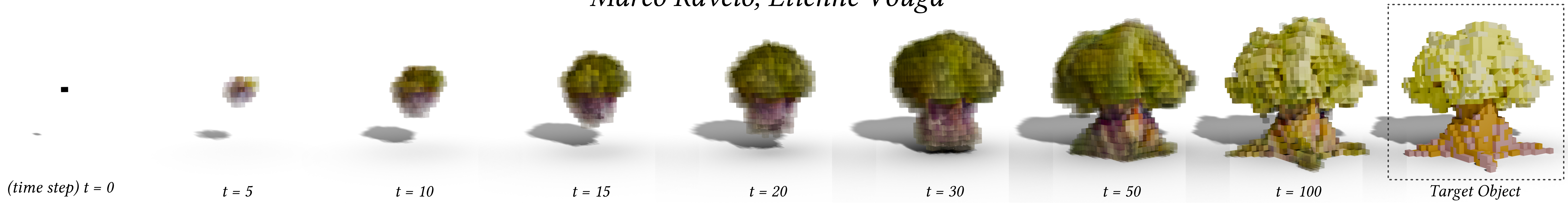
Artificial Morphogenesis via 3D Neural Cellular Automata

Marco Ravelo, Etienne Vouga



This project presents a computational model of morphogenesis, the biological process by which an organism grows and takes its form. We employ three-dimensional neural cellular automata (NCA) to grow from a starting seed state into a target object from a series of update steps. NCA, a modern derivative of classical cellular automata (CA), makes use of a neural network within its update step. Akin to real-world cells, each artificial cell acts independently of one another and is only able to perceive adjacent neighboring cells. Despite lacking a global control mechanism, the cells act as a collective such that a trained NCA model displays morphogenic properties such as growth in size, shape, and complexity over time, regeneration after cellular damage, and isotropic self-organization. Understanding the intricacies of morphogenesis has the potential to provide beneficial insights for the development of regenerative medicines, self-organizing robots, and other bio-engineering endeavors.

Our NCA model is built from a collection of cells depicted as unit cubes (voxels) situated at integer coordinates within a regular 3D orthogonal grid. We represent each cell's state as a vector of 16 real number values between *0* and *1*. The update step is what propels an NCA model through time, such that one update step takes a model from time-step *t* to time-step *t+1*. It is comprised of four distinct sub-updates: (1) cell perception, (2) a neural update rule, (3) a stochastic update, and (4) the application of a living cell mask. The training regimen developed to build our NCA models includes three pre-training steps: (1) selecting a starting seed and a target object, (2) creating a sample pool, and (3) selecting model and training parameters, and three per-epoch training steps: (1) sampling randomized batches from the sample pool, (2) applying cellular damage in order to learn regeneration, and (3) calculating the loss from model results in order to correctly tune the neural network parameters.