EcoBrush:
Interactive Control of Visually Consistent Large-Scale Ecosystems

J. Gain[1,2], H. Long[1], G. Cordonnier[2] and M.-P. Cani[2]

[1]Computer Science Department, University of Cape Town
[2]University Grenoble-Alpes, CNRS(LJK), and Inria
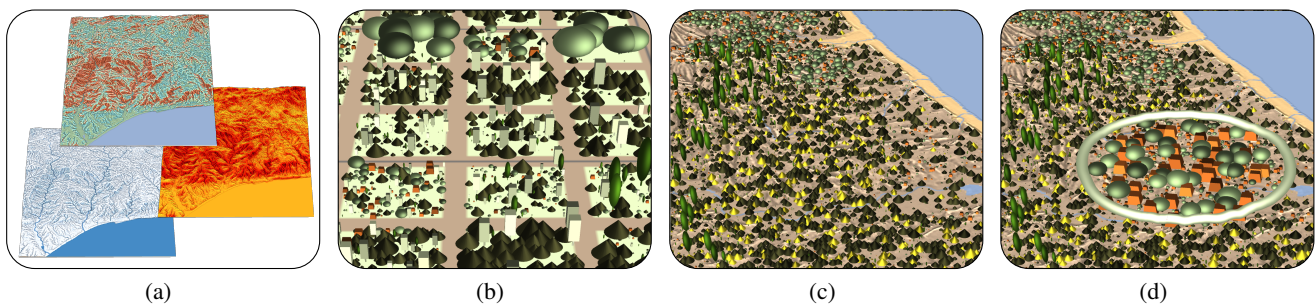
(a) (b) (c) (d)

Figure 1: *Terrain conditions (a), such as temperature, soil moisture, and sunlight exposure, are used to index a database of plant distributions (b) and synthesise an initial complete ecosystem (c), which can then be modified with semantic brushes, to adjust age, density and variability (d).*

Abstract

One challenge in portraying large-scale natural scenes in virtual environments is specifying the attributes of plants, such as species, size and placement, in a way that respects the features of natural ecosystems, while remaining computationally tractable and allowing user design. To address this, we combine ecosystem simulation with a distribution analysis of the resulting plant attributes to create biome-specific databases, indexed by terrain conditions, such as temperature, rainfall, sunlight and slope. For a specific terrain, interpolated entries are drawn from this database and used to interactively synthesize a full ecosystem, while retaining the fidelity of the original simulations. A painting interface supplies users with semantic brushes for locally adjusting ecosystem age, plant density and variability, as well as optionally picking from a palette of precomputed distributions. Since these brushes are keyed to the underlying terrain properties a balance between user control and real-world consistency is maintained. Our system can be be used to interactively design ecosystems up to 5×5 km^2 in extent, or to automatically generate even larger ecosystems in a fraction of the time of a full simulation, while demonstrating known properties from plant ecology such as succession, self-thinning, and underbrush, across a variety of biomes.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-dimensional graphics and realism—

1. Introduction

Natural landscapes serve an important function in computer graphics and virtual environments. Indeed, in applications ranging from computer games and movies to landscape design and VR-based training, virtual landscapes are frequently a dominant visual element. One key aspect, particularly in larger scenes, is the depiction of plant ecosystems, from the modelling of individual plant specimens, to their collective placement, and final rendering. While models of individual plants [DL06], and rendering of large-scale ecosystems [DCDS05] have received significant attention, the eco-placement problem, which extends beyond plant position to include

attributes such as species, age, height, vigour and canopy extent, is less well studied.

Broadly speaking, there are two approaches to eco-placement: bottom-up simulation of individual plant interactions [BSK*15] or top-down randomised synthesis from statistical distributions [EVC*15]. Simulation has the advantage that it explicitly encodes plant dynamics and supports known emergent ecosystem behaviour, such as succession, where slower growing but more suited plants gradually dominate, and self-thinning, where the overall number of plants is reduced in favour of fewer mature specimens. Unfortunately, it can take hours or even days to generate large-scale ecosystems, even with appropriate n-body parallelisation [NHP07]. Synthesis, on the other hand, uses random placement guided by statistical properties, and is capable of interactively populating large terrains with hundreds of thousands of plants. However, the difficulty lies in encoding ecosystem properties in the distributions so that the results are visually consistent with real landscapes.

Our key strategy is to complete the loop between simulation and synthesis, enabling many small-scale but high-fidelity simulations to be fed as inputs to distribution analysis and later synthesis. Figure 1 shows the typical workflow and key components of our system. The user supplies a base terrain and some non-biological (abiotic) input parameters, such as monthly rainfall, from which monthly terrain conditions for temperature, soil moisture, sunlight exposure and slope are derived. Off-line a database is pre-populated with histogram-based disk-distributions, which capture the statistical interactions between plant canopies and are derived from sandbox simulations of a particular biome. These samples capture the biological (biotic) competition for resources between individual plants. Based on the abiotic maps, distributions are drawn from the database and synthesised across the terrain to rapidly populate an initial ecosystem. A user can then overpaint the ecosystem to locally modify age, variability and plant density, while respecting the underlying terrain conditions.

Our technical contributions include:

1. Extensions to the state-of-the-art in both ecosystem simulation and synthesis. On the simulation side, we derive abiotic maps automatically rather than expecting artists to paint complex geomorphological attributes manually, and enhance plant simulation by separating between moisture (root) and sunlight (canopy) competition, as well as by incorporating gradiations of canopy density. On the synthesis side, we depart from the use of point processes, to include disk-based statistics. This is vital to reproducing the canopy-based spacing of plants and for placing shade-tolerant species under cover.
2. Two alternative approaches for using sandbox ecosystem simulations: First, uniform conditions can be sampled from the range of the input biome. Alternatively, terrain conditions can be clustered and cluster means used as simulation inputs, for a manageable set of seed distributions. This second solution enables us to generate very large ecosystems ($10 \times 10 \ km^2$) without the need for a large distribution database.
3. Consideration of the balance between artistic freedom and real-world consistency. Since distributions are indexed by terrain conditions, we can offer freedom to the user - such as overpainting a region with another distribution or arbitrarily tuning

plant density - while providing a suitability map that indicates consistency with terrain conditions. If desired, the user can then use a "healing brush" to progressively shift the new distribution towards the closest consistent state.

These key contributions enable the efficient creation and interactive editing of large-scale landscapes that accord with known emergent properties from plant ecology.

2. Related Work

There are many challenges to effectively portraying natural environments in computer graphics, including modelling the fine-scale geometry of individual plants and rendering the resulting scenes, which exhibit both large scale and high depth complexity. The rendering problem is typically addressed by a custom combination of techniques based on distance from the viewpoint, including instancing [DHL*98], level-of-detail [DCSD02], volumetric textures [DN04], and ray tracing [DHL*98, DCDS05]. Plant modelling also has a long history [DL06], with the use of L-systems as a dominant approach [PL12]. Notable in ecosystem terms is the inclusion of context sensitivity [MP96, PMKL01] in which plants adapt to environmental factors, such as light, moisture, and obstacles. While rendering and plant geometry are vital to computer-generated natural scenes, our focus here is on the problem of plausible plant placement.

If we scope our problem as populating a terrain with a collection of plants, i.e. assigning attribute values to individual specimens, such as position, species, height, canopy extent, and vigour, then there are two broad approaches: bottom-up simulation and top-down statistical synthesis.

Simulation takes the Lagrangian approach of treating plants as particles. Using either simple radial interactions [DHL*98, AD05], L-systems [LP*02] or more complex agent-based models [Ch'13, BSK*15], plants are seeded at a location, grow in size asymmetrically [AD05] under competition for resources, propagate on reaching maturity, and die due to lack of resources or old age. The fidelity of these simulations varies depending on which abiotic features, such as temperature, sunlight, moisture, wind, and slope, are considered and on the biotic complexity of individual plant models and their local interactions. For instance, ecosystem disturbance by humans, grazing animals, and fire is, with a few exceptions [BE03, Ch'13], otherwise neglected. In the simulation literature, Ch'ng's models [Ch'09, Ch'13] are the most complete, incorporating layered soils, various forms of seeding and seasonal differences in evergreen and deciduous species. Nevertheless, the botanical literature [FPR*96, SIK07, SHG*08] goes further to model in detail cycles such as evapotranspiration, photosynthesis and carbon dynamics. With some exceptions [SIK07] this is in the service of Dynamic Global Vegetation Models, which are grid-based Eulerian simulations of climate impact on plant distributions. These typically provide dry-mass proportions for species in cells ranging from 30m to several kilometres on a side, and do not directly enable plant placement.

There is, however, a tension between simulation fidelity and performance. Complex models, even with approximate nearest-neighbour acceleration, do not scale well to large, dense environ-

ments and may take minutes or even hours to execute. A viable approach is to tailor models so that they consider abiotic and biotic factors only to the extent that emergent properties such as succession, self-thinning, clustering and shading are correctly exhibited. We adopt this strategy in precomputing ecosystems over small areas.

Statistical synthesis, on the other hand, generates plant positions randomly to fit a given distribution. Solutions range from straightforward half-toning of a density image [DHL*98] and dart throwing [ACV*14], which exhibit little more than basic plant separation, to Wang tiles, with an underlying Field-of-Neighbourhood plant distribution model [AD06], kernel-based deformation of probability density maps [LP*02] and derived cross-species distribution histograms [EVC*15], which can capture complex interspecies and environmental interactions. These approaches fall under the umbrella of the probability theory of statistical point processes [Dig13]. In fact, plant ecologists have long drawn on this theory [LIB*09] to extract summary statistics for field-based sampling. It has also seen recent application in computer graphics [OG12, EVC*15] for the analysis and synthesis of point distributions. While building on this vein of research, we extend it by explicitly incorporating the extent of plant canopies into distributions. This enables us to capture effects such as the coverage of shade-loving plants.

The issue of user interaction in ecosystem modelling has received less attention than is warranted. The ideal is direct, interactive and intuitive control over the final state of the ecosystem in a way that respects both the users' intentions and physical plausibility, perhaps through a sculpting, painting or sketching interface. Unfortunately, the interfaces of ecosystem simulators [DHL*98, Ch'13] commonly take the form of painting input image maps (e.g., water, soil type, temperature). Users without a deep understanding of geomorphology and plant dynamics have difficulty designing final lengthy simulation outcomes from the poorly understood inputs of these systems. Bradbury et al. [BSK*15] partially address this problem by allowing user-defined regions of a simulation to be paused, edited and then restarted to run at different simulation rates. WorldBrush [EVC*15] provides more direct and immediate interaction through a palette of distributions that can be painted directly onto a landscape, with regions then modified through copy and paste, gradient interpolation, and seam-carved stretching. Unfortunately, WorldBrush neither enforces nor guides realism and instead relies on the user to maintain the link between abiotic conditions and selected distributions.

3. Overview

Our goal is to allow users to interactively design large-scale ecosystems that accord with observed landscapes and botanical principles. Regrettably, interactive simulation of ecosystems larger than a few hundred metres on a side is generally infeasible, because, all other things being equal, the number of plants is proportional to the simulation area. Instead, we use precomputed distributions derived from small sandbox simulations. Analyzing simulation results enables us to derive an ecosystem with the same statistical properties at least two orders of magnitude faster, enabling subsequent interactive editing while preserving consistency.
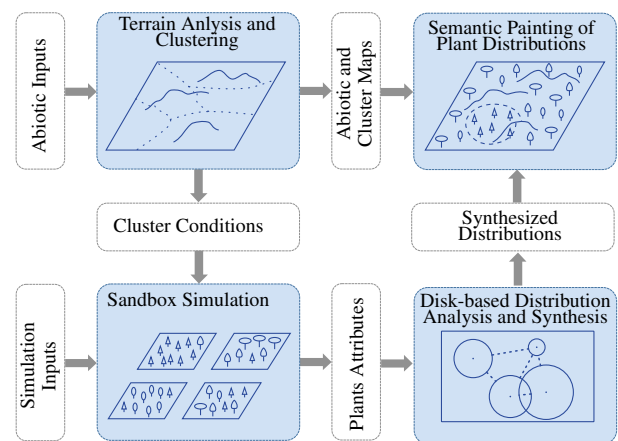

Figure 2: *Method overview: The user provides a heightfield and other abiotic input parameters to a simulation process, which generates abiotic terrain maps for temperature, soil moisture, slope, and sunlight exposure. A clustering process then groups terrain regions according to similar abiotic conditions. As a separate preprocess, the range of possible abiotic conditions are sampled and small scale sandbox simulations are executed and then analysed to create a database of distributions. The abiotic maps, clusters, and distribution database are used to synthesise an ecosystem, which can then be modified by the user using semantic painting and resynthesised interactively.*

More precisely, our system automatically populates the input terrain with an initial ecosystem, where plants match local abiotic conditions computed on the terrain. This ecosystem is generated from a meaningful set of sandbox simulations, analyzed and stored in a distribution database: the right plant distribution is synthesized on each region of the terrain, depending on the local abiotic conditions (see Figure 2). Users are then able to paint with a set of modifying brushes that can increase or decrease ecosystem age, variability, and density, even targeting individual species types. They can also select and paint specific distributions by treating the distribution database as a palette, while being guided by the system to maintain consistency.

To initially populate an ecosystem, the user supplies some simple inputs: a terrain heightfield, latitude, soil type, yearly temperature extremes, and average monthly rainfall. Rather than requiring users to paint abiotic maps, which is difficult to do without domain expertise, we use these inputs to simulate terrain conditions and automatically generate the maps. They include monthly abiotic condition maps for temperature (as affected by altitude), daily sunlight exposure (influenced by latitude and terrain self-shadowing), and soil moisture (a combination of rainfall, slope, soil absorption and runoff). An important insight of our approach is to cluster regions with similar conditions, enabling us to factorize the need for ecosystem simulations. This is done by collectively applying k-means clustering to the abiotic maps. This first stage of the pipeline is detailed in Section 4.

The clusters of abiotic conditions can either be used to uniformly

sample conditions for ecosystem simulations, such as temperature, sunlight exposure, and soil moisture, from the range of the corresponding biome; or alternatively the cluster means can be directly used to trigger a minimal set of such simulations. In both cases, we use small scale ($100 \times 100m^2$) sandbox simulations (See Section 5).

Also supplied to these simulations are viability parameters for a set of plant archetypes (or Plant Functional Types (PFT's) in plant ecosystem parlance), representing collections of similar species, such as "Tropical Broad-leaved Evergreens" or "Boreal Needle-leaved Summergreens". This approach avoids having to exhaustively determine parameters for hundreds of species [FPR*96].

Next, plant attributes (type, position and canopy extent) from simulation results are analysed to derive a new type of statistical distribution, where point processes are extended to distributions of overlapping disks representing plant canopies (See Section 6). Stored as pairwise interaction histograms relating expected distances between and within PFT categories, these distributions are placed in a database ready for subsequent synthesis to reproduce the simulation characteristics on a larger scale. Populating the database is kept manageable by restricting the simulation dimensions and executing them as a pre-process.

The ecosystem we generate from these distributions (where individual species are finally mapped from PFT's using a statistical post-process) can be used for rapidly and automatically populating large terrains without the need for any user intervention. We also believe this initial ecosystem with typical plant coverage is more helpful as a starting point for users than a blank canvas. We then provide a set of high-level painting tools enabling users to interactively and locally modify distributions and hence synthesised plant populations. The challenge is to design tools that provide freedom to the user while maintaining consistency in terms of abiotic constraints. This last interactive editing stage is described in Section 7.

4. Simulation and Clustering of Abiotic Terrain Conditions

In deriving abiotic conditions our aim is to require only a limited set of comprehensible user inputs, while still capturing significant terrain effects that impact ecosystems, including shadowing in defiles, increased runoff on slopes, water table access near rivers, temperature lapse with altitude, and the effects of latitude on growth cycles. Nevertheless, in the interests of efficiency our approach remains approximate and many secondary factors are not considered. In particular, our simulations are forward directed and do not consider feedback cycles, such as evaporation due to canopy interception and soil nutrients from decomposing litterfall.

The product of our simulation of abiotic conditions is a set of monthly abiotic maps over the terrain (see Figure 3) with averaged daily values for maximum temperature, sunlight exposure, and soil moisture, derived as follows:

Temperature – Our consideration of temperature is straightforward: the user provides maximum daily temperatures (averaged over a month) for the middle of winter and summer, at the lowest point on the terrain. These temperatures are linearly interpolated to the remaining months and reduced with altitude (at a user-configurable lapse rate, which defaults to $6.5\,^{\circ}$C per 1000m) to provide average daily maximum temperatures for each month.
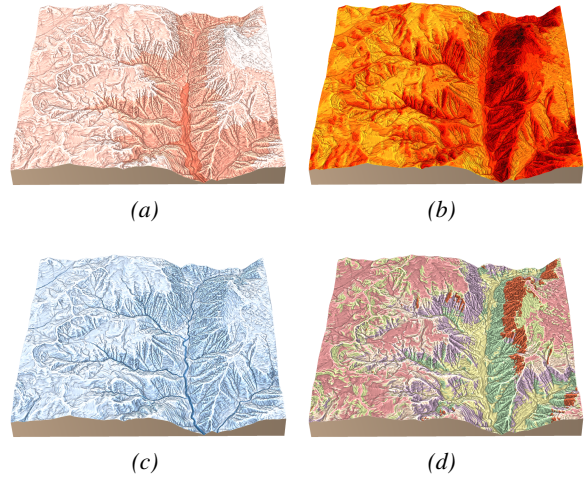

(a) *(b)*

(c) *(d)*

Figure 3: *Abiotic maps and final clusters: (a) temperature from altitude lapse, (b) sunlight exposure from latitude, affected by terrain shadowing, (c) soil moisture as a consequence of rainfall, absorption and runoff, and (d) clusters formed by grouping combined abiotic conditions. The simulation uses monthly abiotic maps, which are averaged here for illustration purposes.*

Sunlight Exposure – From a latitude and compass direction provided for the terrain, and factoring in the earth's axial tilt, it is possible to derive the sun's trajectory at the middle of a given month [SIK07]. We cast rays from the sun, sampled at hourly intervals, to each heightfield position. If a ray is intersection free then the terrain position is not in shadow for that hour. This allows us to derive a map of the daily hours of direct sunlight on a monthly basis.

Soil Moisture – In reality the water content of soil is governed by a complex combination of precipitation, evaporation, runoff, seepage, soil type and water uptake by plants, among other factors. We approximate this with an equilibrium model that dynamically adjusts a single plant uptake value k_p so that over multiple yearly cycles with the same precipitation the soil moisture achieves a rough balance. Also, instead of modelling soil type explicitly, the user configures a single maximal monthly soil absorption value, k_a, and overall reservoir capacity, k_c. Slope acts as a proxy for soil type by linearly weighting the absorption to represent the transition from loamy soils in valleys to rock on cliff faces. Any rainfall above this value contributes directly to runoff. Thus, we model the soil moisture $M_{i,m}$ for a given cell i and month m as:

$$M_{i,m} = \min(k_c, M_{i,m-1} + \min(P_m, A_i) - k_p),$$

where P_m is the monthly precipitation provided by the user, and A_i is the per cell absorption threshold. Any excess absorbed moisture, $S_{i,m}$, above the soil capacity is spread as seepage over the next three months. Free surface water is then defined as a combination of runoff and seepage, as follows:

$$F_{i,m} = \max(0, P_m - A_i) + \sum_{j=1}^{3} \frac{1}{3} S_{i,m-j}$$

From this runoff and seepage, a network of waterways can be established. For our purposes, plant growth is prevented in areas of open water but enhanced along riverbanks due to a more accessible water table. We employ the drainage model of Cordonnier et al. [CBC*16], which provides water volume ($F'_{i,m}$) taking into account excess initial moisture ($F_{i,m}$), inflow from neighbouring cells and outflow to the lowest neighbour. Any cell above a threshold ($F'_{i,m} > k_o$) is marked as open water. These waterways are then expanded to neighbouring cells j within distance $d_{ij} < 0.5\,W_i$ using Tucker et al.'s [TH10] rough river-width formula: $W_i = 10\,F_i^{0.5}$ (with F_i in $m^3\,s^{-1}$). A second shell of cells within $d_{ij} < 0.5\,k_m\,W_i = D_m$ has its soil moisture (M_j) increased by a set amount (f_r) that is tailed off in proportion to relative slope and distance from the river, so as to represent water infiltration along riverbanks. For each non-river cell j:

$$M'_{j,m} = M_{j,m} + \max(0, \max_{i \in N_j} \frac{D_m - d_{ij} * (1 + k_s s_{ij})}{D_m} f_r),$$

where s_{ij} is the slope, N_j is a set of cells in the neighbourhood of j, and k_m and k_s are weighting constants for slope and distance, used to adjust the extent of riverbanks.

The final step in this pipeline is to identify clusters over the parameter space of abiotic conditions. These serve as seeds for an initial ecosystem and, in the absence of a distribution database, can supply a usefully constrained set of simulation parameters. Despite the large number of dimensions (12 each for temperature, moisture, illumination, 1 for slope, and 1 for age = 38), conditions are correlated and continuous to the extent that clusters are relatively contiguous and simple k-means clustering [Jai10] suffices. Typically, a choice of $6 - 10$ clusters results in cluster means that do not diverge too markedly from the underlying conditions (as evident in Figure 3).
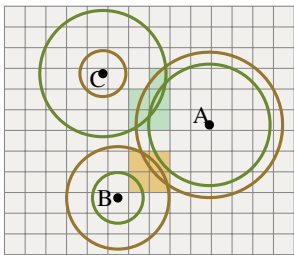
5. Sandbox Simulation

Figure 4: *Competition for sunlight and moisture. The circular canopy (green) and root extent (brown) of a plant are discretised on a grid and plants compete in individual cells of overlap (shaded) to derive averaged scores for moisture uptake and sunlight exposure.*

In order to supply plausible plant distributions to our analysis and synthesis procedure, a series of simulations is undertaken in small-scale $100 \times 100m^2$ regions with uniform input conditions. Conceptually, a sandbox simulation proceeds as follows: In each month, the vigour, $V(p) \in [-1, 1]$, of every plant, p, is calculated. If positive, this vigour value weights the plant's growth rate; if negative, it provides a probability of mortality, as does a plant reaching the age of senescence. Vigour is a function of the Plant Functional Type (PFT) attributes of a specimen, the abiotic conditions in the plant's neighbourhood and biotic competition with other plants that impinge on a specimen's root or canopy extent. Note that unlike previous work we consider these separately.

If a plant has reached maturity and has sufficient vigour it will annually seed new plants randomly out to a radius determined by the PFT-specific propagation (cones, fruit, etc.).
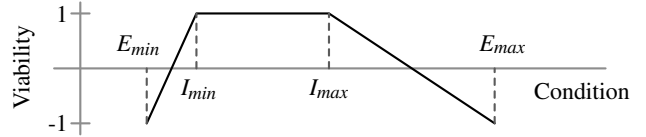

Figure 5: *Piecewise linear functions define PFT-specific viability in terms of climatic extremes $[E_{min}, E_{max}]$ and an ideal range $[I_{min}, I_{max}]$ for each abiotic condition.*

Each Plant Functional Type has a viability function that models response to temperature (B_T), sunlight (B_L), soil moisture (B_M), and slope (B_S), based in part on the corresponding abiotic maps. These are piecewise linear functions (see Figure 5) that range from -1 at the climatic extremes to a plateau of 1 over the optimal inner range. A plant's vigour is then the minimum over all viability function values: $V(p) = \min(B_T(p), B_L(p), B_M(p), B_S(p))$. Taking the minimum in this way is motivated by the consequences of a limiting resource. It matters little, for example, that a plant has abundant water if it is starved of light.

Slope and temperature values are submitted directly to the corresponding viability functions (B_S, B_T), but sunlight and moisture are affected by inter-plant canopy and root interaction (see Figure 4). From a plant's height and type, the circular extent of root and canopy can be derived (such ratios are known as allometries and widely used in plant biology simulation [SIK07, SHG*08]). The intersection of these extents is integrated over a fine-scale grid (we use $20 \times 20cm^2$ cells). For sunlight, the tallest plant in a given cell receives the full sunlight and transmits a proportion, based on its type-specific alpha, to the undercanopy of shorter plants. For moisture, in cases where the minimum needs $\sum_i E_{min}$ of all incident plants i cannot be met, assignment is weighted according to relative root radius. Our motivation is that larger plants tend to dominate through deeper roots and better access to the water table. Finally, a plant's sunlight and moisture values are averaged over all cells within the canopy and root radii, respectively. This allows for partial shading and also seamlessly scales the greater resource requirements of larger plants.

Grass is a special case: a count of grass clumps over a reasonable expanse will likely number in the billions, and synthesising them on the same individual basis as plants is thus clearly infeasible. Rather, as a final pass after plant placement, we subsample the terrain grid and establish grass height per cell. The growth cycle is bypassed and vigour values, based on averaged yearly abiotic conditions, are translated directly into grass height. In terms of biotic factors, we ignore moisture competition on the basis that grass accesses only the topmost layer of soil, but consider sunlight competition by applying a concentric burn of grass heights (moderated by appropriate PFT alpha values) under plant canopies. Because abiotic conditions are sampled more coarsely than grass cells, we also found it necessary to introduce some random variation and low-pass averaging.
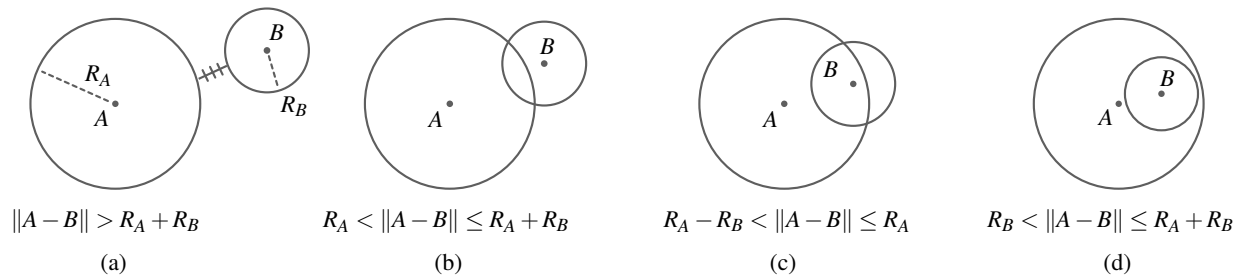
Figure 6: *The different pair-wise disk configurations for plant B with respect to A: (a) bands of distances between the disks, (b) less than half shaded, (c) more than half shaded, (d) fully shaded.*

6. Disk-based Distribution Analysis and Synthesis

One of the main features of our method is to use the ecosystem simulations described above to derive statistical distributions, which can be used to synthesize plants over a much larger terrain at interactive rates. In contrast to conventional point processes, we analyze distributions of possibly overlapping circles of different radii, representing canopies instead of mere points. This extension is an original contribution to point-based analysis and synthesis methods.

Our point of departure is the WorldBrush system of Emilien et al. [EVC*15]. WorldBrush is based on categories $\mathcal{C}_i$ of scene elements (e.g., trees, plants, buildings, and roads) that are ranked to establish both dependencies and an order of evaluation. In our context, each plant functional type is separated by age (seedling, young, mature) into three categories, and all PFT categories are then prioritized according to their relative dominance over resources.

At its core, WorldBrush is built on pair-wise distance histograms between and within categories. Given categories $\mathcal{C}_x$ and $\mathcal{C}_y$ where x has higher priority than y, these encode the probability of a plant in $\mathcal{C}_x$ falling within a certain distance band of a plant from $\mathcal{C}_y$. Such discretised pair correlation functions have proven capable of capturing irregularity, such as clustering and clumping, far better than noise functions [OG12]. Analysis involves counting the incidence of elements of $\mathcal{C}_x$ within concentric annular shells of points in $\mathcal{C}_y$ and normalizing over the total number of elements (n_x, n_y) and the area of the shells to generate a probability distribution. A population of points can then be synthesised by various forms of stochastic search (dart throwing, Metropolis-Hastings) depending on whether the correlation functions are used to score a given distribution or its iterative update.

As mentioned, a pure point process is unable to adequately capture the nuance of plant interaction. Simply drawing a plant's size from a distribution irrespective of position can lead to implausible configurations, such as canopies that intersect the trunks of neighbouring plants or shade-seeking plants that are exposed to direct sunlight. It is thus necessary to simultaneously consider both plant position and canopy radius. (This could, of course, be extended to roots, but we have chosen to focus on the canopy due to its obvious visibility.)

This problem can be classified as a continuous marked point pro-

cess [Dig13], in that additional attributes (marks) are associated with positions (points). An obvious approach would be to shrink or grow a plant's radius based on its local neighbourhood, but this necessitates expensive iterative adjustment spanning all categories that quickly devolves into a form of simulation. Another option would be to divide plant types into subcategories by canopy radius, significantly extending our existing age-based categorisation. In a similar vein, we could develop 2D distribution histograms with distance on one axis and radius on the other. Alas, these are costly solutions: the dependence between categories is $O(N^2)$, while introducing an extra histogram axis will increase distributions by a multiple of the number of canopy bins.

Instead, our strategy is to measure distances between disks rather than points. By jointly considering the positions of two plants (B relative to A), the distance between them ($d = ||A - B||$), and their respective radii (R_A and R_B), we define a new class of histogram bins (see Figure 6) that encodes the degree of canopy overlap: less-than-half shaded, more-than-half shaded and fully shaded. For the non-overlapping case (Figure 6(a)) we populate the distribution histograms according to the closest approach between disks (replacing d with $d - R_A - R_B$ in such calculations). Histogram normalisation also needs to be suitably adjusted to account for R_B in both overlapping and non-overlapping cases. This has the virtues of conceptual simplicity and computational efficiency. The only caveat is that it introduces a form of non-uniform spatial distortion in that the areas represented by histogram bins are no longer consistent between distributions, but this seems to have negligible impact in practice.

We also need to adapt the synthesis process. We begin with a dart-throwing process to match expected plant density, while obeying some simple validity rules ($d > R_A$, $d > R_B$ and $f(X) \neq 0$, where $f(X)$ is the probability density function of the distribution X). This has the benefit of providing rapid initial feedback to the user. Next, eco-placement is refined by iterating between cycles of random position perturbation and radius adjustment of individual plants. A candidate perturbation is accepted with a probability that is proportional to the improvement in the probability density function. In spirit, this is closer to the synthesis of Öztireli and Gross [OG12] than the Metropolis-Hastings sampling of World-Brush [EVC*15].
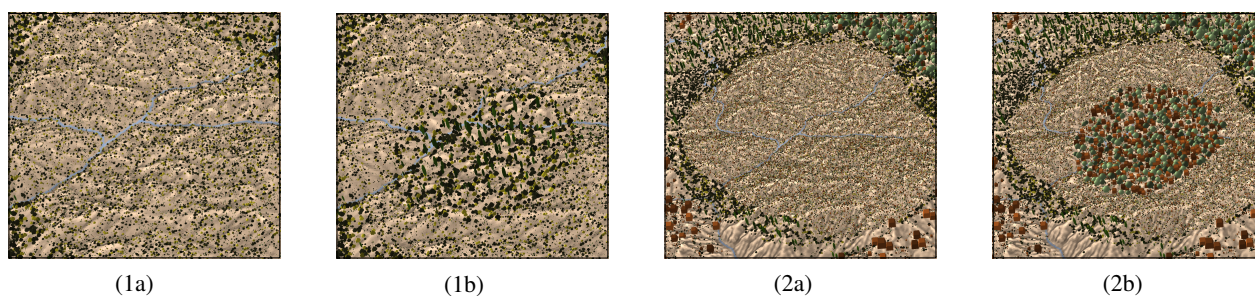
<div align="center">(1a)   (1b)   (2a)   (2b)</div>

Figure 7: *Semantic brushes can be applied locally to an existing ecosystem to, for example, adjust age (increased from (1a) to (1b)), and density (increased (2a) to (2b)).*

7. Semantic Brushes for Plant Distributions

By coupling simulations and distributions we are now able to automatically generate full-scale ecosystems over varied terrain conditions, but it is important to also provide meaningful interactive controls to artists and designers. The WorldBrush solution has users pick a particular distribution from a palette and paint it directly onto the terrain. In our case, this could spoil the plausibility of plant distributions. In contrast, we provide high-level semantic control through a range of context-sensitive brushes to modify the ecosystem age, variability, and plant density while maintaining consistency (see Figures 7 and 9).
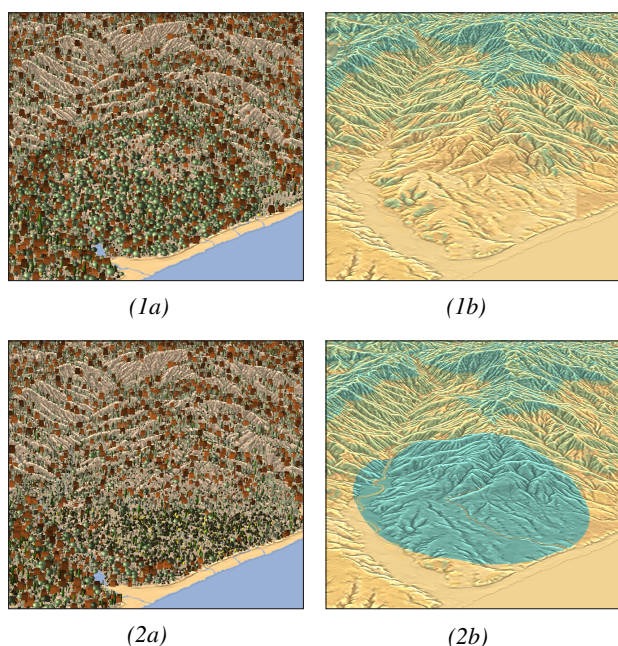

<div align="center">(1a)   (1b)</div>

<div align="center">(2a)   (2b)</div>

Figure 8: *The fit of an ecosystem (1a) is presented to users as a suitability map (1b), where closeness of fit ranges from green (good) to orange (poor). A semantic healing brush enables users to shift ecosystems closer to the underlying terrain conditions (2a), as reflected by improved suitability (2b).*

First a palette of possible distributions needs to be provided to the user. Our sandbox simulation and disk-based distribution analysis pipeline can be interpreted as a mapping ($\mathcal{E} : \mathcal{T} \mapsto \mathcal{D}$) from a space of terrain conditions ($\mathcal{T}$) to distributions ($\mathcal{D}$). To support full ecosystem variety and semantic brush editing we approximate $\mathcal{E}$ by building a database of distributions that sample terrain conditions on a regular grid, $g \in \mathcal{T}$. Since a naive sampling would be intractable due to the high-dimension of the parameter space, we make the following approximations: First, we decrease parameter dimensionality from $\Re^{38}$ to $\Re^5$ using a single seed value for moisture, sunlight and temperature, from which monthly patterns are derived prior to simulation. This approximation sacrifices some of the nuance of monthly variation and captures only broad seasonal effects but reduces the number of sandbox samples reasonably required to cover the search space and hence the cost of pre-processing. Second, we limit the sampling range to the climatic bounds of a particular biome. This makes the database biome specific, but is consistent with our selection of biome specialised Plant Functional Types. Third, we use a coarse sampling only, completed using interpolation between samples: given some non-sampled terrain conditions, we derive a distribution by five-fold linear interpolation of the sampling grid, with a set of interpolations for each dimension, using mass transport of distribution histograms [EVC*15].

In the final ecosystem modelling phase, a design map of conditions over the terrain is drawn from $\mathcal{T}$ and used to index the distribution database. Interactive design starts from an ecosystem automatically populated by synthesising the obtained distributions according to the design map (see §4 and Figure 3). A set of semantic brushes of controllable radius can be used to modify the design map within a distance ($d \leq r_0$) of the brush centre. We support feathering on the edges of the brush by gradiating the weight of any modification between an inner and outer brush radius ($r_0 < d \leq r_1$).

The various brushes function as follows:

- Age – Since ecosystem age is one of the dimensions of $\mathcal{T}$, an ecosystem can be made older or younger by shifting along this axis while holding other coordinates constant.
- Variability – Moving conditions towards or away from the local mean of the design map, enables ecosystem variation to be smoothed or exaggerated.
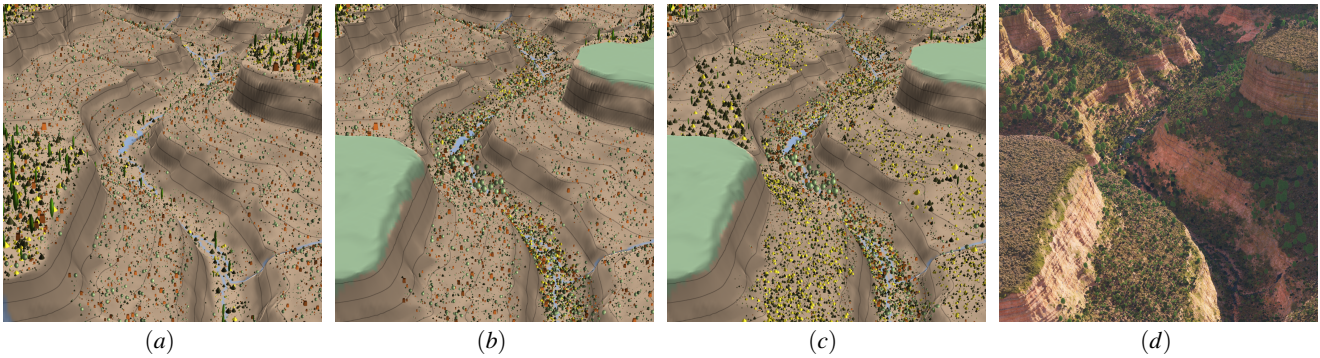- Density – A common summary statistic of distributions is the ex-

Figure 9: *Iterative editing of a Mediterranean canyon landscape. An initial cluster-based plant assignment (a) is modified by the user to introduce grass surrounds and young dense plants along the riverbank (b), followed by painting a varied spread of evergreen shrubs (in yellow and dark green) on the slopes (c), with final Terragen rendering (d).*

pected number of plants. By looking up distributions in the local $\mathcal{T}$-space neighbourhood of a condition from the design map, a $\mathcal{T}$-vector that increases or decreases plant density can be found. This can even be specialised to particular plant types or size categories.

The original abiotic conditions (the source map) can be regarded as a ground truth of sorts. Differences between the design and source maps are necessary for design freedom and can be motivated by ecosystem effects not catered for in our simulations, such as microclimates, local soil types, and fire disturbance. Nevertheless, in the interests of visual consistency it is useful to reflect this divergence back to the user in the form of a suitability map (see Figure 8), which uses colour gradiation based on the normalised L^2 distance between the source and design maps. We also provide a healing brush that shifts overpainted areas of the design map towards the source map.

8. Results and Discussion

In the interests of visual consistency, it is necessary to establish the validity of our underlying simulation engine. In line with previous work [DHL*98, LP*02, BL09, Ch'11] we have checked for expected emergent behaviour, such as self-thinning, where the number of plants is reduced over time as larger specimens crowd out smaller, and succession of a slower-growing but more climatically-suited species over a fast-growing competitor. In figure 10 we show the introduction of a niche-suited invader species to a Mediterranean biome (patterned on Gritti et al. [GSS06]), which results in dominance over other species.

The variety of achievable landscapes is shown in Figures 11-13, all at 5×5 km^2 scale. The Plant Functional Type parameters for these biomes are summarised in Table 2 and the input abiotic maps and clusters appear in Figures 1 and 3. In our interactive editor each PFT is provided a simplified iconic geometric model (see the key of Table 2) and rendered using OpenGL instancing with per instance transformations for size and placement. Export for final rendering is to Terragen.

Figure 11 is a Mediterranean landscape characteristic of the

south of France that shows a variety of plant densities and types, ranging from pastures and scrubland to woodlands and dense forests.
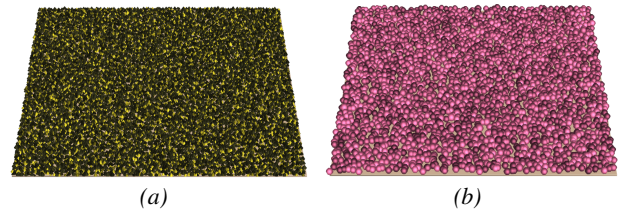

Figure 10: *Simulation validation: (a) a Mediterranean biome, (b) after the introduction of a fast-growing invader species (shown in purple), which overwhelms other plants.*

An important use-case is the generation of terrains in the absence of a database from simulations run on the means of a set of clustered conditions. Figure 12 demonstrates the results of this process, for an Alpine biome with 7 clusters, such as might be found in the Swiss Alps. Of note here is transition from temperate to boreal trees due to changes in mean temperature with increasing altitude.

One shortcoming of our simulations is that they do not cater for disturbance effects. This is particularly evident in Tropical Savannahs, which, without the impact of fires and grazing herds would be relatively dense because of the high rainfall. However, as shown in Figure 13, users can compensate for this with appropriate overpainting of the ecosystem.

The performance of our system was tested on a 3.3 GHz quad-core Intel Haswell, with 7 GB RAM, and an NVIDIA GTX 960 with 4 GB and the results are shown in Table 1. The system performs with reasonable interactive response (< 1s per update) for synthesis with up to $50,000$ plants, which roughly equates to a terrain area of 1×1 km^2, depending on plant density. This does place a limit on brush radius in larger landscapes if interactivity is to be retained. However, it is possible to provide an initial placement estimate (the dart throwing phase) and then run refinements as a background thread (the perturbation phase).

A weakness of our system is the need for manual calibration of plant functional type parameters. Poor calibration tends to result either in homogenization through a lack of niche specialization, or a complete dearth of particular plants. Fortunately, the decision to avoid simulating individual species greatly reduces the search space. Nevertheless, the issue is complicated by the difficulty of mapping botanical data (such as the bioclimatic limits available for Global Vegetation Models) onto our model. This necessitates a cyclic process of adjusting parameters and checking on the emergent properties. Another key parameter is the density of sandbox sampling of terrain conditions. Undersampling leads to particular sub-niches of plant distributions being missed and disparities between the results of our system and a full terrain-spanning simulation, while oversampling unnecessarily increases pre-processing times. This could be improved with a mechanism for scattered sampling of the space of terrain conditions. In general, such parameter tuning is painstaking and could certainly benefit from automation, perhaps through some form of machine learning.

Number of plants	Time (s)	Extent (m)
44,377	0.82	1000
232,921	5.79	2000
543,335	18.49	3000
991,981	49.29	4000
1,568,848	103.60	5000
2,245,103	180.15	6000

Table 1: *Computation cost of synthesis with varying ecosystem size (in number of plants). The peturbation phase of synthesis allows up to* 2 *adjustments per plant.*

9. Conclusions

To summarize: we create a database of plant distributions by sampling the space of abiotic terrain conditions (moisture, sunlight, temperature, slope, and ecosystem age) for a given biome and supplying these samples as parameters to small-scale eco-position simulations. Statistical distributions are then derived from the resulting plant placements via disk-based distance analysis. By synthesising distributions drawn from the database a large ecosystem ($5 \times 5\ km^2$ in extent with up to a million plants) with varying terrain conditions can be populated in under a minute while retaining the visual consistency of the source simulations. Users can locally modify meaningful high-level attributes of the ecosystem, such as age, density, and variability, using an overpainting interface. Modifier brushes are interactive even on larger terrains, requiring 0.6s to update a $500 \times 500 m^2$ subsection of an ecosystem.

There are several possibilities for future work. Most existing simulations, including our own, do no fully address either ecosystem disturbance (the impact of fires, storms, grazing, logging, disease, and the like) or two-way interaction with the terrain (such as litterfall enriching the soil, roots acting as barriers to erosion, and plant canopies intercepting rainfall). We compensate for this to an extent by allowing users to guide the ecosystems. For instance, fire damage can be approximated by applying an age brush, but this fails to account for fire-prompted seeding and germination or varying topkill and plant mortality. Certainly, there is an opportunity

to improve simulation fidelity by borrowing more extensively from plant ecology models [SIK07, SHG*08].

Another challenge lies in parameter selection and validation. It is very difficult to obtain accurate species information even for relatively simple plant models, such as ours. Our strategy has been to tune plant parameters based on expected emergent behaviour, such as ecological specialisation, succession, and self-thinning, but this suffers from being qualitative rather than quantitative. One alternative would be to validate against physical plot samples undertaken by plant ecologists. The difficulty here lies in finding a mapping from the specifics of this data to more generic parameter inputs and plant functional type outputs.

References

[ACV*14] ANDÚJAR C., CHICA A., VICO M. A., MOYA S., BRUNET P.: Inexpensive reconstruction and rendering of realistic roadside landscapes. *Computer Graphics Forum 33*, 6 (2014), 101–117. 3

[AD05] ALSWEIS M., DEUSSEN O.: Modeling and Visualization of symmetric and asymmetric plant competition. In *Eurographics Workshop on Natural Phenomena* (2005), Poulin P., Galin E., (Eds.), The Eurographics Association, pp. 83–88. 2

[AD06] ALSWEIS M., DEUSSEN O.: Wang-tiles for the simulation and visualization of plant competition. In *Computer Graphics International: Advances in Computer Graphics*. Springer, 2006, pp. 1–11. 3

[BE03] BENES B., ESPINOSA E. D.: Modeling virtual ecosystems with the proactive guidance of agents. In *Computer Animation and Social Agents, 2003. 16th International Conference on* (May 2003), pp. 126–131. 2

[BL09] BORNHOFEN S., LATTAUD C.: Competition and evolution in virtual plant communities: a new modeling approach. *Natural Computing 8*, 2 (2009), 349–385. 8

[BSK*15] BRADBURY G. A., SUBR K., KONIARIS C., MITCHELL K., WEYRICH T.: Guided ecological simulation for artistic editing of plant distributions in natural scenes. *Journal of Computer Graphics Techniques 4*, 4 (Nov. 2015), 28–53. 2, 3

[CBC*16] CORDONNIER G., BRAUN J., CANI M.-P., BENES B., GALIN Ã., PEYTAVIE A., GUÃL'RIN Ã.: Large scale terrain generation from tectonic uplift and fluvial erosion. *Computer Graphics Forum 35*, 2 (2016), 165–175. 5

[Ch'09] CH'NG E.: An artificial life-based vegetation modelling approach for biodiversity research. *Nature-Inspired Informatics for Intelligent Applications and Knowledge Discovery: Implications in Business, Science, and Engineering* (2009), 68–118. 2

[Ch'11] CH'NG E.: Realistic placement of plants for virtual environments. *IEEE Computer Graphics and Applications 31*, 4 (July 2011), 66–77. 8

[Ch'13] CH'NG E.: Model resolution in complex systems simulation: Agent preferences, behavior, dynamics and n-tiered networks. *Simulation 89*, 5 (May 2013), 635–639. 2, 3

[DCDS05] DIETRICH A., COLDITZ C., DEUSSEN O., SLUSALLEK P.: Realistic and interactive visualization of high-density plant ecosystems. In *Eurographics Workshop on Natural Phenomena* (2005), Poulin P., Galin E., (Eds.), The Eurographics Association, pp. 73–81. 1, 2
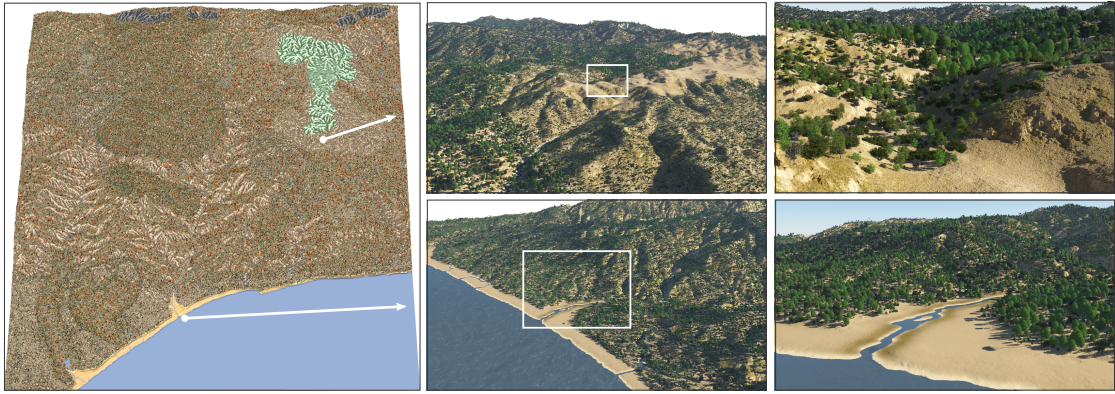
Figure 11: *A Mediterranean biome typical of Southern France, exhibiting a variety of distributions. [scale: 5×5 km^2, input ranges: temperature $4 - 29\,^\circ$C, rainfall $11 - 127$mm, mostly in winter, ~1.8 million plants]*
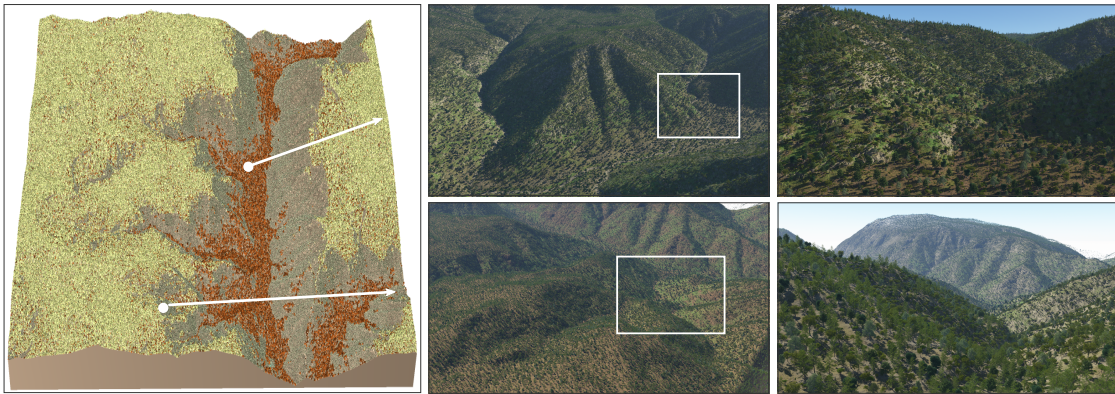

Figure 12: *An Alpine biome based on the Swiss Alps. Plant types are elevation dependent: temperate species in the lower, warmer elevations (brown) and boreal species at higher, colder elevations (light green). [scale: 5×5 km^2, input ranges: temperature $2 - 23\,^\circ$C at base, decreasing to $-6 - 15\,^\circ$C at the higher elevations, rainfall $53 - 161$mm, ~1.5 million plants]*

Figure 13: *A tropical Savannah biome inspired by Central Africa. User intervention is required to reduce the initial high plant density (top) caused by high rainfall and thereby reflect the impact of fire and elephant herds (bottom), which are not explicitly modelled in our simulations. [scale: 5×5 km^2, input ranges: temperature $21 - 26\,^\circ$C, $53 - 161$mm rainfall, ~3.2 million plants reduced to ~1.4 million.]*

	Type	Description	Example Species	Max Height (cm)	Max Age (months)	Moisture	Sunlight	Temperature	Key
Mediterranean	MNE	Mediterranean Needle-leaved Evergreen	Sea Pine	4000	4000	3	9	6	●
	TBS	Temperate Broad-leaved Summergreen	European Beech	3500	3000	5	2	4	▲
	IBS	Shade-intolerant Broad-leaved Summergreen	Silver Birch	2500	1200	5	9	4	■
	TBE	Temperate Broad-leaved Evergreen	Holly Oak	2000	2500	5	4	7	●
	MSEB	Mediterranean Shrub Evergreen Broad-leaved	Lentisk	1000	1000	3	9	8	▲
	MSEN	Mediterranean Shrub Evergreen Needle-leaved	Juniper	800	900	3	9	8	▲
Alpine	TNE	Temperate Needle-leaved Evergreen	Black Pine	4000	4000	2	9	5	●
	TBS	Temperate Broad-leaved Summergreen	European Beech	3500	3000	5	2	3	▲
	TBE	Temperate Broad-leaved Evergreen	Holm Oak	2000	2500	5	2	6	●
	TS	Temperate Shrub	Medlar	800	800	1	1	2	▲
	BNE	Boreal Needle-leaved Evergreen	Scotch Pine	3000	3600	5	9	2	●
	BNS	Boreal Needle-leaved Summergreen	European Larch	4000	2500	3	9	1	■
	BBS	Boreal Broad-leaved Summergreen	Silver Birch	2500	1800	5	2	3	●
	BS	Boreal Shrub	Dogwood	400	1000	5	3	3	▲
Savnnah	PBE	Tropical Broad-leaved Evergreen	Tamarind	1600	2400	7	8	9	●
	PBR	Tropical Broad-leaved Raingreen	Umbrella Acacia	1800	4000	4	5	9	▼
	PBES	Tropical Broad-leaved Evergreen Shrub	African Boxwood	200	1000	6	8	8	▲
	PBRS	Tropical Broad-leaved Raingreen Shrub	Arrow Poison	500	800	4	7	5	●
	AE	Arboreal Evergreen	Tree Aloe	1500	1200	6	6	8	■

Table 2: *A summary of plant functional type parameters for our biome simulations, indicating their general characteristics (form, maximum height, longevity) and relative ranking of response to abiotic factors (on a ten-point scale). The key indicates the shape and colour used for ecosystem visualizations.*

[DCSD02] DEUSSEN O., COLDITZ C., STAMMINGER M., DRETTAKIS G.: Interactive visualization of complex plant ecosystems. In *Proceedings of the Conference on Visualization '02* (Washington, DC, USA, 2002), VIS '02, IEEE Computer Society, pp. 219–226. 2

[DHL*98] DEUSSEN O., HANRAHAN P., LINTERMANN B., MĚCH R., PHARR M., PRUSINKIEWICZ P.: Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1998), SIGGRAPH '98, ACM, pp. 275–286. 2, 3, 8

[Dig13] DIGGLE P. J.: *Statistical analysis of spatial and spatio-temporal point patterns*. CRC Press, 2013. 3, 6

[DL06] DEUSSEN O., LINTERMANN B.: *Digital design of nature: computer generated plants and organics*. Springer Science & Business Media, 2006. 1, 2

[DN04] DECAUDIN P., NEYRET F.: Rendering forest scenes in real-time. In *EGSR04: 15th Eurographics Symposium on Rendering* (Norrköping, Sweden, June 2004), Keller A., Jensen H. W., (Eds.), Mark Ollila, Eurographics Association, pp. 93–102. 2

[EVC*15] EMILIEN A., VIMONT U., CANI M.-P., POULIN P., BENES B.: Worldbrush: Interactive example-based synthesis of procedural virtual worlds. *ACM Trans. Graph. 34*, 4 (July 2015), 106:1–106:11. 2, 3, 6, 7

[FPR*96] FOLEY J. A., PRENTICE I. C., RAMANKUTTY N., LEVIS S., POLLARD D., SITCH S., HAXELTINE A.: An integrated biosphere model of land surface processes, terrestrial carbon balance, and vegetation dynamics. *Global Biogeochemical Cycles 10*, 4 (1996), 603–628. 2, 4

[GSS06] GRITTI E., SMITH B., SYKES M.: Vulnerability of mediterranean basin ecosystems to climate change and invasion by exotic plant species. *Journal of Biogeography 33*, 1 (2006), 145–157. 8

[Jai10] JAIN A. K.: Data clustering: 50 years beyond k-means. *Pattern Recognition Letters 31*, 8 (2010), 651 – 666. 5

[LIB*09] LAW R., ILLIAN J., BURSLEM D. F. R. P., GRATZER G.,

GUNATILLEKE C. V. S., GUNATILLEKE I. A. U. N.: Ecological information from spatial patterns of plants: insights from point process theory. *Journal of Ecology 97*, 4 (2009), 616–628. 3

[LP*02] LANE B., PRUSINKIEWICZ P., ET AL.: Generating spatial distributions for multilevel models of plant communities. In *Graphics Interface '02* (May 2002), Canadian Human-Computer Communications Society, pp. 69–80. 2, 3, 8

[MP96] MĚCH R., PRUSINKIEWICZ P.: Visual models of plants interacting with their environment. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1996), SIGGRAPH '96, ACM, pp. 397–410. 2

[NHP07] NYLAND L., HARRIS M., PRINS J.: Fast n-body simulation with cuda. *GPU gems 3*, 1 (2007), 677–696. 2

[OG12] ÖZTIRELI A. C., GROSS M.: Analysis and synthesis of point distributions based on pair correlation. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 170:1–170:10. 3, 6

[PL12] PRUSINKIEWICZ P., LINDENMAYER A.: *The algorithmic beauty of plants*. Springer Science & Business Media, 2012. 2

[PMKL01] PRUSINKIEWICZ P., MÜNDERMANN L., KARWOWSKI R., LANE B.: The use of positional information in the modeling of plants. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 289–300. 2

[SHG*08] SITCH S., HUNTINGFORD C., GEDNEY N., LEVY P. E., LOMAS M., PIAO S. L., BETTS R., CIAIS P., COX P., FRIEDLINGSTEIN P., JONES C. D., PRENTICE I. C., WOODWARD F. I.: Evaluation of the terrestrial carbon cycle, future plant geography and climate-carbon cycle feedbacks using five dynamic global vegetation models (dgvms). *Global Change Biology 14*, 9 (2008), 2015–2039. 2, 5, 9

[SIK07] SATO H., ITOH A., KOHYAMA T.: Seib–dgvm: A new dynamic global vegetation model using a spatially explicit individual-based approach. *Ecological Modelling 200*, 3–4 (2007), 279 – 307. 2, 4, 5, 9

[TH10] TUCKER G. E., HANCOCK G. R.: Modelling landscape evolution. *Earth Surface Processes and Landforms 35*, 1 (2010), 28–50. 5