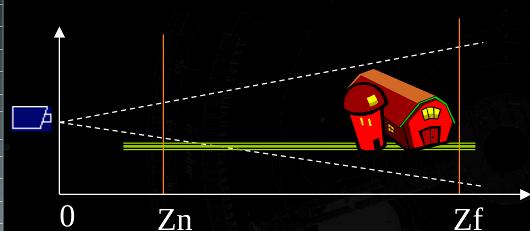
Quasi-linear depth buffers with variable resolution

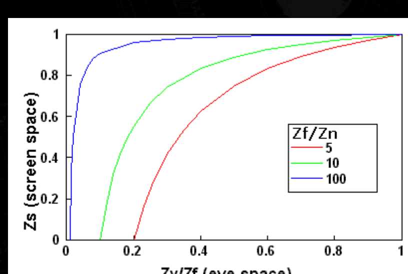
Eugene Lapidous, Guofang Jiao, Jianbo Zhang, Timothy Wilson

Trident Microsystems Inc.

Most popular depth buffer: screen Z



$$Zv \in [Zn, Zf] \Rightarrow Zs \in [0,1]$$



$$Zs = \frac{Zf}{Zf - Zn} * \left(1 - \frac{Zn}{Zv}\right)$$

- preserves lines and planes,
- full range [0,1],
- dZv/dZs > 0

24-bit/pixel screen Z problems

Water (large flat polygon) intersects terrain to form a lake

Trident

Linear and quasi-linear depth buffers: same size, better precision

1. W, integer storage: $W = \dfrac{Zv}{Zf}$

2. rhW, floating point storage: $rhW = \dfrac{Zn}{Zv}$

3. Complementary Z, floating point storage:

$$Zc = 1 - screenZ = \frac{Zn}{Zf - Zn} * \left(\frac{Zf}{Zv} - 1\right)$$

Trident

Why linear depth buffers aren't popular?

- **If 3D application looks fine with 24-bit screen Z, linear depth buffer doesn't make it look better**
  - Most applications benefit from better color: 32bits, sRGB
  - Only applications that don't look right with screen Z need better depth precision: high risk, low volume
  - No visible change – no reason for HW support
- **Motivation: make linear depth buffers benefit majority of 3D applications!**

Trident

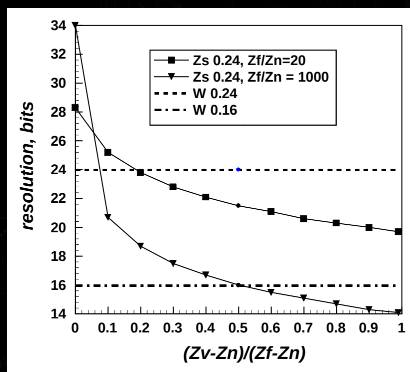
Goals

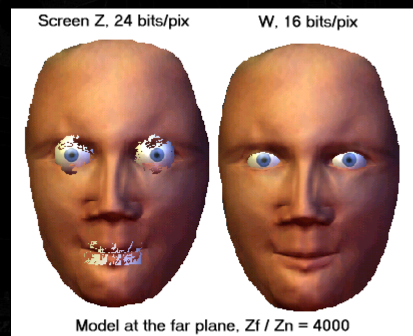
- **Create new class of depth buffers with variable precision-performance balance**
  - Fine-tuning: allow small changes in effective precision
  - Real-time corrections: per application, per frame
- **Define new ways to optimize 3D applications**
  - Measure and modify depth resolution profiles

Trident

Resolution profiles : screen Z vs. W

$$R, bits = \log2\left(\frac{Zf - Zn}{|Zv(Zd + dZd) - Zv(Zd)|}\right)$$

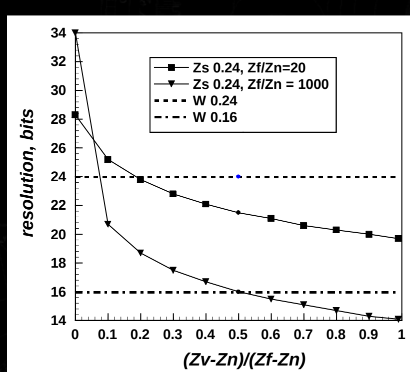




Trident

What depth resolution profile can replace 24-bit/pix screen Z?

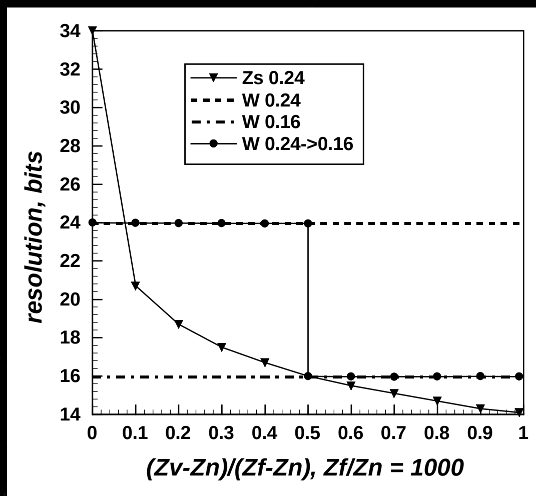
$$R, bits >= \min(24\ bits, R[screenZ])$$



- 24-bit/pix screen Z is most popular
- 24-bit/pix W is a quality reference standard
- Shape of screen Z region with resolution > 24 bits varies with Zf/Zn

Trident

Multi-resolution W buffer



If (new W > W(Zt))
{read and write 16-bit W;}

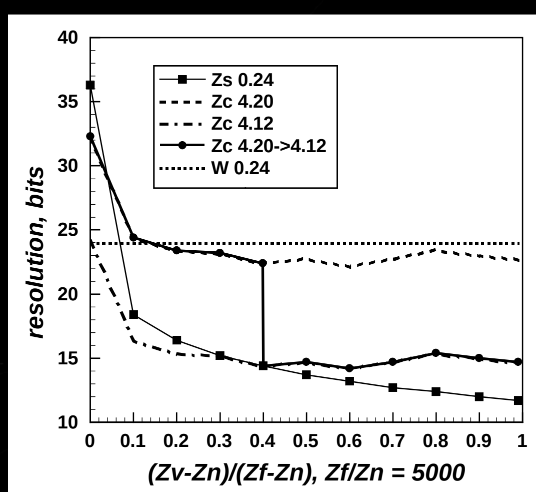
else
{read and write 24-bit W;}

Note:
if (old 24-bit W > W(Zt)),
low 8 bits are undefined,
but we know test result:
(new W < old W).

Main advantage:
entries of different sizes can be read from the same buffer without reading any size flags first!

Trident

Multi-resolution complementary Z

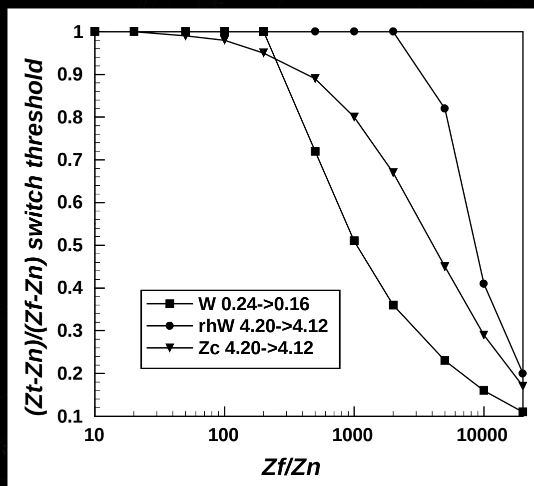


Non-linearity of (1-screenZ) mapping at Zv->Zf is compensated by increase of the floating-point precision at Zd -> 0.

Composite profile is always above 24 bits or screen Z

Trident

When 16-bit/pix depth resolution is better than of 24-bit/pix screen Z?

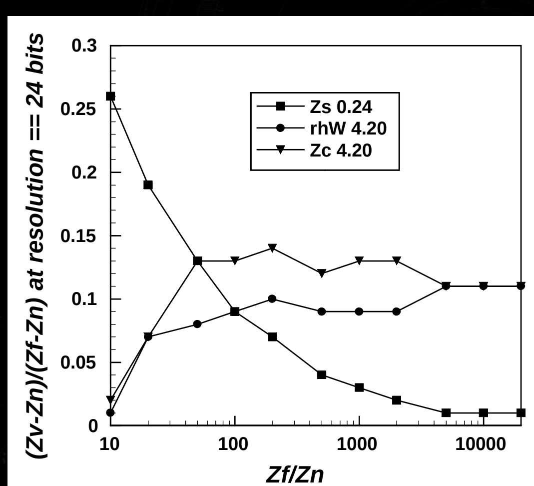


Dynamic ratio suitable for multi-resolution support:

Zf/Zn > 300..500 for W, complementary Z

Zf/Zn > 3000 for rhW

Is 24-bit quasi-linear depth resolution always above 24 bits or screen Z?

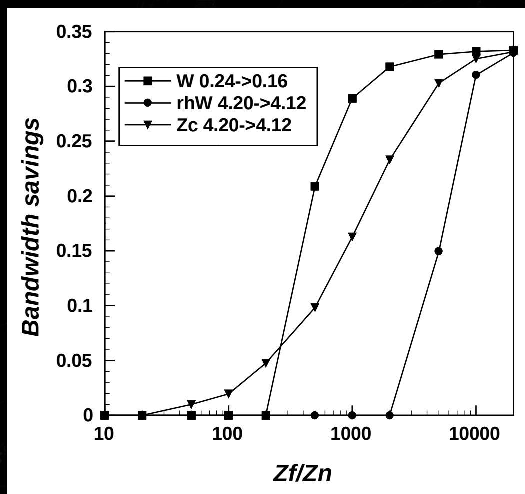


YES if Zf/Zn > 100 !

Decrease of resolution with distance is faster for screen Z than for quasi-linear buffers:

It will stay lower if drops below 24 bits at smaller distance to camera

Bandwidth savings for screenZ-optimized 3D applications



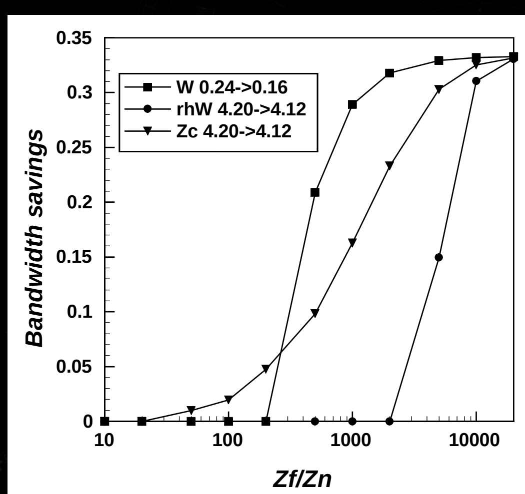
High 16 bits, low 8 bits stored separately for pixels groups:
max saving 33% at Zt = Zn

Sample Zc and W savings:

S = 0.24(Zc)..0.32(W)
at Zf/Zn = 2000

Trident

Which multi-resolution depth buffer is the best?

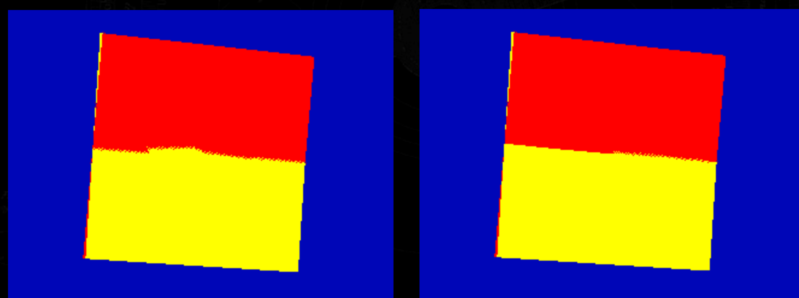


- W: best bandwidth savings, more expensive (1/rhW per pixel). Best for high-end HW.
- rhW: not good enough for Zf/Zn < 7000
- Complementary Z: close to W, less expensive. Best for high-volume HW.

Trident

Artifacts of depth resolution switch at small-angle intersections

Screen Z, 24-bit/pix   W, 24 -> 16-bit/pix

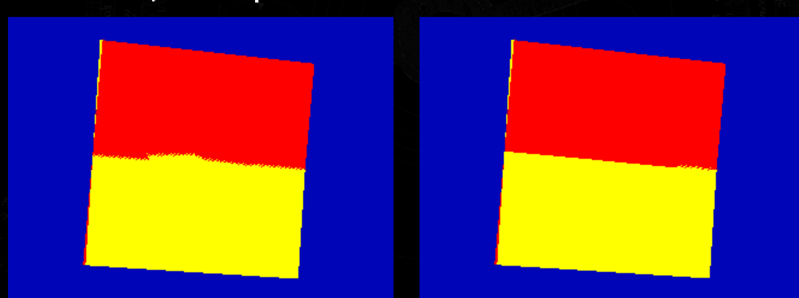


Switch is visible only if application accepts screen Z errors.
Goal: hide switch point.

Trident

Dynamic adjustment of the depth resolution switch threshold

Screen Z, 24-bit/pix   W, 24 -> 16-bit/pix



Motion-based: hash function of object or vertex position.

Trident

Next: application-level control of the depth resolution profile

- Best resolution for every application
  - Any depth resolution profile from Zt = Zn (16 bits/pixel) to Zt = Zf (24 bits/pixel)
  - Multiple windows with different depth resolutions on the same screen
- Dynamic resolution selection
  - Adjustments per scene, user action, field of view.
  - Frame rate control
- Need API hint: desired depth profile.
  - Driver will select buffer type and switch distance

Trident

Conclusions

- New class of depth buffers: multi-resolution storage without size flags
  - Makes linear and quasi-linear buffers useful for screenZ-optimized 3D applications
  - Can save 20..33% of Z bandwidth at Zf/Zn > 500
  - W buffer is best for high-end hardware, complementary Z buffer is recommended for high-volume products.
- New ways to optimize 3D applications: depth resolution profile management
  - Adjustments per application, scene, frame, object
  - 3D API extension request: depth profile hint

Trident


Thank you!
Trident