Large Scale Foliage Animation for *The Ruins*

Fred Chapman Malcolm Humphreys Carsten Kolve Christoph Sprenger
Rising Sun Pictures

Figure 1: Hundreds of vines attacking the main protagonists in the film *The Ruins*. Diagnostic pass used to verify correct collision behaviour and final shot.

Carnivorous vines with an appetite for teenagers are the main digital characters in Dreamworks' *The Ruins*. Animating thousands of leaves and flowers attached to these vines posed a unique challenge – not only did they have to sway, collide and react to external forces in a believable way but at times they were also required to perform in very “un-plantlike” ways. While background leaves were realised using a dynamic hair approach, leaves in the mid and foreground (of which there could be up to 850) needed to be more controllable. In this sketch we present the animation pipeline developed for this type of foliage: a modular control rig system that allowed layering of skeletal deformation derived from procedural- and key-framing techniques. We also show how the results of soft body simulation were incorporated into this work flow.

1. Skeletal Animation Layers

All leaves attached to vines were animated using individual skeleton hierarchies. On the one hand animators demanded total control over the final result but on the other hand the sheer number of leaves prohibited a fully manual approach. To solve this conflict the system *animLayers* was devised. It enabled us to layer animations from different sources. The basis of the system were *rig modules*:

Rig modules were realised as Autodesk Maya node networks that would allow the user to generate animation using different modules: FK controls for hand animating the performance, custom angular spring system for joints used to generate swaying motion derived from underlying vine movement, procedural joint channel animation utilising modulated Improved Perlin Noise [Perlin 2002] and a collision resolution system

Each of these modules allowed blending with input joint animation. This enabled artists to create custom stacks of rigs for each vine appendage based on the current shot needs. Such a *rig-stack* could always be collapsed by caching the resulting performance to disk as an animation cache. The complex stack of rigs could then be replaced by a much lighter single cache-in rig.

Animation data was treated as a series of transformation matrix arrays describing the joint motion and be transferred from one rig module to the next using a custom Maya data type. To ensure that the transformations could be applied even if joint orientation would differ from rig to rig, they were always stored relative to a reference joint chain also used for binding the geometry. Specific interface nodes transformed the animation data in and out of the current rig's joint space.

Custom user interfaces were developed to manage the multitude of rigs and animation caches used in a scene. With hundreds of leaves of various types being visible in a single shot, managing this list became one main objective for the Shot TD.

2. Mesh Driven Bones

Resolving the complex leaf-leaf as well as leaf-prop collisions within the skeletal animation pipeline described above was initially planned to be done manually using an FK rig where needed – a plan that turned out not to be practical when the amount of visible intersections became apparent. Instead a collision rig module based on *Maya's Nucleus* dynamics solver was developed.

We mapped the vertex deformation produced by a soft body simulation onto a joint hierarchy: A mesh was driven by the incoming animation data and served as the target for a second mesh of equal topology deformed by the dynamics simulation. This collision mesh in turn would then be used by the custom node *meshDrivenBones* to produce joint channel animation for the output of the rig module.

MeshDrivenBones drove the complete joint hierarchy using a combination of alignment vectors derived from the collision mesh. Starting at the root and working its way recursively through the skeleton, each joint matrix was constructed using a forward vector pointing from the current joint's base to the mean position of specific target vertices and an up vector derived from the mean vector of specific face normals.

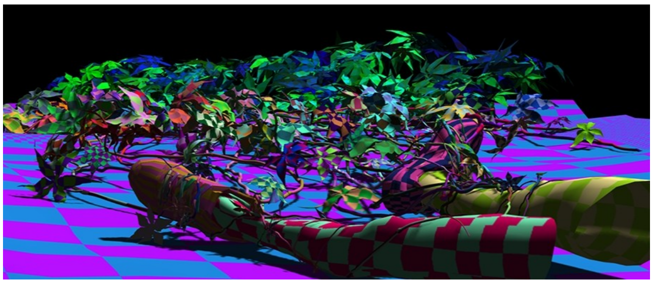

Figure 2: Complex leaf collisions resolved using the *meshDrivenBones* system.

3. Discussion

Using a modular approach to rigging simplified the way animation was used throughout the pipeline and gave us the flexibility to easily adjust the results at every stage of the process. On demand creation and collapsing of control rig stacks allowed us to work with huge numbers of individual vine appendages.

References

PERLIN, K. 2002. Improving Noise. In Computer Graphics Vol. 35 No. 3.