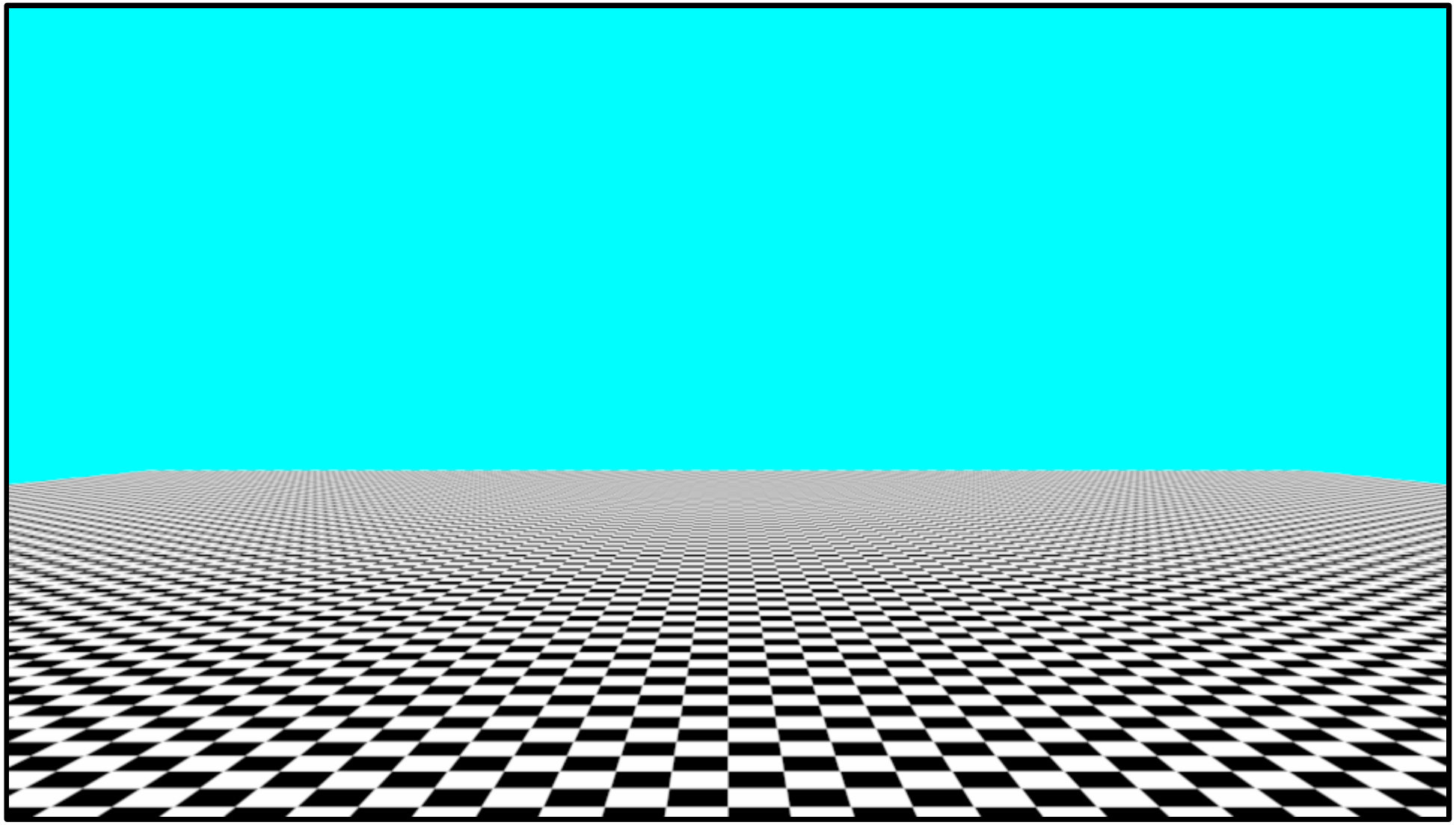


MicroPoint - Non super-sampling aliasing free algorithm -

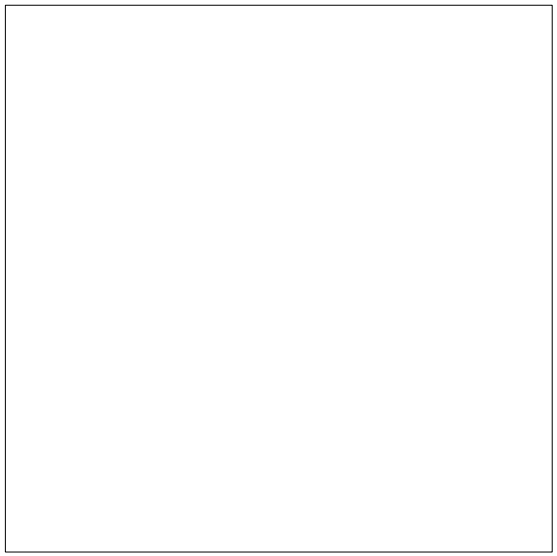
Masao Takakuwa, Koji Iigura

High-frequency component $\xrightarrow{\text{Super-sampling}}$ Pixel

High-frequency component

Super-sampling

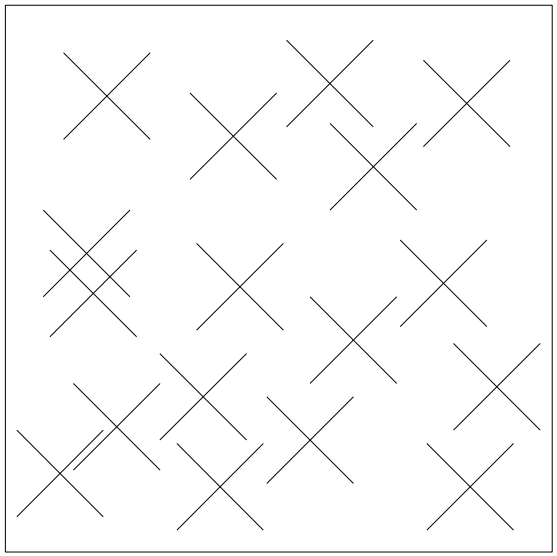
Pixel



High-frequency component

Super-sampling

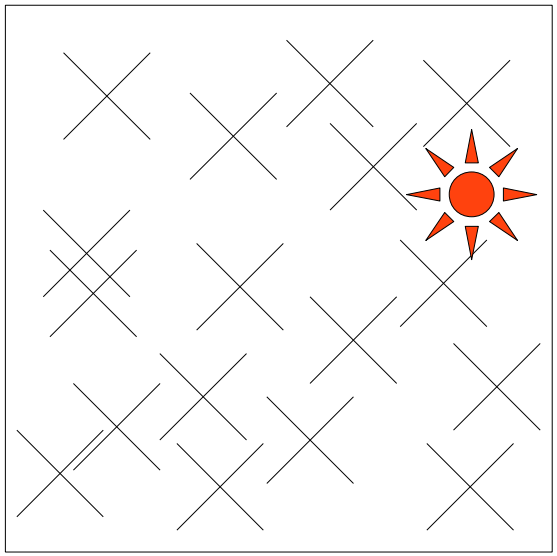
Pixel

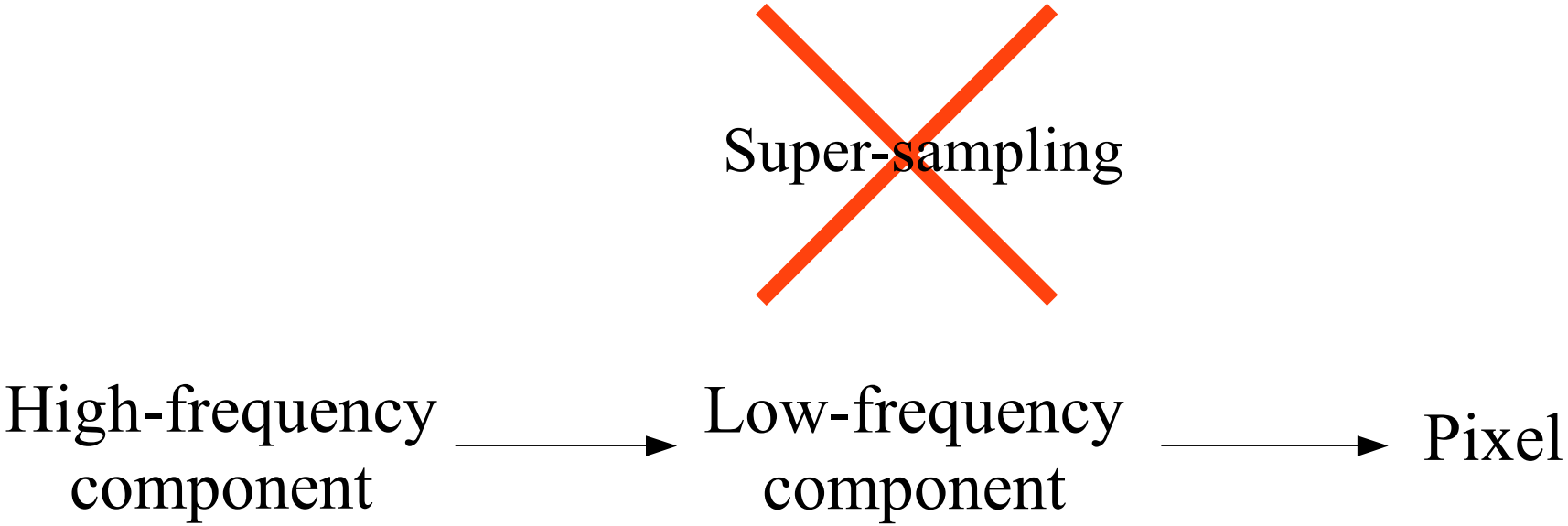


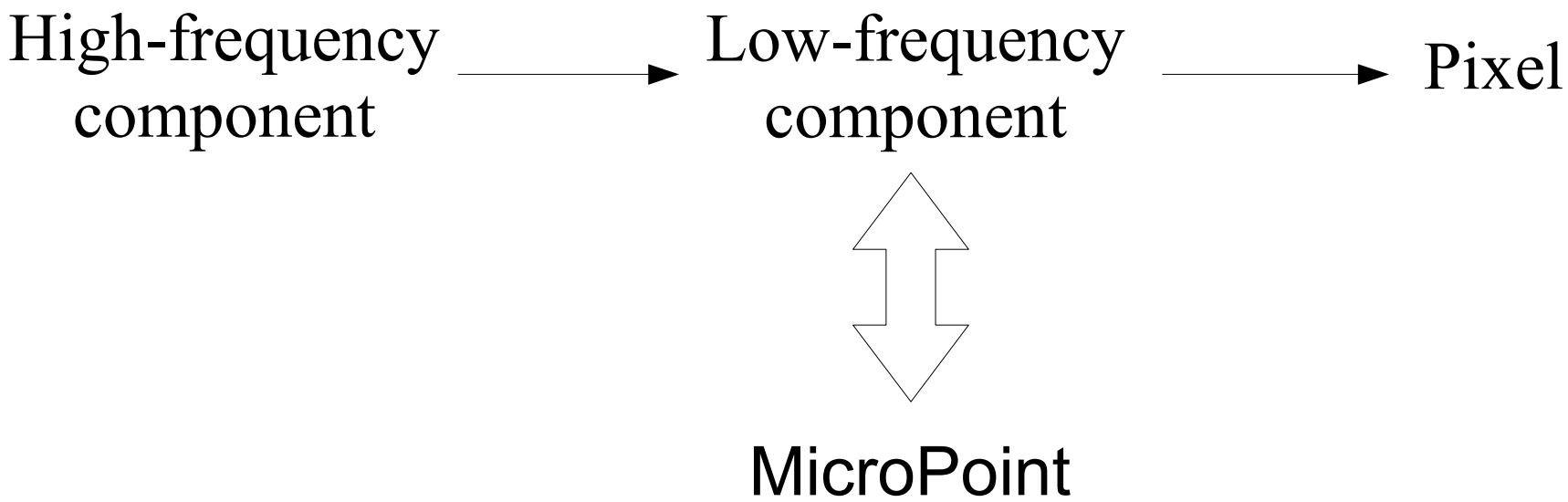
High-frequency component

Super-sampling

Pixel

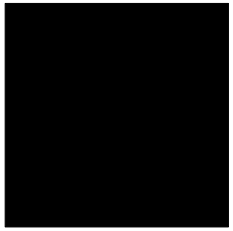



Super-sampling
High-frequency component
Low-frequency component
Pixel


High-frequency component
Low-frequency component
Pixel
MicroPoint

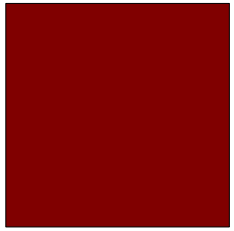
What is a pixel color?

Background



RGB=(0,0,0)

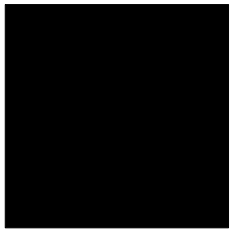
Observation



(128,0,0)

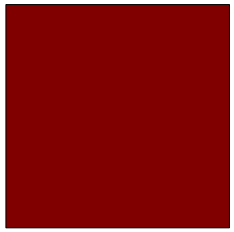
What is a pixel color?

Background



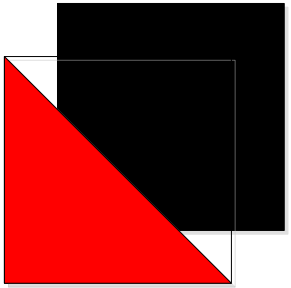
RGB=(0,0,0)

Observation

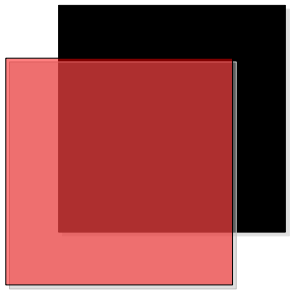


(128,0,0)

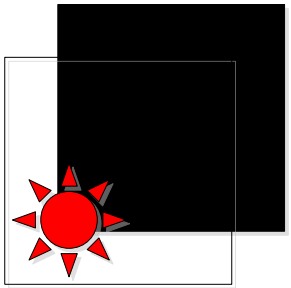
RGB=(128, 0, 0)
Alpha=100%=1
Coverage=100%=1



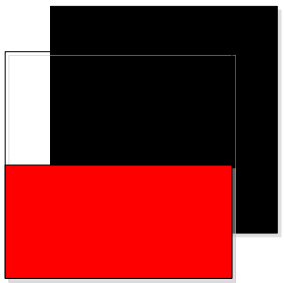
(255, 0, 0)
Alpha=1
Coverage=0.5



(255, 0, 0)
Alpha=0.5
Coverage=1



(1280, 0, 0)
Alpha=1
Coverage=0.1

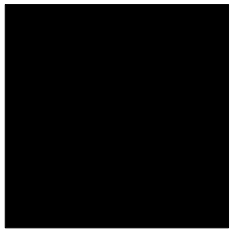


(255, 0, 0)
Alpha=1
Coverage=0.5

?

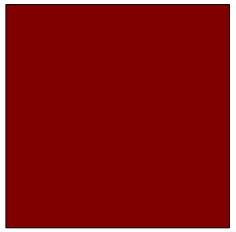
What is a pixel color?

Background



RGB=(0,0,0)

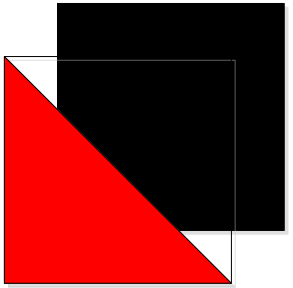
Observation



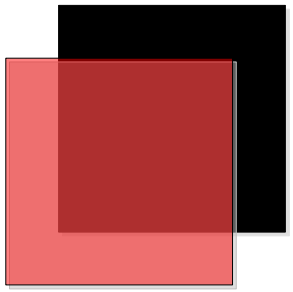
(128,0,0)

Observation = Coverage × Alpha × RGB

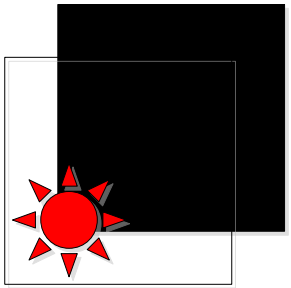
RGB=(128, 0, 0)
Alpha=100%=1
Coverage=100%=1



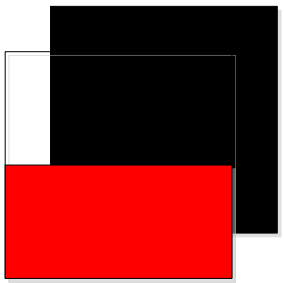
(255, 0, 0)
Alpha=1
Coverage=0.5



(255, 0, 0)
Alpha=0.5
Coverage=1



(1280, 0, 0)
Alpha=1
Coverage=0.1



(255, 0, 0)
Alpha=1
Coverage=0.5

?

Interpretation of the observation

$$\text{Observation} = \text{Coverage} \times \text{Alpha} \times \text{RGB}$$

$$= \underbrace{1.0}_{\text{Normalized Coverage}} \times \underbrace{(\text{Coverage} \times \text{Alpha})}_{\text{Normalized Alpha}} \times \text{RGB}$$

Interpretation of the observation

$$\text{Observation} = \text{Coverage} \times \text{Alpha} \times \text{RGB}$$

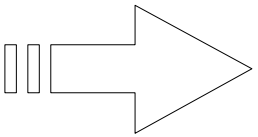
$$= \underbrace{1.0}_{\text{Normalized Coverage}} \times \underbrace{(\text{Coverage} \times \text{Alpha})}_{\text{Normalized Alpha}} \times \text{RGB}$$

High-frequency component → Low-frequency component

High-frequency component

Coverage: 0.000001%
Alpha : 1.0
Intensity : 100,000

Object



Low-frequency component

Coverage: 100%
Alpha : 0.000001
Intensity : 100,000

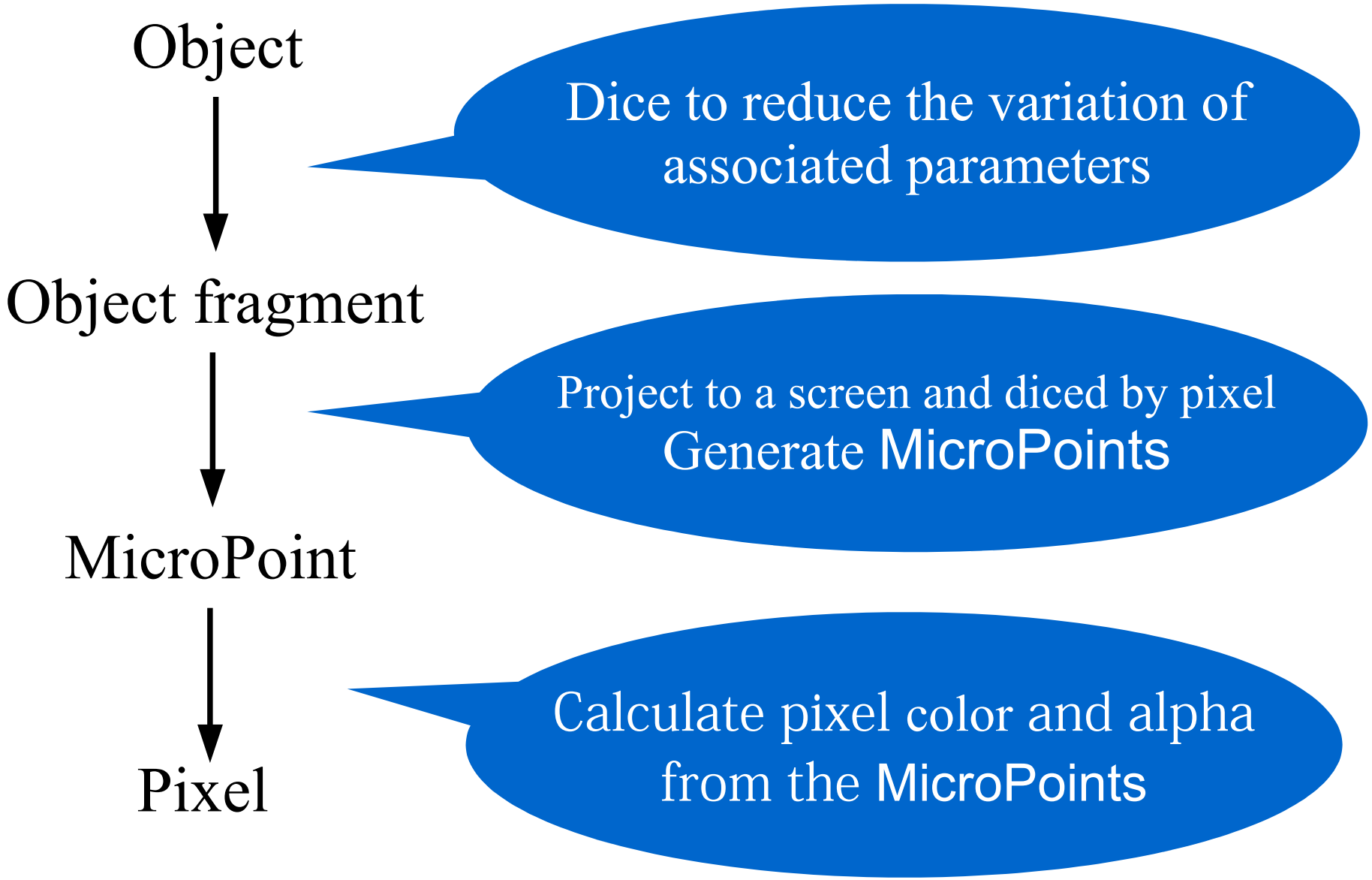
MicroPoint

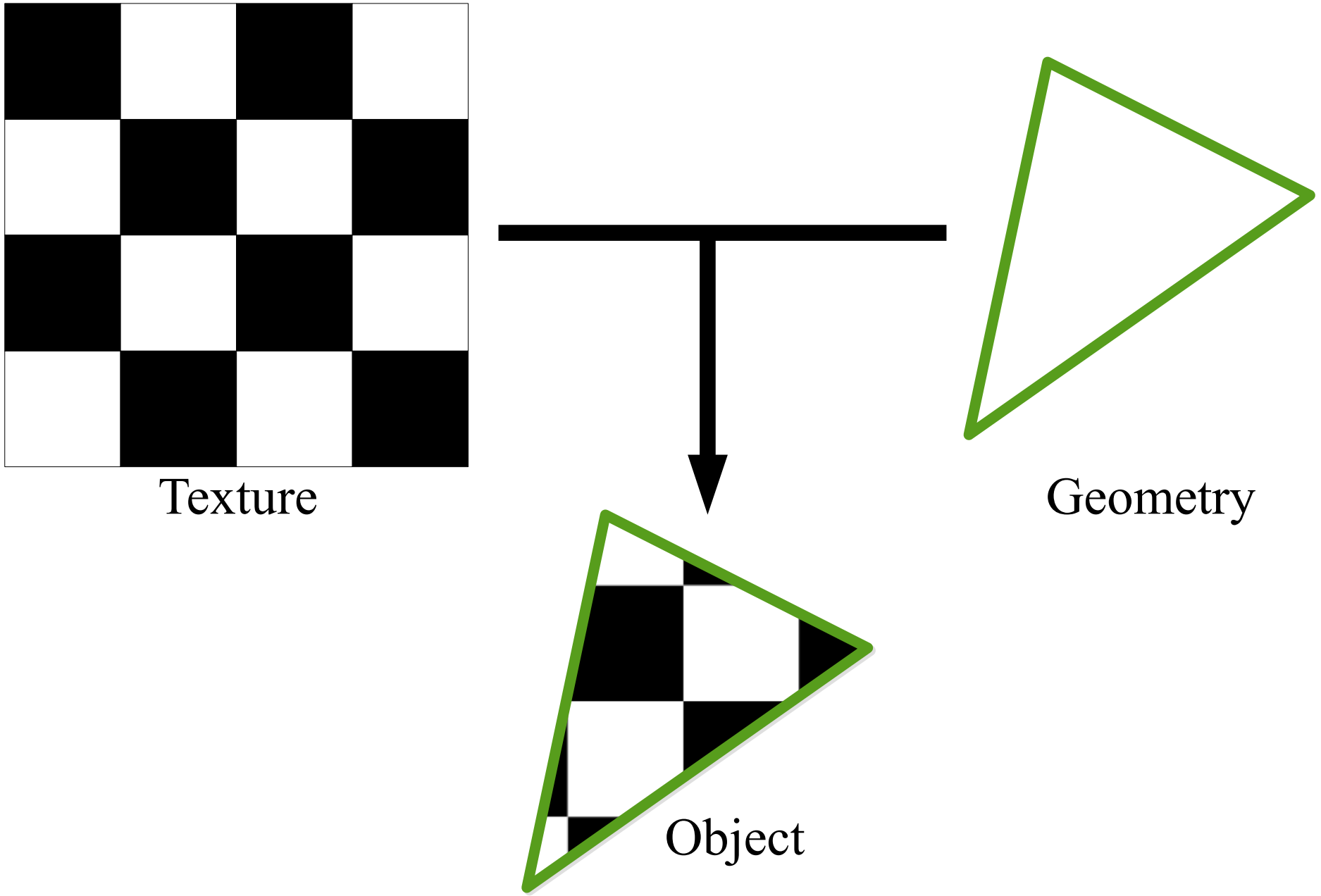
MicroPoint

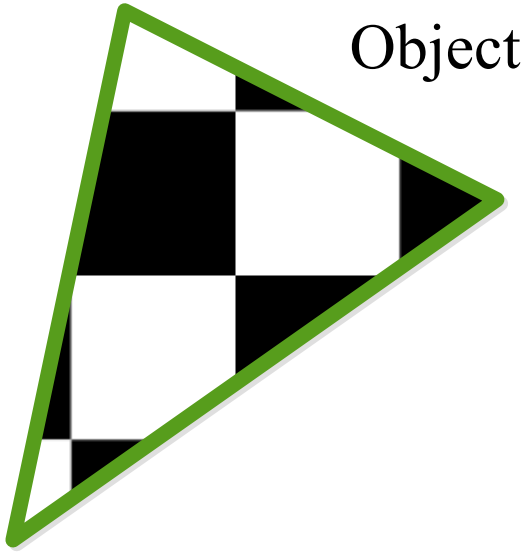
- Color
- Normalized Alpha

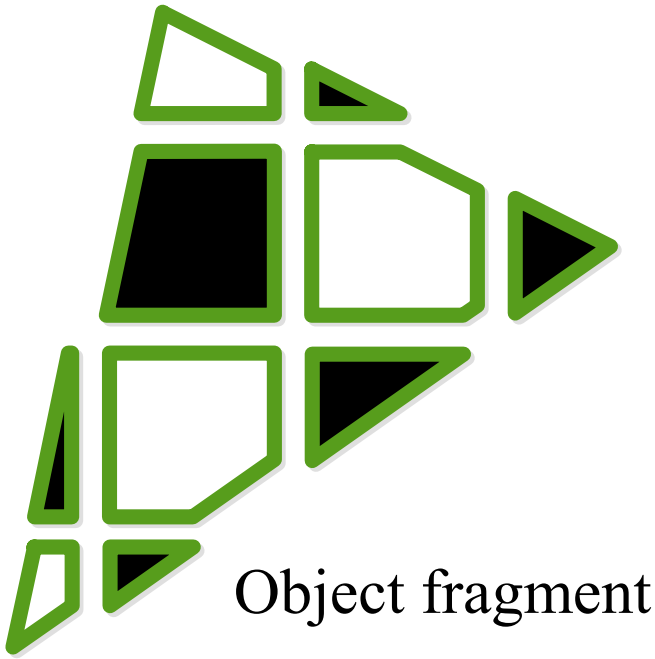
- Screen Position
- Depth info. (Zmin,Zmax)

- Optional info.
- Normal, Light, etc


Object
Dice to reduce the variation of associated parameters
Object fragment
Project to a screen and diced by pixel
Generate MicroPoints
MicroPoint
Calculate pixel color and alpha from the MicroPoints
Pixel

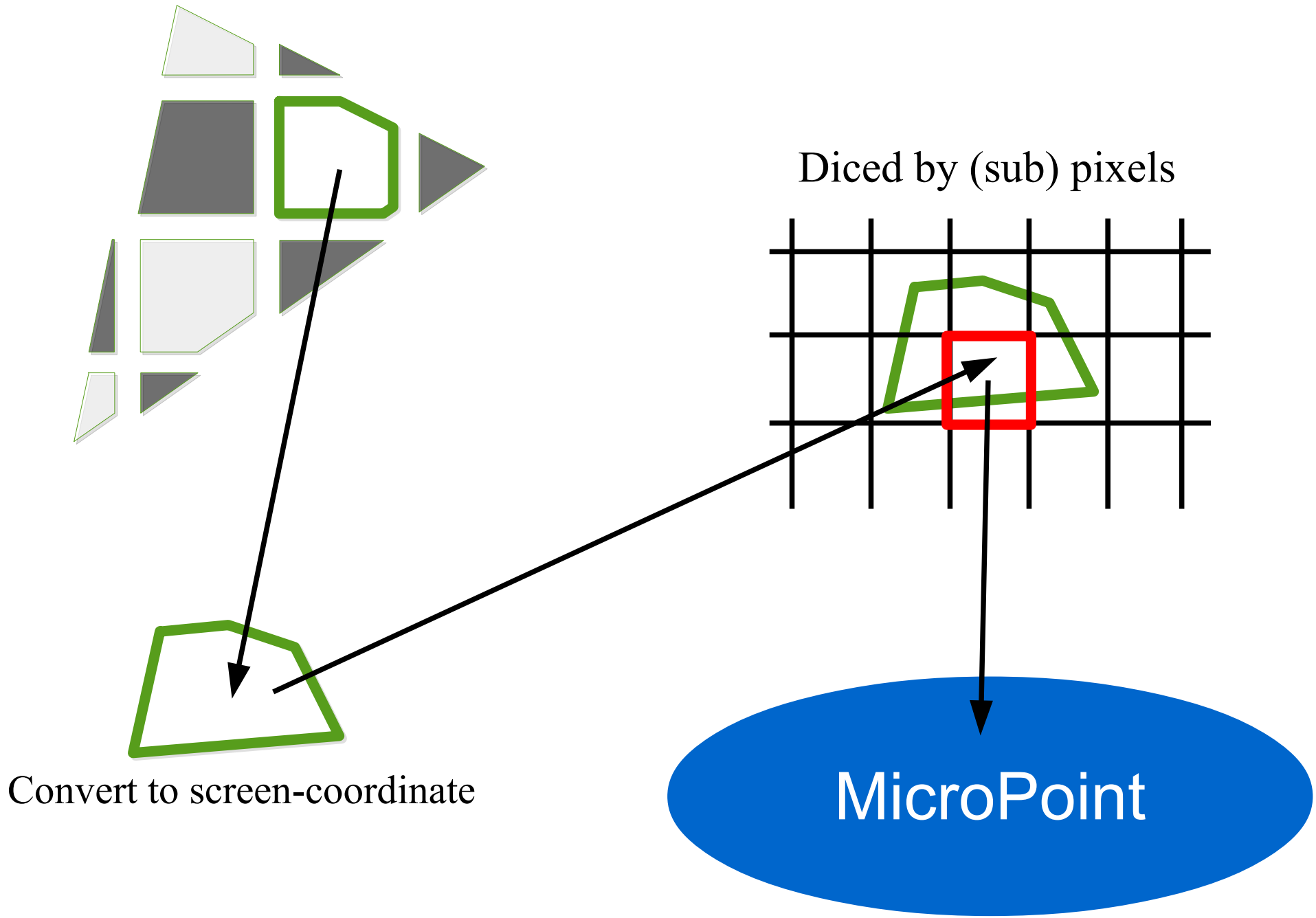

Texture
Geometry
Object

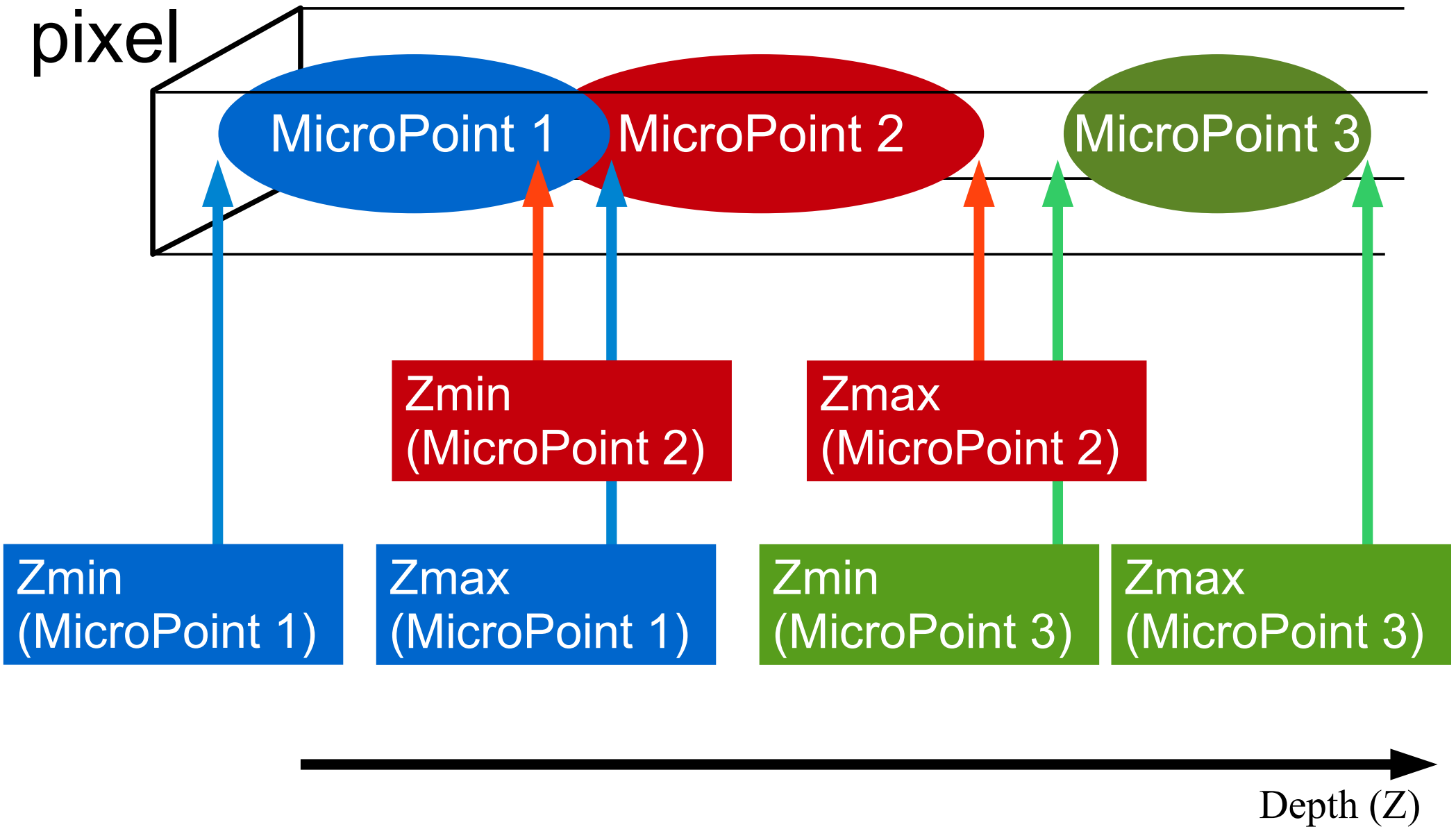


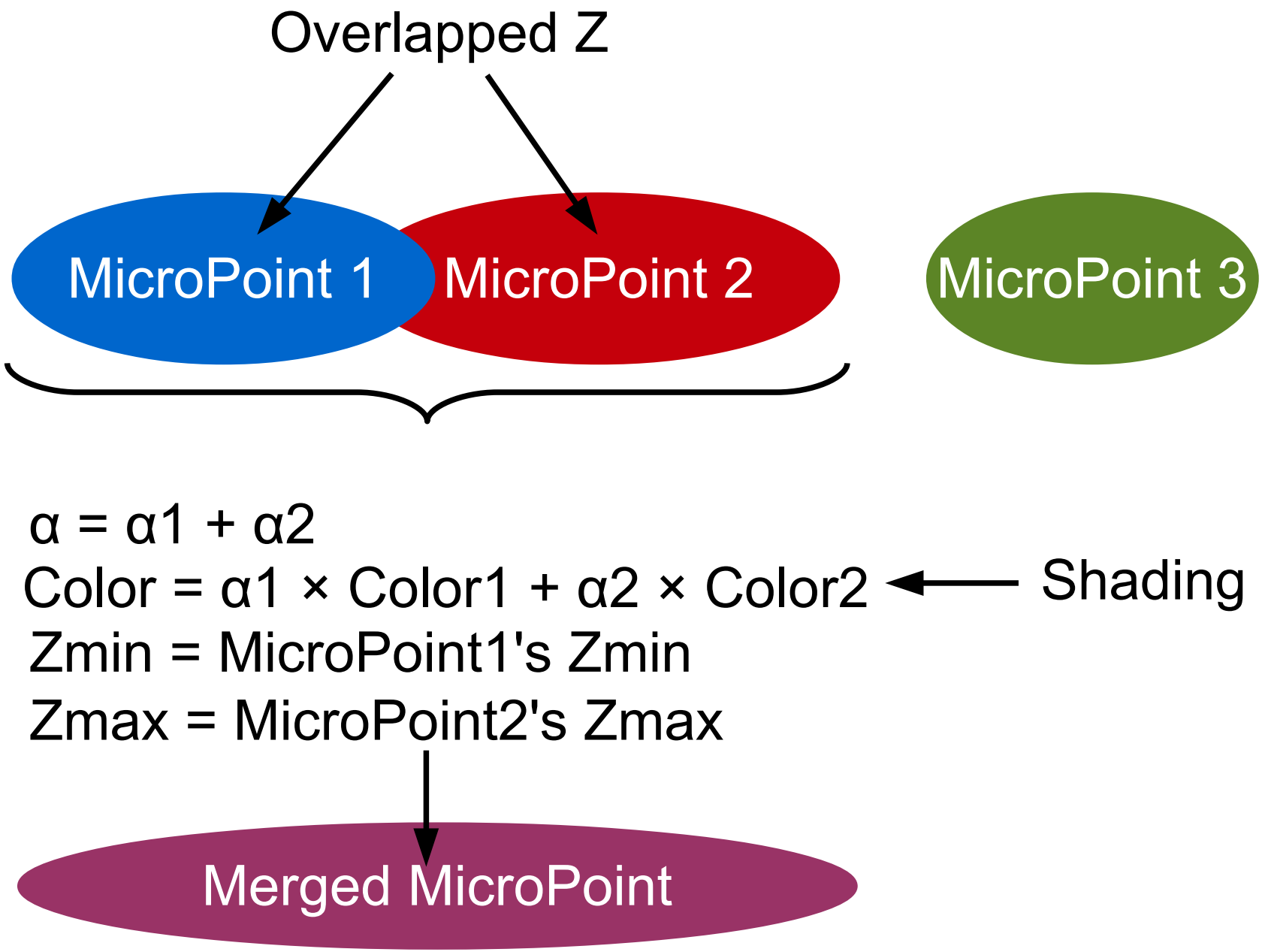


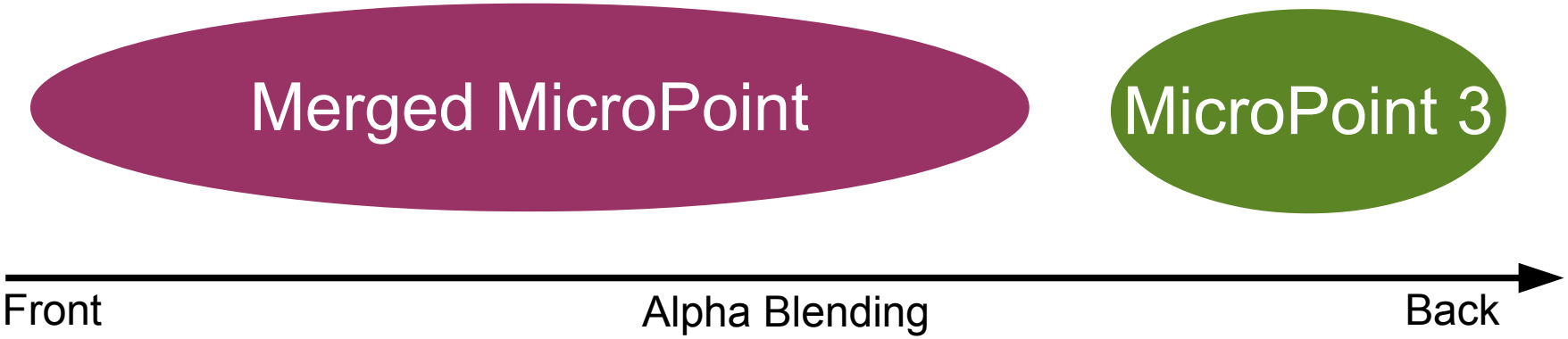
Dice into a object fragment
to reduce the variation of properties
(UV, Zmin, Zmax, Normal, etc) enough.

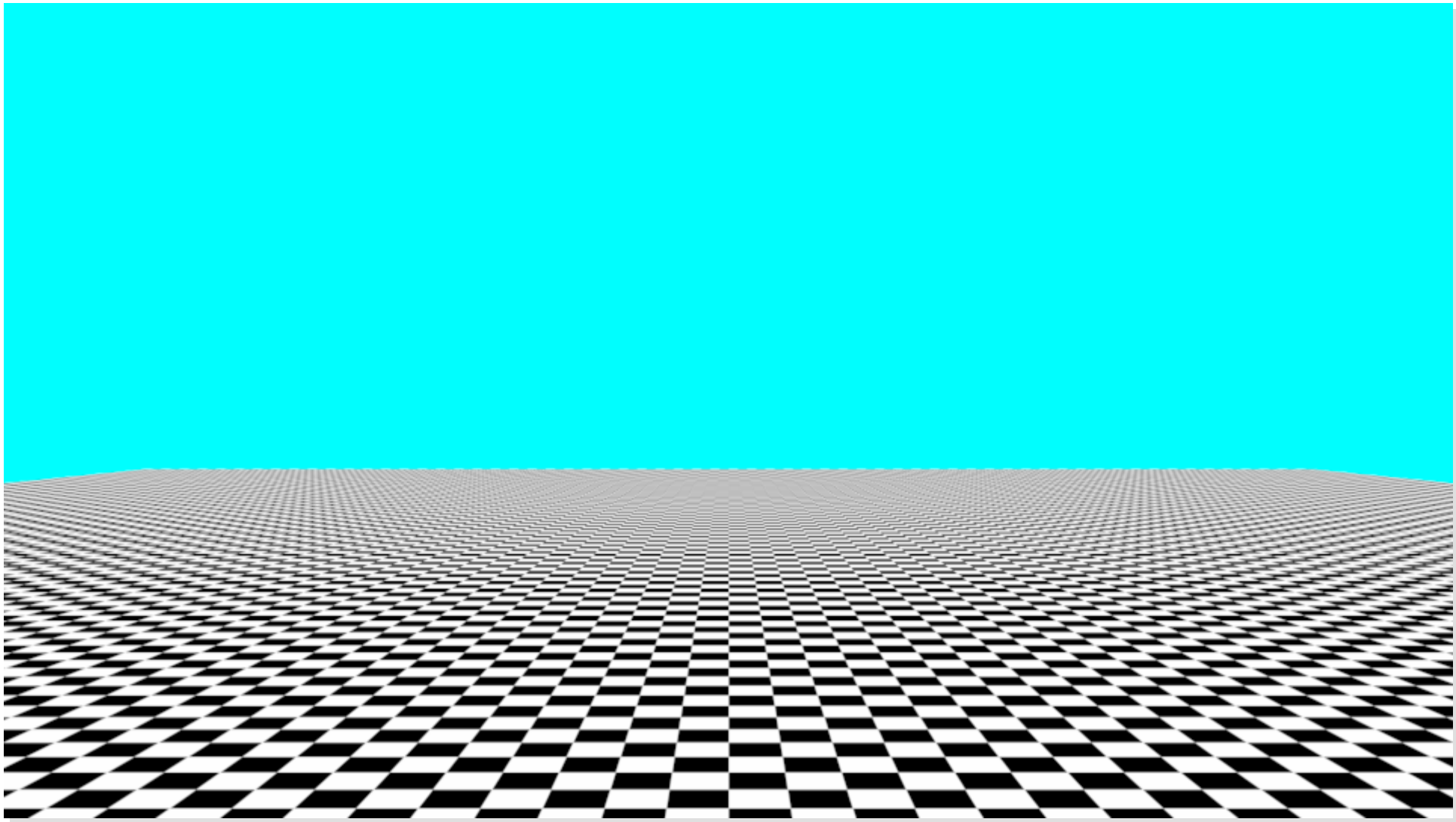
For example,
About UV, dice into object fragments until 1 texel/object-fragment.


Diced by (sub) pixels
Convert to screen-coordinate
MicroPoint

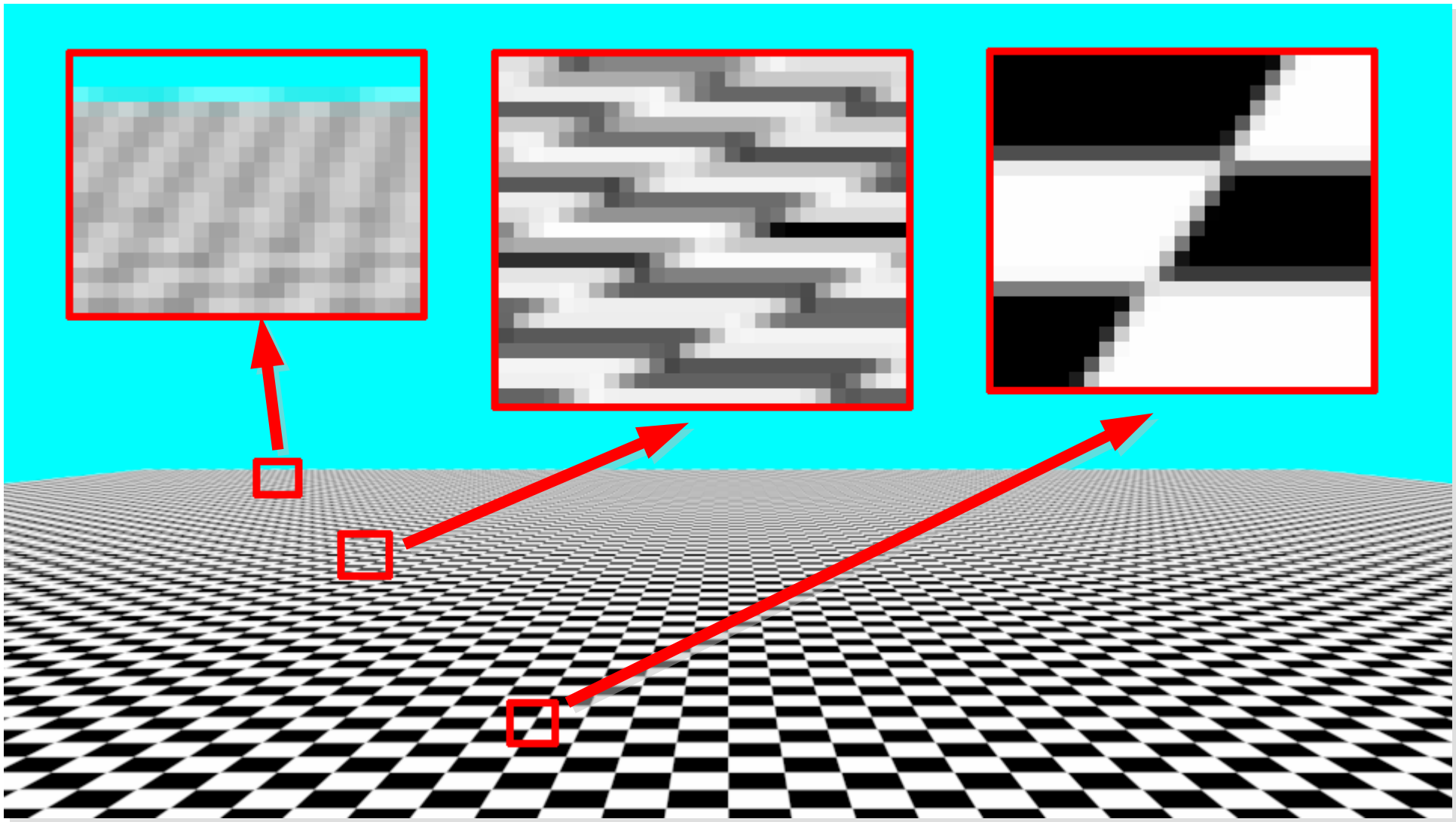

pixel
MicroPoint 1
MicroPoint 2
MicroPoint 3
Zmin (MicroPoint 2)
Zmax (MicroPoint 2)
Zmin (MicroPoint 1)
Zmax (MicroPoint 1)
Zmin (MicroPoint 3)
Zmax (MicroPoint 3)
Depth (Z)

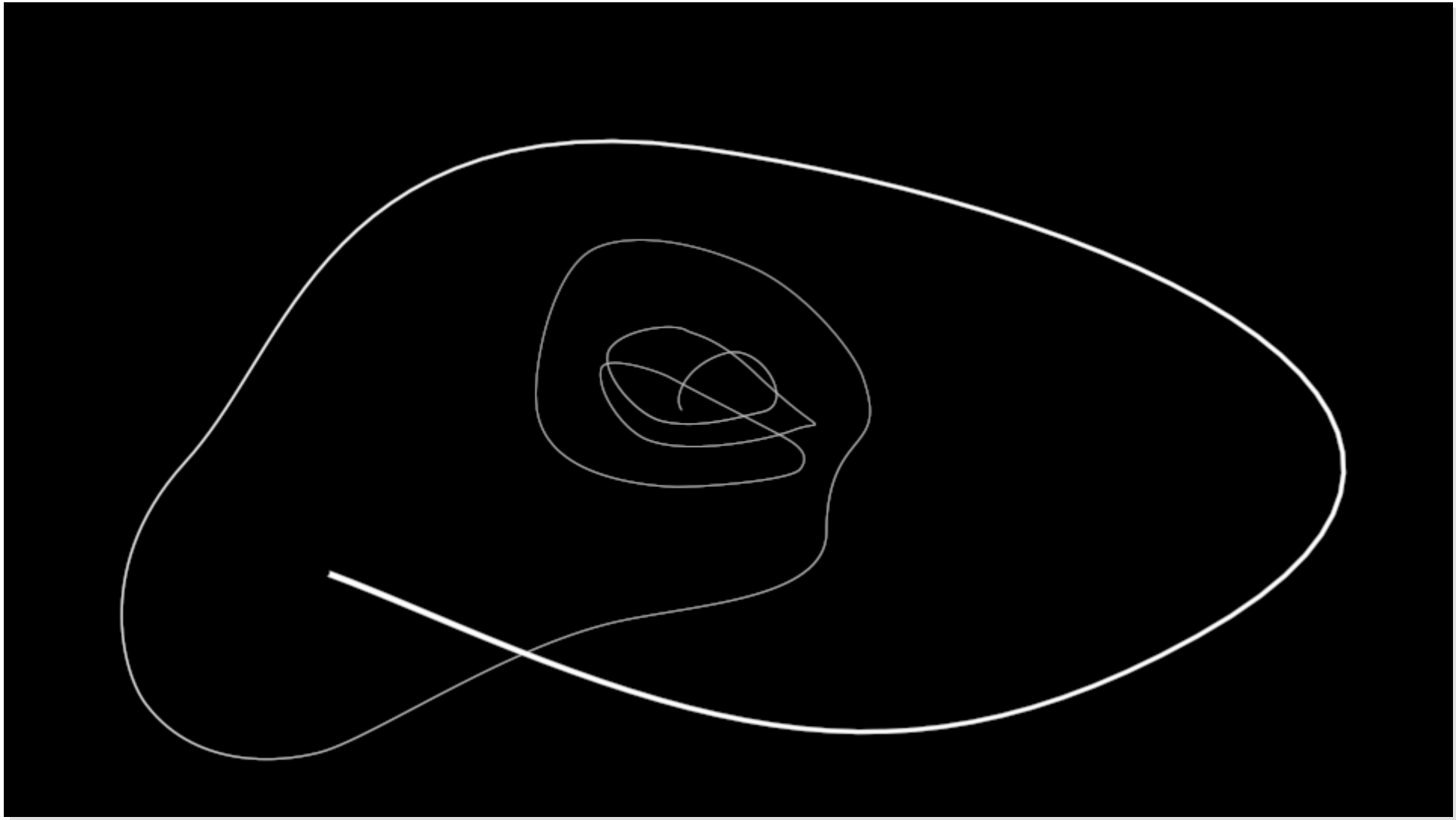

Overlapped Z
MicroPoint 1
MicroPoint 2
MicroPoint 3
α = α1 + α2
Color = α1 × Color1 + α2 × Color2
Shading
Zmin = MicroPoint1's Zmin
Zmax = MicroPoint2's Zmax
Merged MicroPoint


Merged MicroPoint
MicroPoint 3
Front
Alpha Blending
Back

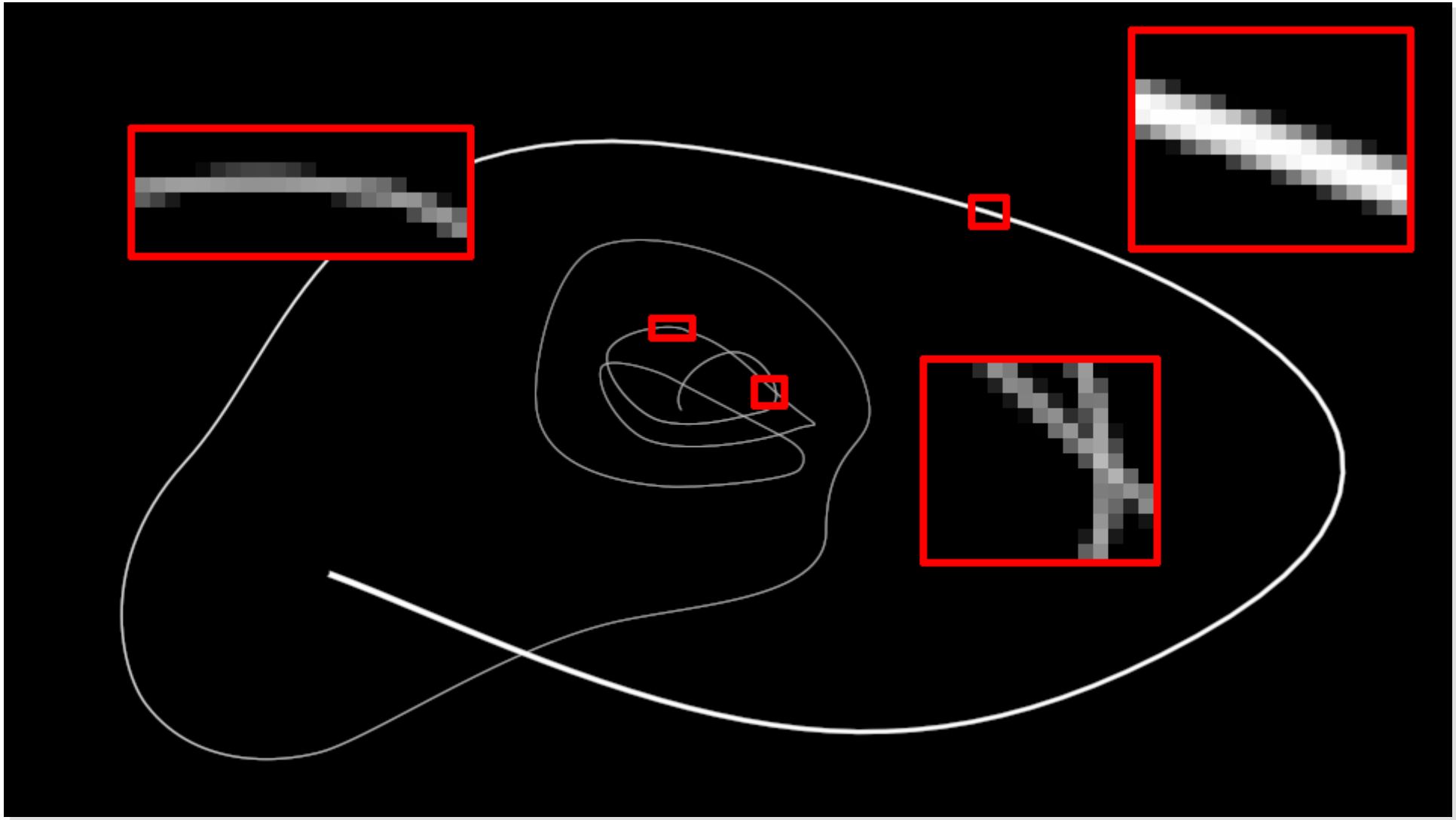


Note: Without super-sampling and any texture filterings.

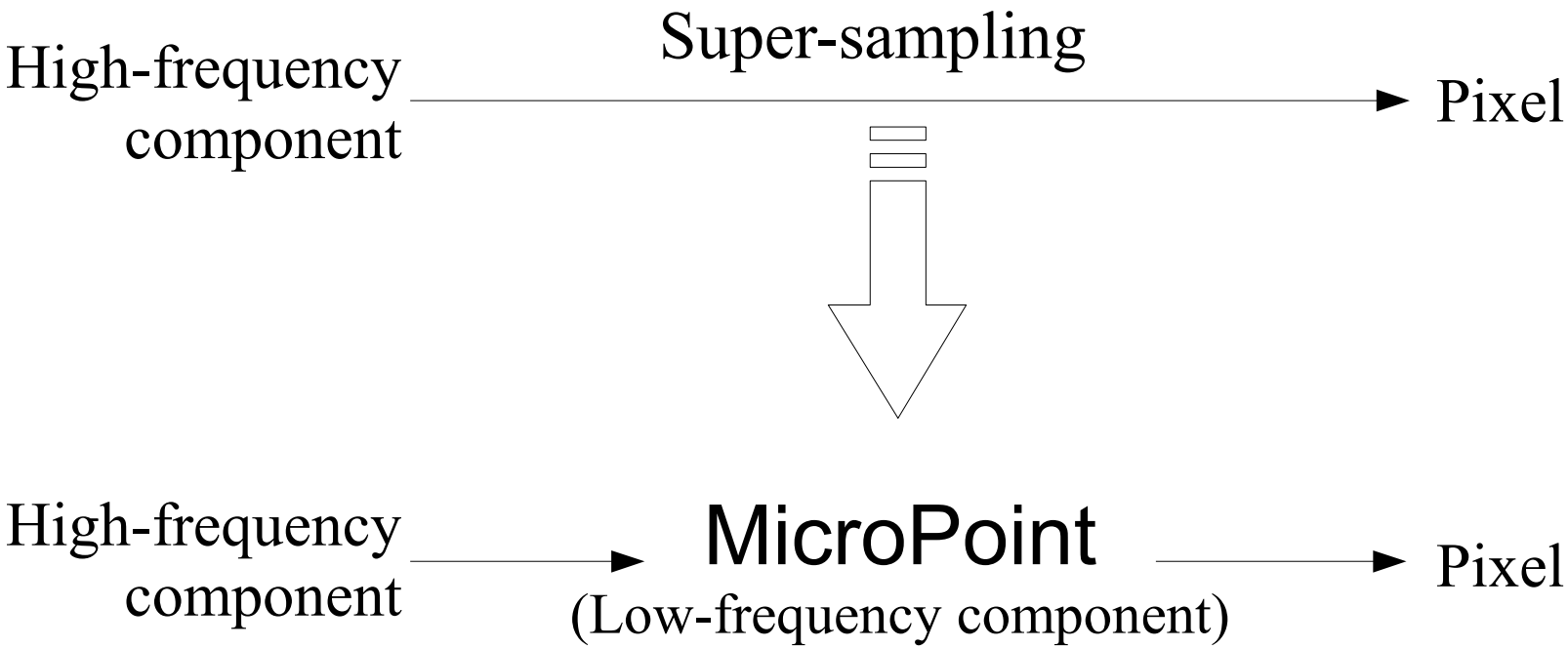




Note: Without super-sampling.



Conclusion



Future tasks

Realize soft shadow, caustics, color bleeding, etc. by MicroPoint.

Note: Hard shadow is already realized by MicroPoint (MicroPoint Based Shadow).
We are writing a this paper now.

http://micropoint.jimdo.com/