseHTTPServer. A Python webserver is very simple to build and deploy, and it allows programs to be run through the command line, which proves useful for playing with the output. The reason for including a web server in our mobile application implementation is a result of exploring the most practical way to implement the application in a limited time frame. Since the javax.sound.midi library is not present in the standard

Android Java libraries. The Android developer tools also do not allow bundling this library into the Android application, therefore music could not be generated directly on the phone without a major modification of our existing code. Instead of rewriting the focal analysis implementation, the code is left as-is and run on an external webserver. This creates latency between the time the picture is taken and the time that music can be played from the image, and it requires the user to have a network connection whenever a new picture is taken within the application. While this latency is only about two seconds, the constant network requirement is not acceptable for many mobile devices. The choice to include a webserver results in one unexpected benefit: modifications and testing are very easy to perform through the application.

5. SYSTEM IMPLEMENTATION

We decided to implement our system in Java for a number of reasons. The abstraction reduced programming time compared with lower-level, more powerful languages like C. There are also several libraries that have been integral to the success of our research. One of these is the JFugue library [10], which is an open-source Java API for programming music without dealing with the complexities of MIDI generation¹. The JFugue library’s Pattern class is particularly useful for using the groupings of notes that have been earlier defined as “chromatic bricks”. The library also supports microtones, intervals not found in the Western system of 12 equal intervals², but instead in more uncommon scales that may be chosen in a case of a picture with a high chromatic index. The library also features a very well written and comprehensive documentation consisting of 11 chapters that include full descriptions of the library classes as well as a basic introduction to the concepts of music composition.

6. RESULTS

As a “sensory” system involving visual and sonic elements, evaluation of success might seem difficult; however, objective measures were used as evaluation criteria. Amazon’s Mechanical Turk ([3]) provided statistics on the accuracy of the main algorithm (focal analysis) compared to the direct mapping algorithm. Mechanical Turk workers were asked to complete our specified Human Intelligence Test (HIT), which will involve viewing an image and listening to a short musical accompaniment (generated from one of the three algorithms), and then determining if the accompaniment is appropriate for the image shown.

We tested one hundred workers with a three question survey regarding the connection between image and output music. Survey participants were asked to listen to a piece for 30 seconds (all the audio outputs were under one minute), and then asked to rate their opinion on whether the generated music was not at all relevant to the image from which the music was generated (a zero), extremely relevant to the image (a ten), or somewhere in between (one to nine). Participants performed this process for two additional image inputs and musical outputs to complete the survey. The first of the image inputs is seen in Figure 7.

This image (Figure 7) has a vibrant array of colors and is realistic, so to cover all avenues of possibility, the next

¹A music file format that specifies music in digital form.

²A standard for musical notation.



Figure 7: Image Input: Question 1

two images for question two and three are chosen carefully. Two’s input image is a artificial painting (not a digital still of a real-life setting) with a more limited number of colors aTimnput image is a larger scale photo of a college campus scene, capturing far more than the dipphoto in imaimage one. It also has an extremely dominant color (green) in comparison to the first image. Choosing these input images in this manner (with respect to diversity in color, contrast, and digital versus real life) is imperative to the validity of these results.

Figures 8 and 9 show the output of the first Question with Direct Mapping and Focal Analysis, respectively.

Our results in Figure 10 show a slight difference between our two algorithms, and this follows from our initial hypothesis that the work will be difficult to distinguish meaning from a different interpretation of less significance. However, our results did show a slight improvement over the initial method, and we believe with further refinement of output to adhere to classical music rules, these results will improve even further.

As seen in the Figure 10, the mean rating for image to musical relevance increased by 0.552 points when comparing the results of Focal Analysis to the Direct Mapping method. These results are difficult to interpret without giving comparison to related work in the field, but our research has not yielded any projects that have given results past a singular example for the project’s output (i.e., one example of an input / output pair for image to music).

7. ETHICS

For the most part, this research remains without ethical reservations. Our data collection was performed anonymously and without obtaining any personally identifiable information. The only information obtained was the respondent’s opinion ratings and his or her Mechanical Turk ([3]) worker ID (used to ensure payment for completing the survey). Applying for IRB approval was still necessary, however, and we completed this process without problems.

Even with the mobile application, ethical issues are non-existent. The phone sends a photo to our server, which performs our algorithm and sends back song data to the phone. A hypothetical situation can be imagined, perhaps, in which the photo is intercepted before reaching the server, but this is a risk any internet user faces when accessing the web.



Figure 8: Direct Mapping: Question 1 Output. See [1] for musical output.

8. FUTURE WORK

Future additions to our focal analysis algorithm will involve several components. Firstly, plans are in place to further refine musical output based on additional classical music theory, which is a collection of rules that guides music composers still to this day. Typically, music that adheres to these rules will sound pleasant and mistake-free to the listener. Focal analysis already follows standard arrangement technique in composing, and the next step would involve incorporating some of the more intricate rules of music theory that guide note changes and counterpoint. One example would be outlawing parallel perfect fifths (as one note changes by a certain frequency, no other note should follow in a similar interval in a similar direction at the same time). This rule, along with others, such as choosing the next chord inversion of least movement, could be implemented as a series of checks on the next step of note generation as the input image is being converted to music.

Additionally, this research will expand into pattern recognition and interpreting patterns from the picture into a melodic line that fits over the chord patterns that we currently generate. Current plans involve generating multiple lines and scoring them individually based on theory rules, choosing the melody with the highest score to fit over the chord changes.

Lastly, our focal analysis implementation is to be modified

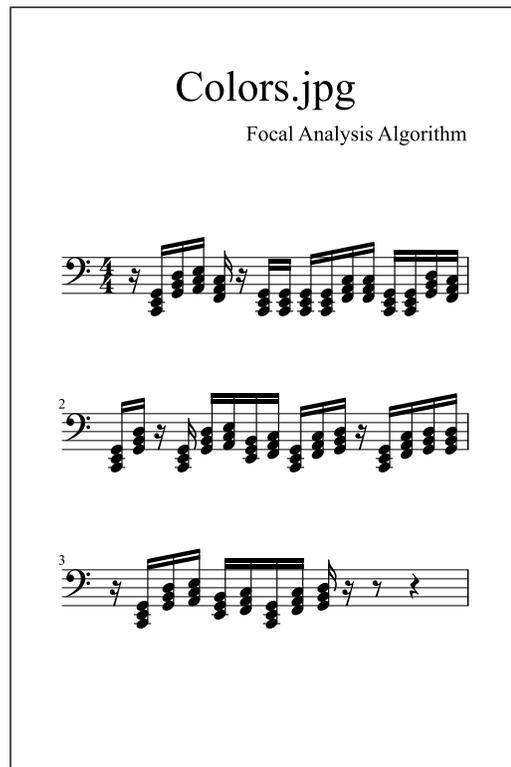


Figure 9: Focal Analysis: Question 1 Output. See [2] for musical output.

Question	Direct Mapping	xCA
1	4.885	5.000
2	5.615	5.571
3	4.462	5.536
Mean	4.987	5.539

Figure 10: Table of Results. The values given are averages of the answers provided by Mechanical Turk ([3]) respondents on a scale of zero (no relevance) to ten (extremely relevant)

to run entirely within the Android operating system. This will allow for a responsive demonstration of image to music conversion without the need for an external web server.

9. CONCLUSIONS

We joined an emerging research topic of image and music analysis, we developed a new method of analyzing an image to determine its important features, and we created a mobile application which allows music to be created from images.

Focal Analysis adds to the emerging research topic of the relationship between images and music. This is a new type of image analysis which is more advanced than current approaches to image to music conversion. The algorithm takes into account the fact that humans do not look at images pixel-by-pixel, from the left to right, moving downwards like one reads a book. Instead, a viewer naturally sees contrasting portions of an image, and his focus shifts in a nonlinear fashion. Additionally, this algorithm produces determinis-

tic results, whereas other existing applications add random elements to generated music. This is important because it adds to the relevance and meaning of the generated output. Different results for a single input image would dilute the significance of an individual result. Finally, the mobile application demonstrates the algorithm in an intuitive way. A potential use case of this system is when a user is outside, and they come across some compelling imagery. With this mobile application, the user can quickly experience the visual imagery in a new way, which will add to the overall experience.

10. REFERENCES

- [1] Direct mapping: Generated music for colors.jpg. <https://dl.dropboxusercontent.com/u/488484/CIS401/dm-colors.mp3>, April 2013.
- [2] Focal analysis: Generated music for colors.jpg. <https://dl.dropboxusercontent.com/u/488484/CIS401/colors.midi.mp3>, April 2013.
- [3] Amazon. Amazon mechanical turk. <https://www.mturk.com/>, 2013.
- [4] Aristotle. *The Complete Aristotle*, chapter Part 3: Human Physics - Sense and sensibilia, pages 906–934. eBooks@Adelaide, The University of Adelaide Library at the University of Adelaide in South Australia, 2007.
- [5] Richard E. Cytowic. *Synesthesia: A Union of the Senses*, pages 13–27. The MIT Press, Cambridge, MA, 2002.
- [6] Dimitrios Margounakis Dionysios Politis. Determining the chromatic index of music. In *Proceedings of the Third International Conference WEB Delivering of Music*, pages 93–100, Sept 2003.
- [7] Dimitrios Margounakis Dionysios Politis and Konstantinos Mokos. Visualizing the chromatic index of music. In *Proceedings of the Fourth International Conference on Web Delivering of Music*, pages 102–109, Sept 2004.
- [8] Michail Karatsoris Dionysios Politis, Dimitrios Margounakis. Image to sound transforms and sound to image visualizations based on the chromaticism of music. In *7th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Databases*, pages 309–317, University of Cambridge, UK, Feb 2008.
- [9] Nicholas Fournel. Audiopaint. <http://www.nicolasfournel.com/>, June 2011.
- [10] David Koelle. Jfugue. <http://www.jfugue.org/>, January 2012.
- [11] Cheng-Te Li and Man-Kwan Shan. Emotion-based impressionism slideshow with automatic music accompaniment. In *MM'07*, pages 839–842, Augsburg, Bavaria, Germany, September 2007.
- [12] Andre Rangel Macedo. Visible and audible spectrums - a proposal of correspondence. In *Proc. of Artech, 5th International Conference on Digital Arts*, pages 168–171, Guimarães, Portugal, April 2010.
- [13] Dimitrios Margounakis and Dionysios Politis. Converting images to music using their colour properties. In *Proceedings of the 12th International Conference on Auditory Display*, pages 198–205, London, UK, June 2006.
- [14] Ernest G. McClain. *The Pythagorean Plato : prelude to the song itself*. Nicolas Hays, Stony Brook, N.Y., 1978.