HAL
open science

Dynamic Painting of Animated 3D scenes

David Vanderhaeghe, Pascal Barla, Joëlle Thollot, François X. Sillion

To cite this version:

David Vanderhaeghe, Pascal Barla, Joëlle Thollot, François X. Sillion. Dynamic Painting of Animated 3D scenes. 4th International Symposium on Non-Photorealistic Animation and Rendering : NPAR, Jun 2006, Annecy, France. inria-00526617

HAL Id: inria-00526617

https://inria.hal.science/inria-00526617

Submitted on 15 Oct 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

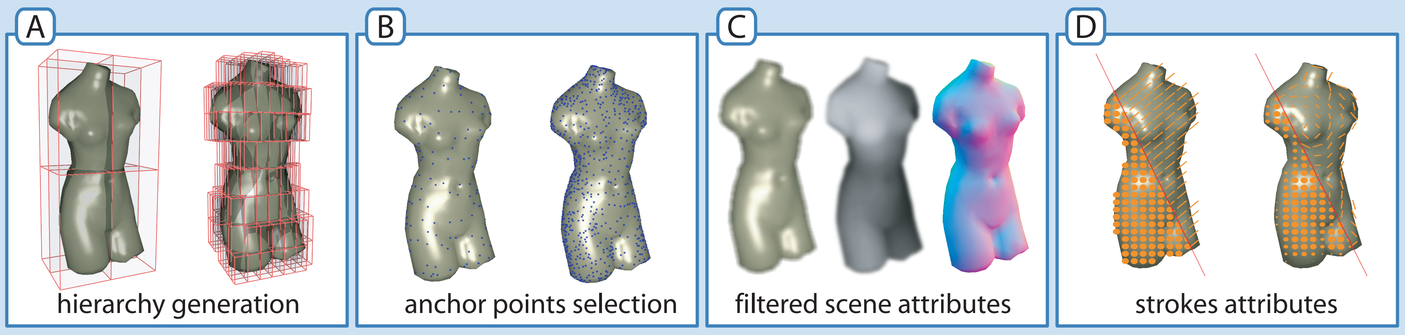
Dynamic Painting of Animated 3D scenes

David Vanderhaeghe, Pascal Barla, Joëlle Thollot, François X. Sillion
ARTIS GRAVIR/IMAG INRIA

Goals and Motivations

Painterly rendering is a technique that takes inspiration from traditional painting, such as oil or acrylic. The main idea is to render a 3D scene with 2D strokes in image space. Creating hand made painterly animations is very time-consuming since each frame of the animation is usually obtained by adding some paint strokes over previous frames. With an automated system, a user can not only build his animation faster but can handle the temporal coherence of strokes via a frame-to-frame correspondence. In our system the strokes follow the motion of objects (as in Meier's system [Mei96]) and faithfully represent some of their properties (depth and reflectance), while enabling the user to specify a painting style.

From scene to strokes



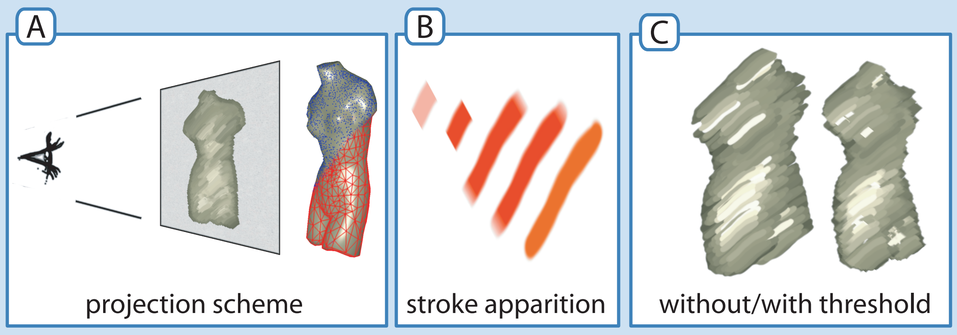
A For each object in the scene, we build a hierarchy of anchor points on its surface using an octree.

B At render time, we choose anchor points depending on the viewpoint and a user supplied density parameter.

C Scene attributes (such as color, depth, normal, curvature…) are rendered into G-buffers. Following a provided abstraction threshold, we filter attributes (with a Gaussian blur) to remove features smaller than this threshold.

D Scene attributes are linked to stroke attributes (color, orientation, size…) by a user specified style. For example one can use a global orientation and choose size depending on normals.

Drawing strokes



A Once we have determined a set of anchor points and stroke attributes, we draw strokes in image space.

B To remove poppings due to density and visibility events during the animation, we use a smooth transition animation.

C The user can constrain strokes to preserve underlying scene features depending on color and depth thresholds.
Drawing stops when a stroke reaches a region with attributes too different from its anchor point attributes.

Results

varying abstraction levels

changing light position

These results show strokes adaptation to the scene, with various styles (global orientation and size on the left, following normals on the right). See how small regions of color are preserved by stroke length cutoff and how jittering emphasizes individual strokes.

[Mei96] Meier, B.J. Painterly rendering for animation, SIGGRAPH 96.

ARTIS is a team of the GRAVIR - IMAG research lab (UMR C5527 between CNRS, INPG, INRIA and UJF), and a project of INRIA.