How the Ai art works?

William Yip
e-mail: mailme@williamyip.com
website: www.williamyip.com

Abstract

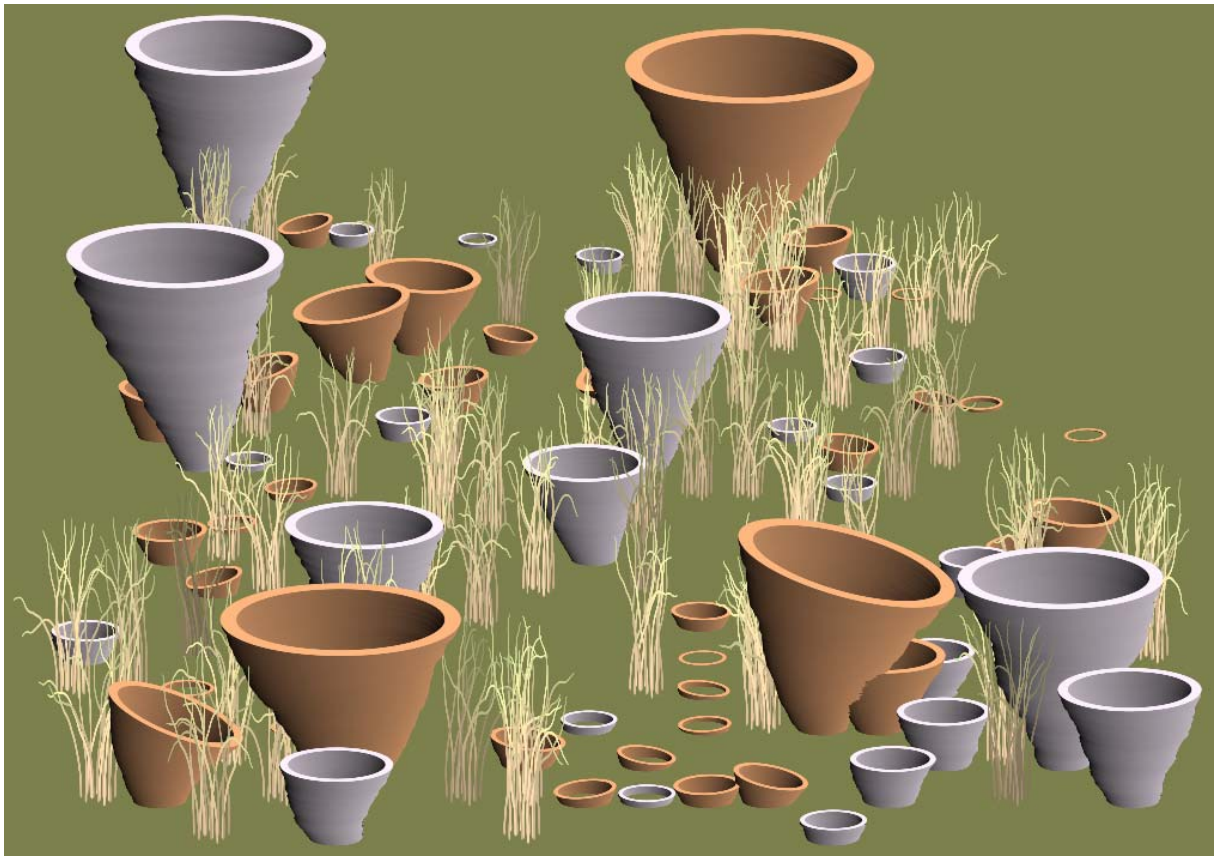
Ai art enables the rendering of arbitrary sized graphical images. It features a two dimensional (2D) graphical space, and some advanced possibilities of graphical definition, for example recursion. It is

1. The Program

It was a command line program for unix-like systems. It is an environment for editing and rendering design grammars Features:

- Simultaneously available for Macintosh, Windows and Posix/Unix.
- Progressive image update: watch it generate
- Save generated images in PNG or SVG format.
- Produce animations
- Edit grammars and re-render easily.
- Render very large images (as large as 100 Mega-pixels).
- Can handle generated images with millions of shapes.
- Carefully tuned graphics rendering

2. The Output



```
//----
//---- T U B U L E   G A R D E N
//----
//----
//----
//----
startshape garden
background { b -0.5 sat 0.4 hue 64 }
rule garden
{
        20* { y 1.5 z -1000 s 0.98 } row { x -25 }
}

rule row
{
        20* { x 2.5 } plant { }
}

rule plant { tubule { sat 0.05 h 300 } }
rule plant { segule { sat 0.5 h 26 } }
rule plant 2 { straw { sat 0.3 h 32 b 1 } }
rule plant 7 { }


// ********** SEGULE **********
```

```
rule segule 0.037
{
        ringule { s 0.95 0.28 b 1 }
        ringule { s 1 0.3 b 1 }
}
rule segule
{
        ringule { s 1 0.3 }
        segule { r 1.2 y 0.02 s 1.01 z 1 }
}

rule segule
{
        ringule { s 1 0.3 }
        segule { r -1.2 y 0.02 s 1.01 z 1 }
}

// ********** TUBULE **********
rule tubule 0.06
{
        ringule { s 0.95 0.28 b 1 }
        ringule { s 1 0.3 b 1 }
}
rule tubule
{
        ringule { s 1 0.3 }
        tubule { x 0.02 y 0.025 s 1.01 z 1 }
}
rule tubule
{
        ringule { s 1 0.3 }
        tubule { x -0.02 y 0.025 s 1.01 z 1 }
}
rule tubule
{
        ringule { s 1 0.3 }
        tubule { x -0.01 y 0.025 s 1.01 z 1 }
}
rule tubule
{
        ringule { s 1 0.3 }
        tubule { x 0.01 y 0.025 s 1.01 z 1 }
}
rule ringule
{
        180* { r 1 b 0.008 } CIRCLE { x 1 s 0.09 z -0.1 }
        180* { r 1 b 0.008 } CIRCLE { x -1 s 0.09 z 0.1 }
}


// ********** STRAW **********
```

```
rule straw { straw { b -0.1 } }
rule straw { straw { b 0.1 } }
rule straw
{
	wheat { s 1 0.3 }
	wheat { x -0.2 y 0.2 s 1 0.3 }
	wheat { x 0.2 y 0.2 s 1 0.3 }
}
rule wheat
{
	root { flip 180 s 0.05 0.5 }
}
rule root
{
	grow { x 0 y 0 b -0.2 }
	grow { x 5 y 1 }
	grow { x -5 y -1 }
	grow { x 10 y 0  b -0.2 }
	grow { x -10 y 0 b -0.1 }
}
rule grow { jump { } }
rule grow { jump { flip 90 } }
rule jump
{
	spot { }
	jump { y -0.2 r 0.5 s 0.9968 h 0.08 }
}
rule jump 0.03
{
	spot { }
	jump { y -0.2 r 0.5 flip 90 s 0.9968 h 0.08 }
}
rule spot
{
	SQUARE { }
}
```