INTERNATIONAL ORGANISATION FOR STANDARDISATION

ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11 M9750
Trondheim, July 2003

Title: **Report on SynthesizedTexture Coding and Animation CE**

Source: **Shlomo Birman (Vimatix), Yoram Elichai (Vimatix), Ehud Spiegel (Vimatix)**
Cyril Concolato (ENST), Alexandre Cotarmanac'h (FT)

Status **Core Experiment input**

Summary

This document presents the results of the SynthesizedTexture Coding and Animation Core Experiment that was defined in N5648 and started after the Pataya meeting. After citing the core experiment stages and the evaluation methods, we present the results and the recommendations.

The recommendations are to use both VIM encoding scheme for objects and Vim interpolator for interpolation, and since both components have no usage outside of ST, it has been decided to further recommend using Vim stream that includes both the Objects (textures) and their interpolation.

We have opened an account that holds the Synthesized Texture CE tools and results. It's details are:

ftp://ftp.vimatix.com

user: vimatix44

pass: 2na#56

You will find there a readme file explaining how to use the tools and content.

Background:

In MPEG#64 March 2003 meeting at Pattaya it has been decided to conduct a CE on SynthesizedTexture compression and animation. See MPEG document N5648.

The CE mandate and stages:

1) To find out where the compression factor of SynthesizedTexture (ST) is coming from.

2) To assess whether and how the techniques that provide the more important gain could be integrated in BIFS, probably by a dedicated encoding node.
3) If the residual gain after integrating these tools is greater that 30%, to define how Vimatix encoding scheme can be introduced to compress SynthesizedTexture stream.
4) Asses if extending XCurve to incorporate SynthesizedTexture ColorProfiles can efficiently replace SynthesizedTextureCurve.
5) Assess the possibility to animate SynthesizedTexture with existing BIFS Interpolators.
6) If animation quality cannot be achieved with BIFS Interpolators, suggest a new Interpolator node for SynthesizedTexture Animations.
7) Assess the efficiency of rendering SynthesizedTextureObjects within a CompositeTexture2D.
8) If CompositeTexture2D does not proof to achieve the required rendering efficiency, propose an equivalent node for SynthesizedTexture.

This document describes:

1. The criteria for evaluating the CE results in correspondence to the above stages
2. The CE results and conclusions

References

- N5648
- M9237

Synthesized Texture Coding and Animation - CE Criteria per CE stage

CE Stage 1: "To find out where the compression factor of SynthesizedTexture is coming from"

Method: A set of examples used in previous CE concerning ST (see M9237) were selected:

<table>
<tr><td>  </td><td>  </td></tr>
<tr><td>1. Basketball.q01.f00.vim</td><td>2. Butterfly.q01.f00.vim</td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><td>3. Liberty.q01.f00.vim</td><td>6. Shlomo.q01.f00.vim</td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><td>5. Pigeons.q01.f00.vim</td><td>4. Palms.q01.f00.vim</td></tr>
</table>

Statistics were gathered from the above example files with regard to various ST compression steps, as defined in M9479 section 3.2.

The ST encoding steps are:

1. Encode Locations
2. Encode Curve Geometries
3. Encode Area Colors
4. Encode Curve Color Profiles
5. Encode Patch Geometry and Colors

Evaluation: Statistics about the various encoding steps shell be used for ranking them by their relative contribution.

Criteria: In case specific steps prove to significantly contribute to the compression more than others, they should be in focus in stage 2 of the CE.
In case no significant contributors were found, stage 2 of the CE will have to consider all steps.

CE Stage 2: " To assess whether and how the techniques that provide the more important gain could be integrated in BIFS, probably by a dedicated encoding node"

Method: Based on most significant ST encoding contributors found in CE Stage 1, BIFS-based alternative coding solutions should be proposed and tested by BIFS experts.

Evaluation: Compare BIFS-based compression methods to proposed ST 'native' coding results. Since the margin for overall decision is small – 30% gain (see Stage 3), the decision to use BIFS-based techniques per ST encoding step should be that it is no more than 30% bigger.

Note: in case this is not achievable for all steps but some of them achieve significantly better results, better steps may compensate for less successful ones – thus, the overall compression result is significant as well.

The overall compression result is calculated by the combined results of BIFS-based techniques, including the new proposed ones, and also old ones that are applicable to the less significant steps (see Stage 1 above).

Criteria: as implied by Stage 3, if the residual gain after integrating these tools is greater than 30%, then the Vimatix proposed encoding scheme will be used for ST.

CE Stage 3: "If the residual gain after integrating these tools is greater that 30%, to define how Vimatix encoding scheme can be introduced to compress SynthesizedTexture stream"

Method: This stage will be based on the results of Stage 2. In case the results of Stage 2 call for introduction of Vimatix encoding scheme, its way of usage should be defined. Since there is no use in MPEG for ST specific nodes outside of ST, it will be possible to integrate the 'native' coding as self-contained methods. Various MPEG integration mechanisms should be examined:

1. Using a Node interface
2. Using a URL
3. Using BitWrapper

Evaluation: The above integration mechanisms would be evaluated considering fulfilling all the needs of ST, which are:

1. Scene updates
2. Integration with an animation mechanism
3. User interaction

Criteria: We will chose the method that fulfills all the above needs while offering the simplest way of integration within MPEG.

Note: Implementation may continue after the CE to be finished prior to MPEG meeting #65.

CE Stage 4: " Asses if extending XCurve to incorporate SynthesizedTexture ColorProfiles can efficiently replace SynthesizedTextureCurve"

Method: In case Vimatix encoding scheme will be introduced as defined by Stage 3 above, stage 4 may not be relevant. The evaluation of XCurve extension to incorporate ST Color Profiles shell be conducted in one or two phases:

Phase 1: Theoretical analysis, leading to first estimation of fulfilling the needed Criteria. Only if definitely positive, phase 2 should be conducted

Phase 2: The visual quality of the sample images from stage 1 should be evaluated. This phase may not be implemented in case BIFS experts it's obvious and won't corrupt the visual results.

Evaluation: The new XCurve must match the entire functionality of STCurve. This must be measured also visually for the reference images.

Criteria:

1. Same visual appearance as using ST Curve color profiles
2. No significant overhead on compression
3. Usability outside of ST
4. No complication of the implementation

CE Stage 5: "Assess the possibility to animate SynthesizedTexture with existing BIFS Interpolators"

Method: Two phases shell be conducted:

> Phase 1: Theoretical analysis - Using the "Synthesized Animation Description" in w5645, specifies what is needed of the interpolators and tracing possible MPEG-interpolators candidates, and filtering the final nominee/s out of them.
>
> Phase 2: If candidates were traced and successfully nominated in Phase 1,we shell proceed to a test implementation. The test set would be:
> YosiBambi.vim, a file that contains all aspects of ST animation (in Vimatix proprietary format: VIM) with photorealistic image (Yosi) and a cartoon (Bambi) carrying full range of motions of all types.

Evaluation: BIFS-based animations have to match the full functionality of 'native' ST Animation (VIM). Visual evaluation should be carried as well.

Criteria:

1. Same visual appearance.
2. No significant overhead on the compression

Note: The implementation itself may proceed after the CE, and should be concluded prior to MPEG#65 meeting.

CE Stage 6: " If animation quality cannot be achieved with BIFS Interpolators, suggest a new Interpolator node for SynthesizedTexture Animations"

Method: If Stage 5 proofs that BIFS animations are good enough for ST, stage 6 is not needed. Otherwise, an interpolation node based on current ST animation methods (VIM) would be suggested. Same test set as described for Stage 5 will be used to evaluate quality.

Evaluation: The new Interpolator must achieve the full functionality of 'native' ST Animation (VIM). This must be measured also visually, comparing to the above test set.

Criteria:

1. Same visual appearance.
2. No significant overhead on the compression

Note: The implementation may continue after the CE; to be concluded prior to MPEG#65 meeting.

CE Stage 7: "Assess the efficiency of rendering SynthesizedTextureObjects within a CompositeTexture2D"

Method: This Stage relates to Stage 2 results. If going for 'native' ST coding, then there is no use for having CompositeTexture2D since it will just call to a self-contained ST node. If not going for 'native' coding, then there's need to evaluate the functionalities offered by Composite2D versus the proposed STObject. Using the above we shell start with the first example, and only after having an equivalent visual result we shell proceed to the rest of the examples. This phase might not be 'physically' implemented in case BIFS experts recommend it's obvious and won't corrupt the visual results.

Evaluation: Using CompositeTexture2D achieves the full functionality of using a dedicated STObject node.

Criteria:

1. Same visual appearance.
2. No significant overhead on the compression
3. Easy integration with the chosen interpolator (Stages 5-6 above).

CE Stage 8: " If CompositeTexture2D does not proof to achieve the required rendering efficiency, propose an equivalent node for SynthesizedTexture"

Method: Based on the results of Stage 7 above. If Stage 7 results in staying with CompositeTexture2D, stage 8 is not needed. If needed, we shell suggest a node based on final implementation of ST per step 2 above.. We shell use the same test set as described for Stage 7.

Evaluation: The new node must achieve the full functionality of 'native' ST. This must be measured also visually, comparing to the above test set.

Criteria:

1. Same visual appearance.
2. No significant overhead on compression
3. No complication of implementation

Note: The implementation itself may proceed after the CE, and should be concluded by MPEG#65 meeting.

Results summary:

CE Stage	Result	See documents (attached to m9750):	Action
1	Five elements, similar importance. All methods are required . The statistics shows that a compression gain of 1.7 is due to Vim special techniques.	STCodingStat2.4.doc	
2	Impossible to implement Vim special techniques in reasonable effort in BIFS. Recommending to use Vimatix encoding scheme.	STCodingStat2.4.doc	STObject uses 'VIM' coding
3	Since the residual gain is 70%, much more than the 30%, it is recommended to use Vim encoding scheme. Use a node, called STObject using a URL, and use exposed control points.		Due to the result in stage 6, it is recommended to use a composite VIM stream. See (*) bellow
4	Use the original ST Curve.	XCurveForSTCurve.03.doc	Use STCurve node (***)
5	Candidate: CoordinateInterpolator . Results shows compression overhead of 6:1. (**)	STInterpolator.03.doc	
6	Due the large overhead on compression it has been decided to recommend using the Vim interpolator.		Due to the result in stage 3, it is recommended to use a composite VIM stream. See (*) bellow
7	CompositeTexture2D seems to be good enough		Use CompositeTexture 2D node.
8	Not needed		

(*) Since it is recommended to use both VIM encoding scheme for objects and Vim interpolator for interpolation, and since both components have no usage outside of ST, it has been decided to recommend using Vim stream that includes both the Objects (textures) and their interpolation, by using an ordered match (i.e. interpolation sequence per object). The exact implementation, that will try to preserve interaction on the object level, will be presented MPEG#65 meeting.

(**) Since VIM interpolation is affecting only the skeleton points that changes at each key-frame, much of the data carried by conventional interpolators is redundant.

(***) Due to the result and recommendation of step3, STCurve becomes a component in STObject, but not a separate node.