Aural-Sight

An Auralization of the Visual World

Gabriel Karpman
Clark Della Silva
Adam Suhl
Overview

- An interactive system that takes video input and converts it into a representation realized in the audio domain.
- Process video for both ambient and intentioned content
- Controls to allow for augmentation / modification of the signals
- Fast, compact IFFT for precise, parametric audio synthesis
Design Block Diagram

- NTSC Camera Input
  - Video Decoder
    - H, S, V
  - Chromatic Spectrographic Analysis
    - HSV addr, H, S, V
    - Object Location
      - Sprite Detection
        - HSV addr, H, S, V
  - Data Converter
    - HSV addr, H, S, V
    - freq, freq_value, tempo
    - Sample Addr
      - Object Location
  - Sequencer
    - IFFT Data
    - Audio Synthesis / IFFT
      - Channel Addr
        - Audio
      - Audio
        - Channel Addr, Audio
  - Sampler
    - Audio
    - Audio
    - Channel Addr, Audio
    - Audio
  - Channel Aggregator / Mixer
    - Audio
    - Audio
  - Post-IFFT FX
    - Audio
    - Audio
  - Pre-IFFT FX
    - IFFT Data
  - Audio Encoder
    - Audio Output
Chromatic Spectrum Analysis
Blob Detection

Identify multiple control objects of predefined types (LEDs of different colors.)

Analyse position and movement.

Generate control parameters.
Blob Detection

In order to achieve manageable computational effort and memory usage
Blob Detection

In order to achieve manageable computational effort and memory usage first identify pixels which may be in objects:

Use threshold on distance from target in LogHS space.
Blob Detection

In order to achieve manageable computational effort and memory usage

first identify pixels which may be in objects:

  Use threshold on distance from target in LogHS space.

Then apply approximated Difference of Gaussians to this bitmap.

  cheap, since multiplying powers of two by 0 or 1.
Video to Audio Conversion

**Hue** (0-360) \( \rightarrow \ f_0 \) (100Hz -- 2kHz)  
Fundamental Frequency

**Saturation** (0-1) \( \rightarrow \ \Phi \) (\( f_0 \) -- 11\( f_0 \))  
Center of Gaussian

**Brightness** (0-1) \( \rightarrow \) (\( f_0 \) -- 5\( f_0 \))  
Std dev of Gaussian
Video to Audio Conversion

$fo$ Harmonics

$\Phi$

Amplitude

$\max$

Frequency

$\Phi$

Amplitude

$\max$

Frequency

$2.5%$ $13.5%$ $34%$ $34%$ $34%$ $13.5%$ $2.5%$

$-3$ $-2$ $-1$ $0$ $1$ $2$ $3$

Standard Deviations
Sequencing
Frequency Domain Reverb

\[ S_{out}(n) = S_{in}(n) + S_{out}(n-1) * e^{-\lambda n} \]

\[ \alpha \Rightarrow \text{Rate of change of video} \]

\[ \alpha \to 0 \text{ then } \lambda \to .1 \]

\[ \alpha \to \infty \text{ then } \lambda \to 10 \]
Audio Generation

Synthesizer (IFFT)

Sampler (playback from RAM/ROM)

Routing/Aggregation

Effects (modular units)

Output

Back to Routing grid
Synthesizer / IFFT

Radix $2^2$ Single Delay Feedback architecture.

Fully multiplex hardware (can fit $2^{16}$ point FFT, for exceptional frequency resolution.)

Use twiddle multipliers to match phase between frames. (for exceptional time resolution.)
Routing and Effects

Routing/Aggregation

- Synth (Channel 0)
- Sampler (Channels 1-15)
- Effects unit outputs (Channels 16-255)

- In 0: Echo -> out 16
- In 1: Low pass -> out 17
- In 2,3: Mix -> out 18
- In 4,5: Stereo pan-> AC'97
  (And so on.)

Back to Routing grid

To AC'97
Timeline

Week of:
11/12 - FFT / Synthesizer finished + tested
- Sampler + channel routing implemented.
- Data Conversion Module and Reverb Module implemented
- Camera + NTSC -> HSV finished and tested.
11/19 - Chromatic analysis implemented
- Blob detection implemented.
- Sequencer Module implemented, Reverb and data conv Tested
- Effects units implemented, routing tested.
11/26 - All video units finished and tested.
- Data Conversion and Sequencer tested,
- integrate data, sequencer, reverb, test as a whole, debug
- All Audio units finished and tested together.
12/3 - Full system integration. Debug module-module issues.
- UI tweaks. Artistic and musical tweaks. Practice using the system.
- Any additional polishing as time allows.