High-Level Techniques for Landscape Creation

Leandro Cruz, Luiz Velho
VISGRAF Lab
Instituto Nacional de Matemática Pura e Aplicada
Rio de Janeiro, Brazil
www.impa.br/∼{lcruz, lvelho}

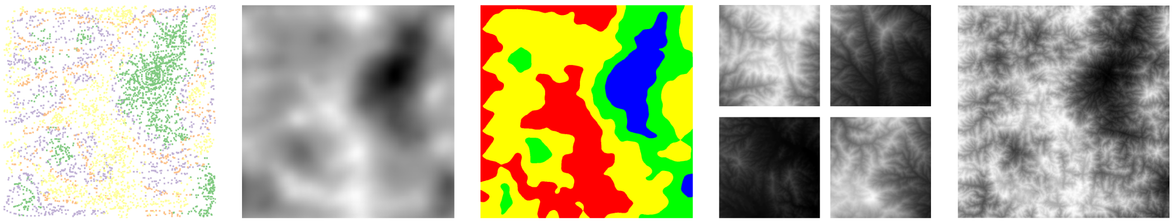

Fig. 1. The pipeline of terrain creation: from a set of seeds (a), the guide (b) was created containing the coarse shape of the terrain; next, the categorization map (c) was defined, associating each part of the target with a class of exemplars; after, the exemplars (d) were chosen according to the control structures; and finally the terrain model (e) was generated.

***Abstract*—A complete solution for creating an entire planet by combining different types of elements from an intuitive specification is still an open problem. High-level techniques are at the core of the solution for this issue. In this context, our work aims to contribute for the development of high-level techniques for landscape creation by the proposal of two approaches. The first deals with the manipulation of the position of landscape elements (according to some criteria), and the second is related to the generation of terrains from two new data-driven methods.**

In particular, we introduced a flexible method for resizing a landscape specification keeping the overall appearance. It is based on the insertion and removal of objects followed by scene adjustment (enlargement or shrinking).

Furthermore, we introduced two data-driven techniques for terrain synthesis. They are developed over the same intuitive control approach, and they take advantage of the geometric nature of the data for improving the synthesis. The first technique is a patch-based algorithm, with new optimization structures for patch choice, and a new patch insertion approach (both based on the geometric nature of the data). The second is a pixel-based method, based on a graph of exemplars in multiresolution.

***Keywords*-Landscape Creation; Terrain Synthesis.** [1]

I. Introduction

In the last three decades, the modeling of virtual landscapes presented significant advances. Nowadays, we can see extremely realistic models in movies and games. Nevertheless, the creation of these models is still very laborious.

The advances in landscape modeling are due to the advances of computer graphics methods (general methods, and those for modeling and visualization); to the use of better hardware, which enable the implementation of more complex algorithms; and the availability of large data sets, from which we can obtain information to make the modeling more efficient.

In this context, our work aims to contribute for the creation of high-level methods for landscape generation. We introduced two techniques: one for handling with vector specification of landscapes, and another for creating terrain based in real data.

Contributions: We performed an analysis of the state of art about techniques for landscape creation, emphasizing the relation between the creation of images, and the generation and manipulation of elements present in a virtual environment; converging to the identification of the main challenges for the development of high level techniques for this purpose. Another contribuion is a technique for resizing a vector-based specification of a landscape satisfying some predefined rules according to the initial model [1]. Furthermore, we introduced two exemplar-based techniques for terrain generation taking advantage of the geometric nature of the data, and improving the control synthesis process [2], [3].

A. Related work

We introduced a method for resizing of a landscape specification keeping the overall appearance [1]. Our approach is based on shifts of the objects according to some inserted or removed paths in the scene. It is similar to the Seam Carving method for image resizing [4]. The difference between our approach and Seam Carving is that we are dealing with a straightforward model, and thus, our method can use a simpler metric, but also producing good results.

We also introduced two data-driven techniques for terrain synthesis inspired on texture synthesis methods. The first method is a patch-based technique [3]. Methods of this class [5], [6] create a model, by covering the target using patches taken from the exemplar, placed with some predefined overlapping over the already synthesized parts of model. Our synthesis approach is similar, but include a more sophisticated control, and improvements on patch choice and insertion based on the geometric nature of the data. The second method is a pixel-based technique [2]. We extended a multiscale approach [7], by improving the synthesis control, and creating automatically the graph of exemplars, from the the control structures and a large database of exemplars.

[1]This paper is based on the PhD Thesis of Leandro Cruz

II. Concepts

A. Terrain and Textures

The main structure of the terrain topography can be represented by a function $f : \mathbb{R}^2 \longrightarrow \mathbb{R}$, called Digital Elevation Model (DEM). There is a seemingly trivial association between a grayscale image and this representation. Thus, it suggests we can use techniques related to images for terrain purpose. Indeed, besides of this representation, terrains have other similarities with textures.

In Computer Graphics, the term *texture* are related to the group of images that represents the material of some object (its appearance). The study about textures comprises problems like synthesis, mapping, reparameterization, etc. We are only concerned about texture synthesis. There are several kinds of textures, with different features. A strategy to distinguish textures is to group them by the type of structures and their organization. The structures can be more or less stationary and the organization can be more or less regular.

A terrain model has similar characteristics to a structured-organized texture. The structures are related to the shape of the landforms and the organization is given by geomorphological phenomena. There is not a satisfactory solution for terrain synthesis that works with both aspects in accordance with natural phenomena. Data-driven approach is good for the creation of local structures, but in general, it does not have an intrinsic control of the organization of the structures.

There are some strategies for terrain synthesis: procedural, by simulation, by sketches, and data-driven [8]. Nowadays, the simulation approach is well developed, i.e., there are many methods able to create terrain some features according to some geomorphological properties. However, most of natural phenomena are not well understood, are extremely complex, or they need of unavailable data. Thus, the created models have some lack of realism. On the other hand, because data-driven methods combine pieces of a real exemplar, they can recreate some terrain features, even without an explicit control. To the best of our knowledge, there are few data-driven terrain synthesis methods, and they have not yet explored all the possibilities. Nevertheless, we believe this type of methods will be better developed in the next years.

B. Landscape and Vector Graphics

As well as terrains are related to textures, there is also a relation between a landscape representation and a vector graphics. A landscape is composed by a set of elements: terrain, trees, rivers, houses, roads, etc. It can be represented by a map containing the position of each element and other attributes (for example, the 3D model, a scale, etc). And, the specification of this map can be similar to a vector graphics.

A Vector Graphics is composed by geometric primitives (like points, curves and polygons), and their respective visual attributes (like color, thickness of the contour, gradient, etc.), and possibly a digital image (or pieces of it). The landscape representation also have the definition of the geometry and attributes of each object into the scene.

This vector-based representation for landscapes is resourceful for high-level approaches. It allows to keep the focus on the main features of the scene, instead to determine geometric details of the object. Smelik et al. [9] proposed a high-level method which combines semantic-based modeling and procedural approaches for populate a large virtual world, described in vector-based layers. As well as, we introduced a technique for landscape specification resizing [1].

C. High Level Techniques

We can split the geometric modeling in two approaches. The first is for modeling of the objects that need of a precise specification (for instance, the CAD-CAM models). And, the second approach works with those objects which the goal is to transmit an idea, i.e., their focus is in the perception of global features. We call the techniques of the first approach by Low-level, because they deal directly with a raw representation of the model (i.e., with a precise information about each part). On the other hand, we call the techniques of the second approach by High-Level. These techniques are mainly based in global representation, by emphasizing the main features of the model (those primarily perceived). In general, they wrap the low-level representation, and only provide for users a more intuitive interface. They are focused on the creation of the macro features, and delegate for specific procedures (of low-level) the creation of the details.

An example of a High-level method for creating a landscape is that whereof we define the topography only by describing where there will be mountains, where it will be smooth, and what is the percentage of water and land. On the other hand, a low-level approach would create a landscape by defining parameters of procedural methods, or by changing of the position of control points.

One of the main focus for high level techniques is to achieve ways for a more intuitive specification. Many methods get inspiring in analogies with activities already practiced outside of the modeling environment (for instance: drawing and painting). In fact, these analogies allow the user to understand easily the behavior of the method, and thus, to obtain the desired result, by applying less effort. From this specification, it is coded many operations whereof the semantic is based on a more abstract representation.

Nowadays, some high-level methods have been achieved good results for very specific purposes. Nevertheless, the creation of high-level techniques for general purpose (even for some specific topics) is still an issue in geometric modeling.

High-level techniques are based in smart approaches for creating abstractions of representations and operations. The intuitive control is one of the most desired characteristic for a high-level technique. A widely used intuitive tool is the *sketch*. Other high-level tool is the *semantic rule*, that is, an operation able to work with low-level representations in a smart way. Furthermore, there are increasingly available *data sets* about landscape elements. We can use them for adding to the synthesis process extra information about structures of the scene that are not explicitly specified.

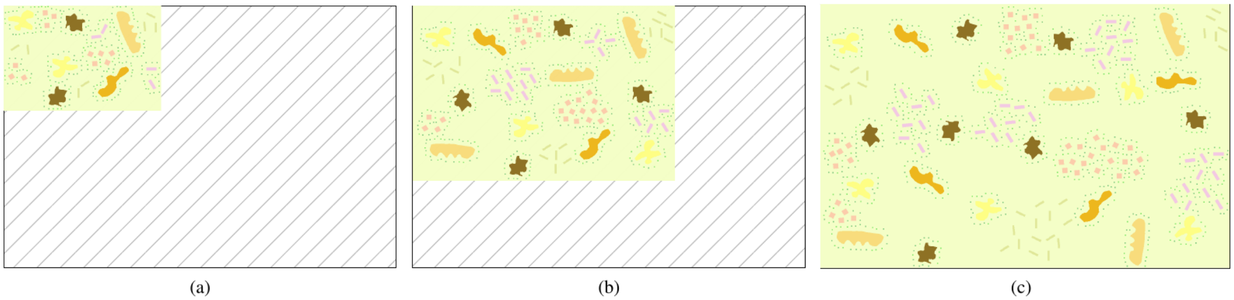

Fig. 2. The landscape map (b) was shrunk (a) and enlarged (c).

III. Vectorial Specification

As introduced in Section II-B, a landscape model is composed by many types of objects. One type of element may be copied in different positions. For instance, some kind of a tree may be placed at many positions of the scene. Because of this, we will refer to *elements* as the set of *objects* (each one placed at different coordinates). Thus, it is possible to specify this scene by a vector model containing the position, and possibly other attributes, of each object.

We presented a technique for resizing a given landscape specification performed on a vector-based model of the scene by changing the dimensions of the model preserving the overall appearance [1]. The motivation for this method is: *Initially, N characters explore a virtual environment. Afterwards, for example, only half of them come back for exploring this scenario. In the second case, whether we keep the initial area, the characters will be too far from each other. Thus, it is interesting to reduce the landscape size in order to return to the initial proportion. Nevertheless, we do not want to change the overall appearance. Hence, the goal is to obtain a smaller scenario by maintaining the same experience.* In general, a simple scaling of the model is not good enough, because it can produce an overlapping of objects. As well, a cropping can destroy global features. For this purpose, a content-aware resizing obtains much better results.

The Seam Carving approach [4] was proposed for image resizing. Our method is inspired in this approach but, because our model is piecewise constant, we obtained good results using metrics simpler than those used on previous methods.

The main contributions of our method are: (1) A flexible method for shrinking and enlargement of a vector-based landscape specification model, keeping the overall appearance; (2) General-purpose approaches for object placement; and (3) Strategies for keeping some properties of object distribution.

The core of our method is similar to seam carving. Its main functionality is to shrink or to enlarge the model by moving the objects according to the removal or insertion of paths. Nevertheless, besides of this displacement, sometimes we have to insert or to remove objects according to some criteria. This method is flexible because we can replace the criteria by any other technique for spreading objects into a landscape.

In general, a 3D landscape is created from a pre-defined vector model, created on the fly using procedural techniques, or by a combination of both approaches. Our method fits in the third category. Our input and output are vector models, and we have some procedural approaches in the resizing pipeline. And from the specification, we create a 2D map containing a set of polygons, each one relative to an object into the scene.

Even we are dealing with a vector model, we define all operations in the pixel space (an image). Most of them are performed in the exterior area of the specification (i.e., in the points that do not belong to any object). Some of these operations are: path creation, computation of the distance field, element and object choice, among others.

Both enlargement and the shrinking are performed similarly. First, we choose the element will be removed or inserted. After, in the enlargement case, we choose a position for inserting a new object associated to the chosen element. Sometimes, it is necessary to create a space for inserting this object. In this case, we create a path going from one side to the opposite side in the exterior area, passing through the chosen position. Analogously, in the shrinking case, we choose an object of the chosen element and remove it. When the removal creates a hole, we define a similar path passing through this region. Finally, for both procedures, we adjust the model performing a displacement of all objects after the path according to the operation.

All steps of this algorithm are based in a cost function defined using a distance field in the exterior area. This function privileges points far of all objects (i.e., points with large values in the distance field). Furthermore, the choice of element and objects are based in criteria that depends on specific purpose related to the type of scene, or application. In our experiments, we used general purpose criteria, like uniform object spreading, clustering, among others. The flexibility of this method is due to the possibility of changing this criteria according to the need.

We presented several results obtained using our approach, with small variations of our methods, and how to evaluate the quality of these results [1]. Figure 2 shows an example where the original model was shrunk and enlarged. Analogously, Figure 3 shows a landscape and its shrunk version.

By combining our approach with good methods to insert and remove objects, we obtained a powerful tool for the creation of huge landscapes. Moreover, by recursively creating structured elements, the proposed approach can be applied in multiresolution. Thus, it could be used for the creation and editing of virtual worlds from a high level perspective: a very difficult problem in this area.

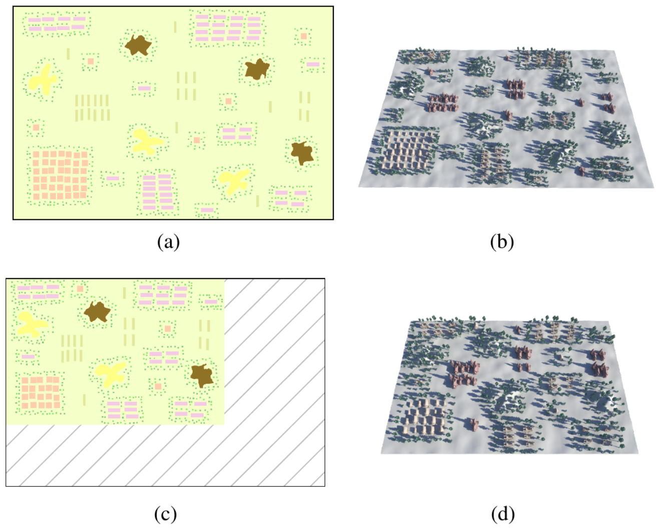

Fig. 3. An example of shrinking of the landscape. In the right column it is shown a 3D version of the scene.

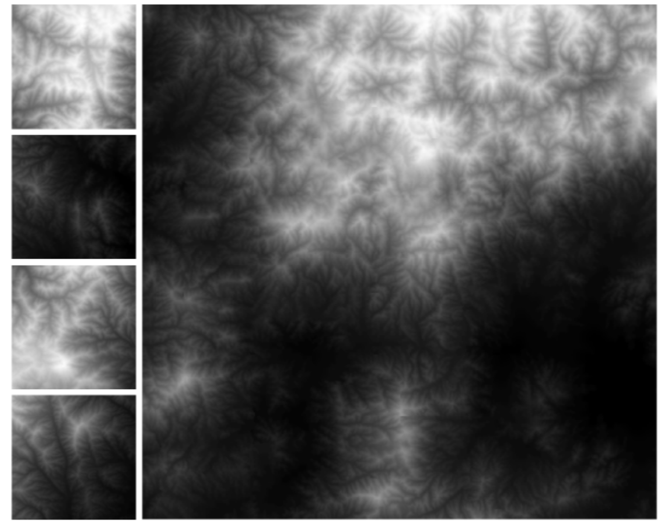

Fig. 4. An example of terrain model generated using our patch-based approach, through the exemplars exhibited in the left column.

IV. Data-driven Terrain Synthesis

In general, terrains are locally homogeneous, but they contain some global structures that are matter of priority for being synthesized (globally heterogeneous). Data-driven approaches have been used for terrain synthesis because their ability of reproducing structures from the exemplar, even we do not have an explicit representation for them. Our proposed methods are able to generate a terrain containing some macro-structures (like large landforms, and the definition of the coarse height average) specified by our control approach, the meso-structures (like ridges, valleys, and big erosion phenomena) by the Markovian approach, and the micro-structures (details) by coping pixels from exemplar.

Most of texture synthesis methods are only based on one small exemplar. For terrain purpose, one small exemplar does not contain enough information for generating the macro and meso structures. Because of this, the use of classical texture synthesis methods produce terrain with spurious artifacts, like strange discontinuities of height and breaking of structures such as ridges and valleys. For solving this problem, we increased the amount of input data (using more than one large exemplars) and, above all, we improved some synthesis steps taking advantage of the geometric nature of the data for enhancing the control.

The novelties of our methods are related to the control of synthesis by the use of some structures, to the choice of exemplars, and to the use of some new criteria for choosing a patch or a pixel during the synthesis. More specifically, the first main contribution is an approach for control the synthesis based on two structures: guide and categorization map. We presented how to create and use the guide (a DEM containing the coarse structures of the target). Furthermore, the synthesis is based on a categorization of exemplars in accordance with some pre-defined features (e.g. similar height, ridges, valleys, etc). We introduced how to create an exemplar categorization and the map for associating regions of the target with these classes. The second contribution is a criterion for validation of exemplars according to the guide, and a rule for choosing a minimum set of exemplars, into a large data set, able to cover the target conforming to the control structures. Another contribution is a controlled patch-based algorithm including a new optimization structure for patch choice (for accelerating the processing, and improve the feature matching), and a new patch insertion approach (reducing the overlapping error and removing seam discontinuities), both based on the geometric nature of the data. Finally, the last contribution is a controlled pixel-based algorithm and a strategy for the creation of the graph of exemplars used in a multiscale synthesis.

A. Synthesis Control

The guide is the most important structure for the control of macro features. It is a DEM with the same dimensions of the desired model containing the coarse shape of the terrain. It also contains an implicit categorization of the exemplars, since it has a wide range of height, and so each exemplars are only used in parts of the model. Also, it guarantees the continuity constraints, i.e., the terrain will be continuous (or discontinuous) where the guide is continuous (or discontinuous). We presented some strategies to create the guide using sketches, brushes and seeds of features.

The map of categories can be based on a subdivision of the guide by clustering of heights. The clustering creates a set of closed regions, and each one is associated to the respective class. This map is created by setting which regions will be associated with each category of exemplars.

Another important strategy to control some features of the model is the choice of an adequate set of exemplars. If the amount of the data is too small there is not enough information to perform an adequate synthesis. However, a huge amount of redundant data increases significantly the processing cost.

We take the exemplars from a large database of candidates. The choice has to be in accordance to the data structures. When it is made manually, it is necessary to validate the

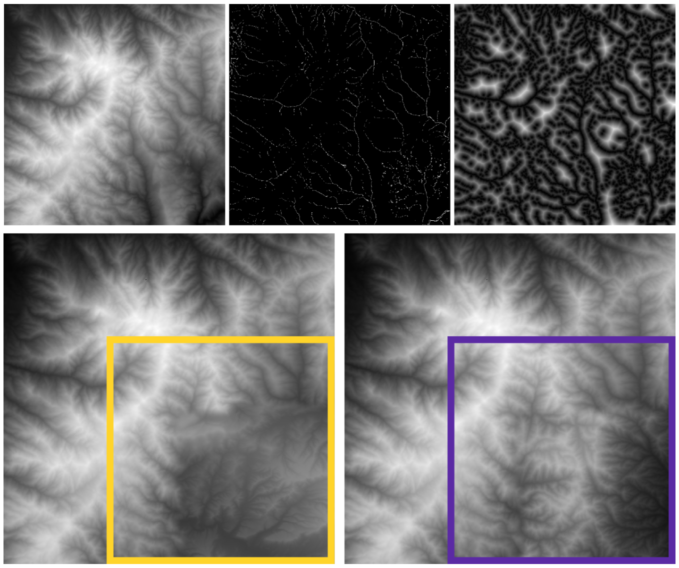

Fig. 5. The first row shows the creation of the Valley Descriptor: (from left to right) the terrain, the valley points (white), and the distance field of the valley points. In the second row we show two examples of patch insertion: without using the valley descriptor (yellow), and using this descriptor (purple).

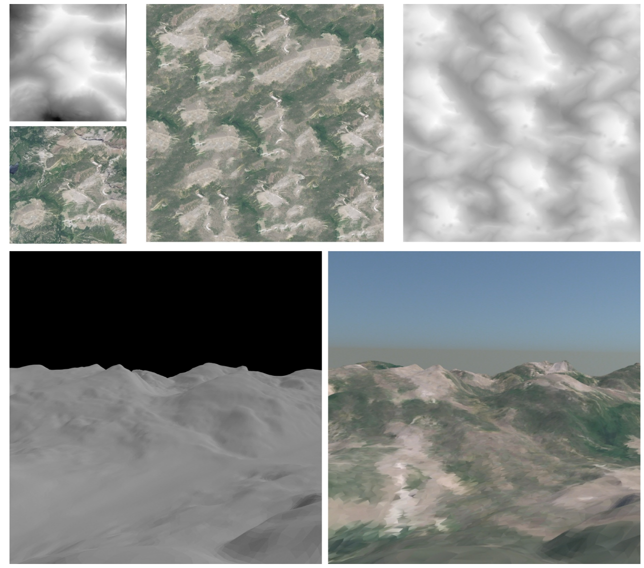

Fig. 6. It is an example of terrain generated using our pixel-based approach. The first row shows the exemplar (DEM and texture), the generated texture and the DEM. The second row shows a 3D view of the geometry and a combination of geometry and texture.

choice. Let P the unity of synthesis (P can be a pixel or a patch), the input validation consists in guarantee that each part of the final model can be generated by at least a minimum amount of P, taken from the input data. The validation is performed using some criteria related to the height variation and the density of meso structures (ridges and valleys).

The automatic choice is performed by choosing the minimum amount of exemplar able to cover the data structures. It is created using a greedy algorithm based on a graph model generated with the exemplars in the database. Each vertex of this graph is one exemplar of the database, and two vertices are connected by an edge if they are compatible (i.e. it exists a range of height R such that there are minimum amount of P in both exemplars whose height belongs to R). Furthermore, for each vertex, we define a value related to how many P in the exemplar are compatible with the control structures (i.e. satisfies the pre-defined validation criteria).

B. Patch-based approach

We will present a patch-based technique for terrain synthesis [3]. This approach is inspired in classical patch-based texture synthesis techniques, above all, the Image Quilting [5] and the Super-Resolution Images [6]. However, when the Image Quilting method is directly applied to generate a terrain, the produced result is not as good as those generated for textures. The overlapped patches are not well fitted and the global topographic features (such as mountains and valleys) have a non natural distribution. It is because the terrain exemplar is a non-stationary model, and so, it does not have information enough to create a new adequate model. Thus, our patch-based approach aims to add some constraints for dealing with the specificities of terrains to obtain better results.

The synthesis of each patch begins by the definition of candidates to be inserted, that is, all possible patches compatible to the respective region of the target (relative to some rule). The initial candidates set has a large number of possible patches, chosen with high level criteria (able to choose the candidates by a low cost comparison) aiming to match the coarse features of the patches and the guide.

After applying these high level criteria for defining the candidates, we will choose the patches with a good matching of features into the overlapping region, by the use of low level criteria, i.e., rules for performing a more detailed comparison (and thus, a more expensive operation). Furthermore, it is necessary to define the cut to split the pre-synthesized part from the chosen patch.

Even though we are using a large amount of input data, in many cases it is not possible to achieve a good overlapping matching that enable an optimal cut. But, because we know that our data is a DEM, we can perform a small vertical translation on the patches, to improve the matching, before the insertion. We achieve it by solving a convex unconstrained optimization problem. Because the candidates patches were chosen to be close to the guide, the optimal translation that improves the overlapping matching is small, and so this operation will not break the compatibility of the patch with the macro features.

Another strategy to improve the seam of patches is to guarantee an adequate alignment of mesostructures of adjacent patches. The favoring of this alignment can produce a small increasing of the cut error, but the visual improvement compensates the possibility of the presence of artifacts related to micro structures. We guarantee this alignment by the use of the Valley descriptor. It is a distance field created from the points of the terrain detected as a valley (a local minimum in some direction).

C. Pixel-based approach

The first idea for texture synthesis using a Markovian approach was based in pixels, instead of patches. Many works present techniques for creating a texture by matching the neighborhood of a given pixel in target texture with all neighborhoods in the exemplar. This approach is very efficient for generating stationary or quasi-stationary models. However, using additional control models it is possible to generate some structured and organized textures.

One of the most advanced pixel-based technique was proposed by Han et al. [7]. They create a texture from a graph of exemplars with one or more samples, and a relation about scale transition. This approach improves the ability of creating structured and organized textures, with heterogeneous patterns. The organization is achieved by coarse scale samples, and the structures are created by finner scales in exemplars.

In our work, we adapted this method to the terrain context, adding a control by our control structures, and present how to create the graph of exemplars. Our method also take advantage of the geometric nature of the data (DEM), because of this, it is not trivial to extend it for colored images. Nevertheless, we can create simultaneously the DEM and the texture of a model applying different metrics for different channels of data.

In our method, the control of macro structures is performed by analyzing the guide, instead of a random process over coarse scale of the graph. The meso structures are generated by matching of pixel neighborhood, and the details is given by copying the exemplar pixels.

V. Conclusions and Future Work

The creation of virtual landscapes has been studied for almost four decades. However, there is not a definitive solution that creates a complex landscape, with many different features, according to nature, and controlled in a reasonably intuitive way. The creation of an entire planet involves many research topics. We are focused on terrain synthesis and on the management of objects into the landscape. Even though our approaches have not been tested on the generation of an entire and complex planet from an intuitive specification, we are contributing to the improvement of the area in direction of achieving these goals.

Our main contribution is, from an analysis of the state of art, pointing to the expected future of the area, presenting its trends and challenges, to introduce a method for landscape specification management [1] and two data-driven terrain synthesis [2], [3].

The main application of our resizing method is to change the dimensions of a vector model keeping the overall appearance. By combining our approach with good methods to insert and to remove objects, we obtained a powerful tool for creation of huge landscapes. Moreover, by recursively creating structured elements, the proposed approach can be applied in multi-resolution, and can be used for the creation and editing of virtual worlds from a high level perspective.

Moreover, we have introduced two data-driven methods for terrain synthesis. Both techniques are inspired on texture synthesis approaches, providing new structures for synthesis control, and taking advantage of the geometric nature of the data for improving the quality of results.

Despite of the variety and quality of methods for creating specific types of elements, few of them consider simultaneously more than one type during the synthesis. We believe the best way for creating a huge landscape, or an entire planet, is by combining of some (or many) techniques related to different kind of objects, in different scales. To the best of our knowledge, the method proposed by Smelik et al. [9] is the most advanced technique in this direction. However, it is not a definitive approach. All introduced combination of elements are limited and hardly specific (they are able to reproduce few phenomena). We also believe that the best approach for this problem is to create a multi-level representation, that would combine elements in a top-down way. In this case, the features of landscape have to be described in several levels (or layers).

Finally, the main goal of this thesis is to contribute for this large landscape creation problem. We proposed some high level techniques for this purpose. Our method for specification resizing contributes to this goal because from a set of provided rules for object insertion, and an initial specification, we can create a larger model with the desired features. Furthermore, our data-driven methods are focused on improvement of the synthesis control (in a high level way). Our control structures provide an intuitive control of coarse features, and we introduced some methods for choosing adequately the exemplars from a large data set. Despite of the advances for terrain synthesis, there is a lack of high level methods for complex models creation, and methods that combines different kinds of nature elements. The main future work for this research is to pursuit these goals. We intend to route our research for creation of entire planets, and identify, in a more precise way, what are the main challenges of this topic.

Acknowledgment

This research has been developed at Visgraf Lab - IMPA and at CNRS-LIRIS. The first author was supported by CAPES, for the development of part of this research at CNRS-LIRIS, and by CNPq for the rest of his PhD at VISGRAF.

References

[1] L. Cruz, L. Velho, D. Lucio, E. Galin, A. Peytavie, and E. Guerin, “Landscape specification resizing.” CLEI, 2014.

[2] L. Cruz, F. Ganacim, L. Velho, L. H. de Figueiredo, and D. Lucio, “Exemplar-based terrain synthesis.” WIP - SIBGRAPI, 2013.

[3] L. Cruz, L. Velho, E. Galin, A. Peytavie, and E. Guerin, “Patch-based terrain synthesis.” GRAPP, 2015.

[4] S. Avidan and A. Shamir, “Seam carving for content-aware image resizing.” ACM SIGGRAPH, 2007.

[5] A. A. Efros and W. T. Freeman, “Image quilting for texture synthesis and transfer.” ACM SIGGRAPH, 2001.

[6] W. T. Freeman, T. R. Jones, and E. C. Pasztor, “Example-based super-resolution.” Computer Graphics and Applications, 2002.

[7] C. Han, E. Risser, R. Ramamoorthi, and E. Grinspun, “Multiscale texture synthesis.” ACM SIGGRAPH, 2008.

[8] R. Smelik, T. Tutenel, R. Bidarra, and B. Benes, “A survey on procedural modeling for virtual worlds.” Computer Graphics Forum, 2014.

[9] R. M. Smelik, T. Tutenel, K. J. de Kraker, and R. Bidarra, “A declarative approach to procedural modeling of virtual worlds.” Computers and Graphics, 2011.