

# Point Normal triangles

## POINT NORMAL TRIANGLES

---

Rick van Veen Laura Baakman  
December 14, 2015  
Advanced Computer Graphics

[Rick] Welcome everybody. Tell people that PN means Point Normal triangles.

# Point Normal triangles



**[Rick]** Why PN triangles? Look at the nice result it gives :-) and we will see that it easy to extend it to the 'existing' pipeline.

# Point Normal triangles

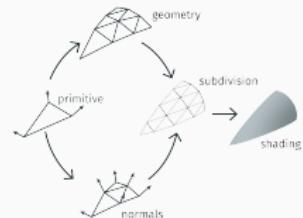
## └ Single PN Triangle

SINGLE PN TRIANGLE

---

[Rick] How does one construct a single PN triangle?

*Overview on the next slide*



2015-12-13

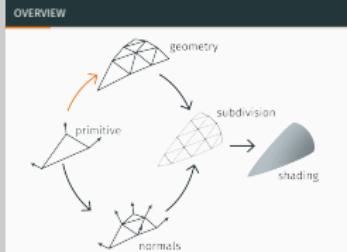
# Point Normal triangles

## └ Single PN Triangle

[Laura] PN triangle is defined by geometry and normal component.

# Point Normal triangles

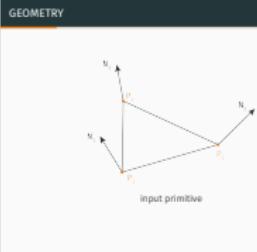
## └ Single PN Triangle



**Laura** From geometric component of input primitive to geometric component of PN triangle

# Point Normal triangles

## └ Single PN Triangle

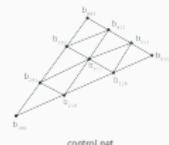


1. **[Laura]** This is a standard triangle primitive, defined by its vertices and normals.
2. **[Laura]** Note that we only have this input primitive, without information about its neighbors.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

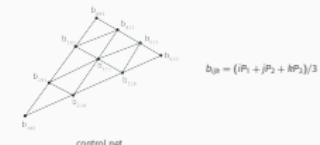


1. [Laura] This is the topology of the final patch
2. [Laura] Mention which vertices are vertex, tangent and center coefficient.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

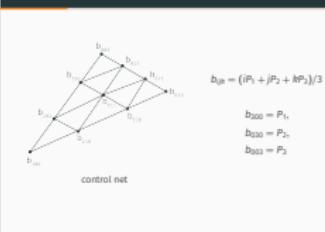


1. [Laura] These are all the initial control point.  
Evenly divided on the triangle. -> formula
2. [Laura] Nice formula

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

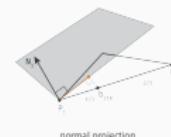


1. [Laura] Stress that the vertex coefficients/control points are the one on the original vertices and that they do not move.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle



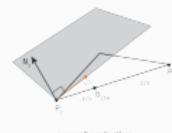
$$\begin{aligned}n_{ij} &= (P_i - P_j) \cdot n_i \in \mathbb{R} \\B_{ij0} &= \frac{n_{ij} + n_{ji}}{2} \\B_{ij1} &= \frac{1}{2} \\B_{ij2} &= \frac{2P_i + P_j - n_{ij}n}{3}\end{aligned}$$

**[Laura]** Define a plane using the closest vertex and its normal. Find the point on this plane that is closest to the uniformly distributed point.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle



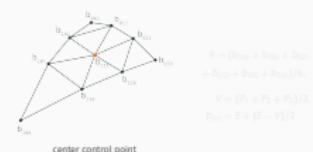
$$w_0 = (P_j - P_i) \cdot N_j \in \mathbb{R}$$
$$b_{210} = \frac{2P_1 + P_2 - w_{12}N_1}{3}$$
$$\vdots$$
$$b_{201} = \frac{2P_1 + P_3 - w_{13}N_1}{3}$$

**[Laura]** Define a plane using the closest vertex and its normal. Find the point on this plane that is closest to the uniformly distributed point.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

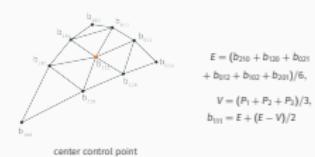


1. [Laura] Note that this is the result of the previous step -> now only center coefficient is left.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

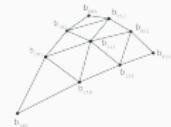


1. [Laura] Average of the tangent coefficients plus half the difference between the tangent and vertex coefficients.

2015-12-13

# Point Normal triangles

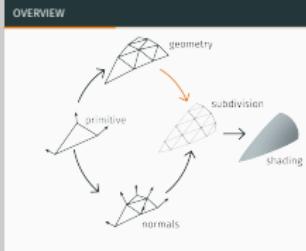
## └ Single PN Triangle



# [Laura] Results

# Point Normal triangles

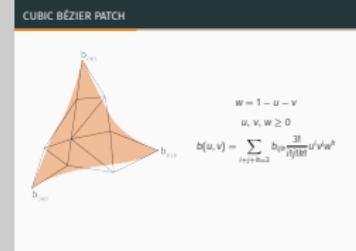
## └ Single PN Triangle



[Rick] Overview -> how to get from this to shading. Sample/subdivide with formula on following slide.

# Point Normal triangles

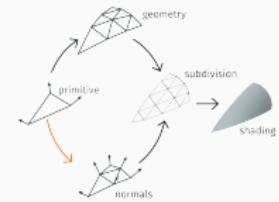
## └ Single PN Triangle



[Rick] Very nice formula with a nice picture.  
Mention **Barycentric coordinates**  $u, v, w$  are a convex combination

# Point Normal triangles

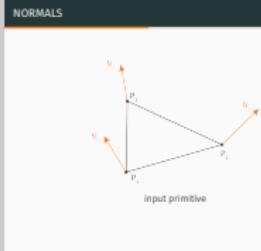
## └ Single PN Triangle



[Rick] From the primitive normals the the PN triangle normals

# Point Normal triangles

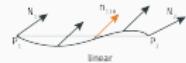
## └ Single PN Triangle



[Rick] Recap input primitive and with emphasis on the normals.

# Point Normal triangles

## └ Single PN Triangle

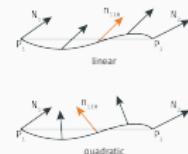


**[Rick]** Stress that there is a need to capture the cubic bezier curve (inflection points) and that this cannot be

2015-12-13

# Point Normal triangles

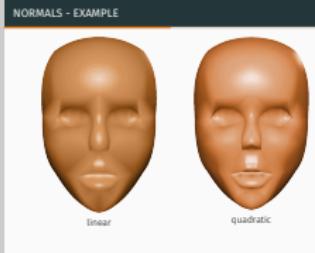
## └ Single PN Triangle



[Rick] Quadratic does capture inflection points.  
Trade off between performance and result  
(maybe?)

# Point Normal triangles

- Single PN Triangle

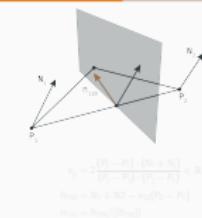


[Rick] Look how pretty.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle

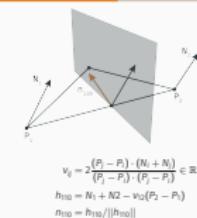


**[Rick]** Formula in words: reflect the averaged normal (average of  $N_1$  and  $N_2$ ) on the plane orthogonal/perpendicular the the edge at the mid point.

2015-12-13

# Point Normal triangles

## └ Single PN Triangle



$$V_0 = 2 \frac{(P_0 - P_i) \cdot (N_i + N_j)}{(P_1 - P_i) \cdot (P_j - P_i)} \in \mathbb{R}$$

$$h_{1|0} = N_1 + N_2 - v_0(P_2 - P_1)$$

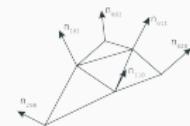
$$n_{1|0} = h_{1|0} / \|h_{1|0}\|$$

**[Rick]** Formula in words: reflect the averaged normal (average of  $N_1$  and  $N_2$ ) on the plane orthogonal/perpendicular the the edge at the mid point.

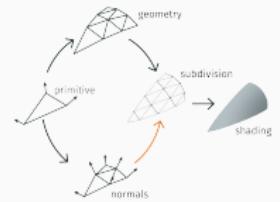
2015-12-13

# Point Normal triangles

## └ Single PN Triangle



[Rick] Result



2015-12-13

# Point Normal triangles

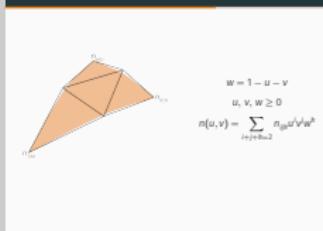
## └ Single PN Triangle

laura

2015-12-13

# Point Normal triangles

## └ Single PN Triangle



$$w = 1 - u - v$$

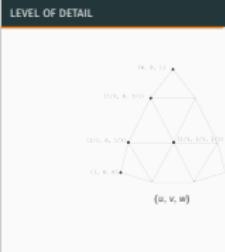
$$u, v, w \geq 0$$

$$n(u, v) = \sum_{i+j+3=2} n_{ij} u^i v^j w^k$$

[Laura]  $u$ ,  $v$  and  $w$  are an convex combination

# Point Normal triangles

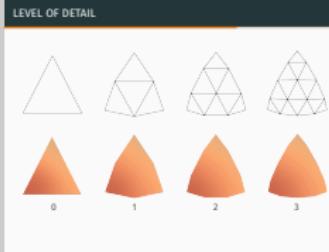
## └ Single PN Triangle



[Laura] LOD = 2

# Point Normal triangles

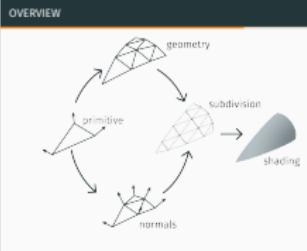
## └ Single PN Triangle



[Laura] Level of detail -> subdivision -> how many triangles go through to the next shaders.

# Point Normal triangles

## └ Single PN Triangle



1. [Laura] Recap of the whole process
2. [Laura] Shading out of the scope of this presentation
3. [Laura] Why quadratic patch for normals, why cubic patch for geometry: We need at least cubic geometry and quadratic normals to capture inflections. There are no additional data to suggest higher degree patches. Simplicity v.s. modeling range

# Point Normal triangles

## └ A Triangle Mesh

A TRIANGLE MESH

---

[rick] blaat

2015-12-13

# Point Normal triangles

## └ A Triangle Mesh

"PN triangles should not deviate too much from the original triangle to preserve the shape and avoid interference with other curved triangles."<sup>1</sup>

<sup>1</sup>slachin2003isurved

**[Rick]** Problem when combining multiple triangles, so this is an important property

2015-12-13

# Point Normal triangles

## └ A Triangle Mesh

PN triangles have:<sup>2</sup>

- $C^1$  continuity in the vertex points
- $C^0$  continuity along the edges
- $C^\infty$  everywhere else

<sup>2</sup>Jan2003parallel

**[Rick]** Continuity  $C^0$  is important -> no gaps.  
Higher is better because this gives a more smooth result.

# Point Normal triangles

## └ A Triangle Mesh



**[Rick]** Curved triangles do not always give the preferred results -> sharp edges. Solution is to insert more triangles at the sharp edges -> model needs to be changed :(

2015-12-13

# Point Normal triangles

## └ A Triangle Mesh



**[Rick]** Beyond the scope of the paper extension exist to overcome the problem what you have when combining multiple meshes. Story about shared vertices.

# Point Normal triangles

## └ Graphics Pipeline

GRAPHICS PIPELINE

---

[Laura] Blaat

2015-12-13

# Point Normal triangles

## └ Graphics Pipeline



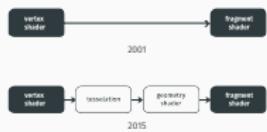
2005

[Laura] Great part of the paper stresses the point that it can easily be implemented as a preprocessing step (CPU.  
2001 pipeline (OpenGL 1.3)

2015-12-13

# Point Normal triangles

## └ Graphics Pipeline



**[Laura]** 2015 we have OpenGL 4.5 with more programmable shaders and the whole process can be done on the GPU. Since PN triangles only uses the primitive, no neighboring primitives, easy in shaders.

# Point Normal triangles

## └ Graphics Pipeline



**[Laura]** Why PN triangles are not suited to rendering for CAD: On real objects, the normal field is determined by the geometry and thus fixed. If you invent fake normals the rendering looks better but the user gets an unpleasant surprise when the product is actually manufactured.

2015-12-13

# Point Normal triangles

## └ Graphics Pipeline