# Poincaré Disk Image Tiling

Sam Brunacini (Advisor: Dr. Jason DeBlois)

Department of Mathematics

## Background

Many people are familiar with the famous mathematical drawings of M.C. Escher. Some of his artworks, known as the "Circle Limit" drawings, depict regular tilings in the Poincaré disk, a model of hyperbolic geometry entirely contained in the open unit disk in $\mathbb{C}^2$.

## Prior Results

Many automatic tiling generators have been created. Notably, [1] and [2] describe tiling programs that support color symmetry up to 4 colors. No programs have been created that use images with a conformal mapping.

Our methods are built upon studies of hyperbolic groups [3], [4] and the Schwarz-Christoffel mapping [5].

## References

[1] Douglas Dunham. Computer design of repeating hyperbolic patterns.

[2] Douglas Dunham. Hyperbolic symmetry. *Comp. Maths with Appls.*, 12B(1):138–153, 1986.

[3] A. M. Macbeath. The classification of non-euclidean plane crystallographic groups: Canadian journal of mathematics. *Cambridge Core*, Nov 2018.

[4] George Hyun. *Hyperbolicity and the word problem*, Aug 2013.

[5] Chamberlain Fong. Analytical methods for squaring the disc.

## Introduction

The goal of this project is to create artistic tilings in the Poincaré disk similar to those created by Escher. We created a program using Javascript and WebGL that can generate $\{p,q\}$ tilings using an arbitrary square image as a starting pattern. We devised a process that maps square images conformally to regular polygons in the Poincaré disk. The program generates the tiling using symmetries of the Poincaré disk applied to the starting polygon.

## Image Mapping

Define the Schwarz-Christoffel map (disk to $p$-gon):

$$f_p(z) := \int_0^z \frac{dw}{(1-w^p)^{\frac{2}{p}}}$$

and the Inverse Schwarz-Christoffel map ($p$-gon to disk):

$$f_p^{-1}(z) := z \sum_{k=0}^{\infty} \frac{z^{pk}}{(pk+1)!} \left(f^{(pk)}(0)\right)^{-pk-1}$$

1. Start with a square picture and parameters $(p,q)$ and consider the point $\mathbf{x}$ inside the rotationally symmetric $p$-gon in the Poincaré disk.
2. Map to the Klein disk using the hemisphere model as intermediate.
3. Send the result to the square picture using the map $f_4^{-1} \circ f_p^{-1}$.
4. Color the point $\mathbf{x}$ accordingly.

While $f_p$ can be calculated directly using typical quadrature methods, the inverse $f_p^{-1}$ has to be calculated using a Maclaurin series as shown above. This does occasionally result in boundary artifacts (where the metric distortion is greatest due to the embedding), but we have found 5 or 6 terms to be sufficient most of the time.

## Results

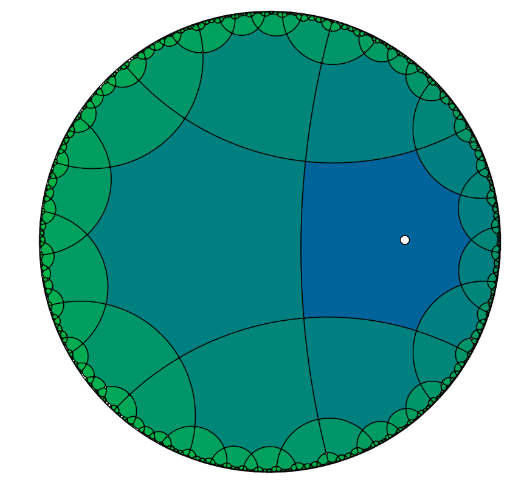

**Figure 1:** A $\{7,4\}$ tiling generated with our program

Figure 1 shows an example tiling generated with the program. You can interact with the tilings in real time for tilings that do not have an image pattern. The coloring in Figure 1 is based on hyperbolic distance from the white circle.

Figure 2 shows an example of what an image tiling looks like. It is much easier to see the size distortion with such a pattern on the tiles. Each tile has the same area, demonstrating the exponentially increasing distortion as you move from the origin to the disk at infinity.

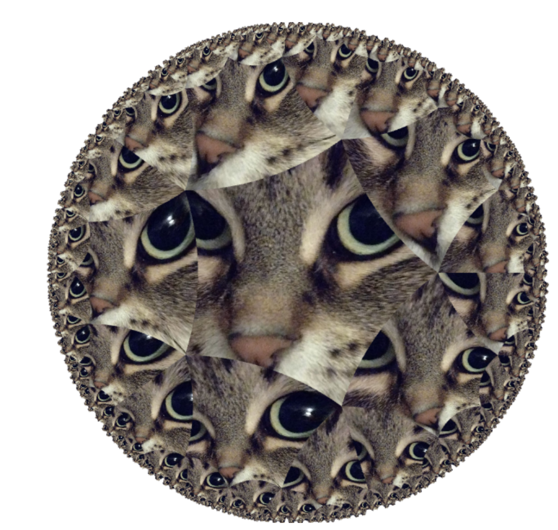

**Figure 2:** A $\{5,5\}$ tiling of the disk with my cat

## Future work

In the future, we would like to be able to interact with image tilings in real time. The program should be able to apply a Möbius transformation to the disk to translate the origin to the mouse position, as with the plain color tilings. The implementation of the conformal mapping in the tiling generator needs to be fixed to avoid artifacts.

## Acknowledgements

Thanks to Dr. Jason DeBlois for mentoring this project, providing extremely useful feedback, and recommending references.

Github: https://github.com/sam-lb/PoincareTessellation