WebGL, WebCL and Beyond!

Neil Trevett
Vice President Mobile Content, NVIDIA
President, The Khronos Group
Topics in this Session

- Quick introduction to 3D
- The need for acceleration APIs
- OpenGL ES and WebGL
- WebGL tools and frameworks
- Beyond 3D – WebCL for compute
- Augmented Reality in the browser?
- Questions

- Next Session – Mo Zhenyao from Google
- Hands on with WebGL
What is Real-time 3D Graphics?

Computer graphics is the science and art of using computers to create and enjoy beautiful, interactive experiences. The processor that makes these amazing experiences possible is the GPU.
3D has evolved over more than 30 years

‘Doom’ on a PC – 1993

id Software

http://www.youtube.com/watch?v=RSXyztq_0uM

‘Samaritan’ Real-time Demo on a PC – 2011

Epic Unreal Engine
3D Pipeline Basics

• The art of “faking” realistic looking scenes or objects using heuristic techniques learned over the years
• The objects making up a scene are held in a database
• Surfaces of objects are broken down into a grid of polygons
• The vertices of the polygons are located in 3D coordinate space - x,y,z
• Each vertex has a “material” – color and reflective properties
• Vertices are positioned in 3D space – matrix math zooms and rotates
3D Pipeline Basics – Pixel Shading

• **Project each polygon onto the screen**
  - Determine which pixels are affected

• **Smooth Shading**
  - Run lighting equation at each vertex
  - Compute vertex color depending on how lights interact with surface angles and properties
  - Interpolate colors between the vertices

• **Texture Mapping**
  - “Wallpaper” each polygon with an image
  - For each pixel compute image coordinates in image to paste

• **Environment Mapping**
  - Paste reflection of image of environment at each pixel
Fundamental 3D Processing Stages

Operations on Vertices
- Traversal: What objects are in current scene?
- Transforms: Where are the polygons?
- Lighting: What color are the polygons?
- Rasterize: What shape are they on the screen?

Geometry

Operations on Pixels
- Color: What color is each pixel?
- Clip: Which pixels are visible?
- Write: Write the pixels to the framebuffer
Actual 3D Pipelines

OpenGL ES 1.x Fixed Function Pipeline

OpenGL ES 2.0 Programmable Pipeline
Khronos - Connecting Software to Silicon

• Creating open, royalty-free acceleration API standards
  - Focus on graphics, dynamic media, compute and sensor hardware

• Low-level - just above raw silicon
  - “Foundation” functionality needed on every platform

• Safe forum for industry cooperation
  - ‘By the industry for the industry’
  - Open to any company to join
  - IP framework to protect members and industry

APIs enable software developers to turn silicon functionality into rich end user experiences
Khronos Ecosystem of Standards

Khronos creates royalty-free specifications to meet real market needs and helps drive industry adoption across multiple platforms.
Mobile Silicon Experiential Processing

Diagram showing various processor units:
- Cortex A9 Processor
- HD Video Decode Processor
- ARM 7
- Audio Processor
- HD Video Encode Processor
- Image Processor
- 2D/3D Graphics Processor

Icons for web browsers and games are also displayed.
Mobile Roadmap Acceleration

PERFORMANCE

Production Devices

2011

2012

2013

2014

TEGRA 2

KAL-EL

WAYNE

LOGAN

STARK

Core 2 Duo - Macbook Air

© Copyright Khronos Group, 2011 - Page 13
A Lot More than Just “More HTML”

How can the Browser rapidly assimilate such diverse functionality?

- Rich Experiential Processing
- Multi-core CPUs
- Rich 2D and 3D GPU
- GPU Computing
- Multiple HD cameras
- Image and vision processing
- Video encode/decode
- Audio encode/decode
- Inertial and positional sensors
What is OpenGL ES?

• **OpenGL for embedded and mobile devices**
  - Eliminates redundant and legacy features
  - Adds extensions to make it mobile-friendly

• **The dominant 3D API for mobile devices**
  - Widely adopted for STB, DTV, automotive,…
  - Hundreds and hundreds of millions shipped

• **Runs high-end content and engines**
  - UE3, Unity, Unigine, Rage
**OpenGL ES Pipelines**

**OpenGL ES 1.x Fixed Function Pipeline**
- API
  - Triangle/Line/Points
  - Primitives
  - Transform and Lighting
  - Primitives Assembly
  - Rasterizer
  - Texture Environment
  - Colour Sum
  - Fog
  - Frame Buffer
- Alpha Test
  - Depth Stencil
  - Colour Buffer Blend
  - Dither

Based on OpenGL 1.5
- Vertex Arrays / Buffer Objects
- Transform & Lighting
- Multi-texturing (min 2 units)

**OpenGL ES 2.0 Programmable Pipeline**
- API
  - Triangle/Line/Points
  - Primitives
  - Vertex Shader
  - Primitives Assembly
  - Rasterizer
  - Frame Buffer
- Vertex Buffer
  - Depth Stencil
  - Colour Buffer Blend
  - Dither

Based on OpenGL 2.0
- Removes fixed function pipeline
- High level language (GLSL ES)
- Super-compact, efficient API
WebGL – 3D on the Web – No Plug-in!

• Historic opportunity to bring accelerated 3D graphics to the Web
  - WebGL defines JavaScript binding to OpenGL ES 2.0

• Leveraging HTML5 and uses <canvas> element
  - Enables a 3D context for the canvas

• JavaScript is easily fast enough now for visual computing
  - Plus OpenGL ES 2.0 enables local geometry caching and GPGPU computation

Being defined by major browsers and GPU vendors working together
OpenGL Ecosystem – 3D Everywhere

OpenGL ES 2.0 on desktop as subset of OpenGL 4.2 for mobile content flexibility – including native support for WebGL

Leading-edge functionality developed first on desktop

WebGL driving new-generation security features into OpenGL family

Mobile functionality subset that is deployed on billions of devices

Pervasive OpenGL ES 2.0 availability enables Browser vendors to build 3D directly into HTML5
WebGL Implementation Anatomy

Content downloaded from the Web. Middleware can make WebGL accessible to non-expert 3D programmers.

Browser provides WebGL functionality alongside other HTML5 specs - no plug-in required.

OS Provided Drivers. WebGL on Windows can use Google Angle to create conformant OpenGL ES 2.0 over DX9.
HTML5 Content Architecture

- **HTML content generated by layout engine ‘on page’**
- **<video> tag**
- **<canvas> tag**
- **Lorem ipsum**
- **Composition of off-screen buffers**
- **CSS Layout and Transforms**
- **Composition needs to be GPU accelerated**
- **Video, Vector Graphics and 3D created off-screen buffers**

JavaScript drives interactivity for 2D and 3D graphics.
WebGL and HTML Interaction

• 3D is not trapped in a rectangular window
  - 3D can overlay and underlay HTML content
  - Easy to make 2D HTML HUDs or 3D user interfaces

• Strong ties with other advanced HTML5
  - WebGL can use HTML5 `<video>`
    or canvas as a texture

• Can use 3D for core Web UI – as well as content
  - Advanced transforms and special effects

• Render HTML DOM sub-tree as texture
  - Mozilla and Google prototyping as extension
  - Support user interaction when in 3D
WebGL Deployment

- **WebGL 1.0 Released at GDC March 2011**
  - Mozilla, Apple, Google and Opera working closely with GPU vendors

- **Typed array 1.0 spec ratified by Khronos in May**
  - Supporting bulk data transfer between threads (workers)
  - Many use cases - background mesh loading, generation, deformation, physics ...

- **1.0.1 release of WebGL spec and conformance suite imminent**
  - 100% robust stance on security
  - Fixing bugs in 1.0.0 conformance suite
  - Implementations will report `getContext("webgl")` (not experimental)

<table>
<thead>
<tr>
<th>Show all versions</th>
<th>IE</th>
<th>Firefox</th>
<th>Chrome</th>
<th>Safari</th>
<th>Opera</th>
<th>iOS Safari</th>
<th>Opera Mini</th>
<th>Opera Mobile</th>
<th>Android Browser</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 versions back</td>
<td>6.0</td>
<td>5.0</td>
<td>12.0</td>
<td>3.2</td>
<td>11.0</td>
<td>3.2</td>
<td>10.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 versions back</td>
<td>7.0</td>
<td>6.0</td>
<td>13.0</td>
<td>4.0</td>
<td>11.1</td>
<td>4.0-4.1</td>
<td>11.0</td>
<td>2.1</td>
<td></td>
</tr>
<tr>
<td>Previous version</td>
<td>8.0</td>
<td>7.0</td>
<td>14.0</td>
<td>5.0</td>
<td>11.5</td>
<td>4.2-4.3</td>
<td>11.0</td>
<td>2.2</td>
<td></td>
</tr>
<tr>
<td>Current</td>
<td>9.0</td>
<td>8.0</td>
<td>15.0</td>
<td>5.1</td>
<td>11.6</td>
<td>5.0</td>
<td>11.5</td>
<td>2.3</td>
<td>3.0</td>
</tr>
<tr>
<td>Near future</td>
<td>9.0</td>
<td>9.0</td>
<td>16.0</td>
<td>5.1</td>
<td>12.0</td>
<td>5.0-6.0</td>
<td>11.5</td>
<td>2.3</td>
<td>3.0</td>
</tr>
<tr>
<td>Farther future</td>
<td>10.0</td>
<td>10.0</td>
<td>17.0</td>
<td>6.0</td>
<td>12.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

WebGL is not enabled by default in Safari

http://caniuse.com/#search=webgl
Aquarium Demo

• On PC and Android

• http://webglsamples.googlecode.com/hg/aquarium/aquarium.html
Frameworks and Tools

- WebGL is deliberately low level to enable the full power and flexibility of OpenGL ES 2.0
- If you are not an expert 3D programmer – don’t panic!
- WebGL is perfect foundational layer for JavaScript middleware frameworks
- Lots of utilities and tools already appearing

http://www.khronos.org/webgl/wiki/User_Contributions
WebGL Security

• Any new functionality in the browser increases exposure to attack
  - True since the beginning of the web – the new functionality becomes hardened

• ANY graphics in the browser need the GPU drivers to be hardened
  - HTML, Canvas, WebGL, Adobe Molehill, Silverlight 5 ...

• WebGL is designed with security as the highest priority
  - Hardening is being strongly promoted and enabled

• Short term – browser vendors will maintain white and black lists
  - Compromised system can have WebGL disabled until mitigation developed

• Longer term – GPUs provide increasingly robust security and multi-tasking
  - GPU becoming a first-class computing platform alongside CPU
WebGL Security in the Press!

• Confusion in the industry as we start this hardening process
  - Shader programs cannot access general system resources or perform out of range memory access!

• Issues in the Press
  - Cross domain image access – timed loop attack – SOLVED!
    - WebGL and HTML spec updates - mandating CORS for video, images and audio
    - Servers have to grant cross-domain access to media resources
  - DOS Attacks and general hardening
    - ARB_robustness extensions that provide additional protection being mandated
    - New robustness spec limits the side-effects of a GPU reset after a DOS attack
    - ANGLE shader validator improved; more improvements coming
Web Apps versus Native Apps

- **Mobile Apps have functional and aesthetic appeal**
  - Beautiful, responsive, focused

- **HTML5 with accelerated APIs can provide the same level of “App Appeal”**
  - Highly interactive, rich visual design

- **Using HTML5 to create ‘Web Apps’ has many advantages**
  - Portable to any browser enabled system
  - Same code can run as app or as web page
  - Web app is searchable and discoverable through the web
  - Not a closed app store – no app store ‘tax’
Web Apps - Wider Ecosystem

- **OS capability access before in HTML5**
  - Execution with no browser UI
  - Packaging standalone apps

- **OS Independent App stores**
  - Discovery and payment

- **Language and JavaScript Tools**
  - Native code compilation to JavaScript
  - JavaScript libraries

- **Authoring Tools**
  - Bringing Flash-grade authoring to HTML5
Declarative 3D for the Web

• Need to enable ‘non-expert’ web programmers with layers over WebGL
  - 10,000s of 3D programmers worldwide versus millions of web developers
  - Middleware and layered architectures play a vital role

• W3C Incubator for Declarative 3D
  - "easy way to add interactive high-level declarative 3D objects to the HTML-DOM”
  - X3DOM (www.x3dom.org/) and XML3D (www.xml3d.org/)

• Bind 3D even closer into the browser stack
  - Use as much HTML5 machinery as possible – DOM, JavaScript, CSS
  - Focus on driving optimized WebGL/OpenGL ES 2.0 back-end
  - Use Typed Arrays and drive for optimal performance
Leveraging Native API Investment into HTML5

- HTML5 evolving into cross-platform programming platform
  - Gradually exposing complete system capabilities
- Opportunity to synergize Web and native APIs development
  - Leverage native API investments, reduce developer learning cycles
- Khronos and W3C creating close liaison

Native APIs shipping or working group underway
JavaScript API shipping or working group underway
Possible future JavaScript APIs
Processor Parallelism

CPUs
Multiple cores driving performance increases

GPUs
Increasingly general purpose data-parallel computing

Emerging Intersection

Multi-processor programming – e.g. OpenMP

Heterogeneous Computing

Graphics APIs and Shading Languages

OpenCL is a programming framework for heterogeneous compute resources
OpenCL – Heterogeneous Computing

• Framework for programming diverse parallel computing resources in a system

• **Platform Layer API**
  - Query, select and initialize compute devices

• **Runtime API**
  - Execute compute kernels – gather results

• **Kernel Language Specification**
  - Subset of ISO C99 with language extensions

• **OpenCL has Embedded profile**
  - No need for a separate “ES” spec
WebCL – Parallel Computing for the Web

• Khronos launching new WebCL initiative
  - First announced in March 2011
  - API definition already underway

• JavaScript binding to OpenCL
  - Security is top priority

• Many use cases
  - Physics engines to complement WebGL
  - Image and video editing in browser

• Stay close to the OpenCL standard
  - Maximum flexibility
  - Foundation for higher-level middleware
Visual Computing Ecosystem

High performance compute and graphics interop – buffer and events

Compute and mobile APIs interoperate through EGL

JavaScript bindings to OpenCL
Parallel computation in HTML5
WebCL Open Process and Resources

- **Khronos open process to engage Web community**
  - Public specification drafts, mailing lists, forums
  - [http://www.khronos.org/webcl/](http://www.khronos.org/webcl/)
  - [webcl_public@khronos.org](mailto:webcl_public@khronos.org)

- **Khronos welcomes new members to define and drive WebCL**
  - [info@khronos.org](mailto:info@khronos.org)

- **Nokia open sourced prototype for Firefox in May 2011 (LGPL)**
  - [http://webcl.nokiaresearch.com](http://webcl.nokiaresearch.com)

- **Samsung open sourced prototype for WebKit in July 2011 (BSD)**

- **Deformation Demo:**
  - Calculates and renders transparent and reflective deformed spheres on top of photo background
  - Performance comparison on Mac
    - JS: ~1 FPS
    - WebCL: 87-116 FPS
  - [http://www.youtube.com/user/SamsungSISA#p/a/u/1/9Ttux1A-Nuc](http://www.youtube.com/user/SamsungSISA#p/a/u/1/9Ttux1A-Nuc)
Visual-based Augmented Reality

Camera video stream sent to the compositor

Camera images used to track the camera’s location and orientation

Camera Tracking

Camera-to-scene transform locks the 3D rendering to the real world

3D Augmentation Rendering

3D augmentations composited with video stream
OpenSL ES – Advanced Audio

• OpenSL ES does for audio what OpenGL ES does for graphics
  - Advanced audio functionality from simple playback to 3D audio

• Object-based native audio API for simplicity and high performance
  - Reduces development time

• Same API regardless of underlying implementation
  - Software or hardware accelerated

• Cross OS portability
  - Preserves application investment
StreamInput Connects Sensors to Apps

Advanced Sensors Everywhere
Standard cameras, depth cameras
motion and position, touch, microphones
wireless controllers

Apps request semantic sensor information
StreamInput defines list of possible semantic requests
"Am I in an elevator?"  "Give me gestures and face position"

Universal Timestamps

Sensor graph created to provide sensor information
StreamInput defines graph creation API and node interconnects
Low-level sensor processing encapsulated in nodes – unleashes fusion innovation
Apps gain ‘magical’ situational awareness

Apps Need Sophisticated Access to Sensor Data
Without coding to specific systems or sensor hardware
Khronos Computer Vision Standard

- OpenCV is widely use open source project for COMPUTER VISION
- Khronos Hardware Abstraction Layer
  - Will enable hardware vendors to provide accelerated imaging and vision modules
- CV HAL can be used by high-level libraries or applications directly
Augmented Reality Functionality

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

- Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV

Augmented Reality Functionality

Control Camera, Preprocess and generate video streams

EGL

Sync

Sensor fusion

Audio Rendering

StreamInput

- Positional and GPS Sensor Data
- Computer Vision and Tracking
- Synchronization and sensor fusion

Positional Sensors

Camera Processing

- Control Camera, Preprocess and generate video streams

Camera

EGLStream

Video stream to GPU

3D Rendering and Video Composition

- 3D Rendering and Video Composition

Audio Rendering

- Position and Tracking Semantics

Application on CPU

OpenSL ES

Much more flexibility than just “overlay augmentations over background”

OpenMAX AL

Video TAP to CPU

OpenCV
Augmented Reality in the Web

Semantic, synchronized sensor fusion

StreamInput

Positional and GPS Sensor Data

Computer Vision and Tracking

Synchronization and sensor fusion

WebCL

Application on CPU

Audio Rendering

WebAudio?

Camera Processing

Camera

Sufficiency of sophisticated camera control

Positional Sensors

Video TAP to CPU

Video stream to GPU

EGLStream

3D Rendering and Video Composition

WebGL

Need to explore whether HTML composition can handle all AR use cases
Industry Cooperation
Get Involved!

- Engage with the WebGL working group on Khronos forums and mailing lists
- Let us know if you have news or links that Khronos can help highlight
  - info@khronos.org or edit the Wiki
- Join Khronos to have a voice in how the specs evolve!
  - Any company is very welcome

http://www.khronos.org/webgl/wiki/Main_Page
In Summary

- WebGL brings another vital piece of system capability into the HTML5 browser for web apps – 3D graphics
- WebGL is being deployed right now on PC – soon on mobile – and is being strongly supported by browser and GPU vendors
- WebGL is a low-level, secure technology – that can be used directly and will support a rich ecosystem of tools and frameworks
- WebGL and WebCL show how to take well proven native APIs and bring them to the web – with more to come!
Questions?