I,Robot: Character Pipeline, Tools, and Methods

Serge Sretschinsky Steve Preeg Jonathan Gerber Paul George
Erick Miller
Digital Domain

The journey which took place at Digital Domain called I, Robot all began with a very simple realization: This is the largest character based feature the studio has ever realized. This realization surfaced many concerns about our job structure and our toolset for such a large-scale character film. These new concerns raised ideas, and from these ideas were born new beginnings: For the first time in almost 10 years, a radically new job system would be created. For the first time ever, a full asset management system, an entirely new character pipeline, many new character animation and deformation tools, and new animation and rendering pipelines would begin development. With less than one year to implement and roll out these systems, (and still complete the film on time) our adventurous tale begins.

Striving For A Neoteric System

From concept to completion, the new job system and character pipeline strove for mutative vicissitude; it was therefore based entirely on relatively referenced data assets, and relied wholly on the context of the environment that the software ran within to derive it's data. All character rig creation was procedurally automated with executable build scripts, facilitating the update or clean re-creation of a character rig at any time on demand. Characters were checked-in as assets to a relative location containing multiple levels of detail where all animators, lighters and dynamics teams used file-referencing tools to load them into their scenes with relative paths. Animation was published from an animator's file, and the high-resolution character with lights and animation were loaded back into the lighter's file as separately referenced connecting file components. All data (models, character rigs, animation, and light rigs) were checked in from the user's work location to a shared parallel asset tree, which contained version-less data assets symbolically linking to the latest versions.

Novel Character Tools: Overview

Not only was a completely new job system architecture and check-in/check-out development underway, but also a vast number of completely new character based tools were being developed from scratch at the same time. Many tools were pipeline based, such as tools for setting up the environment, baking and publishing animation for lighting, an XML-based file format for shot content definitions, and GUIs for building final lighting scenes from the XML files.

Many more tools were new or unique character animation based projects, such as advanced character animation curve warping tools. One such tool allowed for re-mapping any animated motion into the parameter space of an implicit NURBS surface. The NURBS surface could then be used as an interactive 3d animation f-curve deformer, acting on full (or partial) transformation space, including rotations. Another animation curve warping tool allowed for interactive key frame warping of dense curves in the curve editor. A unique derivative of Gaussian for filtering animation curves was implemented to remove motion pops, but keep valid spiked motion. Motion capture matching tools were written to automate the animation process, allowing the animators to attach motion capture animation as key frames applied to the abstract character controls on the actual character setup rigs.

Last but not least, many new geometric deformation nodes were developed. Pose based deformation allowed further control of facial and character deformations, so they could be modeled and driven by any definable character pose. Geometric shape morphing using radial basis functions for smoothly interpolated n-dimensional morphing within facial - space was the primary nodal component of the facial deformation system for the hero character of our film, Sonny. A real time geometric wrap deformer node using linear transformations of vectors (based on normals and texture space) was developed to allow any geometry's deformation to be driven by any other geometry with no hit in compute performance. A stable semi-dynamic hampered spring algorithm for an "animator friendly" muscle jiggle node was implemented, which allowed animating the per-vertex jiggle in real-time, without needing to run a dynamic simulation. A cluster tracked deformation control system was developed, which permitted for weighted cluster controls that move as the mesh deforms, yet still allow additional deformation in local alignment to the current deformed transformation. An adaptation for real time faked eye refraction (inspired by previously presented works) was created, based on geometric deformations driven by an eye to camera vector and localized spherical geometry projections. The final iris refraction & dilation was animated and viewed in real time by the animator.

Conclusion

While creating I, Robot, Digital Domain developed a mixture of innovative solutions and novel adaptations of techniques, as well as refining our internal job system and character pipelines. Final retrospect revealed opportunities for improvement to our relative structure, and many of our tools and methodologies can still be improved, but the work that was done for this project is guaranteed to make a lasting impact on the future of our visual effects pipelines at Digital Domain.