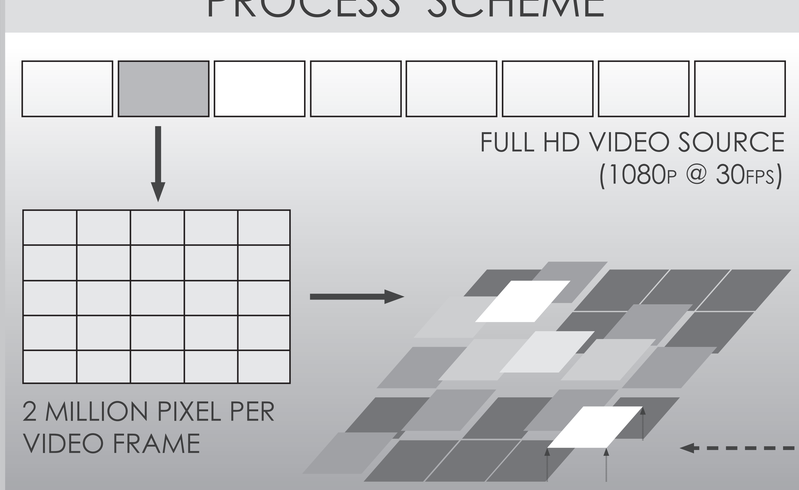
2 MILLION PIXEL EXPERIMENT

MAPPING A 1080P VIDEO SOURCE INTO 3D SPACE IN REALTIME

DESCRIPTION OF CONCEPT

2 MILLION PIXEL EXPERIMENT has been created as a piece of computational art using visual computing technologies. Developed as an software application this experiment maps a full HD video source (1920 x 1080 @ 30 fps) into 3D space. Each video frame is processed in realtime by a CUDA kernel. Every pixel in a frame is scaled in the Z-axis by its luminance value and given its original color from the video source.
This C#/.NET application uses DirectShow.NET for frame grabbing, SlimDX.NET for DirectX11 rendering and CUDA.NET for interfacing NVIDIA CUDA. Fast graphics interoperability functions are used to reduce redundant memory transfers.

PROCESS SCHEME



IMPLEMENTATION DETAILS

VIDEO PLAYBACK FRAME GRABBER [DIRECTSHOW.NET]

```
// CODE EXCERPT for the frame grabber event at a new frame
void ISampleGrabberCB.BufferCB(double SampleTime, IntPtr pBuffer, int BufferLen) {
    if (cudabuffer != null && (BufferLen == Math.Abs(m_stride) * m_videoHeight))
        cudabuffer.UpdateBytes(byteinput, BufferLen);
}
```

```
// CODE EXCERPT for launching the CUDA KERNEL to update vertices

// Copy parameters and the bytes of the last video frame to device
cuda.CopyHostToDevice<float>(memparams, parametervalues);
cuda.CopyHostToDevice<byte>(meminput, byteinput);

// Map the DirectX vertices to a CUDA array as a graphics ressource
// to make use of fast DirectX / CUDA interoperability
cuda.MapGraphicsResources(resource);
mem = cuda.GetGraphicsResourceMappedPointer(resource[0]);

// Set kernel parameters and launch
cuda.SetParameter(kernel, IntPtr.Size * 0, mem.Pointer);
cuda.SetParameter(kernel, IntPtr.Size * 1, memparams.Pointer);
cuda.SetParameter(kernel, IntPtr.Size * 2, meminput.Pointer);
cuda.SetParameterSize(kernel, (uint)(IntPtr.Size * 3));
cuda.SetFunctionBlockShape(compute, sizeBlock[0], sizeBlock[1], 1);
cuda.Launch(kernel, sizeGrid[0], sizeGrid[1]);

// Unmap the graphics ressource
cuda.UnmapGraphicsResources(resource);
```

GRAPHICS INTEROPERABILITY DIRECTX / CUDA

CUDA KERNEL VERTEX MAPPING [CUDA.NET]

RENDER LOOP DISPLAY RESULTS [SLIMDX.NET]

```
// CODE EXCERPT for the CUDA KERNEL*
// to achieve the vertex mapping according to a video frame

extern „C" __global__ void compute(
    float* OUTPUT,
    float* PARAMS,
    unsigned char* BYTES)
{
    unsigned int x = blockIdx.x*blockDim.x + threadIdx.x;
    unsigned int y = blockIdx.y*blockDim.y + threadIdx.y;
    unsigned int w = __float2uint_rd(PARAMS[1]);
    unsigned int h = __float2uint_rd(PARAMS[2]);
    unsigned int s = 8;
    float wf = PARAMS[1];
    float hf = PARAMS[2];

    float aspect = wf/hf;
    float u = (((float)x / wf)-0.5f)*2.0f*aspect;
    float v = (((float)y / hf)-0.5f)*2.0f;
    float z = 0.0f;
    float rgb[3] = {0.0f, 0.0f, 0.0f};
    for(unsigned int i=0; i<3; i++)
    {
        float bfloat = __uint2float_rd(
            (unsigned int)BYTES[y*w*3 + (w-x)*3 + i]);
        rgb[i] = bfloat / 255.0f;
        z += rgb[i];
    }
    z /= 3.0f;

    OUTPUT[y*w*s + x*s + 0] = u;
    OUTPUT[y*w*s + x*s + 1] = v;
    OUTPUT[y*w*s + x*s + 2] = z;
    OUTPUT[y*w*s + x*s + 3] = 1.0f;

    for(unsigned int i=4; i<7; i++)
        OUTPUT[y*w*s + x*s + i] = rgb[2-(i-4)];
    OUTPUT[y*w*s + x*s + 7] = z;
}
```

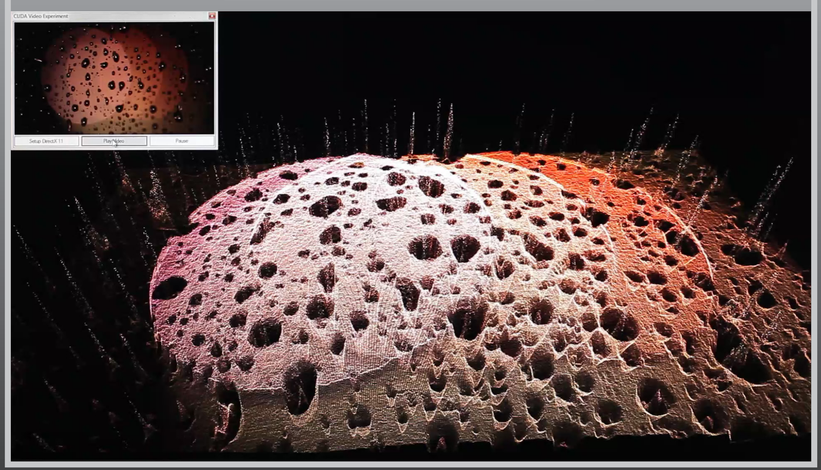
KERNEL INITIALIZATION: SIZE PARAMETERS

MAPPING VERTICES INTEROP POSITION WITH COLOR SCALED Z-VALUE

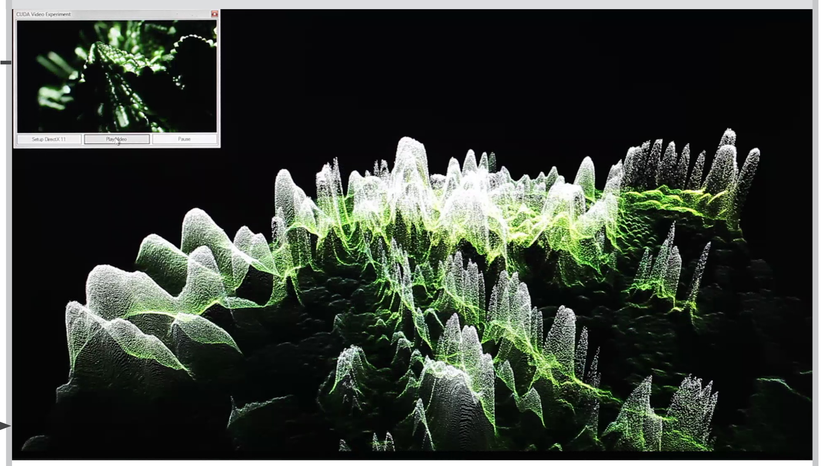
COLORING VERTICES FROM THE VIDEO FRAMES

SPACE MOUSE CONTROLLING VIRTUAL CAMERA

* ALL CODE EXCERPTS COULD BE FURTHER OPTIMIZED



POSSIBLE PURPOSES



By mapping a video source into 3D space, each pixel can be displayed highly differentiated. This approach could be utilized for checking video quality on lower pixel level like on a realtime video waveform monitor. For digital content creation this experiment may provide new possibilities for particle-based visual effects that allow to dissolve a video source into a highly dynamic 3D point cloud.

CONTACT INFORMATION

Author: © Philipp Drieger 2010 - 2012
Project website: VISUALCOMPUTE.COM
Company website: NOUMENTALIA.COM

NOUMENTALIA is a digital arts company that offers innovative IT services, rapid software development with focus on visual computing technologies, sophisticated IT consulting as well as fine art content creation.