

Appendix C

Supplementary Material: Images, Papers and Videos (Electronic)

The enclosed DVD-ROM (affixed to the back cover) contains a collection of paintings, papers and videos which support this thesis.

C.1 Paintings

High resolution versions of paintings presented in this thesis are contained in the /paintings directory:

bathabbey.tif	Bath Abbey (Figure 4-21)
chilepickup.tif	Pickup Truck (Figure 4-14)
dragon.best.tif	Chinese dragon post-relaxation (Figure 4-1)
dragon_gen1.tif	Chinese dragon after 1 iteration of GA relaxation
dragon_gen30.tif	Chinese dragon after 30 iterations of GA relaxation
dragon_gen70.tif	Chinese dragon after 70 iterations of GA relaxation
modelface.tif	Man on a rock (detail on face before sharpening)
relaxation.mpg	Video of the GA relaxation process for dragon
rock.tif	Man on a rock (Figure 4-13)
still-life.tif	Still-life, kitchen (Figure 4-8)
sunflowers.tif	Sunflowers (Figure 4-15)
cubist/CharlesClark.pdf	Front page of the THES feat. Cubist portrait (Figure 3-18)
cubist/guitar.tif	Cubist still-life (guitar) (Figure 3-17d)
cubist/portrait_jpc.tif	Cubist portrait (of author) (Figure 3-17b)
cubist/portrait_pmh.tif	Cubist portrait (of supervisor) (Figure 3-21)

C.2 Papers

Copies of publications arising from this thesis [22, 23, 24, 25, 26, 27] and the state of the art video painting algorithm used in the comparison of Chapter 8 [103] are included in the `/papers` directory.

C.3 Videos

The following source videos and animations are included in the `/videos` directory:

<code>ballet_streak.avi</code>	<i>BALLET</i> sequence with streak-lines, demonstrating depth ordering.
<code>basket_rendered.avi</code>	<i>BASKETBALL</i> sequence with augmentation and deformation cues, demonstrating occlusion handling.
<code>basket_source.avi</code>	Source <i>BASKETBALL</i> video sequence.
<code>bounce_deformationonly.avi</code>	<i>BOUNCE</i> sequence exhibiting only squash and stretch deformation, illustrating collision handling.
<code>bounce_flatshade.avi</code>	<i>BOUNCE</i> sequence, flat shaded with no outlines. Exhibits squash and stretch deformation.
<code>bounce_fullcartoon.avi</code>	<i>BOUNCE</i> sequence, full cartoon flat shaded with sketchy outlines. Exhibits both augmentation and deformation motion cues.
<code>bounce_gradshade.avi</code>	<i>BOUNCE</i> sequence, gradient shaded with no lines.
<code>bounce_mixedmedia.avi</code>	<i>BOUNCE</i> sequence, mixed media effect — actor photorealistic, background sketchy. Exhibits both augmentation and deformation cues.
<code>bounce_motiononly.avi</code>	<i>BOUNCE</i> sequence exhibiting both augmentation and deformation cues, but no shading.
<code>bounce_painterly_ourmethod.avi</code>	Our painterly video algorithm (Chapter 8) applied to <i>BOUNCE</i> .
<code>bounce_painterly_SoA-.avi</code>	State of the art painterly video algorithm [103] (without stroke density regulation) applied to <i>BOUNCE</i> .

<code>bounce_painterly_SoA.avi</code>	State of the art painterly video algorithm [103] applied to <i>BOUNCE</i> .
<code>bounce_reframe.avi</code>	Demonstrating the attachment of a rigid reference frame for painting in <i>BOUNCE</i> .
<code>bounce_source.avi</code>	Source <i>BOUNCE</i> video sequence.
<code>bounce_watercolourwash.avi</code>	<i>BOUNCE</i> sequence with watercolour effect.
<code>contraption.avi</code>	Tracked <i>CONTRAPTION</i> sequence, used to test pivot point recovery algorithms. Here, pivot points have been identified and articulated pose recovered automatically (pose vector on right hand side of frame).
<code>cricket_mblur.avi</code>	<i>CRICKET</i> sequence exhibiting motion blur (tightly packed ghosting lines).
<code>cricket_source.avi</code>	Source <i>CRICKET</i> video sequence.
<code>cricket_streakghost.avi</code>	<i>CRICKET</i> sequence exhibiting streak lines and ghosting.
<code>metro_anticipate.avi</code>	<i>METRONOME</i> sequence exhibiting “time and pose” cues (specifically, anticipation).
<code>metro_qmapped.avi</code>	<i>METRONOME</i> sequence exhibiting both augmentation cues and coherent application of Q-mapped textures [65].
<code>metro_source.avi</code>	Source <i>METRONOME</i> video sequence.
<code>metro_streaks.avi</code>	<i>METRONOME</i> sequence exhibiting streak-lines and ghosting.
<code>metro_warp_accel.avi</code>	<i>METRONOME</i> sequence exhibiting emphasised inertia by non-linear deformation.
<code>metro_warp_anticipate.avi</code>	<i>METRONOME</i> sequence exhibiting both anticipation, and deformation motion cues.
<code>metro_warp_veloc.avi</code>	<i>METRONOME</i> sequence exhibiting emphasised drag by non-linear deformation.
<code>panorama.avi</code>	Demonstrating how video frames are registered to one another via homography, so producing a camera motion compensated sequence (uses <i>VOLLEY</i> footage).
<code>pooh_angrybear.avi</code>	Example of rotoscoping (an illustration is rotoscoped onto the head in <i>POOHBEAR</i> sequence).

<code>pooh_cartoon.avi</code>	<i>POOHBEAR</i> sequence, cartoon flat shaded with solid brush lines.
<code>pooh_coherentshade.avi</code>	<i>POOHBEAR</i> sequence, demonstrating coherent setting of interior region attributes (colour). See Figure 8-13.
<code>pooh_flatshade.avi</code>	<i>POOHBEAR</i> sequence, flat shaded no lines.
<code>pooh_gradshade.avi</code>	<i>POOHBEAR</i> sequence, gradient shaded no lines.
<code>pooh_incoherentshade.avi</code>	<i>POOHBEAR</i> sequence, demonstrating simplistic (incoherent) setting of interior region attributes (colour). See Figure 8-13.
<code>pooh_painterly_falsecolour.avi</code>	<i>POOHBEAR</i> painterly rendering of head, in false colour to demonstrate stroke coherence.
<code>pooh_painterly_truecolour.avi</code>	<i>POOHBEAR</i> painterly rendering of head in true colour.
<code>pooh_reframe.mpg</code>	Demonstrating the attachment of a rigid reference frame for painting in the <i>POOHBEAR</i> sequence.
<code>pooh_source.avi</code>	Source <i>POOHBEAR</i> video sequence.
<code>pooh_watercolourwash.avi</code>	<i>POOHBEAR</i> sequence with watercolour wash effect.
<code>pooh_wobblyflatshade.avi</code>	Introducing controlled incoherence into the Stroke Surfaces in <i>POOHBEAR</i> (flat shaded).
<code>pooh_wobblygradshade.avi</code>	Introducing controlled incoherence into the Stroke Surfaces in <i>POOHBEAR</i> (gradient shaded).
<code>sheep_flatsegment.avi</code>	Cartoon flat-shaded <i>SHEEP</i> sequence (no lines).
<code>sheep_painterly_falsecolour.avi</code>	<i>SHEEP</i> painterly rendering of sheep in false colour to demonstrate stroke coherence.
<code>sheep_painterly_SoA-.avi</code>	State of the art painterly video algorithm (without stroke density regulation) applied to <i>SHEEP</i> .
<code>sheep_painterly_SoA.avi</code>	State of the art painterly video algorithm as described in [103] applied to <i>SHEEP</i> .

<code>sheep_painterly_truecolour.avi</code>	<i>SHEEP</i> painterly rendering of sheep in true colour.
<code>sheep_rotomatte.avi</code>	Demonstrating rotoscoping and video matting in the <i>SHEEP</i> sequence.
<code>sheep_sketchcartoon.avi</code>	<i>SHEEP</i> sequence, flat shaded with sketchy lines.
<code>sheep_source.avi</code>	Source <i>SHEEP</i> video sequence.
<code>sheep_wobblycartoon.avi</code>	Introducing controlled incoherence into the holding line in the <i>SHEEP</i> sequence, but maintaining coherence in interior regions (flat shaded with solid brush lines).
<code>spheres_painterly_ourmethod.avi</code>	Our proposed painterly video algorithm applied to <i>SPHERES</i> .
<code>spheres_painterly_SoA-.avi</code>	State of the art painterly video algorithm (without stroke density regulation) applied to <i>SPHERES</i> .
<code>spheres_painterly_SoA.avi</code>	State of the art painterly video algorithm as described in [103] applied to <i>SPHERES</i> .
<code>spheres_sketchywash.avi</code>	<i>SPHERES</i> with watercolour wash effect and sketchy lines.
<code>spheres_source.avi</code>	source <i>SPHERES</i> video sequence (synthetic test sequence used in Chapter 8).
<code>spheres_wobblyflatshade.avi</code>	<i>SPHERES</i> sequence, flat shaded and after introduction of controlled incoherence into the region boundary.
<code>stairs_exaggerate.avi</code>	<i>STAIRS</i> sequence with motion exaggeration (filled feature polygons only)
<code>stairs_exaggerate_polys.avi</code>	<i>STAIRS</i> sequence with motion exaggeration, textured to produce Monty Python style animation.
<code>stairs_source.avi</code>	Source <i>STAIRS</i> video sequence.
<code>volley_motiononly.avi</code>	<i>VOLLEY</i> sequence with augmentation and deformation cues, demonstrates camera motion compensation.
<code>volley_source.avi</code>	Source <i>VOLLEY</i> video sequence, exhibits large scale camera motion.
<code>vpshowreel.avi</code>	Video Paintbox show-reel (uncompressed RGB — very large, but full quality).

<code>vpshowreel_lowres_divx5.avi</code>	Video Paintbox show-reel (highly compressed, 52Mb approx DIVX-5 compression).
<code>wand_cartoon.avi</code>	<i>WAND</i> sequence, begins with original footage, then adds augmentation cues, then adds deformations, then adds video shading (flat shaded cartoon with sketchy lines).
<code>wave_sketchonly.avi</code>	<i>WAVE</i> sequence, sketchy lines only.
<code>wave_sketchycartoon.avi</code>	<i>WAVE</i> sequence, full cartoon flat shaded with sketchy lines.
<code>wave_source.avi</code>	Source <i>WAVE</i> video sequence.
<code>wave_thicklinecartoon.avi</code>	<i>WAVE</i> sequence, cartoon flat shaded with solid brush lines.