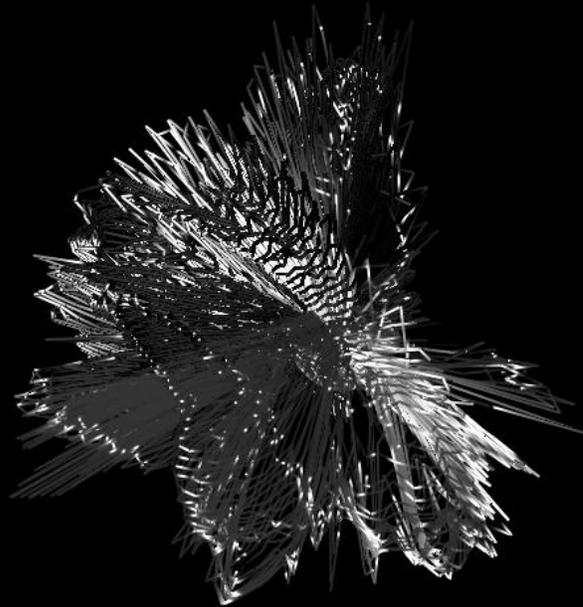


Modification of 3D Shapes by Means of Movie Color Data in Jitter



A tutorial written by Sabina Covarrubias, PhD. **2017**

Second edition 2018
Compatible with both Max 7 and 8 versions

Copyright Sabina Covarrubias,
Paris FRANCE, 2018

Contents

FOREWORD.....	3
Getting acquainted with the patch.....	4
Analyzing the patch.....	5
The big picture.....	5
Detailed analysis of the patch.....	7
Phase 1: load a movie file.....	7
Phase 2: extract the colors from this movie as numeric values.....	7
Phase 3: sum the color components of the movie.....	8
Phase 4: map the color values into a “geometry matrix”.....	9
Phase 5: merge the values obtained from the movie into a second 3D shape.....	12
Phase 6: make the shape visible and render.....	15
Phase 7: “aesthetic” modifications.....	18
Conclusion.....	20

FOREWORD

This tutorial is divided into two main sections. In the first part, you will learn from watching a video how to use the patch “Shapes6” as a user. In the second section, you will understand in depth the architecture and the operational system of the software.

There are two modes for reading this tutorial: a superficial (fast) mode, or a thorough one. If you need to understand the general principles of the patch and the objects that compose it, go over the information found outside the blue boxes. Otherwise, if you aim to have deep understanding about the attributes and arguments of each object, then go through the information contained in the blue boxes as well.

The patch is divided into 7 sections. Each section is a part or a phase of the process. I have assigned different background colors to each phase for better identification. Please open the “Shapes6” patch and identify the 7 different colors. Note that each phase in the patch corresponds to a section in the tutorial:

- Phase 1: load a movie file
- Phase 2: extract the colors from this movie as numeric values
- Phase 3: sum the color components of the movie
- Phase 4: map the color values into a “geometry matrix”
- Phase 5: merge the values obtained from the movie to obtain a second 3D shape
- Phase 6: make the shape visible and render
- Phase 7: “aesthetic” modifications

For optimal use of this material, I suggest you having the patch open while reading the tutorial and regularly see there the objects I am describing in the text. Since I will describe each object in a very detailed way, it is important to zoom out constantly and remain focused on the general process.

Enjoy your learning experience!

Getting acquainted with the patch

In this chapter, you will learn how to set up and use the patch “Shapes6”. In addition, you will get familiarized with the range of visual “styles” that it can create. Note that this first approach will be at the level of a software user.

What can this patch do?

This software allows to create reactive visual content from a movie file. In other words, it modifies a 3D shape by means of the “color values” from a movie file. The patch can create different types of visuals in many styles, so you will be able to personalize your own work (Fig. 1).

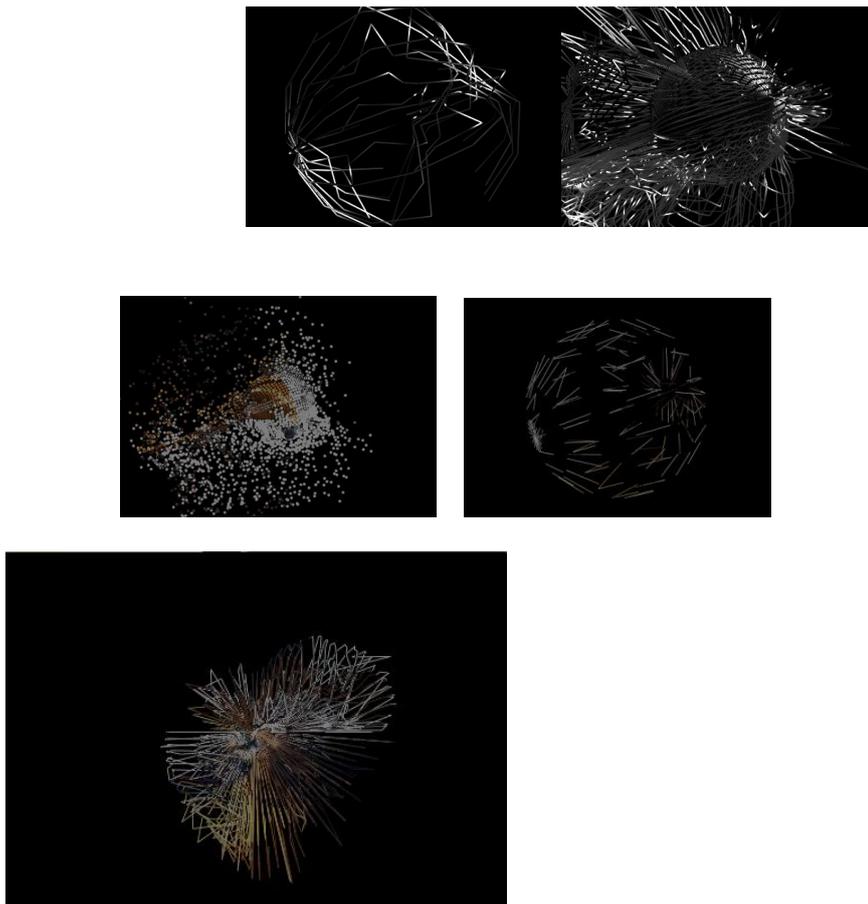


Figure 1. Examples of visual content generated by the patch “Shapes6”. By changing the `jit.gl.mesh` object attribute `@draw_mode` value you can visualize a geometrical surface as points, lines, polygons, etc.

Since the learning process for using a given software is easier through videos, I prepared a guided demonstration for you. Watch now “Jitter Patch Shapes6 , user guide” to learn how to use the patch.

<https://youtu.be/8QcZ1Qc7qlw>

Analyzing the patch

In this chapter, you will learn about the architecture and the working principles of this patch. You will understand each part of the process and the function of each element on it. First, I will explain the whole process in a general way. Then, I will describe each feature of it .

The big picture

Before analyzing this patch in detail, we need to have a simple and general idea about how it works. The sequence of the main tasks that this patch performs is the following:

- 1. Load a movie file.**
- 2. Extract the colors from this movie as numeric values.** In Jitter, each color can be composed of 3 numeric values: R (red), G (green) and B (blue).
- 3. Sum the 3 components of each color (RGB) from the movie to obtain one single value for each color.** In other words, it means that, at this point, each color is now composed of one single value, instead of the three values R, G and B. As a result, the movie will be displayed in grayscale.
- 4. Map the color values into a “geometry matrix” so that each value represents a point in a 3D space.** It means that the values are no longer color features, instead, they are points in a three-dimensional coordinate system. Hence, we are transforming the color information into a 3D shape. Accordingly, we'll now define each point in the three coordinates x , y and z (Figure 2).

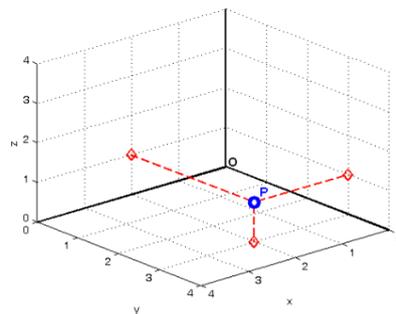


Figure 2. A representation of one single point in a three-dimensional Cartesian coordinate system. To define a position, the point needs the three coordinates x , y and z .

- 5. Merge the values obtained from the movie to a second 3D shape** (e.g. a plane or a sphere). It means that the incoming data from the movie (originally color data and now “space information”) will modify that second 3D shape in real time.