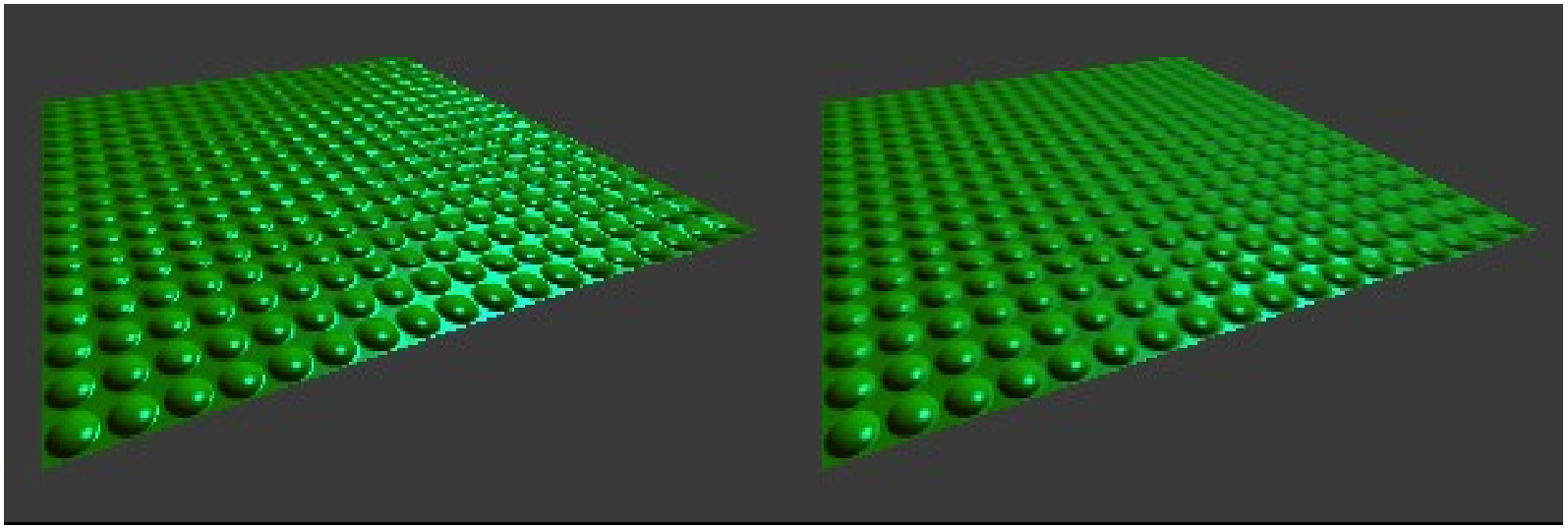
Mipmapping Normal Maps

- Technical brief by Michael Toksvig of Nvidia.
- Published in April 2004.
- Covers Geforce3 / Radeon 8500 generation hardware.
  - Requires DOT3 texture combine.
  - Requires dependent texturing.

Background

- Traditional mipmap calculations fail on normal maps.
  - Averaging multiple normals together gives a vector less than unit length, which results in lighting artifacts.



Theory

- By knowing how much shorter the averaged normal is, we can improve the lighting.
  - The paper suggests creating a table of scale factors and shininess exponent modifiers.
  - The new lighting equation becomes, where N_a is the averaged normal and s is the shininess exponent:

$$f_t = \frac{|N_a|}{|N_a| + s\left(1 - |N_a|\right)}, K_s = \frac{1 + f_t s}{1 + s}\left(\frac{N_a \cdot H}{|N_a|}\right)^{f_t s}$$

Implementation

- Since $N_a \cdot N_a$ and $N_a \cdot H$ must be between 0 and 1, all possible values of each can be plugged into the preceding equation and stored in a 2D texture.
  - The texture is accessed using $N_a \cdot N_a$ and $N_a \cdot H$ as texture coordinates.
  - Note that N_a is **not** normalized.

Questions?

Legal Statement

- This work represents the view of the authors and does not necessarily represent the view of IBM or the Art Institute of Portland.
- OpenGL is a trademark of Silicon Graphics, Inc. in the United States, other countries, or both.
- Khronos and OpenGL ES are trademarks of the Khronos Group.
- Other company, product, and service names may be trademarks or service marks of others.