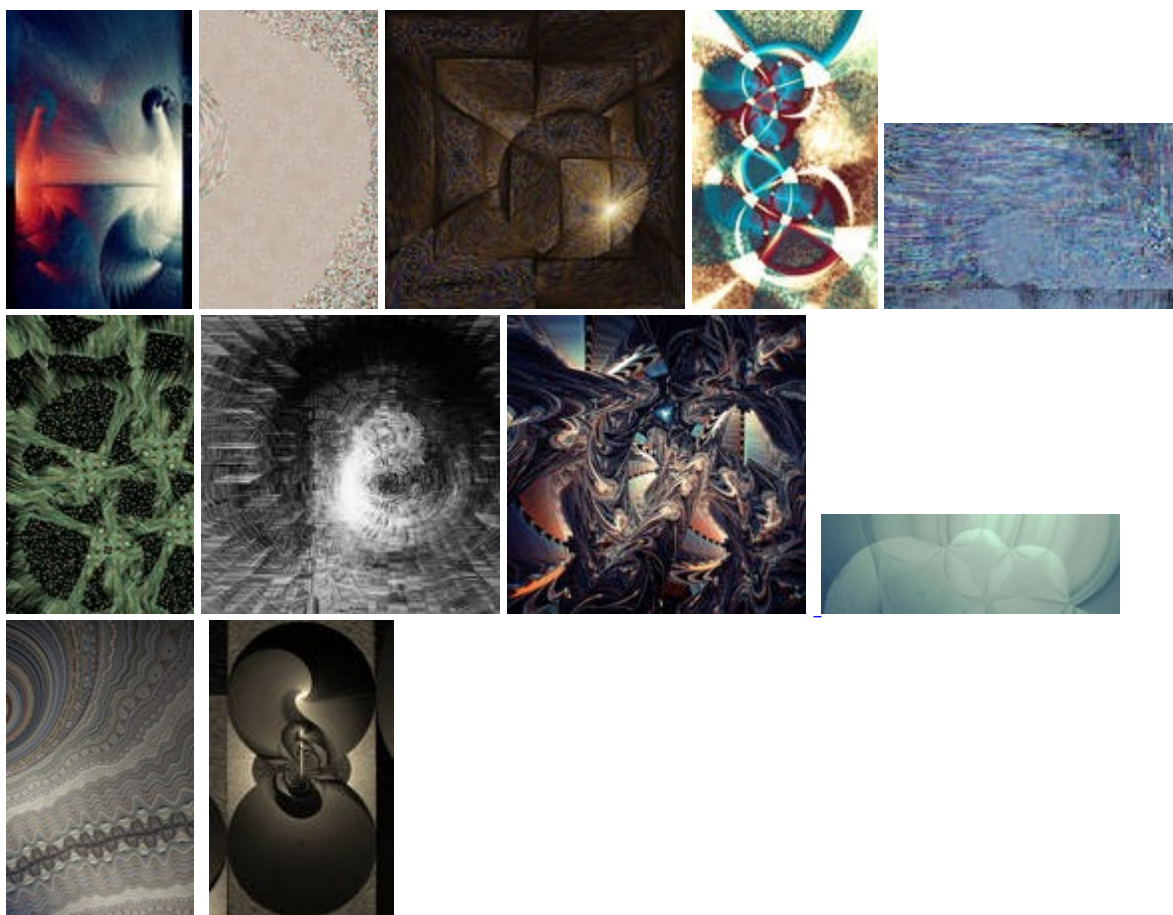


What is Glitch Style?

by FarDareisMai, Aug 13, 2013, 8:17:54 PM

This question, like "What is the biological definition of life?" and "What images and sets count as 'fractals'?" is not as straightforward as it first appears!

In the end, glitch style is really determined by its look: patterns and textures that almost never occur naturally in apophysis. These patterns and textures can take many forms:



How are these patterns and textures achieved?

- There is usually a strict deterministic chaos loop: Xform 1 feeds only to xform 2, which feeds only to xform 3, which feeds only back to xform 1. There is no branching, as in a more typical IFS. Sometimes, extremely unbalanced weights can give a similar effect.
- Transforms often only alter the input sample slightly. Often, they are only slight alterations of the identity transform (linear=1 in default position at origin), which does not alter the input sample at all. For instance, a transform with rectangles=1 and very small values for the rectangles x and y variables, or a linear transform with .001 of some other variation added, or a transform that is only scaled or translated .001

units from the default position.

- Spherical and other plane-inverting transforms obviously alter the input sample a lot, BUT can be arranged so that the overall behavior is stable, i.e. the sample alternates between two locations but does not move unpredictably around the plane. It is to this stable behavior that you would add one of the "slight alterations" detailed above.
- Glitch style fractals usually have very few transforms, sometimes only one.
- Glitch style fractals are generally rendered at much lower density than usual, as the patterns get too "smoothed out" at a high density.
- Glitch style fractals are VERY difficult to color. Oftentimes all transforms are set to color speed=1. Non-black background colors and chaotica's curves can also be very helpful.
- The following variations have been very useful in making glitch-style fractals: linear, spherical, y, disc, lazysusan, lazyjess, lazytravis, rectangles, mobius, and wedge. However, glitch style can probably be made with almost any variation!

Now, none of the above are *strictly* necessary for something to count as glitch style. I've made glitchy-looking fractals without a chaos loop, with transforms that chaotically alter the input sample, and even glitch style fractals that look okay with high density/sampling level. However, they are very useful to keep in mind; it is definitely easier to get a glitchy-looking render if you are following these guidelines. But, as stated at the beginning, glitch style is really determined by how the final result *looks*. Thus, it can be hard to identify something as "glitch style" if you aren't yet familiar with what apo *usually* looks like. In the end, you may just have to solicit the opinion of someone who has already worked with glitch style a lot (so, basically me or [tatasz](#)). But... hopefully this little writeup will help quite a bit to further the understanding of glitch style!

I've always had the idea of releasing a little glitch-style param pack to help people get started, as it seems many people have difficulty just going off my description. It's not really assembled yet, but in the meantime if you'd like some glitch style sample params you can always note me.



ETA: [Param pack](#) has been uploaded.