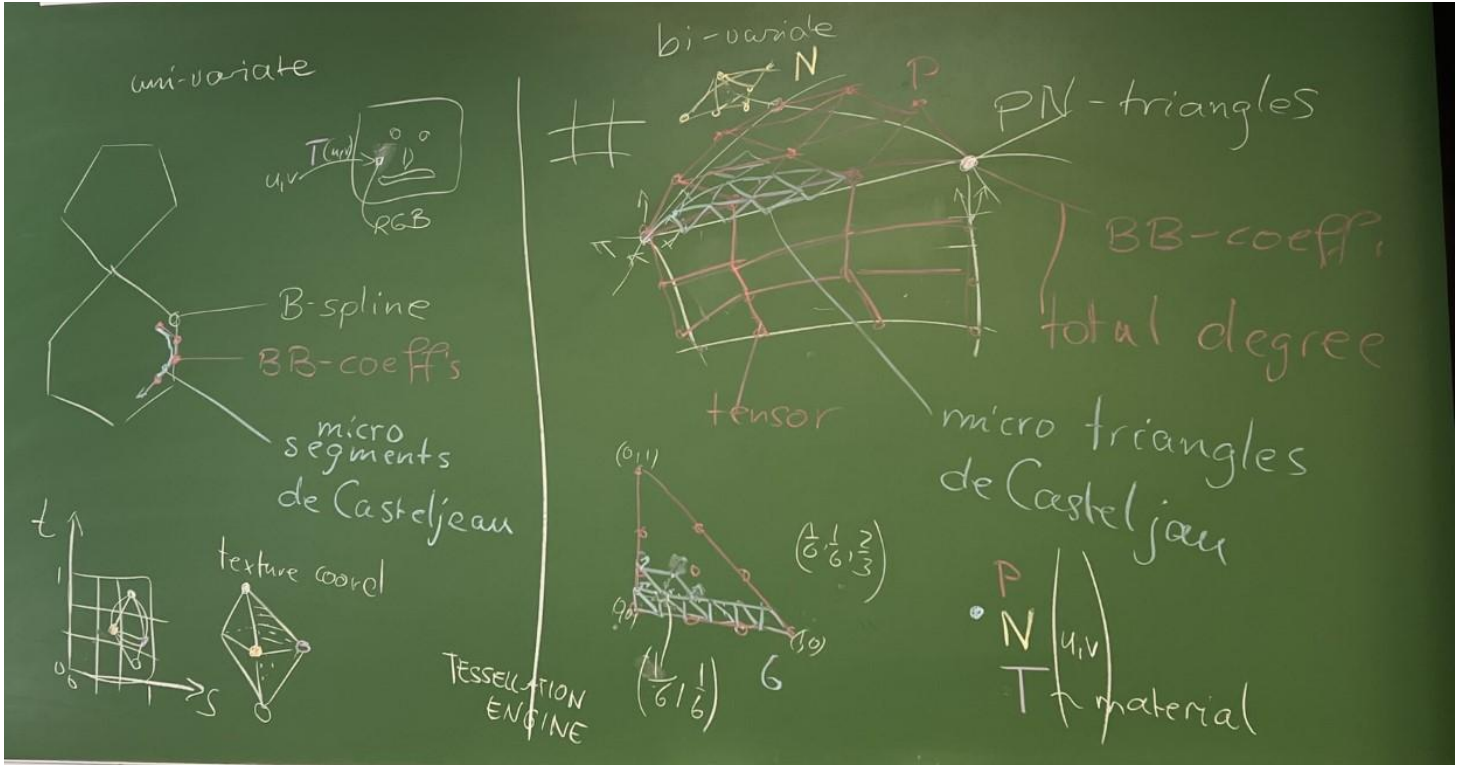
# Textures




uni-variate
T(u,v)
u,v
RGB
B-spline
BB-coeff's
micro segments de Casteljeau
t
texture coord
s
TESSELLATION ENGINE
bi-variate
N
P
PN-triangles
BB-coeff's
total degree
tensor
micro triangles de Casteljeau
(0,1)
(1/6, 1/6, 2/3)
(1/6, 1/6)
6
P
N
T
u,v
material

# Textures



Fragment shader:
Textures, bump maps, normal maps, parallax mapping

Vertex shader: Displacement maps

# Textures



bump maps

imitate $\mathbf{p}^{\text{new}} = \mathbf{p} + d\mathbf{n}$.

$$\frac{\partial \mathbf{p}^{\text{new}}}{\partial u} \times \frac{\partial \mathbf{p}^{\text{new}}}{\partial v} = \left(\frac{\partial \mathbf{p}}{\partial u} + \frac{\partial d}{\partial u}\mathbf{n} + d\frac{\partial \mathbf{n}}{\partial u}\right) \times \left(\frac{\partial \mathbf{p}}{\partial v} + \frac{\partial d}{\partial v}\mathbf{n} + d\frac{\partial \mathbf{n}}{\partial v}\right)$$

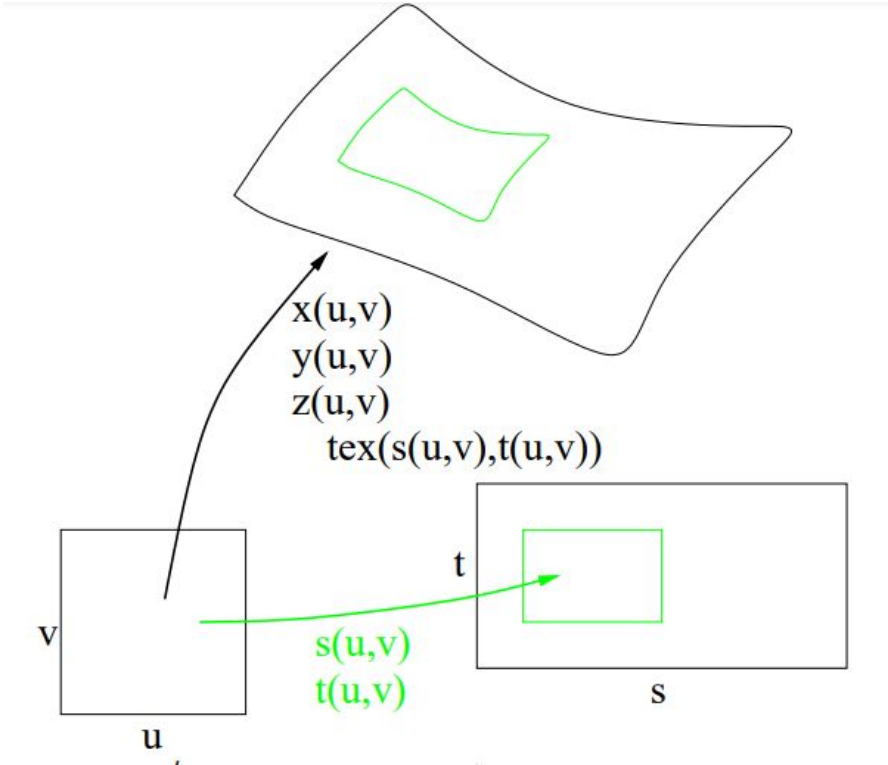
$$= \mathbf{n} + \underbrace{\frac{\partial d}{\partial u}\mathbf{n} \times \frac{\partial \mathbf{p}}{\partial v} - \frac{\partial d}{\partial v}\mathbf{n} \times \frac{\partial \mathbf{p}}{\partial u}}_{\partial \mathbf{n}} + O\left(\frac{\partial \mathbf{n}}{\partial u}, \frac{\partial \mathbf{n}}{\partial v}\right).$$

# Textures



Fragment shader: Texture mapping

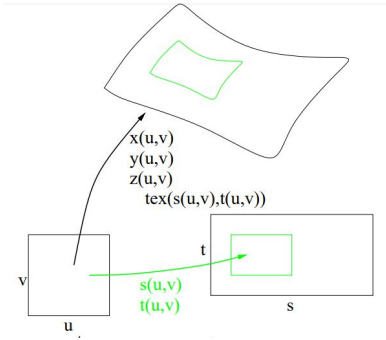
**2D texture:**
pasting an image onto a surface (challenges: distortion and aliasing)



# 2D Texture



**texels** (texture pixels)
fill, by unchangeable default, the unit square
When accessing the texture plane with s, t outside the unit square, texture wrapping rules clip s,t back to the unit square.




Many bit patterns (formats) (gimp exports C-arrays!)

Texture wrapping
GL LINEAR, GL NEAREST
Mipmapping

# Textures



Fragment shader: Texture mapping

**2D texture:** pasting an image onto a surface
**Transfer texture** from an intermediate object (sphere or cylinder) for better parametrization
**Video texture**

# Textures



Fragment shader: Texture mapping

**2D texture:** pasting an image onto a surface

**Environment Map**, **cube map**:
place viewer at object center. Transfer resulting image as texture (possibly via intermediate)

**3D texture**: generated (random), x,y,z direct, discrete grid

# Texture Mapping Challenges



- Distortion (flat → sphere): fundamental!

  Peters projection

  Other projections

# Texture Mapping Challenges



- Distortion
- Want to color pixel
  - map screen coordinates ←→ texture coordinates

# Texture Mapping Challenges



- Distortion
- screen / texture coordinates
- Areas, not points should be mapped →
  *bilinear interpolation*

- Aliasing (Moire pattern)
  - pointwise: might miss, average, smears out