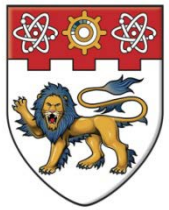


NANYANG TECHNOLOGICAL UNIVERSITY

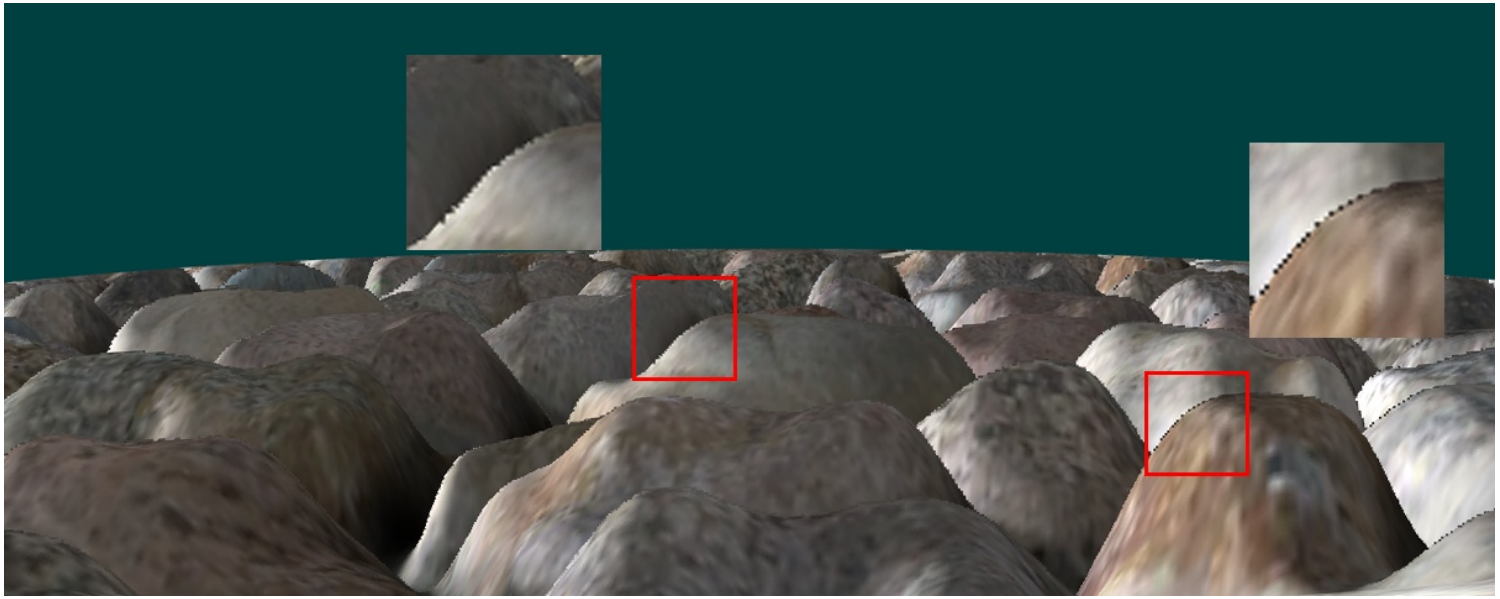
3D Chinese ink painting: texture antialiasing with diffusion equations

Presenter: Cheng Peng
School of Computer Engineering & Institute of Media Innovation
Supervisors: Miao Chunyan, Nadia Thalmann

Outline

- Introduction
- Texture antialiasing
- Combined anisotropic diffusion filters
- Experiments and analysis

aliasing



Introduction

- Why texture?

  Computer graphics applications often use surface textures to give the illusion of fine detail without explicit geometric modeling.

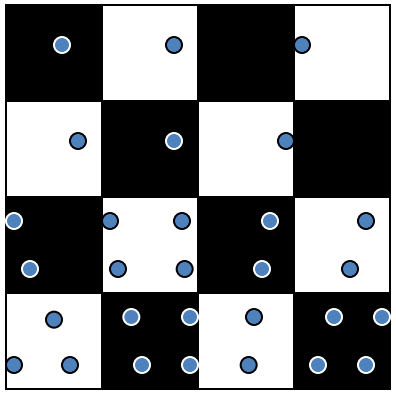
- What is and why synthesis?

  Use one or more example textures to texture the surface. It is a flexible way to generate textures with any size.

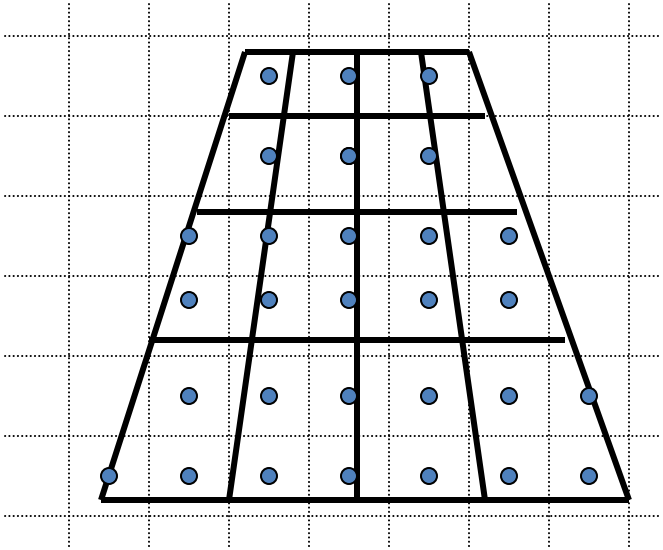
Texture and aliasing

- Textures are subject to aliasing
  - Sampling texture as point when mapping polygon pixels into texture image
  - Image wrap defined by the mapping and projection
- Approaches
  - Pre-filtering: filter the texture down before use it
  - Post-filtering: take multi-pixels from texture and filter them before applying to polygon fragment

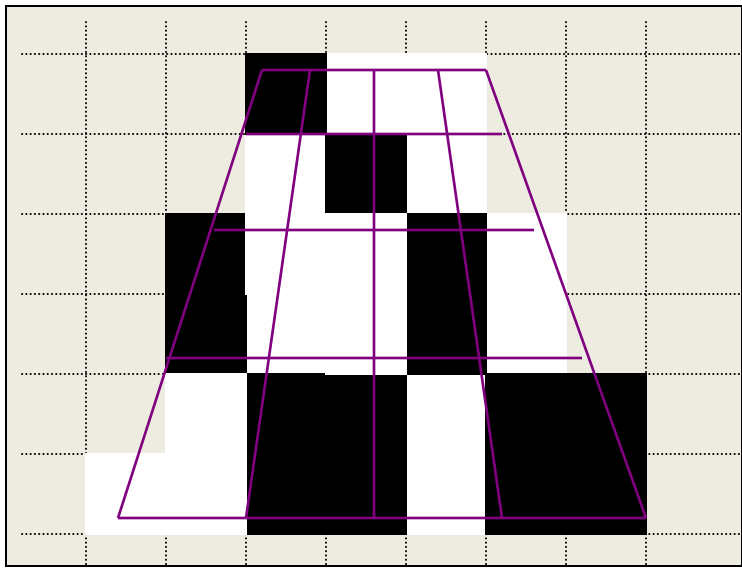
Point sampled texture aliasing



Texture map



Polygon far from the viewer
in perspective projection



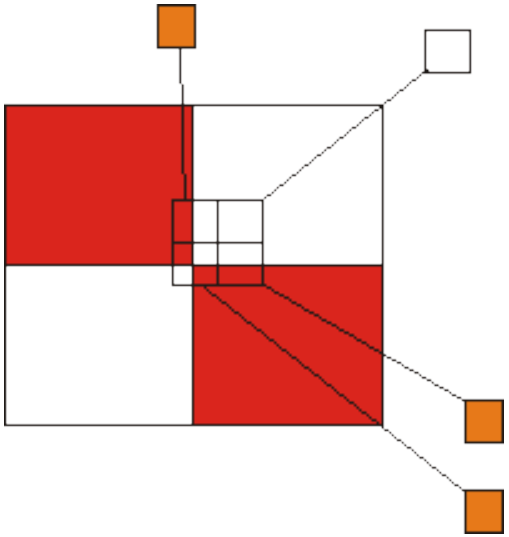
Rasterized and textured

- Note that the back row is a very poor representation of the true image

Outline

- Introduction
- Texture antialiasing
- Combined anisotropic diffusion filters
- Experiments and analysis

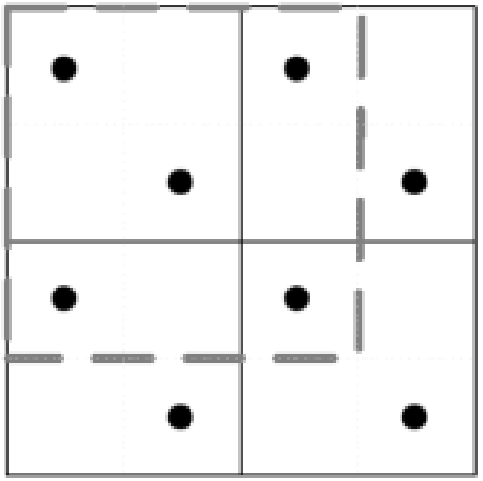
Supersampling



- taking the corners of each pixel and creating what would be the average colour

- effectively smudging the image and averaging out the colour along a curve

Multisampling



taking multiple samples for each pixel

Taking into account the colours around the pixel

happens when a pixel is covered by more than one colour

Filters

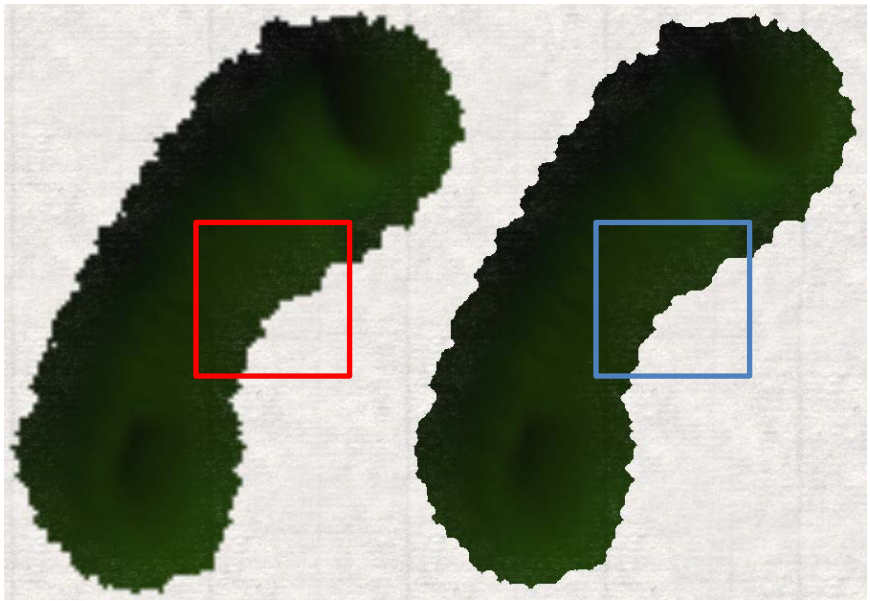
- Linear filters
- Nonlinear filters
- Anisotropic filters
-
- Texture filtering is mature
  - Fast on graphics hardware
  - Anisotropic filtering is available

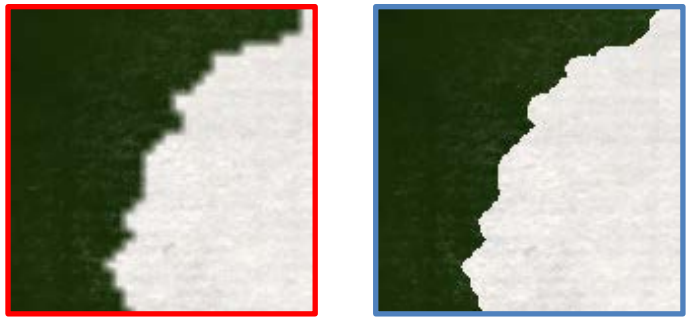
Outline

- Introduction
- Texture antialiasing
- Combined anisotropic diffusion filters
- Experiments and analysis

ink painting aliasing

- Hardware interpolation
- Color map for high resolution (shimmering)
- Texture aliasing intensified





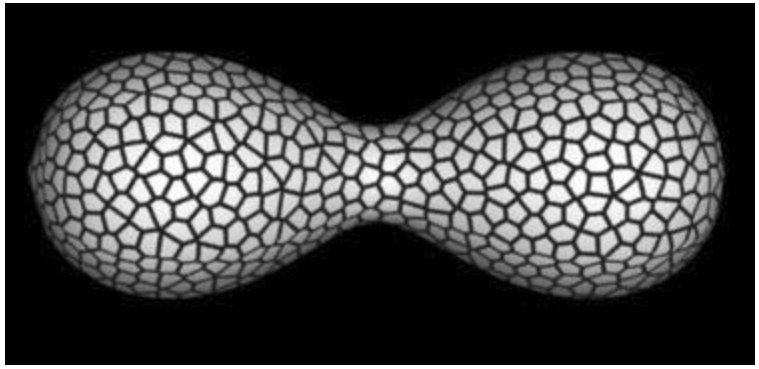
Geometry aliasing

- Surface geometry details
- Ink interaction between brush and object surface

- Model geometry into texture (anisotropic filter)
- Employ hardware antialiasing benefits

Reaction-diffusion on surface

- Turk's method[Turk1991]:

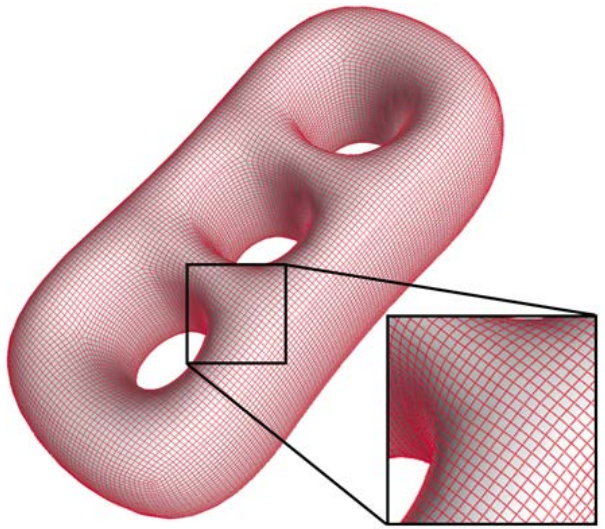


Sparkle a number of points onto the surface and jitter these points until they are evenly distributed. Turk uses these points as the rd cell.

- Lo's method [Lo2007]:

Use quad parameterization to remesh the surface and regard each quad as a rd cell

Easy to extend to GPU



Linear model

$$\begin{cases} u_t = \triangle_M u, \\ \frac{\partial u}{\partial \vec{n}} |_{\partial M} = 0, \\ u(p,0) = f(p), \end{cases}$$

M: polygon mesh

u(p): piecewise linear function

∇_M: gradient operator

Δ_M: Laplace-Beltrami operator

f(p): initial function

Nonlinear model

$$\begin{cases} u_t = \nabla_M \cdot (g(|\nabla_M u|)\nabla_M u), \\ \frac{\partial u}{\partial \vec{n}}|_{\partial M} = 0, \\ u(p,0) = f(p), \end{cases}$$

Taking $g(s) = \frac{1}{\sqrt{s^2+\beta}}$,

Degenerates to total variation model

Anisotropic model

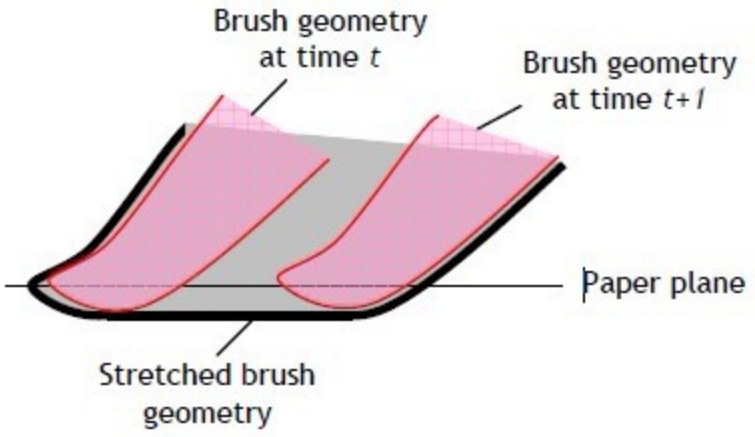
$$\begin{cases} u_t = \nabla_M \cdot (g_1(\nabla_M u \cdot e_1)e_1 + g_2(\nabla_M u \cdot e_2)e_2), \\ (g_1(\nabla_M u \cdot e_1)e_1 + g_2(\nabla_M u \cdot e_2)e_2) \cdot \vec{n}|_{\partial M} = 0, \\ u(p,0) = f(p). \end{cases}$$

Choosing appropriate functions g_1 and g_2

Indicating differential geometry over the neighborhood

Combined reaction-diffusion

- Object surface geometry
  - Principle directions and curvatures
- Brush surface
  - Pressure, tilt, orientation
- Ink concentration
  - Concentration



- "add" brush and ink dimensions to object surface

Numerical discretization

$$
\begin{cases}
\frac{\partial u_1}{\partial t} = \nabla_M \cdot \left(g_{11}(|\nabla_M u_1|)(\nabla_M u_1 \cdot e_1)e_1 + g_{12}(|\nabla_M u_1|)\right. \\
\left.(\nabla_M u_1 \cdot e_2)e_2\right) + F(u_1, u_2), \\
\frac{\partial u_2}{\partial t} = \nabla_M \cdot \left(g_{21}(|\nabla_M u_2|)(\nabla_M u_2 \cdot e_1)e_1 + g_{22}(|\nabla_M u_2|)\right. \\
\left.(\nabla_M u_2 \cdot e_2)e_2\right) + G(u_1, u_2), \\
(g_{11}(|\nabla_M u_1|)(\nabla_M u_1 \cdot e_1)e_1 + g_{12}(|\nabla_M u_1|)(\nabla_M u_1 \cdot e_2)e_2) \\
\cdot \vec{n}|_{\partial M} = 0, \\
(g_{21}(|\nabla_M u_2\|)(\nabla_M u_2 \cdot e_1)e_1 + g_{22}(|\nabla_M u_2|)(\nabla_M u_2 \cdot e_2)e_2) \\
\cdot \vec{n}|_{\partial M} = 0,
\end{cases}
$$

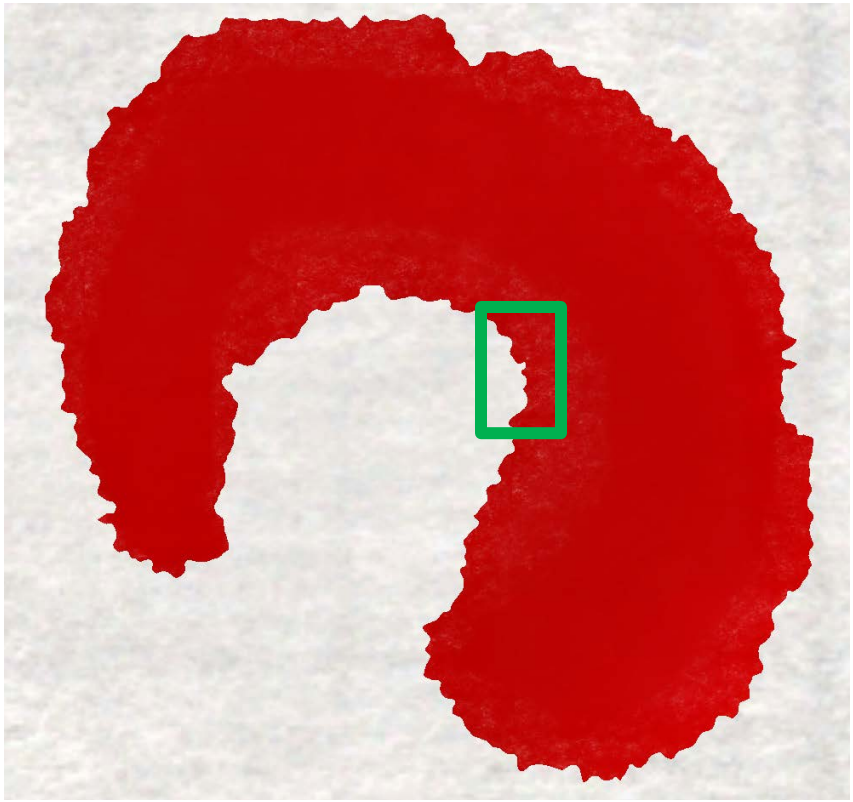
e^1 and e^2 : principle directions

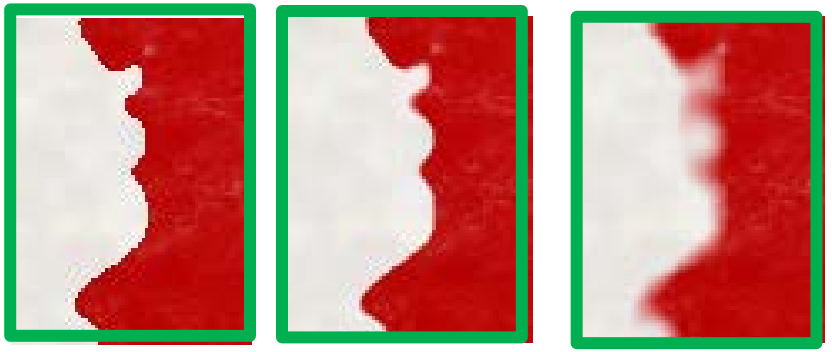
g_{ij} : curvatures indicating diffusion rates

Outline

- Introduction
- Texture antialiasing
- Combined anisotropic diffusion filters
- Experiments and analysis

Simple antialiasing results





a=0 a=0.05 a=0.3

Properties of combined anisotropic filtering

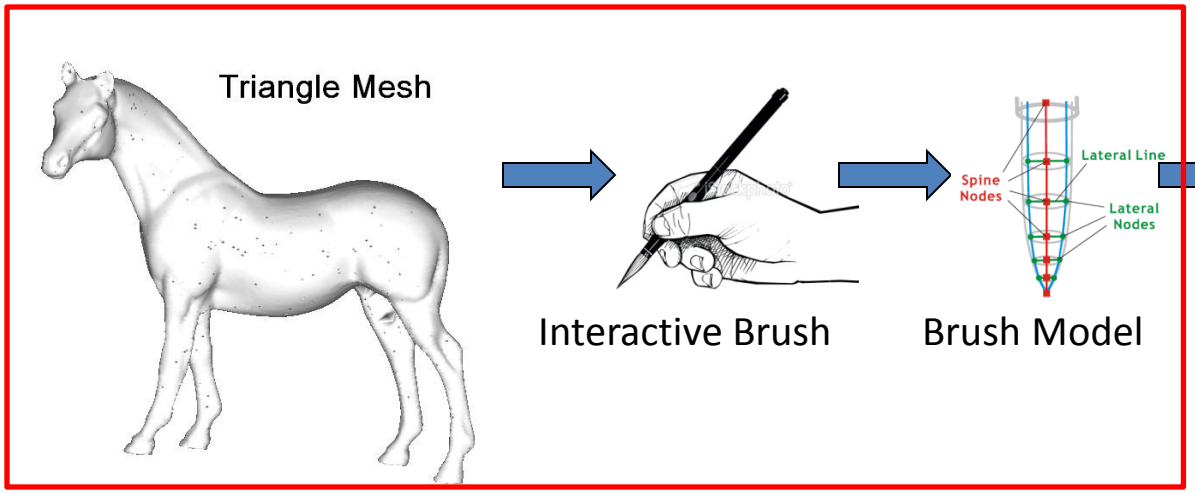
- Reduce color shimmering while keep color map
- Sharpen boundaries without “staircase”
- Flexible to work with arbitrary object surface and pigment

Limitations of diffusion equations

- Difficult to use: Unknown equation and parameters
  - Keep trying
- Too slow: Takes 3-4 secs to complete on a moderate-sized model
  - Localized reaction-diffusion [2008]
- Surface parameterization bound
  - Tetrahedral mesh [Takayama 2008]

Thank You.

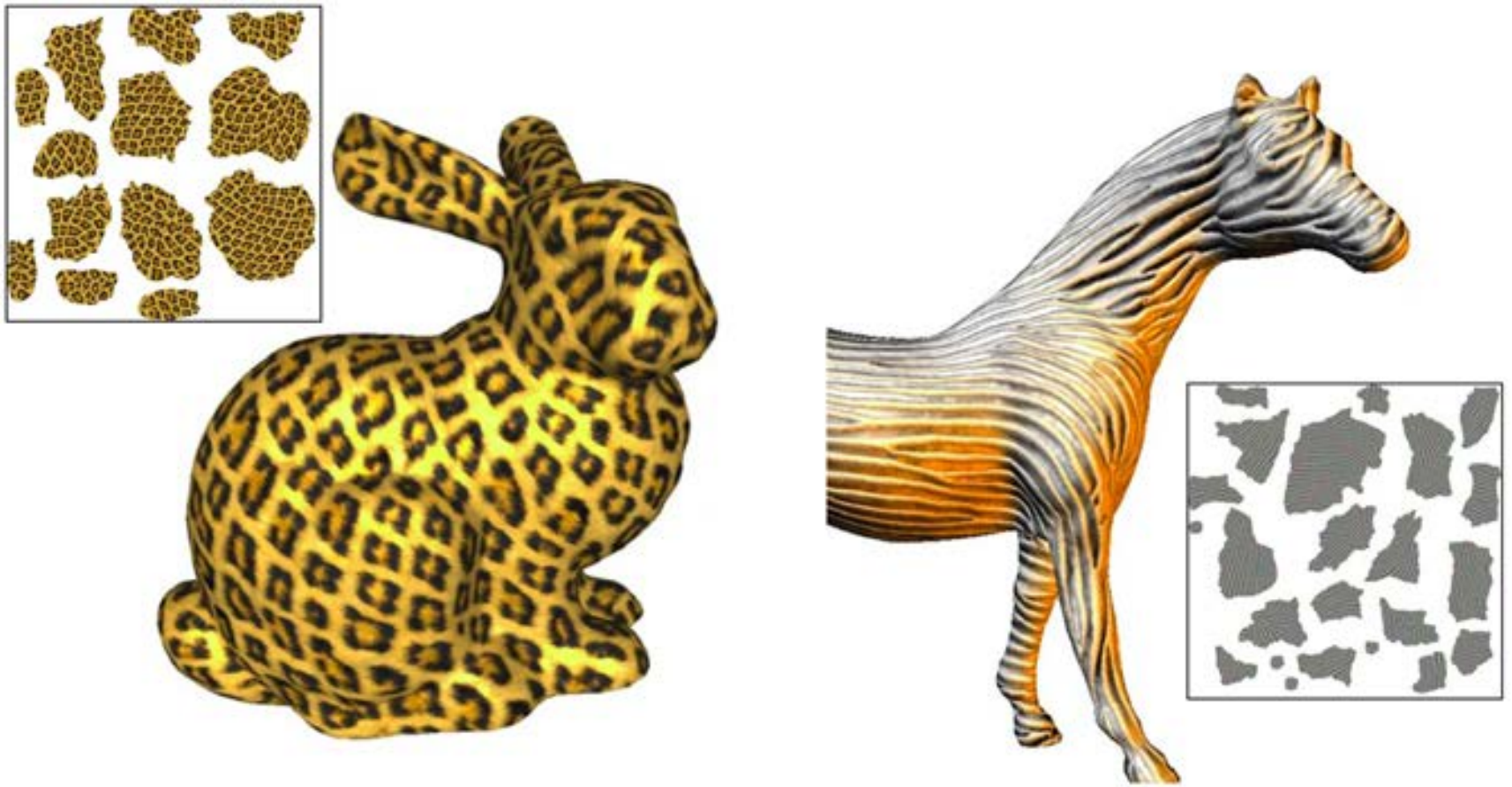
3D Chinese ink painting



institute for media innovation
The Avant-Garde Place...

NANYANG TECHNOLOGICAL UNIVERSITY

Texture mapping



Texture mapping contains two phases, surface parameterization that maps texture space to object space and followed by standard modeling and viewing transformation that map object space to screen space.

Reaction-diffusion

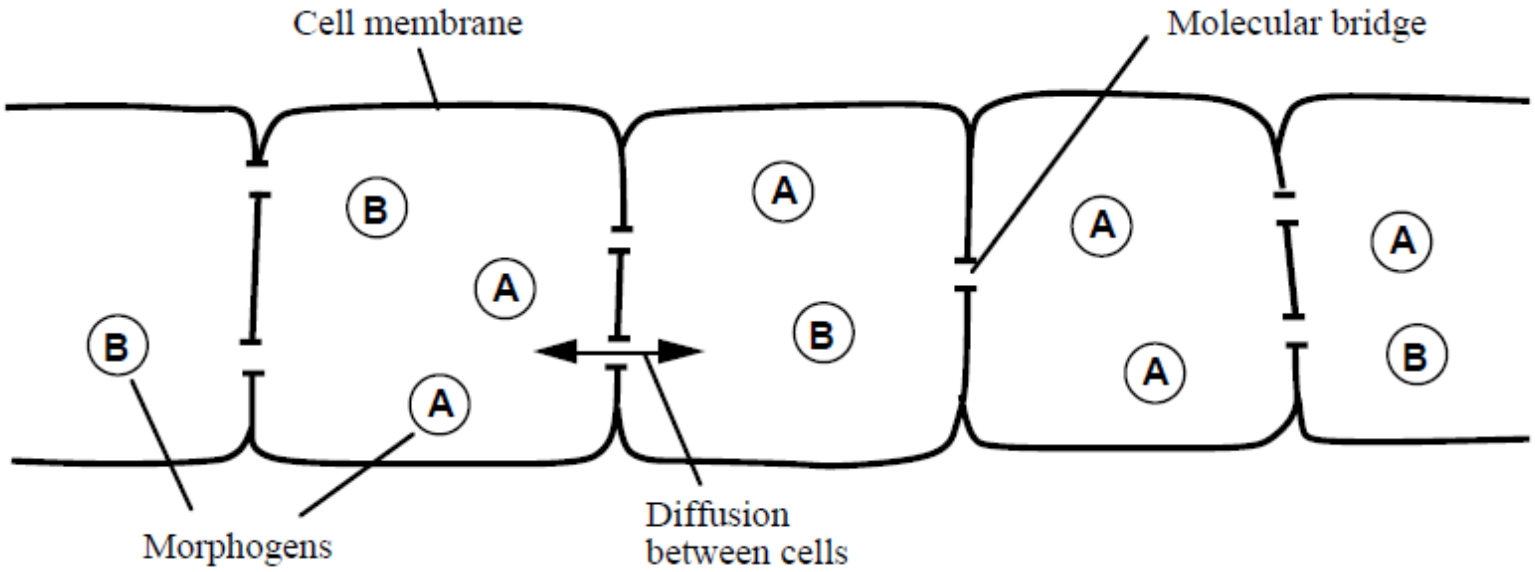
reaction within one cell

diffusion between every neighboring cells

$$\frac{\partial a}{\partial t} = F(a,b) + D_a \nabla^2 a$$

$$\frac{\partial b}{\partial t} = G(a,b) + D_b \nabla^2 b$$

[Turing 1952]



Reaction-diffusion results

- Results